

**UNIVERSIDAD INTERNACIONAL DE LAS AMÉRICAS
ESCUELA DE INGENIERÍA INFORMÁTICA**

**TRABAJO FINAL DE GRADUACIÓN PARA OPTAR POR EL GRADO DE
LICENCIATURA EN INGENIERÍA EN INFORMÁTICA CON ÉNFASIS EN
GERENCIA**

**Prototipo Funcional Para La Gestión De Ventas De La Empresa Zapatitos Lindos S.A,
Ubicada En San José**

Daniel David Quesada Fallas

Máster Carlos De La O Fonseca

TUTOR

Sede Central

MARZO, 2024

CONTENIDO

DEDICATORIA	2
AGRADECIMIENTOS	3
CARTA DE APROBACIÓN DEL TUTOR	4
SOLICITUD DE DEFENSA	5
CARTA DE APROBACIÓN DEL TRIBUNAL EXAMINADOR	6
CARTA DE AUTORIZACIÓN DE LA DIRECCIÓN DE CARRERA	7
CARTA DEL LECTOR	8
CÓDIGO DE ÉTICA	9
CARTA DEL FILÓLOGO	10
DECLARACIÓN JURADA	11
TABLAS	17
FIGURAS	21
RESUMEN EJECUTIVO	24
CAPÍTULO I: INTRODUCCIÓN	25
Planteamiento Del Problema.....	25
Objetivos.....	27
<i>Objetivo General</i>	27
<i>Objetivos Específicos</i>	27
Justificación.....	27
<i>Viabilidad Técnica</i>	28
<i>Viabilidad Operativa</i>	31
<i>Viabilidad Económica</i>	32
<i>Viabilidad Legal</i>	34
Proyecciones.....	35
<i>Alcance Funcional</i>	35
<i>Alcance Metodológico</i>	38
<i>Alcance Tecnológico</i>	40
CAPÍTULO II: MARCO REFERENCIAL	41
Gestión De Ventas.....	41
Prototipo Funcional.....	42

Software Web.....	43
Sistema Tecnológico De Información.....	45
Requerimientos Funcionales y No Funcionales	46
Diagramas UML.....	47
<i>Diagrama De Casos De Uso</i>	47
<i>Diagrama De Secuencia</i>	47
<i>Diagrama De Contexto</i>	48
<i>Diagrama De Actividad</i>	48
<i>Diagrama De Clases</i>	48
Arquitectura De Software.....	49
Entorno De Desarrollo Integrado	51
<i>Lenguaje de programación</i>	52
<i>Asp.NET Core</i>	52
<i>JavaScript</i>	53
<i>Hojas De Estilo CSS</i>	53
<i>JQuery</i>	53
<i>ORM</i>	53
<i>GitHub</i>	54
Bases De Datos.....	56
<i>Diagrama Entidad Relación</i>	57
<i>Claves Primarias y Claves Foráneas</i>	58
<i>Diccionario de Datos</i>	59
<i>Lenguajes De Base De Datos</i>	59
Facturación Electrónica.....	59
<i>Factura Electrónica</i>	63
<i>Tiquete Electrónico</i>	64
<i>Notas De Crédito y Débito Electrónicas</i>	64
<i>Notas De Crédito y Débito Financieras</i>	65
<i>Archivo XML</i>	66
<i>Firma Electrónica</i>	66
<i>Catálogo CABys</i>	68

Gestión De Cuentas Por Cobrar y Cuentas por Pagar	69
<i>Cuentas Por Cobrar</i>	69
<i>Cuentas Por Pagar</i>	70
Conceptos Contables	70
<i>Activos</i>	70
<i>Pasivos</i>	71
<i>Patrimonio</i>	71
<i>Asientos Contables</i>	72
<i>Libro De Diario</i>	72
<i>Balance General</i>	72
<i>Catálogo Contable</i>	72
Control De Inventarios	73
<i>Inventario</i>	73
<i>Tipos De Inventarios</i>	74
Test Driven Development	76
Clean Code	76
<i>SOLID</i>	77
SCRUM.....	77
<i>INVEST</i>	78
CAPITULO III: MARCO METODOLÓGICO	80
Enfoque De La Investigación	80
<i>Enfoque Cualitativo</i>	81
<i>Enfoque De Investigación Seleccionado</i>	82
Método De La Investigación	83
<i>Investigación Descriptiva</i>	83
<i>Tipo De Investigación Seleccionado</i>	84
Fuentes De Información	85
<i>Fuente De Información Primaria</i>	85
<i>Fuente De Información Secundaria</i>	86
<i>Fuente De Información Terciaria</i>	86
<i>Fuente De información utilizada</i>	87

Variables o Unidades de Análisis.....	87
<i>Variable Conceptual</i>	88
<i>Variable Operacional</i>	88
<i>Variable Instrumental</i>	89
Instrumentos	92
Procesos Para La Recolección Y Análisis De Datos.....	93
CAPITULO IV: ANÁLISIS DE RESULTADOS	94
Entrevista.....	94
Observación.....	104
Análisis Integral	106
CAPITULO V: CONCLUSIONES Y RECOMENDACIONES	107
Conclusiones	107
Recomendaciones	108
CAPITULO VI: PROPUESTA	111
Análisis.....	111
<i>Análisis Detallado Del Software Desarrollado</i>	111
<i>Análisis Detallado De Telecomunicaciones</i>	119
<i>Descripción Detallada De las Herramientas Técnicas Utilizadas</i>	120
<i>Descripción Detallada Del Personal Requerido Para El Uso Del Sistema</i>	123
<i>Casos De Uso</i>	124
Diseño.....	198
<i>Arquitectura Del Sistema</i>	198
<i>Arquitectura Del Software</i>	199
<i>Diseño De Entradas</i>	200
<i>Diseño De Base De Datos y Su Respectivo Diccionario De Base De Datos</i>	222
<i>Diseño De Procesos</i>	254
<i>Diseño De Salidas</i>	260
<i>Diagramas UML</i>	265
Programación	271
<i>Entradas y salidas</i>	271

<i>Procesos</i>	286
<i>Validaciones</i>	298
Pruebas	303
<i>Scripts De Pruebas Unitarias</i>	303
<i>Casos De Prueba</i>	311
REFERENCIAS	320
ÁPENDICES	326
Apéndice A: Guía de entrevista semiestructurada aplicada	326
<i>Entrevista Semiestructurada Aplicada</i>	326
Apéndice B: Guía de Observación	327
<i>Entrevista Semiestructurada Aplicada</i>	327

TABLAS

Tabla 1.	30
Tabla 2.	30
Tabla 3.	33
Tabla 4.	33
Tabla 5.	40
Tabla 6.	90
Tabla 7.	117
Tabla 8.	118
Tabla 9.	120
Tabla 10.	125
Tabla 11.	128
Tabla 12.	128
Tabla 13.	130
Tabla 14.	132
Tabla 15.	136
Tabla 16.	138
Tabla 17.	142
Tabla 18.	145
Tabla 19.	147
Tabla 20.	149
Tabla 21.	152
Tabla 22.	155
Tabla 23.	157
Tabla 24.	160
Tabla 25.	162
Tabla 26.	163
Tabla 27.	167
Tabla 28.	169
Tabla 29.	170
Tabla 30.	172
Tabla 31.	174
Tabla 32.	176
Tabla 33.	178

Tabla 34.	180
Tabla 35.	182
Tabla 36.	184
Tabla 37.	192
Tabla 38.	195
Tabla 39.	223
Tabla 40.	223
Tabla 41.	223
Tabla 42.	224
Tabla 43.	224
Tabla 44.	224
Tabla 45.	224
Tabla 46.	225
Tabla 47.	225
Tabla 48.	225
Tabla 49.	225
Tabla 50.	226
Tabla 51.	227
Tabla 52.	227
Tabla 53.	227
Tabla 54.	228
Tabla 55.	228
Tabla 56.	228
Tabla 57.	229
Tabla 58.	229
Tabla 59.	229
Tabla 60.	230
Tabla 61.	230
Tabla 62.	230
Tabla 63.	230
Tabla 64.	231
Tabla 65.	231
Tabla 66.	231
Tabla 67.	231

Tabla 68.	232
Tabla 69.	233
Tabla 70.	233
Tabla 71.	233
Tabla 72.	234
Tabla 73.	234
Tabla 74.	234
Tabla 75.	234
Tabla 76.	235
Tabla 77.	235
Tabla 78.	236
Tabla 79.	236
Tabla 80.	238
Tabla 81.	238
Tabla 82.	239
Tabla 83.	239
Tabla 84.	240
Tabla 85.	240
Tabla 86.	240
Tabla 87.	241
Tabla 88.	241
Tabla 89.	242
Tabla 90.	244
Tabla 91.	245
Tabla 92.	246
Tabla 93.	247
Tabla 94.	248
Tabla 95.	249
Tabla 96.	250
Tabla 97.	250
Tabla 98.	250
Tabla 99.	250
Tabla 100.	251
Tabla 101.	251

Tabla 102.	252
Tabla 103.	252
Tabla 104.	252
Tabla 105.	253
Tabla 106.	303
Tabla 107.	304
Tabla 108.	305
Tabla 109.	306
Tabla 110.	307
Tabla 111.	308
Tabla 112.	309
Tabla 113.	310
Tabla 114.	311
Tabla 115.	312
Tabla 116.	314
Tabla 117.	315
Tabla 118.	316
Tabla 119.	317
Tabla 120.	318
Tabla 121.	327

FIGURAS

Figura 1.	124
Figura 2.	198
Figura 3.	199
Figura 4.	201
Figura 5.	202
Figura 6.	203
Figura 7.	204
Figura 8.	204
Figura 9.	205
Figura 10.	205
Figura 11.	206
Figura 12.	207
Figura 13.	208
Figura 14.	208
Figura 15.	209
Figura 16.	209
Figura 17.	210
Figura 18.	210
Figura 19.	211
Figura 20.	211
Figura 21.	212
Figura 22.	212
Figura 23.	213
Figura 24.	213
Figura 25.	214
Figura 26.	214
Figura 27.	215
Figura 28.	215
Figura 29.	216
Figura 30.	216
Figura 31.	217
Figura 32.	217

Figura 33.....	218
Figura 34.....	218
Figura 35.....	219
Figura 36.....	219
Figura 37.....	220
Figura 38.....	220
Figura 39.....	221
Figura 40.....	221
Figura 41.....	222
Figura 42.....	254
Figura 43.....	255
Figura 44.....	256
Figura 45.....	257
Figura 46.....	259
Figura 47.....	260
Figura 48.....	261
Figura 49.....	262
Figura 50.....	263
Figura 51.....	264
Figura 52.....	265
Figura 53.....	266
Figura 54.....	267
Figura 55.....	268
Figura 56.....	269
Figura 57.....	270
Figura 58.....	272
Figura 59.....	273
Figura 60.....	274
Figura 61.....	275
Figura 62.....	276
Figura 63.....	277
Figura 64.....	278
Figura 65.....	279
Figura 66.....	280

Figura 67.....	281
Figura 68.....	282
Figura 69.....	283
Figura 70.....	284
Figura 71.....	285
Figura 72.....	286
Figura 73.....	287
Figura 74.....	288
Figura 75.....	289
Figura 76.....	290
Figura 77.....	291
Figura 78.....	292
Figura 79.....	293
Figura 80.....	294
Figura 81.....	294
Figura 82.....	295
Figura 83.....	296
Figura 84.....	297
Figura 85.....	298
Figura 86.....	299
Figura 87.....	300
Figura 88.....	301
Figura 89.....	302
Figura 90.....	303
Figura 91.....	304
Figura 92.....	305
Figura 93.....	307
Figura 94.....	308
Figura 95.....	309
Figura 96.....	310
Figura 97.....	310

RESUMEN EJECUTIVO

El presente proyecto tiene como objetivo desarrollar un prototipo funcional para la gestión de ventas de la empresa Zapatitos Lindos S.A., ubicada en San José. Se busca mejorar la generación de facturas, la gestión de inventarios y las órdenes de trabajo, con el objetivo de aumentar la eficiencia y rentabilidad de la empresa.

Para llevar a cabo la investigación, se seleccionó el enfoque cualitativo, ya que permite una comprensión profunda de las problemáticas y necesidades expuestas. Además, se utilizará el tipo de investigación descriptiva, lo que proporcionará una base sólida para el diseño, la programación y las pruebas del prototipo funcional.

En el proceso de desarrollo de este prototipo, la administradora y la gerente aportarán su experiencia y conocimiento para asegurar que el prototipo cumpla con las necesidades y objetivos específicos del proyecto. Su colaboración será esencial para recolectar datos precisos y garantizar que el sistema final sea adecuado y eficiente.

Este proyecto representa un avance sustancial hacia la mejora de los procesos de ventas en Zapatitos Lindos S.A. La combinación de herramientas de investigación efectivas y enfoques de desarrollo sólidos ha culminado en un prototipo funcional altamente prometedor que aborda con éxito los desafíos identificados.

Se recomienda incorporar funcionalidades adicionales que no se contemplaron en el alcance del prototipo funcional, como la capacidad de calcular importaciones y la implementación de un portal para la aceptación de facturas electrónicas. Estas adiciones ampliarán la utilidad del prototipo con beneficios tangibles en seguridad, costos, productividad y gestión de datos.

CAPÍTULO I: INTRODUCCIÓN

Planteamiento Del Problema

Zapatitos Lindos S.A. es una empresa fundada en Costa Rica en el año 1996 con el fin de ofrecer al consumidor una alternativa diferente en calzado ortopédico y medicado, tanto para adultos diabéticos y artríticos, como para niños y jóvenes en las diferentes etapas de formación y corrección del pie. Con respecto al calzado medicado la empresa se encarga de realizar las adaptaciones o modificaciones requeridas al calzado seleccionado por el cliente para lo cual se genera una orden de trabajo que finaliza con el proceso de facturación.

Zapatitos en la actualidad tiene 6 empleados y cuenta con dos sucursales: la principal se ubica en Barrio Don Bosco, avenida 2, calle 34, costado norte del Colegio María Auxiliadora y la otra sucursal se ubica en Escazú, en el Centro Comercial el Cruce.

El horario de las zapaterías es de lunes a viernes de 9:00 a.m. a 5:30 p.m. y los días sábado de 9:00 a.m. a 3:00 p.m. en jornada continua.

Los problemas que presenta la empresa Zapatitos Lindos S.A. son los siguientes:

1. Excesiva carga de trabajo administrativo, pérdida de tiempo y recursos valiosos a la hora de generar las facturas: el proceso de facturación actualmente es bastante tedioso y presenta una serie de inconsistencias, lo cual se traduce en una mayor carga de trabajo administrativo para los vendedores. El sistema actual presenta las siguientes inconsistencias: desconexiones, cálculo incorrecto de impuestos, saltos de consecutivos, rebajos incorrectos de inventarios. Las situaciones anteriores generan desconfianza y una carga extra de trabajo al tener que estar revisando los cálculos y rebajos de inventarios. Adicionalmente, el sistema actual es bastante lento y pierde conexión frecuentemente lo que obliga a iniciar el proceso desde cero.
2. Pérdida física de productos e inconsistencias del inventario: los inventarios de la empresa presentan inconsistencias con respecto a las cantidades que se indican en el sistema y las cantidades físicas; adicionalmente, se presenta pérdida física de productos por falta de control y funcionalidad del sistema actual. Se identifica que el proceso de anulación de facturas implementado en el sistema informático actual presenta varias inconsistencias, entre las que se pueden citar las siguientes: imposibilidad para anular parcialmente una factura, problemas de integridad, ya que el proceso se realiza de manera incorrecta, lo que genera inconsistencias en el manejo del inventario.

Se ha presentado el escenario en donde una factura con 10 líneas debe generar una nota de crédito por una única línea; y este proceso no se encuentra disponible en el sistema, por lo que deben anular completamente la factura, realizar ajustes de inventario manuales y así poder emitir una nueva factura; perdiendo el control y la trazabilidad de las unidades de inventario.

3. Pérdida de dinero: no se cuenta con un sistema automatizado que ayude a identificar, hacer seguimiento y gestionar de manera adecuada los saldos por cobrar que se emiten cuando se realiza una factura con la condición de venta "Crédito". El control se lleva de forma manual, esto ha generado inconsistencias, tales como atrasos en la recaudación del dinero, doble cobro a los clientes y pagos no registrados en el archivo. Todas estas situaciones afectan la rentabilidad de la empresa, restan capacidad de pago y afectan la operación.
4. Pérdida de oportunidades de negocio y falta de visibilidad: no se cuenta con un sistema automatizado para gestionar las cuentas por pagar; actualmente la gestión de los pagos a los proveedores de materias primas y zapatos se realiza manualmente; esto causa lentitud en el proceso y lo hace propenso a errores. Así mismo, la empresa puede estar perdiendo oportunidades de negocio o descuentos por pronto pago; adicionalmente, sin un sistema automatizado que gestione las cuentas por pagar es difícil tener una visión global de lo que se debe pagar, lo que puede dificultar la toma de decisiones y la planificación financiera.
5. Descontento de clientes y retraso en la producción: la empresa ofrece calzado ortopédico y medicado para adultos con diabetes o artritis, así como para niños y jóvenes en las diferentes etapas de formación y corrección de pie; por lo que en su mayoría los productos pasan por un proceso de modificación y adaptación según las necesidades de cliente.

Actualmente no se cuenta con el inventario de insumos necesario para la adaptación o modificación de los zapatos, lo cual genera retrasos en la producción y, por ende, en la entrega de los productos a los clientes que se traduce en un descontento y pérdida de confianza de los clientes. Por otra parte, al no contar con un proceso automatizado de órdenes de trabajo, no se tiene información en tiempo real sobre su estado; esto podría influir, también, en una disminución de ventas y los ingresos.

6. Falta de visibilidad financiera y pérdida de tiempo en el registro manual de asientos: la empresa no cuenta con un sistema contable automatizado que permita la consolidación de la información importante para la continuidad del negocio. Actualmente el registro contable se

realiza de manera manual y la contabilidad no se lleva al día, por lo que no se tiene visibilidad financiera y existe una pérdida de tiempo en el proceso de registro de asientos.

Objetivos

Objetivo General

Desarrollar un prototipo funcional para la gestión de ventas de la empresa Zapatitos Lindos S.A, ubicada en San José.

Objetivos Específicos

- Analizar en detalle los requerimientos técnicos y funcionales para el prototipo propuesto, utilizando la información recopilada de los involucrados en el proceso de ventas de la empresa.
- Diseñar un prototipo funcional para la gestión de ventas de la empresa, que cumpla con los requerimientos específicos definidos, garantizando un sistema eficiente y escalable.
- Programar los módulos necesarios solventando la problemática que se presenta en la empresa y cumpliendo con los requerimientos solicitados.
- Probar de manera exhaustiva y sistemática el prototipo de gestión de ventas de la empresa, evaluando y determinando su correcto funcionamiento.

Justificación

El presente proyecto tiene como objetivo desarrollar un prototipo funcional para la gestión de ventas de la empresa Zapatitos Lindos S.A., ubicada en San José. En el contexto de la obtención del grado de licenciatura en Ingeniería en Informática con énfasis en Gerencia, esta investigación busca abordar los desafíos que enfrenta la empresa en su gestión de ventas, mediante la implementación de soluciones tecnológicas y prácticas de desarrollo de software eficientes. Se hará hincapié en la adopción de metodologías ágiles, como Scrum, y el enfoque del Test Driven Development.

El prototipo propuesto permite aplicar los conocimientos adquiridos durante la formación en ingeniería informática y gerencia en un caso práctico y real. La gestión de ventas es una actividad fundamental en cualquier empresa y su optimización requiere de habilidades técnicas y administrativas. Este proyecto proporciona la oportunidad para combinar ambos campos de

conocimiento y demostrar la capacidad de abordar problemas empresariales desde una perspectiva integral.

El desarrollo del prototipo funcional implica la adopción de tecnologías avanzadas y mejores prácticas de desarrollo; esto contribuye a la innovación tecnológica y abre nuevas oportunidades para mejorar los procesos de gestión en la empresa.

La implementación de metodologías ágiles, como Scrum, facilita la adaptación ágil a los cambios y mejora la eficiencia en la gestión de proyectos, lo que se traduce en una ventaja competitiva y una mayor satisfacción por parte de los clientes. Además, la adopción de tecnologías y prácticas de desarrollo de *software* avanzadas promueve la innovación tecnológica tanto dentro de la empresa Zapatitos Lindos S.A. como en el campo de la ingeniería informática en general.

El enfoque del Test Driven Development en el desarrollo del prototipo funcional asegura la calidad del producto final. Al escribir pruebas automatizadas desde el inicio, se garantiza que cumpla con los requisitos establecidos, siendo más confiable y robusto. Esto resulta en una reducción de errores y problemas en la gestión de ventas, mejorando así la eficiencia y la rentabilidad.

Zapatitos Lindos S.A. obtendrá notables ventajas al utilizar el prototipo funcional para la gestión de ventas. Este avance optimizará la eficiencia y exactitud en la creación de facturas, el control de inventarios y el manejo de órdenes de trabajo, además de incorporar la facturación electrónica conforme a las normativas fiscales de Costa Rica.

Viabilidad Técnica

El proyecto de desarrollo del prototipo funcional para la gestión de ventas de Zapatitos Lindos S.A. es factible desde el punto de vista técnico, debido a las siguientes consideraciones:

- Características tecnológicas: el uso de ASP.NET Core MVC en conjunto con el lenguaje de programación C# como base de desarrollo brinda una plataforma sólida y flexible para la implementación del prototipo. Estas tecnologías permiten construir aplicaciones web robustas, escalables y de alto rendimiento. Además, el entorno de desarrollo integrado (IDE) seleccionado, Visual Studio 2022 Community, proporciona herramientas poderosas y una interfaz intuitiva para facilitar el desarrollo y la depuración del *software*.
- Base de datos y almacenamiento: la elección de SQL Server Express como motor de base de datos garantiza un soporte confiable y eficiente para el almacenamiento y gestión de los datos

de la empresa. Esta versión de SQL Server es adecuada para proyectos de tamaño mediano y permite el desarrollo sin restricciones significativas. Asimismo, el espacio de almacenamiento proporcionado por el servicio de hospedaje SmarterASP.NET es suficiente para manejar los datos de la empresa, con un límite de 10 GB en la base de datos.

- **Hospedaje y recursos:** la elección de SmarterASP.NET como proveedor de hospedaje se basa en su oferta de precios competitivos y características adecuadas para el proyecto. Los servidores con Windows Server 2022 y soporte a tecnologías de Microsoft ASP ofrecen un entorno estable y confiable para alojar la solución. Además, el hospedaje incluye espacio web ilimitado y ancho de banda ilimitado, lo que asegura un rendimiento óptimo y la capacidad de manejar un volumen de usuarios y transacciones sin problemas.
- **Licencias:** se utilizarán licencias gratuitas y complementos de terceros de código abierto, eliminando la necesidad de adquirir licencias adicionales. Además, al realizar el pago del plan de alojamiento, se incluyen tanto el motor de base de datos, así como el sistema operativo del servidor, lo cual proporciona una solución integral y costos previsibles en términos de infraestructura. Esta estrategia permite optimizar los recursos económicos y asegura la viabilidad técnica del proyecto al utilizar tecnologías accesibles y ampliamente respaldadas por la comunidad.
- **Experiencia y conocimiento:** se cuenta con experiencia en desarrollo de sistemas y los conocimientos necesarios para llevar a cabo el prototipo.
- **Recursos propios:** Zapitos Lindos S.A., dispone de recursos propios de *hardware* y *software* que cumplen con las especificaciones requeridas. Tanto los componentes de *hardware* como las aplicaciones de *software* cumplen con los requisitos técnicos necesarios, lo que proporciona un entorno propicio para el desarrollo del prototipo.

Con esta sólida base de recursos, se pueden cumplir los objetivos establecidos y asegurar la viabilidad técnica del proyecto de gestión de ventas de Zapitos Lindos S.A."

A continuación, se presenta la tabla que detalla los recursos de hardware que se utilizarán en la creación del prototipo funcional para la gestión de ventas de Zapitos Lindos S.A.

Cada uno de los elementos de la tabla cumplen con los requisitos técnicos para el desarrollo del prototipo.

Tabla 1.

Recursos de Hardware

Recurso	Detalle
1 laptop	Accer Aspire A515-56
Procesador	Intel(R) i7-1165G7 @ 2.80GHz 2.80 GHz
Memoria RAM instalada	20 GB
Almacenamiento	SDD 512 GB

Fuente: Elaboración propia (2024).

En cuanto a los recursos de *software*, se cuenta con las aplicaciones especializadas y licencias necesarias que facilitarán la codificación del prototipo. A continuación, se presenta la tabla que detalla los recursos de *software* que se utilizarán.

Tabla 2.

Recursos de Software

Recurso	Detalle
Sistema Operativo	Windows 11 Home
Motor de Base de Datos	Microsoft® SQL Server® 2017 Express
Entorno de desarrollo	Visual Studio Community 2022

Fuente: Elaboración propia (2024).

Considerando las características tecnológicas de ASP.NET Core MVC, el lenguaje de programación C#, el uso de Microsoft SQL Server Express como motor de base de datos, el IDE Visual Studio 2022 Community y el hospedaje proporcionado por SmarterASP.NET, se dispone de una combinación óptima de herramientas y tecnologías que garantizan la viabilidad del desarrollo e implementación del prototipo funcional destinado a la gestión de ventas de la empresa Zapatitos Lindos S.A.

Viabilidad Operativa

La viabilidad operativa del proyecto se basa en la probabilidad de que el prototipo funcional para la gestión de ventas sea utilizado de manera efectiva después de su desarrollo, y en la capacidad de los usuarios para adaptarse y utilizar el nuevo sistema sin dificultades significativas, por lo que a continuación, se presentan los aspectos clave que respaldan la viabilidad operativa de la investigación:

- **Análisis de requerimientos:** durante la etapa inicial del proyecto, se llevará a cabo un análisis exhaustivo de los requerimientos técnicos y funcionales. Esta fase permitirá comprender las necesidades específicas de los usuarios y diseñar un prototipo que satisfaga sus expectativas. Al involucrar a los interesados y captar sus opiniones e ideas, se garantiza una mayor adopción y aceptación del sistema.
- **Diseño centrado en el usuario:** durante el proceso de diseño del prototipo, se seguirá una metodología centrada en el usuario para garantizar que la interfaz y la funcionalidad del sistema sean intuitivas y fáciles de usar.
- **Pruebas exhaustivas:** antes de la implementación del prototipo, se realizarán pruebas exhaustivas y sistemáticas para evaluar y garantizar su correcto funcionamiento. Esto incluirá pruebas de funcionalidad, rendimiento y seguridad para detectar y corregir cualquier problema antes de que los usuarios finales interactúen con el sistema. Las pruebas rigurosas contribuirán a la confiabilidad y eficacia del sistema en el entorno operativo de la empresa.
- **Optimización de tareas:** el prototipo funcional no provocará reducción de personal ni cambios drásticos en la forma como se realizan las tareas. El objetivo de este proyecto es optimizar y mejorar los procesos existentes. Esto implica que los empleados podrán utilizar el sistema de manera efectiva, lo cual conducirá a una mayor productividad y a un uso más optimizado de los recursos de la empresa. Esta eficiencia operativa se traducirá en beneficios tangibles para Zapatitos Lindos S.A., lo cual permite una mejor gestión de sus procesos y un aprovechamiento eficaz de los recursos humanos y materiales.

- Personal calificado y entrenamiento: el personal de la empresa cuenta con la calificación necesaria para utilizar el prototipo. No se requerirá un entrenamiento, debido a la colaboración y la participación activa del usuario experto; este colaborador no solo proporcionará los requerimientos funcionales y de diseño necesarios para el prototipo, sino que también realizará pruebas y desempeñará un papel clave en la transmisión del conocimiento a los demás colaboradores.
- Enfoque colaborativo: se promoverá un enfoque colaborativo donde se considerarán las sugerencias de mejora para optimizar el sistema.

Estos factores aseguran que el prototipo funcional para la gestión de ventas sea efectivamente utilizado tras su desarrollo, facilitando a los usuarios superar las posibles dificultades en la adopción del nuevo sistema. Por lo tanto, desde una perspectiva operativa, la elaboración del prototipo es viable.

Viabilidad Económica

La empresa Zapatitos Lindos S.A. se beneficiará de no incurrir en gastos relacionados con *hardware* o *software*, ya que para el desarrollo de este proyecto se utilizarán equipos y herramientas propias, así como soluciones de código abierto que no requieren licencias. Sin embargo, es importante mencionar que la mano de obra sí conlleva un costo, el cual no será asumido por la empresa, ya que es un proyecto universitario para optar por el grado de licenciatura en Ingeniería en Informática con énfasis en Gerencia; por lo que los costos asociados a los recursos humanos no se transferirán a Zapatitos Lindos.

A continuación, se muestra una tabla con los costos del Recurso Humano que se necesitará para la elaboración del prototipo para la gestión de ventas de la empresa Zapatitos Lindos S.A. donde el precio por día corresponde a ¢29,460.00 para un bachiller universitario; esto según el Ministerio de Trabajo y Seguro Social sobre la lista de salarios mínimos por ocupación del periodo del 2024. Por lo que al tipo de cambio de venta de dólar del día 23 de marzo del 2024, 505.68 colones, el precio por hora del trabajado es de \$7.28.

Tabla 3.*Costo del recurso humano*

Actividad	Tiempo en horas	Total
Análisis de requerimientos	40	\$291.2
Diseño del prototipo	100	\$728
Desarrollo del prototipo	542	\$3945.76
Pruebas del prototipo	24	174.72
Total	706	\$5139.68

Fuente: Elaboración propia (2024).

En cuanto a la implementación del prototipo funcional, es necesario contratar un plan de *hosting* que sea compatible con las tecnologías y herramientas utilizadas. Es importante destacar que, durante la etapa de desarrollo, los costos del hosting serán responsabilidad del estudiante. Sin embargo, una vez que el proyecto sea aceptado y el prototipo esté en funcionamiento, estos serán asumidos por la empresa Zapatitos Lindos.

A continuación, se detallan los costos de alojamiento del proveedor sugerido SmarterASP.net con actualización de precios al 23 de marzo de 2024.

Tabla 4.*Costos de alojamiento*

Descripción	Costo Mensual	Costo Anual
Plan .NET PREMIUM	\$7.95	\$95.4
IP Estática	\$2	\$24
Nombre de Dominio	\$1	\$12
Privacidad de Dominio	\$0.66	\$8
Totales	\$11.61	\$139.4

Fuente: Elaboración propia (2024).

En lo que respecta a los costos asociados al alojamiento, es esencial destacar que la empresa está dispuesta a asumirlos. Esta disposición evidencia el compromiso de la organización con la eficiencia operativa y el éxito del proyecto, asegurando así una transición fluida y exitosa hacia el

nuevo sistema, una vez entre en producción.

Viabilidad Legal

La realización del prototipo funcional para la gestión de ventas de la empresa Zapatitos Lindos S.A. en ámbitos legales cumplirá con la ley vigente en Costa Rica, apegándose a las siguientes leyes:

- Ley 8148 - Adición de los artículos 196 BIS, 217 BIS y 229 BIS al Código Penal: La Ley 8148 es una adición al Código Penal de Costa Rica que introduce nuevos artículos relacionados con delitos y explotación sexuales. Estos artículos (196 BIS, 217 BIS y 229 BIS) abordan aspectos como el abuso sexual, la explotación sexual comercial y el turismo sexual. La ley establece sanciones penales para quienes cometan estos delitos y busca fortalecer la protección de los derechos sexuales y la integridad personal de las personas.
- Ley N° 4573 - Delitos informáticos: La Ley N° 4573 es una legislación diseñada para reprimir y sancionar los delitos informáticos en Costa Rica. Esta ley establece disposiciones legales relacionadas con la protección de los sistemas informáticos, la confidencialidad de los datos, el acceso no autorizado, el sabotaje informático, el fraude electrónico, entre otros delitos cibernéticos. El objetivo principal de esta ley es prevenir y combatir las actividades delictivas en el ámbito digital.
- Ley de Derechos de Autor 6683: Esta ley establece los derechos y protecciones legales para los autores de obras literarias, artísticas y científicas, así como para los titulares de derechos conexos, como los artistas intérpretes o ejecutantes y los productores de fonogramas. La ley regula aspectos relacionados con la protección de los derechos de autor, la duración de los derechos, las excepciones y limitaciones y la gestión colectiva de los derechos de autor.
- Ley 8968 - Protección de datos personales: La Ley 8968 es una legislación que tiene como objetivo proteger los datos personales de las personas en Costa Rica. Esta ley establece los principios y requisitos para el tratamiento de datos personales por parte de entidades públicas y privadas. La ley establece que las personas tienen derecho a conocer, actualizar y rectificar sus datos personales, así como a solicitar su eliminación en ciertas circunstancias. Además, se establecen obligaciones para las organizaciones en cuanto a la seguridad y confidencialidad de los datos personales que maneja.

Proyecciones

Se tiene como objetivo presentar el desarrollo de un prototipo funcional que permita a Zapatitos Lindos S.A. mejorar su gestión de ventas, superar la problemática actual y buscar eficiencia y eficacia en sus procesos. El enfoque principal será entregar un producto de calidad que brinde beneficios significativos a la empresa.

El prototipo consistirá en un *software* completo y funcional que abordará los desafíos identificados en la gestión de ventas. Uno de los aspectos clave del prototipo será asegurar que el sistema sea cien por ciento funcional y cumpla con la legislación tributaria de Costa Rica. Para lograr esto, el prototipo será capaz de emitir y validar facturas electrónicas, asegurando el cumplimiento legal y simplificando el proceso administrativo.

Además, el prototipo ofrecerá otros beneficios a la empresa. Por un lado, mejorará la calidad del producto entregado al implementar un seguimiento adecuado de los pedidos de los clientes. Esto permitirá asegurar una gestión eficiente de los procesos y garantizar una entrega oportuna y precisa de los productos, lo que a su vez fortalecerá la satisfacción del cliente y la reputación de la empresa. También, se incluirá un módulo de gestión de inventario que permitirá llevar un control preciso de los productos disponibles. Esto evitará faltantes y optimizará los niveles de *stock*, lo que contribuirá a una mayor eficiencia en la gestión de inventario y reducirá los costos asociados a la falta de control.

Otro beneficio importante es que el *software* permitirá la automatización de tareas repetitivas, ahorrando tiempo y recursos, mejorará la eficiencia operativa y permitirá a los empleados enfocarse en actividades de mayor valor añadido.

Alcance Funcional

El prototipo funcional en desarrollo está compuesto por los siguientes módulos:

- Generar facturación: este módulo se encargará de generar la facturación de contado y crédito contemplando la integración con los inventarios por lo que rebajará cantidades y alimentará un Kardex para tener un mayor control y confianza en las transacciones; en caso de no completar el proceso de facturación la información se podrá guardar para continuar posteriormente.

Además, contará con integración al módulo de Cuentas por Cobrar, cuando la venta sea crédito y se integrará con el módulo de órdenes de trabajo para optimizar el tiempo y que no se tenga que volver a digitar toda la información.

Este módulo emitirá facturas y tiquetes electrónicos por lo que se validará que las facturas cumplan con la regulación fiscal esto incluye la generación de documentos XML, la firma electrónica, la correcta aplicación de impuestos y cálculos adicionales tales como: descuentos totales o por línea y registro de las exoneraciones cuando sean requeridas. Una vez que el documento XML de la factura se encuentre firmado y validado se procederá con él envío de este al Ministerio de Hacienda para su recepción y aceptación; una vez que se obtenga la respuesta por parte de hacienda se generará un correo electrónico que adjuntará los XML del documento y aceptación, así como la representación gráfica en formato PDF.

- Anular facturación: este módulo se encargará de generar notas de crédito electrónicas asociadas a las facturas de venta. El concepto de notas de crédito sustituye la anulación ya que para efectos de la regulación fiscal los documentos electrónicos no se pueden anular.

El módulo permitirá crear notas de crédito completas o parciales asociadas a la factura o tiquete electrónico correspondiente. En el proceso se contempla la integración con los inventarios y el registro del Kardex recalculando los costos y utilizando el método del costo promedio ponderado y la reversión de la venta.

Cuando la condición de venta de la factura o tiquete electrónico que se asocia a la nota de crédito es crédito el módulo tendrá integración con las cuentas por cobrar y rebajará el saldo del cliente. Al permitir crear las notas de crédito parciales no es necesario que todos los artículos regresen al inventario y tampoco se tiene que volver a emitir una nueva factura, lo que solventaría el problema actual.

El módulo contempla la creación del documento XML de la nota de crédito, la firma electrónica, el cálculo de totales, impuestos, descuentos, la aplicación de exoneraciones, envío y recepción al Ministerio de Hacienda y a los clientes.

- Gestionar las cuentas por cobrar: este módulo se encargará de llevar el control de los saldos de los clientes, controlar los pagos entrantes y asegurar los ingresos de la empresa verificando y registrando recibos, notas de crédito y débito de tipo financieras y electrónicas.

La creación de documentos incluye el cálculo de impuestos, descuentos, exoneraciones y totales. Durante la creación de documentos se habilitará la opción para que se liquiden automáticamente los saldos de los clientes y se actualice el límite de crédito de estos.

Adicionalmente el módulo incluye una opción para el registro de abonos el cual emitirá un recibo. El módulo contará con un *Dashboard* que le permitirá a la empresa conocer cuánto

dinero representan las cuentas por cobrar en la compañía, cuánto y cuándo se les debe cobrar a los clientes tomando en cuenta la fecha de vencimiento de la factura, la antigüedad de los saldos y los saldos próximos a vencer, así como un historial de pagos.

- Gestionar las cuentas por pagar: este módulo se encargará de llevar el control de las cuentas por pagar a los proveedores de zapatos y de materias primas de la empresa, el módulo contempla integración con el proceso de compra por lo que el registro de factura de compra se realizará automáticamente cuando la condición de la compra sea de crédito, también se permitirá el registro de facturas que no tengan relación directa con el inventario o materias primas, el registro de pagos y notas de crédito y débito financieras y electrónicas.

El módulo contará con *Dashboard* que mostrará información clave como proyecciones de flujo de efectivo, proveedores claves con el monto total pendiente de pago y la fecha de vencimiento, así como información de incumplimiento.

- Generar órdenes de trabajo: este módulo se encargará de llevar el control de las órdenes de trabajo pendientes, en proceso de producción, finalizadas y facturadas; el registro de órdenes de trabajo tendrá integración con el módulo de inventarios para comprometer los productos y poder rebajar y llevar un control de los insumos que se utilizan para la modificación o adaptación de los diferentes tipos de calzado.

Las órdenes de trabajo finalizadas serán visibles en el módulo de facturación para que no exista la necesidad de volver a registrar toda la información que se tiene desde que se inicia el proceso.

El módulo contará con una cola de trabajo que se puede priorizar para efectos de producción y se podrá conocer el estado de las ordenes de trabajo en cualquier momento.

Al llevar el control de insumos y poder priorizar las órdenes de trabajo, así poder llevar un control histórico permitirá ser más eficientes y cumplir con las necesidades de los clientes, así poder prever retrasos en la producción.

- Controlar ingresos y gastos: este módulo se encarga de llevar el control de los ingresos y gastos de la empresa integrando la información de los módulos del sistema informático propuesto.

Para cumplir con el requerimiento necesario de la empresa de manera directa y consecutiva el módulo permitirá el registro de asientos contables y manejo del catálogo contable permitiendo la comprobación general y detallada de los gastos e ingresos, también incluye registro automático de asientos diarios provenientes del inventario, ventas, cuentas por

cobrar y cuentas pagar. El registro diario y automático de asientos permitirá a la empresa tener un control de gastos e ingresos, le ahorrará tiempo a la empresa al no tener que hacerlo manualmente y ayudará al negocio en la toma de decisiones.

Alcance Metodológico

El alcance metodológico de este proyecto se basará en la metodología ágil Scrum para el desarrollo de *software*. Scrum es un marco de trabajo ágil que se utiliza para gestionar proyectos de desarrollo de *software*. Dicha metodología se centra en la flexibilidad, la adaptación al cambio y la entrega continua de incrementos.

Los principales elementos de Scrum incluyen roles, artefactos y eventos. Respecto de los roles, Scrum define tres principales, a saber:

- **Scrum Master:** es el facilitador del proceso Scrum. Su responsabilidad principal es asegurarse de que se sigan las prácticas y principios de Scrum, y eliminar cualquier impedimento que pueda afectar al equipo.
- **Product Owner:** es el representante del cliente y define los requisitos del producto. Su función principal es gestionar el *backlog* del producto y priorizar las funcionalidades.
- **Equipo de Desarrollo:** es el equipo encargado de desarrollar el producto. Trabaja en colaboración para entregar los incrementos de *software*.

Los principales artefactos en Scrum son:

- **Product Backlog:** es una lista ordenada de requisitos o funcionalidades del producto. Se prioriza y actualiza continuamente.
- **Sprint Backlog:** Es una lista de elementos seleccionados del Product Backlog que se deben completar durante un sprint.
- **Incremento:** es el resultado tangible y potencialmente entregable del trabajo realizado durante un sprint.

Finalmente, en relación a los eventos principales en Scrum se encuentran los siguientes:

- **Planificación del Sprint:** reunión donde se seleccionan los elementos del Product Backlog para incluir en el Sprint Backlog.
- **Reunión Diaria:** breve reunión diaria en la que el equipo de desarrollo se sincroniza sobre el progreso y los próximos pasos.

- **Revisión del Sprint:** reunión al final de cada sprint para revisar y demostrar el incremento desarrollado.
- **Retrospectiva del Sprint:** reunión para evaluar el sprint anterior y buscar mejoras en el proceso.

En el ciclo de vida del *software*, Scrum se puede aplicar de la siguiente manera:

- **Planificación del proyecto:** se define el alcance del proyecto y se crea el *Product Backlog*, que contiene todos los requisitos y funcionalidades del producto.
- **Planificación del Sprint:** se seleccionan los elementos del *Product Backlog* para incluir en el *Sprint Backlog*, que representan el trabajo por realizar durante el sprint.
- **Desarrollo del Sprint:** el equipo de desarrollo trabaja en la implementación de los elementos del *Sprint Backlog*. Se llevan a cabo reuniones diarias para sincronizar el progreso y abordar cualquier impedimento.
- **Revisión del Sprint:** al final del sprint, se realiza una reunión para revisar y demostrar el incremento desarrollado al Product Owner y a los stakeholders. Se recopilan comentarios y se realizan ajustes según sea necesario.
- **Retrospectiva del Sprint:** se lleva a cabo una reunión para evaluar el sprint anterior y buscar oportunidades de mejora en el proceso. Se identifican acciones para implementar en futuros sprints.

Este ciclo de vida se repite en cada sprint, lo que permite la entrega continua de incrementos de *software* y adaptación a medida que se obtienen nuevos conocimientos y se incorporan cambios.

Alcance Tecnológico

El prototipo funcional para la gestión de ventas de la empresa Zapatitos Lindos S.A., ubicada en San José, es un sistema web y se utilizarán las siguientes herramientas:

Tabla 5.

Herramientas de desarrollo

Lenguajes de desarrollo.	C#, JavaScript, SQL
Entorno de desarrollo integrado (IDE)	Visual Studio Community 2022, Visual Studio Code
Motor de base de datos	Microsoft® SQL Server® 2017 Express SQL Server Management Studio 19.0.2
Tipo proyecto	ASP.Net Core 6 MVC
Otros	Bootstrap, Html, CSS, Json, XML, Ajax, jQuery, jQuery UI

Fuente: Elaboración propia (2024).

El alcance tecnológico del prototipo web abarca un conjunto diverso de herramientas, incluyendo diversos lenguajes de desarrollo como C#, JavaScript y SQL. Estas se integran en un entorno de desarrollo integrado (IDE), compuesto por Visual Studio Community 2022 y Visual Studio Code. Además, se emplea el motor de base de datos Microsoft® SQL Server® 2017 Express, respaldado por SQL Server Management Studio 19.0.2. El proyecto se enmarca dentro del tipo ASP.Net Core 6 MVC. Complementariamente, se utilizan tecnologías como Bootstrap, HTML, CSS, Json, XML, Ajax y jQuery UI. Este conjunto de herramientas se combina estratégicamente para generar una solución tecnológica que se ajuste a las necesidades de la empresa.

CAPÍTULO II: MARCO REFERENCIAL

El presente marco referencial tiene como objetivo proporcionar una base teórica sólida y conceptualizaciones válidas para encuadrar el desarrollo del prototipo funcional destinado a la gestión de ventas de la empresa Zapatitos Lindos S.A.

Se realizará un análisis detallado de las teorías y conceptos relevantes que permitirán comprender en profundidad la estructura y los problemas que se pretenden solucionar mediante desarrollo de este proyecto. Además, se establecerán las implicaciones del estudio y su relación con otras áreas del conocimiento, para brindar, así, una visión integral.

Gestión De Ventas

La gestión de ventas es un proceso fundamental para cualquier empresa, ya que implica el manejo eficiente de las transacciones comerciales y la generación de facturas precisas y confiables.

Según da Silva, (2023) “La gestión de ventas en una empresa es un conjunto de prácticas, técnicas y estrategias estandarizadas cuyo objetivo es ayudar a los vendedores a cerrar más tratos.” (párr.7); por lo que realizar una gestión de ventas eficiente es fundamental, imprescindible e ineludible para el éxito sostenible de toda empresa, dado que no solo contribuye a maximizar los ingresos y garantizar la rentabilidad, sino también a cultivar y fortalecer una relación sólida, duradera y satisfactoria con los clientes; esto genera confianza y fidelidad a largo plazo.

En el contexto de Zapatitos Lindos S.A., se identificó una serie de problemas significativos y desafiantes, principalmente en su proceso de facturación, control de inventarios, gestión de saldos por cobrar y saldos por pagar, generación de órdenes de trabajo y el control de ingresos y gastos contables. Estos problemas requieren una atención inmediata y abarcan una amplia gama de áreas críticas que afectan directamente a la organización. Entre los desafíos más destacados se encuentran la carga administrativa excesiva, lo cual implica una inversión considerable de tiempo y recursos valiosos.

Además, las inconsistencias en el sistema actual representan una fuente constante de pérdida de tiempo y energía, lo cual obstaculiza el funcionamiento fluido y eficiente del proceso de ventas. Uno de los problemas de mayor importancia es el de los cálculos incorrectos de impuestos por parte del sistema a la hora de generar facturas de venta que incluyen servicios, como la mano de obra cuya tarifa de impuesto es del trece por ciento; para corregir esta problemática los usuarios deben realizar varias acciones en el sistema, por ejemplo, eliminar y volver agregar la

línea que produce la inconsistencias; dicha situación genera más trabajo al personal de ventas. Es importante mencionar que la mayoría de las ventas que realiza la empresa son exentas; de acuerdo con la legislación vigente y el Sistema Costarricense de Información Jurídica:

Está exenta la venta local o la importación de los siguientes bienes y la prestación de servicios destinados a mejorar la funcionalidad y garantizar la autonomía de las personas con discapacidad, conforme lo dispuesto en el artículo 2 de la Ley No. 7600 del 02 de mayo de 1996, denominada "Igualdad de Oportunidades para las Personas con Discapacidad". (Reglamento de la Ley del Impuesto sobre el Valor Agregado N° 41779, Artículo 11, Exenciones)

Asimismo, los rebajos incorrectos de inventarios también se han convertido en una preocupación constante, estos afectan directamente la precisión de los datos y generan desafíos adicionales para la gestión adecuada de los recursos.

Las desconexiones frecuentes en el sistema plantean un problema adicional, pues interrumpen la continuidad y el flujo de trabajo, lo que resulta en pérdida de productividad. Estas situaciones, en conjunto, generan un ambiente de desconfianza tanto interna como externamente, afectan la reputación de la empresa y debilitan las relaciones con los clientes; de igual manera, aumentan la carga de trabajo, por lo que disminuye la eficiencia operativa y limitan la capacidad de la empresa para cumplir con los objetivos establecidos.

Por lo anterior, es crucial abordar de manera integral y prioritaria estos problemas en el proceso de facturación de la empresa Zapaticos Lindos S.A.; así, se logrará restablecer la confianza, mejorar la productividad, garantizar la rentabilidad y aumentar la satisfacción tanto de los clientes como del equipo interno.

Prototipo Funcional

Este concepto desempeña un papel fundamental en el proceso de elaboración de un sistema para la gestión de ventas. Al desarrollar un prototipo funcional, se busca estandarizar los procesos de ventas, mejorar la gestión de clientes, optimizar el control de inventarios, gestionar eficientemente las cuentas por pagar y cobrar, y realizar un seguimiento preciso del control de ingresos y gastos de la empresa.

El prototipo permite visualizar y evaluar de manera práctica la solución propuesta, identificando posibles fallos y ajustes necesarios antes de una implementación definitiva; lo cual fomenta una mayor eficacia operativa y asegura que el sistema final esté completo y acorde a las

necesidades y objetivos del negocio. Según Medina Cruz *et al.* (2019), “los prototipos son medios de comunicación entre analistas, clientes o usuarios, que muestran las decisiones tomadas, con el fin de validarlas y permitir la resolución de los problemas de comprensión presentes en la etapa de levantamiento de requisitos” (p.331).

Adicionalmente, la representación temprana y tangible de un sistema o producto en desarrollo, permite evaluar su viabilidad y funcionalidad y desempeña un papel esencial en el proceso de desarrollo de *software*, ya que permite a los desarrolladores y a los interesados tener una idea clara de cómo se desarrollará el sistema en su totalidad; esto proporciona una plataforma para la identificación de problemáticas, realización de pruebas y la obtención de retroalimentación temprana de los usuarios y partes interesadas.

Su utilización también se justifica debido a que se pueden identificar y abordar de manera proactiva los posibles errores o deficiencias del sistema, con lo cual se evita costosos retrasos y revisiones en etapas posteriores del desarrollo. Como menciona Martín (2020) “Los prototipos son de gran utilidad para los equipos de desarrollo ya que permiten comprender de manera visual el proyecto y su magnitud, ayudando a explorar otras opciones y permitiendo anteponerse a problemas en el futuro”. (párr.3).

Esto resalta la importancia de contar con un prototipo funcional como herramienta fundamental en el proceso de desarrollo, pues no solo mejora la comunicación, sino que también permite una interacción más ágil y eficiente, con lo cual se asegura la entrega de un producto final de calidad.

Software Web

El prototipo funcional por desarrollar consiste en un *software web*; que de acuerdo con Palomo, et al. (2020)

El software es todo lo que no es hardware. El Software incluye, por supuesto, los programas que gobiernan el funcionamiento del sistema, pero también incluye otros elementos tales como documentos, bases de datos o algo tan inmaterial como son los procedimientos de operación o de mantenimiento periódico. (p.20)

Por lo que el *software* juega un papel crítico en prácticamente todos los aspectos de la vida moderna. Está presente en dispositivos móviles, computadoras personales, sistemas de entretenimiento, sistemas de transporte, infraestructura empresarial y muchas otras áreas; facilita la automatización de tareas, mejora la eficiencia, aumenta la productividad y proporciona

soluciones tecnológicas para diversos problemas y necesidades.

El *software*, en su sentido más amplio, abarca todo tipo de programas informáticos, desde aplicaciones de productividad y entretenimiento hasta sistemas operativos y controladores de dispositivos en donde su objetivo principal es proporcionar funcionalidad y resolver problemas específicos como los que presenta la empresa Zapatitos Lindos S.A.

La transición del *software* hacia los sistemas web ha ampliado nuestras posibilidades, permitiéndonos acceder a aplicaciones y servicios sin restricciones geográficas o limitaciones en términos de *hardware*. Este avance ha facilitado enormemente nuestra experiencia digital, eliminando la necesidad de instalaciones complicadas y brindando la flexibilidad de utilizar herramientas poderosas en cualquier momento y lugar. En simples palabras, TechTarget (2023) “una aplicación web es un programa de aplicación que se almacena en un servidor remoto y se entrega a través de la interfaz de navegador.” (párr.1).

En el contexto del desarrollo del prototipo funcional para la gestión de ventas de la empresa Zapatitos Lindos S.A., es esencial fusionar los conceptos de prototipo, *software* y web. Al combinar las funcionalidades de un prototipo con las capacidades de un *software* web conlleva múltiples beneficios; en comparación con el *software* que actualmente la empresa posee y que requiere una compleja instalación en cada equipo en el cual desean utilizarlo y que se limita únicamente al sistema operativo Windows. Al optar por un sistema web, la empresa puede disfrutar de una mayor accesibilidad y flexibilidad, ya que el *software* podrá utilizarse desde cualquier dispositivo con conexión a Internet, incluyendo tabletas, dispositivos móviles y computadoras con diversos sistemas operativos, sin restricciones.

Esto aumenta significativamente la portabilidad y la movilidad para los empleados, al permitirles acceder a la plataforma desde cualquier lugar y en cualquier momento. Además, la administración y actualización del sistema se vuelven más sencillas, ya que no se requiere la instalación manual en cada dispositivo, sino que se accede directamente a través del navegador web. También, se reducen los costos asociados con el mantenimiento de múltiples instalaciones, ya que todo se centraliza en un servidor web.

Asimismo, el sistema web facilita la colaboración en tiempo real y el intercambio de información entre los diferentes usuarios de las dos sucursales con las que la empresa cuenta; mejorando la eficiencia y la comunicación interna.

Sistema Tecnológico De Información

De acuerdo con Beynon-Davies (2018)

Un sistema tecnológico de información es un sistema técnico. Con frecuencia calificamos este tipo de sistemas de ejemplo de sistemas “hard” en el sentido de que existen físicamente. Un sistema tecnológico de información es una colección organizada de hardware, software, y tecnología de las comunicaciones y los datos diseñada para sustentar ciertos aspectos de un determinado sistema de información. Las entradas de un sistema tecnológico de información son datos, realiza alguna manipulación sobre ellos como parte de un proceso y devuelve los datos manipulados para que sean interpretados dentro de algún sistema de actividad humana. (p.6)

Las entradas de información consisten en la recolección de datos sobre clientes, productos, transacciones y cualquier otro elemento relevante para el proceso de ventas. Una vez que se ingresa a la información, se almacena en una base de datos o sistema de almacenamiento adecuado. El almacenamiento de información garantiza la disponibilidad y preservación de los datos a largo plazo, lo cual permite su posterior recuperación y consulta. Esto facilita la gestión y el análisis de la información para tomar decisiones estratégicas en el ámbito de las ventas.

El procesamiento de información es otro componente esencial en el marco del sistema informático. Consiste en la manipulación y transformación de los datos ingresados para generar resultados útiles y relevantes. Esto implica realizar cálculos, aplicar reglas de negocio y llevar a cabo operaciones lógicas que permitan obtener información procesada y significativa. El procesamiento de información se realiza mediante algoritmos y programas específicos diseñados para el contexto de la gestión de ventas.

Una vez que se ha realizado el procesamiento de los datos, el sistema debe proporcionar salidas de información que permiten visualizar y comunicar los resultados del procesamiento de datos. Estas salidas pueden ser informes, gráficos, tablas o cualquier otro formato que facilite la comprensión y la toma de decisiones. Las salidas de información son de vital importancia, ya que brindan a los usuarios y tomadores de decisiones una visión clara y concisa de los indicadores clave de desempeño y resultados del proceso de ventas.

Es importante destacar que el uso de un sistema informático de información en la gestión de ventas ofrece numerosos beneficios, como la automatización de tareas, la mejora de la precisión

y la agilidad en la toma de decisiones. Además, permite una mayor eficiencia operativa y la optimización de los recursos disponibles.

Requerimientos Funcionales y No Funcionales

Complementando la información anterior para el desarrollo de un prototipo funcional de *software*, es esencial tener en cuenta tanto los requerimientos funcionales como los no funcionales. Al hablar de los requerimientos funcionales, es primordial enfatizar su papel crucial en asegurar el éxito del prototipo, así como la plena satisfacción de los usuarios. Estos requisitos definen las funcionalidades y características específicas que el prototipo debe tener para así cumplir con las necesidades y expectativas de los usuarios. Según PMOinformatica (2018) “Los requerimientos funcionales son las descripciones explícitas del comportamiento que debe tener una solución de *software* y que información debe manejar.” (párr.6).

Los requerimientos funcionales especifican las acciones y capacidades que el sistema debe tener, como por ejemplo el registro de compras y ventas, el procesamiento de pagos y la generación de informes. Estos requerimientos definen la funcionalidad principal del *software* y aseguran que cumpla con las necesidades y expectativas de la empresa y sus clientes. Sin una clara comprensión y documentación de estos requerimientos, el prototipo para la gestión de ventas de la empresa Zapatitos Lindos S.A. puede carecer de las características esenciales que se esperan de él.

Por otro lado, como se menciona en Ken (2023)

Los requisitos no funcionales, también conocidos como "requisitos de calidad" o "requisitos de atributos del sistema", son características y criterios que describen cómo debe ser el rendimiento, la seguridad, la usabilidad y otros aspectos de un sistema o *software* más allá de su funcionalidad básica. (párr.2).

Un prototipo que cumple con los requerimientos no funcionales brinda una experiencia de usuario mejorada, asegura la confidencialidad de los datos y proporciona un rendimiento óptimo en términos de tiempo de respuesta y capacidad de manejo de carga.

La adecuada identificación y consideración de estos requerimientos, tanto funcionales como no funcionales, permiten una planificación precisa y una implementación efectiva del prototipo; adicionalmente contribuyen a evitar problemas futuros, como la falta de funcionalidades clave, problemas de rendimiento o incompatibilidades de seguridad.

En última instancia, los requerimientos de *software* desempeñan un papel crucial en la calidad y el éxito del prototipo, ya que establecen una base sólida para su desarrollo y entrega final.

Diagramas UML

En relación con el tratamiento de los conceptos esenciales en el desarrollo de software, es relevante mencionar la creación de diagramas, los cuales desempeñan una función fundamental en la comprensión y transmisión de los requisitos, así como en la construcción de un prototipo operativo.

La utilización del UML (Lenguaje Unificado de Modelado) es de vital importancia para representar y visualizar los diferentes aspectos de un sistema de manera gráfica. El UML proporciona un conjunto de notaciones y herramientas que permiten crear diagramas que describen desde la estructura hasta el comportamiento del software. De acuerdo con el sitio Lucidchart.com (s.f.)

El Lenguaje Unificado de Modelado (UML) fue creado para forjar un lenguaje de modelado visual común y semántica y sintácticamente rico para la arquitectura, el diseño y la implementación de sistemas de software complejos, tanto en estructura como en comportamiento. (párr.2.)

A continuación, se presentan algunos de los diagramas más utilizados y su relevancia en este contexto.

Diagrama De Casos De Uso

El diagrama de casos de uso describe las interacciones entre los actores, usuarios o sistemas externos y el prototipo en desarrollo. Es útil para identificar y comprender los requerimientos funcionales del *software*, ya que muestra las principales funciones o acciones que el sistema debe realizar desde la perspectiva del usuario. Según IONOS (2020),

El diagrama de casos de uso es una forma de diagrama de comportamiento en lenguaje de modelado unificado (UML, del inglés Unified Modelling Language), con la que se representan procesos empresariales, así como sistemas y procesos de programación orientada a objetos. Por lo tanto, UML no es un lenguaje de programación, sino un lenguaje de modelado, es decir, un método estandarizado para representar sistemas planificados o ya existentes. En este diagrama, todos los objetos involucrados se estructuran y se relacionan entre sí. (párr.1)

Diagrama De Secuencia

El diagrama de secuencia, también parte del UML, representa la interacción entre los objetos del sistema a lo largo del tiempo. Este diagrama es útil para comprender cómo se

comunican los distintos componentes del sistema y cómo se desarrollan las secuencias de acciones. Ayuda a identificar los flujos de trabajo y las interacciones clave, lo que contribuye a una mejor comprensión de los requerimientos del *software* y a la planificación del desarrollo del prototipo.

Como lo describen Ramos & Segura (2022) “El diagrama de secuencia nos indica cómo se transfiere la información entre cada uno de los elementos que intervienen en la ejecución de un programa”. (párr.1)

Diagrama De Contexto

El diagrama de contexto representa la relación del sistema en desarrollo con su entorno externo. Este diagrama se centra en mostrar las interacciones entre el sistema y otros sistemas, usuarios u otros factores ambientales relevantes, según Cuofano (2023)

Un diagrama de contexto es un modelo que ilustra la interacción entre un producto y personas, organizaciones o sistemas externos. Los analistas de negocios utilizan el diagrama de contexto para comprender el contexto y los límites de los sistemas dentro de un proyecto. Los elementos estructurales de un diagrama de contexto comprenden el producto, las entidades externas y los flujos de datos. (párr.1)

Lo que permite visualizar los límites del sistema y cómo se relaciona con otros elementos externos, lo que ayuda a comprender los requerimientos del *software* en términos de interoperabilidad y comunicación con sistemas externos.

Diagrama De Actividad

El diagrama de actividad es utilizado para modelar el flujo de actividades y acciones dentro de un proceso o función. Este diagrama es especialmente útil para comprender las acciones y flujos de trabajo complejos, y permite identificar requerimientos específicos relacionados con la secuencia de actividades.

Diagrama De Clases

El diagrama de clases es utilizado para modelar la estructura de objetos y las relaciones entre ellos, mostrando las clases, atributos y métodos que componen el sistema, contribuyendo a una mejor comprensión de los requerimientos en términos de estructura y comportamiento del *software*.

Además del levantamiento de requerimientos y la creación de diagramas es imprescindible tener en cuenta la arquitectura de *software*. Esta arquitectura es fundamental para la estructuración

del sistema y su ausencia en etapas tempranas puede tener consecuencias negativas, limitando severamente la capacidad del producto final para satisfacer las necesidades de los clientes.

Arquitectura De Software

De acuerdo con Software Engineering Institute (SEI)

La arquitectura de software de un sistema representa las decisiones de diseño relacionadas con la estructura y el comportamiento general del sistema. La arquitectura ayuda a las partes interesadas a comprender y analizar cómo el sistema logrará cualidades esenciales como modificabilidad, disponibilidad y seguridad (parr.1)

La Arquitectura de Software es el esqueleto conceptual que define la estructura y organización del sistema. Se ocupa de los componentes principales, sus interacciones y la distribución de responsabilidades entre ellos. Al diseñar una arquitectura sólida desde el inicio, se establecen las bases para un desarrollo coherente y eficiente.

Una arquitectura bien definida permite identificar los principales módulos o componentes del sistema, así como las interfaces y la comunicación entre ellos. También facilita la integración con sistemas externos y la reutilización de componentes, lo que ahorra tiempo y esfuerzo en etapas posteriores.

Al no dar la debida importancia a la Arquitectura de Software desde el principio, se corre el riesgo de enfrentar problemas durante el desarrollo y la implementación del prototipo funcional. Esto puede resultar en un código inmanejable y propenso a errores, que afectará negativamente la calidad del producto final. Adicionalmente, una arquitectura inadecuada puede llevar a que un prototipo funcional no pueda evolucionar con los requerimientos cambiantes del cliente, lo cual genera insatisfacción y la necesidad de rehacer gran parte del desarrollo.

Existen diferentes tipos de arquitectura, cada uno con sus propias características y ventajas; para el desarrollo del prototipo funcional de Zapatitos Lindos S.A., se ha realizado una cuidadosa selección de los elementos arquitectónicos más adecuados.

En primer lugar, se ha elegido utilizar el patrón de capas. Este patrón divide el sistema en diferentes capas, cada una con una responsabilidad específica. Según menciona Huaman (2018) “Este patrón se puede utilizar para estructurar programas que se pueden descomponer en grupos de subtarear, cada una de las cuales se encuentra en un nivel particular de abstracción. Cada capa proporciona servicios a la siguiente capa superior” (párr.3).

Por lo general se definen las siguientes capas: la capa de presentación que se encarga de la interfaz de usuario, la capa de lógica de negocio que maneja la lógica y el procesamiento de datos, y la capa de acceso a datos se ocupa de interactuar con la base de datos.

El proyecto de prototipo funcional para la gestión de ventas de la empresa Zapatitos Lindos se estructura de la siguiente manera: la capa llamada BC contendrá todas las reglas de negocio y validaciones de datos, la capa llamada Abstracciones está compuesta por todos los modelos e interfaces utilizados por todo el sistema, en la capa llamada BW se incluirá toda la lógica y el procesamiento de datos; sumado a estas capas tendremos la capa DA o de acceso a datos la cual permitirá el acceso y la interacción con la base de datos, la capa llamada UI o interfaz de usuarios y, por último, se adicionará una división que se llamará utilitarios en la que se agregan funciones comunes que se podrán utilizar desde cualquiera de las capas anteriores; dentro de estos utilitarios se agrega el proyecto para firmar documentos electrónicos en formato XML con firma digital y otro tipo de funciones estáticas las cuales no requieren de una instanciación para poder ser utilizadas. Esta separación permitirá una mayor modularidad y facilita la comprensión y el mantenimiento del sistema.

Además, se ha optado por utilizar el patrón Modelo-Vista-Controlador (MVC), ampliamente utilizado en el desarrollo de aplicaciones web. Según Herrera (2020),

MVC es un patrón de diseño que se estructura mediante tres componentes: modelo, vista y controlador. Este patrón tiene como principio que cada uno de los componentes esté separado en diferentes objetos, esto significa que los componentes no se pueden combinar dentro de una misma clase. Sirve para clasificar la información, la lógica del sistema y la interfaz que se le presenta al usuario. (párr.2)

El modelo se encarga de representar los datos y la lógica de negocio, la vista se ocupa de la presentación de la información al usuario y el controlador coordina las interacciones entre el modelo y la vista, lo que proporciona una clara separación de responsabilidades y facilita la escalabilidad y el mantenimiento del prototipo funcional.

La elección de estos patrones arquitectónicos se fundamenta en su demostrada utilidad en el desarrollo de sistemas similares. La utilización de los patrones de capas y MVC ofrece una estructura organizada, modularidad, reutilización de código y flexibilidad para adaptarse a necesidades futuras. Además, estos patrones promueven una alta cohesión y un bajo acoplamiento

en el sistema. Su implementación proporcionará una estructura clara, modularidad y flexibilidad, lo que permite así el desarrollo de un sistema sólido y adaptable para la gestión de ventas, incluyendo el control de inventarios, la gestión de las cuentas por pagar y cuentas por cobrar, el control de ingresos y gastos, así como la administración de órdenes de trabajo.

Entorno De Desarrollo Integrado

En adición a la selección de los patrones arquitectónicos mencionados previamente, se ha optado por utilizar Visual Studio en su versión Community como entorno de desarrollo integrado (IDE). Según el sitio Amazon.com (s.f) un IDE es:

Es una aplicación de software que ayuda a los programadores a desarrollar código de software de manera eficiente. Aumenta la productividad de los desarrolladores al combinar capacidades como editar, crear, probar y empaquetar software en una aplicación fácil de usar. Así como los escritores utilizan editores de texto y los contables, hojas de cálculo, los desarrolladores de software utilizan IDE para facilitar su trabajo. (párr.1).

El uso de un IDE es de suma importancia en el proceso de desarrollo de *software* debido a los numerosos beneficios que brinda. Es una herramienta que proporciona un conjunto de características y funcionalidades que facilitan la escritura, la depuración y la gestión del código. Estas características incluyen un editor de código avanzado con resaltado de sintaxis, autocompletado, refactorización de código, navegación rápida y muchas otras utilidades.

Referente a la documentación disponible, en el sitio learn.microsoft.com (2023) se nos indica que:

Visual Studio es una herramienta de desarrollo eficaz que permite completar todo el ciclo de desarrollo en un solo lugar. Es un entorno de desarrollo integrado (IDE) completo que puede usar para escribir, editar, depurar y compilar el código y, luego, implementar la aplicación. Aparte de la edición y depuración del código, Visual Studio incluye compiladores, herramientas de finalización de código, control de código fuente, extensiones y muchas más características para mejorar cada fase del proceso de desarrollo de software. (párr.1)

Para el desarrollo del prototipo funcional se decidió utilizar Microsoft Visual Studio Community 2022 como entorno de desarrollo integrado (IDE). Esta versión gratuita de Visual Studio ha ganado reconocimiento y popularidad en la industria del desarrollo de *software* debido

a sus características y herramientas de alta calidad. Esta edición cuenta con soporte para múltiples lenguajes de programación, lo cual facilita a los desarrolladores trabajar en proyectos que empleen diferentes lenguajes de programación.

Lenguajes de programación

Según Mendoza (2020):

Un lenguaje de programación, en palabras simples, es el conjunto de instrucciones a través del cual los humanos interactúan con las computadoras. Un lenguaje de programación nos permite comunicarnos con las computadoras a través de algoritmos e instrucciones escritas en una sintaxis que la computadora entiende e interpreta en lenguaje de máquina. (párr.3)

Dentro del entorno de desarrollo integrado (IDE) Microsoft Visual Studio Community, también se puede trabajar con diversas herramientas y lenguajes que permiten desarrollar un prototipo funcional que cumpla con todos los requerimientos y necesidades del proyecto. Uno de los lenguajes principales que se utilizará es C#; según Bill Wagner (2023)

C# (pronunciado "si Sharp" en inglés) es un lenguaje de programación moderno, basado en objetos y con seguridad de tipos. C# permite a los desarrolladores crear muchos tipos de aplicaciones seguras y sólidas que se ejecutan en .NET. C# tiene sus raíces en la familia de lenguajes C, y a los programadores de C, C++, Java y JavaScript les resultará familiar inmediatamente. (párr.1)

Al emplear C# como lenguaje de programación, se podrá aprovechar su sintaxis de fácil comprensión y su enfoque en la programación orientada a objetos, lo que facilitará la creación de un código organizado y modular. Además, este lenguaje cuenta con una amplia gama de bibliotecas y *frameworks* que simplificarán la implementación de funcionalidades complejas y la integración con otras tecnologías. De acuerdo con Assembler Institute (2022) “un framework es un marco de trabajo que tiene como objetivo facilitar la solución de problemas que pueden surgir al programar” (párr.1)

Asp.NET Core

Asimismo, se utilizará ASP.NET Core; de acuerdo con Microsoft (2023) “ASP.NET Core es un marco multiplataforma de código abierto y de alto rendimiento que tiene como finalidad compilar aplicaciones modernas conectadas a Internet y habilitadas para la nube.” (párr.1).

Mediante el uso de ASP.NET Core, se obtendrán beneficios como la capacidad de

desarrollar aplicaciones web rápidas, escalables y de alto rendimiento. Este *framework* es multiplataforma, lo que permitirá el despliegue de las aplicaciones en diversos sistemas operativos. Asimismo, ASP.NET Core ofrece características avanzadas de seguridad para proteger las aplicaciones contra ataques y vulnerabilidades.

JavaScript

Para complementar el desarrollo del proyecto, se emplearán otras herramientas y lenguajes. Por ejemplo, se utilizará JavaScript para agregar interactividad y dinamismo a las aplicaciones.

Tal como se menciona en Amazon (s.f.):

JavaScript es un lenguaje de programación que los desarrolladores utilizan para hacer páginas web interactivas. Desde actualizar fuentes de redes sociales a mostrar animaciones y mapas interactivos, las funciones de JavaScript pueden mejorar la experiencia del usuario de un sitio web. Como lenguaje de scripting del lado del servidor, se trata de una de las principales tecnologías de la World Wide Web. (párr.1)

Con JavaScript, será posible crear efectos visuales, manipular elementos del interfaz de usuario en tiempo real y realizar peticiones al servidor sin necesidad de recargar la página.

Hojas De Estilo CSS

También se utilizará CSS (Cascading Style Sheets) para el diseño y la presentación visual de las aplicaciones. De acuerdo con el sitio NextU Latam (2022) “Cascading Style Sheets (CSS) que en español se traduce como “hojas de estilo en cascada”, es un lenguaje de programación que permite crear y realizar cambios en un documento HTML de forma rápida y sencilla”. (párr.2)

CSS permitirá definir el aspecto y formato de los elementos HTML, brindando un mayor control sobre la apariencia del prototipo funcional.

JQuery

Otra herramienta importante que se utilizará es jQuery que, de acuerdo con NextU Latam (2022), “es una biblioteca de JavaScript rápida, pequeña y rica en funciones que permite añadir interacciones en un sitio web.” (párr.2). Esta biblioteca ofrece una amplia gama de funciones y efectos predefinidos que agilizarán el desarrollo y mejorarán la interactividad con la aplicación.

ORM

Además, se empleará Entity Framework, un ORM que simplificará el acceso a la capa de datos. De acuerdo con campusMVP (2020):

Un ORM (Object Relational Mapping) es un tipo de biblioteca de acceso a datos que intenta hacer que esta tarea sea más natural para los desarrolladores. Así, a la hora de acceder a datos, en lugar de utilizar otro lenguaje (generalmente SQL), un ORM permite que puedas utilizar los paradigmas habituales de la programación orientada a objetos: clases y objetos. En lugar de pensar en tablas y relaciones, piensas en objetos y propiedades. (párr.1)

Entity Framework, dentro del alcance tecnológico del prototipo, desempeña un papel crucial al simplificar las operaciones de lectura y escritura en la base de datos. Esta herramienta proporciona una capa de abstracción sobre la persistencia de datos, lo que implica que los módulos de alto nivel del sistema pueden interactuar con entidades de objeto en lugar de lidiar directamente con consultas SQL. Esta abstracción permite que los desarrolladores se centren en la lógica de negocio y la funcionalidad del sistema, en lugar de preocuparse por los detalles de implementación específicos de la base de datos. En consecuencia, se fomenta la modularidad y la independencia de los componentes del sistema, ya que los módulos de alto nivel pueden interactuar con las abstracciones proporcionadas por Entity Framework sin estar directamente vinculados a la estructura y la sintaxis de la base de datos subyacente. Este enfoque facilita la escalabilidad y el mantenimiento del sistema a medida que evoluciona con el tiempo, ya que los cambios en la estructura de la base de datos pueden ser gestionados de manera más eficiente a través de las abstracciones proporcionadas por Entity Framework.

Continuando con Visual Studio Community 2022 uno de los aspectos destacados de es su potente herramienta de depuración. Se puede rastrear el código paso a paso, establecer puntos de interrupción y examinar variables y objetos en tiempo de ejecución lo cual agiliza la identificación y solución de errores.

Visual Studio en su edición Community en su versión de IDE gratuito cuenta con una comunidad de desarrolladores activa y colaborativa. Esta comunidad proporciona conocimientos, soluciones y recursos que facilitan la resolución de problemas y el aprendizaje. Además, se ofrecen extensiones y complementos adicionales que amplían las capacidades de este.

GitHub

El entorno de desarrollo se integra sin problemas con otros servicios y plataformas de Microsoft, lo cual simplifica el desarrollo de aplicaciones para la nube, bases de datos y entornos empresariales. Esta integración perfecta permite aprovechar las capacidades adicionales que ofrece

la suite de productos de Microsoft.

Dentro de esta integración, se utilizará GitHub, una plataforma de alojamiento de repositorios de código fuente basada en la nube. Según Souto (2023) “GitHub es un servicio en la nube que permite a los desarrolladores y desarrolladoras almacenar y administrar su código, así como a llevar el registro y control de cualquier cambio que se realice sobre este código.” (párr.1) Para comprender la utilidad y la importancia de GitHub, resulta relevante explorar otro concepto estrechamente vinculado llamado Git; que de acuerdo con Souto (2023):

Git es un sistema de control de versiones que se utiliza para mantener un control sobre nuestros proyectos. Se puede decir que nos permite guardar el progreso de nuestro código de tal forma que, si en algún momento cometemos algún error irreversible en una versión posterior, siempre podremos recuperar una versión anterior en la cual todo funcionaba correctamente y retomar el proyecto desde ese punto. (párr.2)

La utilización de GitHub en el desarrollo del prototipo funcional para la empresa Zapatos Lindos S.A. brinda diversas ventajas, centrándose especialmente en el control de fuentes, la revisión de código y el seguimiento de problemas.

En cuanto al control de fuentes, GitHub ofrece un sistema de control de versiones distribuido altamente confiable. Esto significa que se puede mantener un historial completo de cambios y actualizaciones realizadas en el proyecto, lo que facilita la gestión y el seguimiento de las diferentes versiones. Además, permite trabajar en diferentes ramas del proyecto de forma simultánea, lo que posibilita el desarrollo de nuevas características o experimentos sin afectar la versión principal del prototipo funcional.

La revisión de código es otra ventaja destacada de GitHub. Los desarrolladores pueden realizar anotaciones y comentarios en el código fuente, lo que promueve la detección temprana de posibles errores, la optimización del rendimiento y la adhesión a las mejores prácticas de programación. Esta funcionalidad facilita una revisión colaborativa y exhaustiva del código, asegurando su calidad y eficiencia.

El sistema de seguimiento de problemas de GitHub resulta fundamental para el desarrollo del prototipo funcional. Permite registrar y asignar tareas, problemas y mejoras en un entorno organizado. Se pueden añadir comentarios, adjuntar archivos y realizar un seguimiento actualizado de los avances en la solución de cada problema. Esto asegura que todos los aspectos del desarrollo

se gestionen de manera efectiva y se resuelvan de manera oportuna.

Además de estas ventajas específicas, GitHub también proporciona otras características útiles para el desarrollo del prototipo, como la integración con herramientas de compilación y despliegue continuo, la posibilidad de automatizar tareas y pruebas, y la disponibilidad de un ecosistema de complementos y extensiones que pueden ampliar las funcionalidades de la plataforma.

Bases De Datos

Continuando con la integración de herramientas de la suite de Microsoft al desarrollar el prototipo funcional, es imprescindible contar con una sólida gestión de bases de datos para almacenar y manipular la información de manera eficiente. En este sentido, se ha optado por utilizar Microsoft SQL Server 2017 en su edición Express para efectos del desarrollo y una vez que esté publicado se utilizará la versión 2017 pero bajo la licencia estándar que proporciona el *hosting* de alojamiento llamado SmarterAsp.NET; esta es una poderosa herramienta de base de datos que se integra perfectamente con el entorno de desarrollo.

Esta elección permitirá aprovechar las características de este sistema de gestión de bases de datos, como su alta disponibilidad, rendimiento optimizado y seguridad avanzada. Además, SQL Server 2017 cumple con los principios ACID (Atomicidad, Consistencia, Aislamiento, Durabilidad), asegurando la integridad y confiabilidad de los datos. Esta versión de SQL Server es una opción confiable y escalable para administrar bases de datos, incluso en entornos de menor escala. Entre las características destacadas de SQL Server 2017 se podemos mencionar que ofrece opciones para garantizar la disponibilidad continua de los datos. En su edición Express de SQL Server permite aprovechar las funcionalidades de optimización de consultas y rendimiento para lograr una ejecución más eficiente de las consultas y transacciones.

En cuanto a seguridad proporciona un conjunto de características de seguridad sólidas para proteger los datos, como autenticación integrada, cifrado de datos y auditoría.

Con la combinación del entorno de desarrollo integrado, la integración con otros servicios de Microsoft, y el uso de SQL Server 2017 con sus características, se establece una sólida base para el desarrollo del prototipo; asilo que permite así, una gestión y almacenamiento de los datos de manera confiable y eficiente.

A continuación, se podrán explorar conceptos adicionales relacionados con las bases de datos para una comprensión más profunda y un desarrollo exitoso de la aplicación por lo que se

iniciará con el concepto de diagrama entidad-relación.

Diagrama Entidad Relación

Según lucidchart.com (s.f) “Un diagrama entidad-relación, también conocido como modelo entidad relación o ERD, es un tipo de diagrama de flujo que ilustra cómo las "entidades", como personas, objetos o conceptos, se relacionan entre sí dentro de un sistema.” (párr.1).

Este diagrama proporciona una base sólida para el diseño de la base de datos, ya que permite identificar las entidades, los atributos y las relaciones entre ellas de manera clara y estructurada. Esto ayuda a garantizar la integridad y coherencia de los datos, evitando redundancias e inconsistencias; también sirve como una guía para el proceso de normalización de la base de datos. Al visualizar las entidades y las relaciones, es posible identificar y eliminar redundancias y dependencias no deseadas, lo que conduce a una base de datos más eficiente y fácil de mantener.

En el contexto del modelo entidad-relación, existen tres elementos fundamentales: entidad, atributos y relaciones. Cada uno de estos elementos desempeña un papel crucial en el diseño y desarrollo de una base de datos.

Entidades

En primer lugar, las entidades representan los objetos o conceptos del mundo real que se desea modelar en la base de datos. De acuerdo con Monteserín (2020) “Una entidad es una «cosa» u «objeto» en el mundo real que es distinguible de otros objetos” (párr.1). Pueden ser personas, lugares, objetos, eventos o cualquier otra cosa que sea relevante para el sistema. Las entidades se representan mediante rectángulos en los diagramas de entidad relación y sirven como contenedores para almacenar información relacionada. En el contexto de la base de datos la entidad se les conoce como tablas.

Atributos

En segundo lugar, tenemos los atributos que, por su parte, son las características o propiedades de las entidades y si se refiere a la base de estos, estos serían los campos. Según keepcoding (2022) los atributos “Son elementos que en conjunto conforman una entidad en el modelo entidad-relación. Puedes imaginarlos como las características que describen una entidad, o sea los adjetivos y los tipos de sustantivos” (párr.11). Estos describen los detalles específicos y los datos asociados a cada entidad. Por ejemplo, en la base de datos del prototipo por desarrollar se tiene la entidad clientes y los atributos de esta incluyen el nombre, la dirección y el número de teléfono entre otros. Los atributos se representan como óvalos conectados a las entidades en los diagramas entidad- relación.

Relaciones

Por último, las relaciones definen las asociaciones o vínculos entre diferentes entidades en la base de datos. Según Monteserín (2020) “Una relación es una asociación entre varias entidades” (párr.3). Estas relaciones indican cómo se relacionan y se comunican las entidades entre sí. Pueden ser uno a uno, uno a muchos o muchos a muchos, según la complejidad de la relación dentro del contexto del problema. Las relaciones se representan mediante líneas que conectan las entidades en los diagramas entidad- relación, y su tipo se indica mediante símbolos como una línea simple, una línea con un rombo o una línea doble, según corresponda.

Claves Primarias y Claves Foráneas

Asimismo, en el contexto de base de datos, las relaciones se definen mediante el uso de claves primarias y claves foráneas. La llave primaria es esencial para garantizar la integridad y la unicidad de los datos almacenados, ya que evita la duplicación y asegura que cada registro sea único y distinguible de los demás. De acuerdo con la documentación en IBM (2021)

Una clave primaria es una columna o un conjunto de columnas en una tabla cuyos valores identifican de forma exclusiva una fila de la tabla. Una base de datos relacional está diseñada para imponer la exclusividad de las claves primarias permitiendo que haya sólo una fila con un valor de clave primaria específico en una tabla. (párr.2)

Además, actúa como punto de referencia para establecer relaciones con otras tablas dentro de la base de datos. Al ser único y permanente, este identificador permite una rápida y eficiente recuperación de datos a través de operaciones de búsqueda y acceso.

Por su parte, las llaves foráneas, desempeñan un papel complementario y estrechamente relacionado en el diseño de bases de datos. Según la documentación en la página web de IBM (2021)

Una clave foránea es una columna o un conjunto de columnas en una tabla cuyos valores corresponden a los valores de la clave primaria de otra tabla. Para poder añadir una fila con un valor de clave foránea específico, debe existir una fila en la tabla relacionada con el mismo valor de clave primaria. (párr.3)

Una llave foránea es un atributo o conjunto de atributos en una tabla que establece una conexión o relación con una llave primaria en otra tabla. Esta relación permite vincular entidades de diferentes tablas y establecer la integridad referencial.

Además de las clave primarias y foráneas, otro concepto importante en el ámbito de las bases de datos es el diccionario de datos SQL; tal y como se explica en Master SQL (2023):

Un diccionario de datos SQL es una herramienta importante para la gestión de bases de datos. Esencialmente, es una colección de metadatos que describen la estructura de la base de datos y los datos que se almacenan en ella. Con esta información, los administradores de bases de datos pueden asegurarse de que la base de datos esté bien organizada y sea fácilmente accesible para los usuarios finales. (párr.1)

Diccionario de Datos

Elaborar un diccionario de datos es importante, ya que proporciona documentación detallada de los elementos de datos utilizados en la base de datos, lo que facilita la comprensión de su estructura y significado. Además, ayuda a establecer estándares y convenciones para la nomenclatura y estructura de los datos, asegurando la consistencia y facilitando la colaboración entre los equipos. También, actúa como una referencia centralizada para el diseño, desarrollo y administración de la base de datos, lo que permite una gestión más eficiente y una toma de decisiones informadas.

Lenguajes De Base De Datos

También, es crucial comprender los diferentes lenguajes utilizados en la base de datos, como el DML (Lenguaje de Manipulación de Datos), que permite insertar, actualizar y eliminar datos; el DDL (Lenguaje de Definición de Datos), que se utiliza para crear, modificar y eliminar objetos de la base de datos; y el DQL (Lenguaje de Consulta de Datos), que permite realizar consultas y extraer información específica.

Además, es esencial entender el DCL (Lenguaje de Control de Datos), que se utiliza para otorgar y revocar permisos de acceso, y el TCL (Lenguaje de Control de Transacciones), que permite controlar y administrar transacciones en la base de datos.

Los conceptos mencionados anteriormente son fundamentales para comprender y desarrollar la base de datos del prototipo funcional para la gestión de ventas para Zapatitos Lindos S.A.

Facturación Electrónica

Después de una inmersión exhaustiva en los aspectos técnicos y de desarrollo, este proyecto toma un nuevo rumbo estratégico. Su foco se desplaza hacia la concreción y ampliación

de los componentes esenciales que conformarán el prototipo funcional diseñado para optimizar la gestión de ventas de Zapatitos Lindos S.A., ubicada en San José. En esta fase avanzada de la investigación, se hace hincapié en la delineación minuciosa y la expansión meticulosa de los módulos que integran el prototipo en desarrollo. Cada uno de estos módulos se concibe con la finalidad de potenciar la eficacia y la eficiencia en los procesos de ventas, así como en las áreas adyacentes que inciden en el desempeño operativo de la organización.

El proceso de llevar a cabo este prototipo se inicia con la introducción de la facturación electrónica, un pilar esencial en la transformación de las transacciones comerciales. En esta fase, se realizará un análisis exhaustivo de los requerimientos legales y tecnológicos inherentes a la facturación electrónica, destacando su capacidad para reducir errores y fortalecer la transparencia en las operaciones comerciales. Asimismo, se llevará a cabo un análisis detallado del procedimiento relativo a las anulaciones o notas de crédito electrónicas. Es importante destacar que la trascendencia de este proceso va más allá de los aspectos financieros, ya que ejerce un impacto directo en la satisfacción del cliente y en la garantía de una trazabilidad contable precisa y fiable.

La iniciativa de la implementación de la factura electrónica fue y ha sido impulsada por la Dirección General de Tributación Directa y tiene como objetivo principal mejorar la eficiencia en la emisión y registro de comprobantes de venta. A través del tiempo, se han realizado diversas actualizaciones y mejoras en el sistema de facturación electrónica para adaptarse a las necesidades cambiantes y a los avances tecnológicos.

La facturación electrónica en Costa Rica tiene una historia que se remonta a más de una década. Con el pasar de los años, se ha consolidado como una herramienta eficiente y segura para la emisión y registro de comprobantes de venta, contribuyendo a la modernización y transparencia en los procesos fiscales del país.

En el año 2003, mediante la resolución N° 04-03, se autorizó el uso de comprobantes electrónicos en el país, con el objetivo de que estos pudieran respaldar los gastos e ingresos de una manera lógica por medio del almacenamiento de estos por computadora, en sustitución de los comprobantes físicos o de papel. Sin embargo, no fue hasta el año 2007 que se introdujo por primera vez la facturación electrónica como tal en Costa Rica, esto como parte de los esfuerzos por modernizar los procesos administrativos y fiscales.

Con la publicación de la resolución N.º DGT-22-07 de la Dirección General de Tributación

en San José, a las nueve horas del día diecisiete de septiembre de dos mil siete; el país contó con la primera normativa para el uso de tecnología en la emisión y recepción de documentos electrónicos. Esta primera normativa estaba compuesta por cinco capítulos, en donde, por ejemplo, en el capítulo uno llamado “Autorización para el uso de factura electrónica y documentos electrónicos asociados” se dieron todas las definiciones iniciales sobre el tema de factura electrónica, y se abordaron términos como los siguientes: Documento electrónico, Emisor-Receptor, Nota de Crédito y Notas de Débito electrónicas, Receptor Electrónico-No emisor, entre otros; que más adelante y referenciando la última resolución vigente se explicarán con mayor detalle.

También, entre los diferentes artículos que conformaron esta resolución ya derogada se citaban los requisitos para la emisión y entrega de la factura electrónica, el proceso de aceptación y rechazo de documentos, los lineamientos para el almacenamiento y conservación de estos, el procedimiento para la anulación o corrección de documentos electrónicos, así como los lineamientos para la impresión de la factura electrónica. Consecuentemente, los otros cuatro capítulos contaban con información referente a registro especial de emisores-receptores electrónicos, receptores electrónicos-no emisores y sus obligaciones, así como la autorización para el uso de sistemas de facturación electrónica en sitios web entre otras disposiciones y anexos.

Posteriormente, para el año 2009 mediante resolución de esta Dirección General, N° DGT-02-09 de 9 de enero de 2009, publicada en La Gaceta N.º 25 del 5 febrero de 2009, se regula la "Autorización para el uso de factura electrónica y documentos electrónicos asociados". Ahí se establecía la obligación para las personas físicas o jurídicas de inscribirse ante la Administración Tributaria en diferentes roles; ya sea como emisores-receptores electrónicos, receptores electrónicos-no emisores o proveedores de soluciones de facturación en plataformas web.

Otro hito importante es que esta resolución estableció que, con fines tributarios, estaba permitido utilizar certificados electrónicos para llevar a cabo la firma digital de los documentos electrónicos designados por la Administración Tributaria y se hacía la observación de que esa disposición se mantendría vigente hasta que la entidad certificadora oficial implementara la emisión de certificados para agentes electrónicos, situación que hasta la fecha actual de este documento se mantiene vigente.

Para el año 2017, se implementó una nueva versión del sistema, conocida como Factura Electrónica 4.3, con el fin de cumplir con los estándares internacionales y facilitar la

interoperabilidad con otros países. Fue así como en Costa Rica se estableció la obligatoriedad de uso de la factura electrónica, un 20 de setiembre del año 2017; se publicó en la Gaceta N° 178 la comunicación de las fechas de inicio del uso de comprobantes electrónicos. Dicho comunicado detalló por sector, así como por tipo de profesionales independientes las fechas en que cada uno debía utilizar los comprobantes electrónicos. El primer sector que inició el proceso fue el sector salud, al cual se le dio como fecha de inicio el 15 de enero del 2018. El segundo sector impactado fue el financiero contable, el cual inició un 1 de febrero del 2018. Consecuentemente ingresaron el sector legal un 1 de marzo del 2018 y el sector ingeniería, arquitectura e informática un 2 de abril del 2018 y, para finalizar, los demás sectores debían iniciar con el proceso un 1 mayo de 2018.

También, en la publicación de La Gaceta ya mencionada se resaltó la atención a los contribuyentes obligados, cuyas operaciones comerciales no estaban contempladas en las tablas publicadas por sectores que, de acuerdo con el último dígito de su número de cédula se establecían las fechas de cumplimiento, quedando la programación de la siguiente manera: aquellos con terminaciones en uno, dos o tres debieron adoptar el sistema de facturación electrónica, a partir del primero de septiembre de 2018. En contraste, los contribuyentes con el último dígito de su número de cédula en cuatro, cinco y seis tenían como plazo máximo el primero de octubre de 2018 para hacerlo. Por otro lado, las personas con un último dígito en cero, siete, ocho y nueve debían migrar al método electrónico para la facturación antes del primero de noviembre de 2018.

Desde entonces, la facturación electrónica ha experimentado una amplia adopción en Costa Rica, tanto por parte de las grandes empresas como por los pequeños y medianos negocios. Se ha convertido en una herramienta fundamental para agilizar los procesos contables, mejorar el control fiscal y reducir la evasión de impuestos.

Es importante destacar que la legislación relacionada con la facturación electrónica en Costa Rica ha ido evolucionando a lo largo del tiempo. Se han establecido requisitos específicos en términos de formatos, firma electrónica y almacenamiento de los comprobantes electrónicos, con el objetivo de garantizar su validez y autenticidad. El fundamento legal del uso de comprobantes electrónicos se sustenta en dos decretos y tres resoluciones que a continuación se citarán:

- Decreto N° 41820 – H Reglamento de comprobantes electrónicos para efectos tributarios.

- Decreto N° 42195 – H Reforma al Reglamento de comprobantes electrónicos para efectos tributarios.
- Resolución MH-DGT-RES-0013-2023 Reforma Resolución general sobre las técnicas de los comprobantes de electrónicos para efecto tributarios.
- Resolución DGT-R-033-2019 Resolución general sobre las disposiciones técnicas de los comprobantes electrónicos para efectos tributarios.
- Resolución DGT-R-012-2018 Obligatoriedad general para el uso de los comprobantes electrónicos.

Factura Electrónica

El prototipo desarrollado para la empresa Zapatitos Lindos S.A. deberá cumplir con los requisitos legales y técnicos dentro del marco legal establecido para la facturación electrónica en Costa Rica y garantizará un sistema eficiente y escalable que solucione la problemática identificada, como cálculos incorrectos, rebajos inexactos de inventarios y pérdida de conexión, por lo que es importante abordar los siguientes conceptos como por ejemplo que es una factura electrónica; según GS1 Costa Rica (s.f) una factura electrónica

“Es un documento comercial con efectos tributarios, generado, expresado y transmitido en formato electrónico en el mismo acto de la compra-venta o prestación de servicio, ésta sustituye por igual a la ya tradicional factura en papel y la puede utilizar cualquier tipo de empresa.” (párr.1).

No se puede hablar de facturación electrónica sin mencionar los distintos tipos comprobantes electrónicos que se utilizan para respaldar transacciones comerciales y cumplir con las obligaciones fiscales. Estos documentos reemplazan a los tradicionales comprobantes impresos en papel, brindando una alternativa más eficiente, segura y ecológica. También, contienen información detallada sobre una transacción, como la descripción de los bienes o servicios, montos, impuestos, datos del emisor y receptor, entre otros.

De acuerdo con el sitio web Hacienda.go.cr (2023) una factura electrónica se define como “Comprobante electrónico para respaldar la venta de bienes y la prestación de servicios, que debe ser generado y transmitido en el mismo acto de la compraventa o la prestación del servicio” (párr.2).

El uso de la factura electrónica permite agilizar y simplificar el proceso de emisión, envío y almacenamiento de comprobantes, lo que genera ahorro de tiempo y costos. Además, ofrece

mayor seguridad y validez legal gracias a su tecnología de firma digital, evitando alteraciones o falsificaciones. Facilita la gestión tributaria al proporcionar un registro preciso de las transacciones y simplificar la presentación de informes. También, mejora el control y la trazabilidad de las operaciones comerciales, y contribuye a la sostenibilidad ambiental al reducir el consumo de papel.

Tiquete Electrónico

Otro comprobante que es muy utilizado son los tiquetes electrónicos; los cuales son documentos generados y transmitidos en formato electrónico, utilizado específicamente en operaciones con consumidores finales. A diferencia de la factura electrónica, el tiquete no está obligado a contener la información del receptor, ya que su finalidad principal es respaldar la venta de bienes o la prestación de servicios a consumidores finales. Según Hacienda.go.cr (2023) define Tiquete Electrónico de la siguiente manera “Documento generado, expresado y transmitido en formato electrónico XML, autorizado únicamente para operaciones con consumidores finales y que no puede ser usado para justificación de gastos o créditos.” (párr.3).

Es importante destacar que el tiquete electrónico no puede ser utilizado para justificar gastos o solicitar créditos fiscales. A diferencia de la factura electrónica, que brinda la posibilidad de utilizar los impuestos pagados como créditos en las declaraciones fiscales, el tiquete electrónico está limitado a operaciones con consumidores finales y no otorga beneficios fiscales adicionales.

Notas De Crédito y Débito Electrónicas

Además de los comprobantes electrónicos mencionados anteriormente, existen las notas de crédito y débito electrónicas. Estos desempeñan un papel crucial al permitir anular o modificar los efectos contables de las facturas o tiquetes electrónicos, sin alterar la información del documento original. Las notas de crédito electrónicas se utilizan para corregir errores que impliquen un monto a favor del receptor, mientras que las notas de débito electrónicas se emplean para ajustar montos adicionales a cobrar al receptor. Estos documentos ofrecen una forma precisa y transparente de corregir y ajustar transacciones, manteniendo la trazabilidad con los documentos originales. En el sitio Web hacienda.go.cr (2023) se definen las notas de crédito y débito electrónicas como: “Comprobantes electrónicos que permiten anular o modificar los efectos contables de la factura o tiquete electrónicos, sin alterar la información del documento origen.” (párr.4).

En el desarrollo del prototipo para la gestión de ventas de la empresa Zapaticos Lindos S.A., ubicada en San José, se aborda de manera profunda el proceso de anulación de facturas como parte de la estrategia central para abordar una problemática crucial: las pérdidas tangibles de

productos y las persistentes inconsistencias que aquejan al sistema de inventario. A nivel de comprobantes electrónicos el concepto de anulación no existe, por lo que para efectos contables y tributarios este proceso se realiza mediante la generación de notas de crédito o débito electrónicas.

Notas De Crédito y Débito Financieras

El módulo de anulaciones propuesto, también, incorpora el concepto de notas de crédito financieras, las cuales no se integrarán con la plataforma de comprobantes electrónicos proporcionada por el Ministerio de Hacienda. Estas notas de crédito financieras tienen como función principal ajustar saldos que no están vinculados con los registros reportados al Ministerio de Hacienda. Por ejemplo, al facturar a una institución gubernamental, se retiene una cantidad por impuestos específicamente el impuesto de renta. Sin embargo, al liquidar la factura, pueden quedar saldos correspondientes a dicha retención, los cuales deben ser ajustados y registrados adecuadamente en los movimientos contables.

Otro aspecto importante que agrega funcionalidad a las notas de crédito financiera es el siguiente: de acuerdo con Resolución N° DGT-R-033-2019 de 2019 [Dirección General de Tributación]. Resolución General sobre las disposiciones técnicas de los comprobantes electrónicos para efectos tributarios:

Quando el mensaje de confirmación por parte de la Dirección General de Tributación es de rechazo, el comprobante emitido carece de validez, por lo que según el artículo 18 del DGT-R-033-2019 Página 10 de 11 Reglamento de Comprobantes Electrónicos para efectos tributarios no debe realizarse la respectiva nota de crédito. (Art.9)

Esto quiere decir que, en el ámbito de la facturación electrónica, cuando un documento electrónico es rechazado por la Dirección General de Tributación, este queda inválido, lo que descarta la necesidad de generar una nota de crédito electrónica y remitirla al Ministerio de Hacienda. Sin embargo, resulta esencial instaurar una transacción en el sistema que sea capaz de revertir los impactos derivados de la emisión original de la factura o el tiquete electrónico. En este contexto, se implementará un enfoque específico en el módulo, donde estas notas de crédito financieras no se integrarán con la plataforma proporcionada por el Ministerio de Hacienda de Costa Rica.

En conjunto, estas funcionalidades servirán tanto para la gestión de documentos rechazados por la Dirección General de Tributación como para la liquidación de saldos, como los de retención

de renta. Esto asegurará una respuesta eficiente a diversas necesidades tributarias, lo que permite un manejo integral de las circunstancias particulares que puedan surgir.

Adicionalmente, este módulo no solo se concibe como una respuesta eficiente a la problemática planteada, sino también como una solución integral que se entrelaza con otros aspectos críticos de la operativa empresarial. La interconexión de este módulo con el sistema de control de inventarios, la gestión de cuentas por cobrar y el registro de ingresos y gastos, emerge como un aspecto fundamental para garantizar la coherencia y la eficiencia de la gestión integral de la empresa. Así, esta propuesta no solo apunta a mitigar las pérdidas y las discrepancias en el inventario, sino que también se erige como un pilar central para la optimización de los flujos financieros y la toma de decisiones estratégicas informadas.

Archivo XML

Otro concepto por resaltar cuando se habla de facturación electrónica son los archivos con formato electrónico XML; tal y como se menciona en la documentación disponible en hacienda.go.cr (2023) el archivo XML

“Es un documento en el formato denominado, por sus siglas en inglés, “Extensible Markup Language” (XML) o Lenguaje Extensible de Etiquetado, el cual permite la transmisión de documentos entre diversas plataformas, así como su validación e interpretación entre aplicaciones y organizaciones.” (párr.13)

La función principal de los archivos XML en la facturación electrónica es la representación estructurada y estandarizada de los datos de una factura. El formato XML permite organizar la información de manera jerárquica, utilizando etiquetas y atributos para definir los diferentes elementos y atributos de la factura, como el emisor, el receptor, los detalles de los productos o servicios, los montos y otros datos relevantes.

Firma Electrónica

En el contexto de los archivos XML, utilizados en diversos procesos, como la facturación electrónica o la transmisión de documentos de forma segura, la firma electrónica permite garantizar la autenticidad, integridad y no repudio de los archivos XML; esto brinda confianza y respaldo legal a las transacciones electrónicas.

La firma electrónica aplicada a los archivos XML consiste en la generación de un conjunto de datos criptográficos que se adjuntan al archivo XML. Estos datos contienen información única y exclusiva del emisor del archivo, lo que permite verificar su identidad y asegurar que el contenido

del archivo no ha sido modificado desde su firma. La Dirección General de Tributación (DGT), establece que la autenticación de los documentos electrónicos se lleva a cabo mediante la inclusión de una firma digital en los archivos XML.

La firma digital se implementa mediante la inclusión de una etiqueta Signature que sigue las directrices del estándar XMLDSig, que es una especificación del W3C utilizada para garantizar la autenticidad, integridad y no repudio de documentos en formato XML. La etiqueta contiene una firma XAdES-EPES conforme a las especificaciones del estándar ETSI TS 101 903, versión 1.3.2 o posterior, y se utiliza un empaquetado ENVELOPED en conjunto con la firma electrónica. Esto significa que la firma se coloca como un elemento dentro del propio documento electrónico como una etiqueta o un conjunto de etiquetas XML que rodean los elementos que están siendo firmados. Esta técnica garantiza que la firma esté directamente asociada con el contenido que se está autenticando, lo que facilita la verificación de la firma y asegura que cualquier modificación en el contenido también afecte a la firma, lo que contribuye a mantener la integridad del documento.

En la generación de firmas digitales, se requieren certificados respaldados por algoritmos de encriptación como RSA 2048 o RSA 4096, junto con algoritmos hash criptográficos pertenecientes a la familia Secure Hash Algorithm (Algoritmo de Hash Seguro), como lo son SHA-256 o SHA-512. El Ministerio de Hacienda brinda a los contribuyentes un método de seguridad sólido y confiable al emplear llaves criptográficas RSA 2048 y el algoritmo de hash SHA-256 para realizar firmas XAdES-EPES en documentos electrónicos. Esta combinación de tecnologías criptográficas avanzadas no solo garantiza la autenticidad y la integridad de los documentos, sino que también proporciona un alto nivel de seguridad en el intercambio de información financiera y tributaria.

La elección de RSA 2048 para las llaves criptográficas asegura una robusta protección contra ataques de fuerza bruta y factores de seguridad adicionales, mientras que SHA-256, al generar valores hash únicos de los datos, ofrece una capa adicional de seguridad al detectar cualquier alteración o manipulación de los documentos firmados. Esta implementación tecnológica no solo cumple con los estándares de seguridad exigidos por las regulaciones tributarias, sino que también proporciona a los contribuyentes la confianza necesaria para realizar transacciones electrónicas seguras con el Ministerio de Hacienda.

Referente al almacenamiento y conservación de los documentos electrónicos, tanto el Ministerio de Hacienda como los contribuyentes asumen responsabilidades significativas para

garantizar la integridad y la seguridad de la información. Siguiendo lo estipulado en la legislación pertinente, los documentos electrónicos se enviarán por correo electrónico tanto al receptor como al emisor para su respectivo respaldo. Además, se implementará un sistema de almacenamiento en la base de datos SQL Server, utilizando un campo de tipo TEXT codificado en Base64, lo que permite una gestión eficiente del espacio de almacenamiento en comparación con la guardia directa del XML.

Este proceso asegura la inalterabilidad, privacidad, legibilidad, accesibilidad, preservación, confidencialidad, autenticidad, integridad y consulta posterior de la información contenida en todos los archivos XML. Cabe destacar que el contribuyente también está obligado a asumir la responsabilidad de almacenar y conservar adecuadamente todos los documentos electrónicos emitidos durante un período de 5 años, como lo indica la legislación vigente, para cumplir con los requisitos legales de archivo y preservación de registros.

Catálogo CABys

Continuando con los conceptos por resaltar en el proceso de facturación electrónica uno de los últimos ajustes que se incorporaron al proceso de facturación electrónica fue el catálogo de bienes y servicios conocido de aquí en adelante como CABys. Con respecto a este catálogo, el Banco Central de Costa Rica en coordinación con el Ministerio de Hacienda establecieron un procedimiento estándar para la clasificación de productos y servicios comercializados a nivel nacional. Según la información que se encuentra en el sitio Web hacienda.go.cr (2019)

El CAByS constituye un catálogo que agrupa los bienes y servicios en categorías jerarquizadas, el cual deberá utilizarse al momento de emitir los comprobantes electrónicos. Las tarifas del impuesto sobre el valor agregado (IVA) que se publiquen en el CAByS son únicamente una referencia informativa. Por lo tanto, será responsabilidad del contribuyente asegurarse que coincidan con la normativa vigente. Ningún comprobante será rechazado por utilizar una tarifa diferente a la indicada en el CAByS. (párr.2)

El catálogo CAByS organiza los productos y servicios en categorías jerarquizadas y les asigna la correspondiente tarifa de Impuesto sobre el Valor Agregado (IVA), de acuerdo con la legislación vigente. Es obligatorio que los contribuyentes incluyan en los comprobantes electrónicos el código de identificación de cada bien o servicio, conocido como "Código de

Producto/Servicio", asignado en el Catálogo de bienes y servicios CAByS. De esta manera, se garantiza la correcta identificación y aplicación del IVA en los documentos fiscales.

En el contexto empresarial, como el de Zapatitos Lindos S.A., es esencial realizar una correcta categorización de los productos conforme al Código CABys, ya que la mayoría de los productos se facturan como exentos. Esto se vuelve fundamental para evitar cualquier inconsistencia en la información que se remite al Ministerio de Hacienda. El catálogo CABys, que abarca más de veinte mil productos y servicios, se identifica mediante un código numérico de trece dígitos, donde cada número denota una categoría y subcategoría específica.

Desde el primero de diciembre del año dos mil veinte, la aplicación del CABys se ha vuelto obligatoria, con el propósito primordial de establecer un riguroso control y trazabilidad fiscal. Además, el sistema busca facilitar la comparabilidad, el cálculo y el análisis de las estadísticas económicas del Banco Central de Costa Rica, entre otros objetivos. En este sentido, el correcto manejo y uso del CABys se convierte en un aspecto crítico para las operaciones comerciales, asegurando la transparencia y precisión en los registros tributarios y contribuyendo a la eficiencia del sistema fiscal en su conjunto.

Gestión De Cuentas Por Cobrar y Cuentas por Pagar

Simultáneamente a los módulos de facturación y anulación de documentos, se dedicará una atención especial a la gestión de cuentas por cobrar y cuentas por pagar. Estos componentes críticos de la gestión financiera no solo influyen en la salud económica de la empresa, sino que también tienen un impacto directo en el flujo de efectivo y en las relaciones comerciales. Es crucial explorar con minuciosidad la administración de plazos de pago, la implementación de recordatorios de cuentas pendientes y la optimización de la comunicación con proveedores y clientes, subrayando su efecto en la sostenibilidad y la rentabilidad de la empresa.

Cuentas Por Cobrar

El propósito fundamental de la Gestión de cuentas por cobrar radica en convertir estas en recursos líquidos en el menor tiempo posible, para revertir, así, el impacto adverso en el flujo de efectivo o en el Capital de Trabajo de la empresa. Por lo que cualquier proceso relacionado con la gestión de cuentas por cobrar requiere de una supervisión y medición constantes, por lo que es esencial utilizar informes que muestren en tiempo real la evolución de los indicadores clave de cobranza. Estos indicadores, también conocidos como KPIs, juegan un papel vital al permitir la adaptación de la estrategia de gestión de cobranza en función de los resultados obtenidos. Con

todo esto es posible implementar un plan de cobro más efectivo y eficiente, para optimizar, de esta manera, las tareas realizadas por el equipo a cargo de la gestión de cuentas por cobrar.

Cuentas Por Pagar

Con respecto a las cuentas por pagar, se hace referencia al procesamiento de pagos que la empresa debe realizar. Esta gestión representa un proceso estratégico que involucra la administración y seguimiento de las obligaciones financieras de la empresa con sus proveedores y acreedores. Comprende tareas esenciales como el registro de facturas pendientes, el establecimiento de plazos de pago, la verificación de la concordancia entre las facturas y los bienes o servicios recibidos y la planificación eficiente de los recursos para cumplir con los compromisos económicos de manera puntual.

La gestión de cuentas por pagar no solo impacta la relación con los proveedores, sino que también influye en la liquidez y estabilidad financiera de la empresa, lo que evita posibles sanciones y genera un ambiente propicio para el mantenimiento de relaciones comerciales sólidas. En esta perspectiva, el módulo de gestión de cuentas por pagar dentro del prototipo es de gran valor para la optimización de los procesos financieros de la empresa Zapatitos Lindos S.A.

Conceptos Contables

Adicional a todo lo mencionado anteriormente, la visión global de la gestión empresarial se cristaliza a través de la integración contable que el prototipo por desarrollar ofrece. Esta sinergia esencial entre la gestión de ventas y la contabilidad asegura la coherencia de los registros financieros y la generación automática de informes precisos, fundamentales para la toma de decisiones estratégicas y la planificación a largo plazo. Este módulo permitirá automatizar el registro de asientos contables y el manejo del catálogo contable, lo que agilizará y optimizará el proceso de contabilización de las transacciones financieras de la empresa.

Activos

En el ámbito financiero, comprender y definir con precisión los términos clave es fundamental para construir una base sólida en la gestión empresarial. Uno de estos conceptos primordiales es el de Activo. Según GBM (2022)

Los activos son todos aquellos bienes, recursos y servicios que puede poseer una empresa. Estos elementos deben haber sido adquiridos en su totalidad para que su posesión genere recursos a largo plazo. Los activos de una empresa pueden ser muy variados, y tienen una importancia central en el cálculo del patrimonio neto de la

empresa. (párr.3).

A través de esta definición, se despliega la importancia intrínseca del activo como un componente esencial en la evaluación del valor económico de una entidad y su capacidad para generar beneficios y crecimiento. Los activos se reflejan en el balance general y proporcionan información esencial para la toma de decisiones de inversión.

Pasivos

Continuando en el ámbito contable, tenemos otro concepto “los pasivos”, que al igual que los activos desempeñan un papel crucial en la evaluación de la situación financiera de una empresa. Como menciona Euroinnova Business School (2022): “Los pasivos son las obligaciones y deudas que tiene la empresa, de entre las cuales figuran: pagos a bancos por créditos, impuestos, salarios, entre otros. Los pasivos están compuestos por las obligaciones que tiene la empresa con tercer” (párr.13). Conocer y comprender los pasivos contables es fundamental para analizar la solvencia y la capacidad de pago de la empresa, así como para tomar decisiones financieras acertadas.

Patrimonio

Una vez abordados los conceptos de Activo y Pasivo, se procederá a la definición del concepto Patrimonio. Según García (2017) “El patrimonio de una empresa, particular o colectiva, está compuesto por los bienes, derechos y obligaciones que constituyen los medios económicos y financieros a través de los cuales ésta puede cumplir sus fines.” (p.27).

El patrimonio es una medida clave de la riqueza y el valor económico de la organización. Comprender y evaluar el patrimonio permite a los inversionistas y a los propietarios de la empresa tomar decisiones informadas sobre la asignación de recursos y la toma de riesgos.

Dentro del proceso contable, el libro de diario desempeña un papel central. Este registro sistemático y cronológico de las transacciones financieras proporciona una visión detallada de las operaciones diarias de la empresa. Según Dobaño (2023),

El libro diario, también conocido como diario contable o diario general, es un registro financiero donde se anotan todas las operaciones comerciales y financieras de una empresa en el orden en que ocurren.

Cada transacción se registra en términos de debe (débito) y haber (crédito), y se proporciona una breve descripción de la transacción, además de los correspondientes importes. (párr.3)

Tal y como se menciona anteriormente, cada transacción se registra en forma de asiento

contable, los cuales incluyen información adicional tales como la fecha, una descripción de la operación, las cuentas afectadas y los montos correspondientes.

Asientos Contables

Los asientos contables son registros específicos que se realizan en el libro de diario para reflejar el impacto de las transacciones financieras en las cuentas contables.

Cada asiento contable consta de al menos dos partes: un cargo debe y un abono haber. Siguiendo el principio contable de partida doble, cada transacción afecta al menos dos cuentas de manera igual y opuesta. Los asientos contables permiten documentar de manera precisa y sistemática cada operación financiera, lo que facilita la preparación de informes financieros y la generación de análisis contables más detallados. A través de los asientos contables, se registra y rastrea el flujo de dinero, las deudas, los ingresos y los gastos, así, se contempla, el panorama de control y análisis financiero de la empresa.

Libro De Diario

El libro de diario es una herramienta esencial para mantener un seguimiento preciso de las transacciones y facilitar la generación de informes financieros claros y precisos.

Uno de los informes financieros más importantes generados a partir del libro de diario es el balance general.

Balance General

Según Guzmán (2023) “El balance general es un estado financiero que brinda información sobre los activos, pasivos y patrimonio neto de una empresa, durante un período de tiempo determinado.” (párr.1)

También conocido como estado de situación financiera, el balance general ofrece una instantánea de la posición financiera de la empresa en un momento específico. Este informe muestra los activos, pasivos y el patrimonio de la empresa.

Catálogo Contable

Para poder generar estos informes, es esencial contar con un catálogo contable. Dicho catálogo se trata de una lista organizada y estructurada de todas las cuentas contables utilizadas por la empresa para clasificar y registrar las transacciones financieras. Cada cuenta en el catálogo tiene un código único y una descripción que indica su naturaleza y función.

El catálogo de cuentas proporciona una estructura coherente y consistente para el registro contable, lo cual facilita la generación de informes financieros precisos y la realización de análisis

contables más profundos. Además, permite un seguimiento eficiente de las transacciones financieras y una gestión adecuada de los activos, pasivos y el patrimonio de la empresa.

Estos conceptos proporcionan las bases necesarias para un registro preciso, una presentación clara de los estados financieros y un análisis detallado de la salud financiera de la empresa. Al comprender y aplicar adecuadamente estos conceptos, se puede proporcionar información valiosa para la toma de decisiones estratégicas y el éxito a largo plazo de la empresa.

Control De Inventarios

A medida que se desglosa los módulos y funcionales del prototipo por desarrollar, se logra ir adentrándose en el panorama integral del Control de Inventarios; para lo cual se explorará en detalle los distintos tipos de inventarios y sus métodos de valoración, delineando cómo el prototipo establecerá un control exhaustivo y cómo alertará sobre niveles críticos; de esta manera se evitará tanto el exceso como la carestía de existencias.

La trascendencia de mantener un equilibrio óptimo en los niveles de inventario se manifestará en el análisis de su relevancia en el funcionamiento eficiente del sistema. Se explorará cómo el prototipo se erigirá como una herramienta vital para prevenir tanto los excesos como los faltantes en el inventario, lo que promoverá la optimización de los recursos y la continuidad de las operaciones. Esta exploración resaltarán la importancia de evitar los inconvenientes inherentes a los niveles de inventario desequilibrados, ya que los excesos inmovilizan capital y los faltantes interrumpen la producción y el servicio al cliente.

En el contexto de una administración precisa de existencias, el prototipo surgirá como un recurso esencial al ofrecer una solución integral. Esta solución permitirá el registro meticuloso de las existencias, proporcionando un panorama claro y actualizado en todo momento. Además, la incorporación de alertas automáticas para niveles críticos fortalecerá la capacidad de respuesta del sistema ante situaciones potencialmente disruptivas. A través de esta funcionalidad, el prototipo no solo mitigará los riesgos asociados con la escasez repentina, sino que también evitará los inconvenientes de una inversión excesiva en inventario, fomentando así una operatividad equilibrada y eficiente.

Inventario

De acuerdo con el sitio IBM.com (s.f.):

El inventario es el conjunto de artículos o materiales que un negocio tiene la intención de vender a los clientes con fines lucrativos. La gestión de inventario, un

elemento crítico de la cadena de suministro es el seguimiento del inventario desde el momento de su fabricación hasta los almacenes, y desde estas instalaciones hasta el punto de venta. El objetivo de la gestión de inventario es tener los productos correctos en el lugar adecuado y en el momento preciso. (párr.1)

En el contexto de la empresa Zapaticos Lindos el inventario está compuesto por productos ortopédicos, los cuales se encuentran exentos, según la Ley 7293 de 1992. Ley Reguladora de Exoneraciones Vigentes, Derogatorias y Excepciones en donde se indica lo siguiente:

Exonérense de todo tributo y sobretasas la importación y la compra local de equipo médico, de sillas de ruedas y similares, de camas especiales para hospitales, de equipo ortopédico, de equipo para laboratorios químico-clínicos y de investigación agrícola, de equipos odontológicos, de prótesis en general y toda clase de equipo usado por parte de personas con problemas auditivos, así como el que se usa en programas de educación especial para discapacitados.

Asimismo, exonérense de todo tributo y sobretasas, excepto de los derechos arancelarios, las materias primas, los insumos y todo producto intermedio o final que se utilice en la elaboración de medicamentos. Se incluyen los reactivos o catalizadores, la maquinaria y los equipos requeridos para la producción de medicinas, envases y materiales de empaque de ellos. (Asamblea Legislativa de la República de Costa Rica, Ley Reguladora de Exoneraciones Vigentes, Derogatorias y Excepciones N° 7293, 1992, Abril 03).

Para una buena gestión del inventario es necesario controlar el inventario en otras palabras controlar las existencias, el objetivo de controlar el inventario es poder satisfacer las demandas de productos de los clientes y maximizar los beneficios.

Tipos De Inventarios

Existen varios tipos de inventarios de los cuales se consideran como los principales el sistema de inventario periódico y el inventario perpetuo o permanente. El método de Inventario Periódico presenta la flexibilidad de ejecutarse en intervalos variables, que pueden ser mensuales, semestrales o anuales. A través de esta práctica, la empresa logra obtener una comprensión precisa de su inventario actual y los costos relacionados con las ventas. El proceso implica la suma de las adquisiciones iniciales y su posterior resta en función del inventario final. En esencia, este enfoque brinda a la entidad una visión actualizada de sus existencias y la contribución que estas hacen al

cálculo de los costos de ventas.

Por otro lado, el enfoque de Inventario Perpetuo emerge como una opción crucial para fortalecer la gestión logística empresarial. Este método, altamente recomendado, se distingue por su capacidad para establecer un control constante sobre la disponibilidad de existencias en el almacén o bodega. Su utilidad reside en la capacidad de mantener un seguimiento continuo y detallado de cada importe y la cantidad de productos que la empresa resguarda.

Al optar por este enfoque, la empresa Zapaticos Lindos S.A. puede mantener un orden más fluido y sistemático en sus operaciones, lo que resulta fundamental para una logística eficaz y una toma de decisiones informada. La implementación de un registro permanente de las existencias fortalece la visión actualizada y precisa del *stock*, lo que contribuye, en última instancia, a optimizar la gestión de recursos y la eficiencia operativa.

Las ventajas primordiales del sistema de gestión de inventarios son de gran envergadura. En primer lugar, se destaca la elevación del nivel de servicio al cliente, lo cual conlleva a una reducción significativa en las pérdidas de ventas originadas por la falta de productos. Esta mejora en el servicio no solo conserva la satisfacción del cliente, sino que también fomenta una mayor fidelidad hacia la empresa.

Adicionalmente, la adopción de este sistema tiene un impacto positivo en el flujo de efectivo de la entidad. Esto se traduce en una adquisición más eficiente y una rotación optimizada de inventarios. Como resultado, los recursos financieros no se encuentran inmovilizados en la bodega, sino que se destinan a actividades que contribuyen directamente al crecimiento y operatividad de la empresa.

Este sistema se convierte en una herramienta valiosa para la identificación y prevención de situaciones adversas como robos y mermas. La capacidad de monitoreo y registro detallado de las existencias permite detectar de manera temprana cualquier irregularidad, de esta forma, se contribuye a la seguridad y la eficiencia en la gestión de inventarios.

Para efectos de la valuación del costo, en el caso de la empresa Zapaticos Lindos S.A., se utilizará el método del promedio ponderado, este método se utiliza para determinar el costo de venta de un producto cuando se implementa el sistema de inventario permanente o perpetuo. Mediante esta técnica, se procede a calcular un promedio del costo de las diversas unidades adquiridas. Este cálculo implica la suma de los valores presentes en el inventario y los valores de las adquisiciones (compras) más recientes. Luego, este total se divide entre la cantidad total de

unidades presentes en el inventario, abarcando tanto las existentes desde el inicio como las que se añadieron con la nueva compra.

El enfoque del promedio ponderado ofrece un costo de venta más equilibrado, precisamente debido a que representa el promedio de los precios pagados para obtener o construir dicho inventario, por lo que este método de cálculo refleja una visión más justa y balanceada de los costos incurridos en las distintas compras. Así, se contribuye a una gestión financiera más precisa y eficaz. Cada vez que se realiza una compra, se debe recalcular este promedio, a razón de que la nueva compra puede tener un precio distinto y, adicionalmente, incrementa la cantidad de unidades disponibles. El control del inventario utilizando el promedio ponderado se realizará mediante un Kardex de inventario en donde con cada compra o venta se van determinando los diferentes movimientos que permitirán determinar el costo de venta promedio en todo momento.

Test Driven Development

Dentro de los conceptos por implementar en el desarrollo del prototipo funcional, se encuentran, en primer lugar, el enfoque Test Driven Development (TDD); que según Herranz (2023) “TDD o Test-Driven Development (desarrollo dirigido por tests) es una práctica de programación que consiste en escribir primero las pruebas (generalmente unitarias), después escribir el código fuente que pase la prueba satisfactoriamente y, por último, refactorizar el código escrito.” (párr.1). A manera de resumen, se basa en la creación de pruebas automatizadas antes de escribir el código de producción. Este enfoque fomenta un ciclo de desarrollo iterativo y colaborativo, en el que los desarrolladores definen primero los requisitos funcionales en forma de pruebas y luego implementan el código necesario para que las pruebas pasen satisfactoriamente. TDD mejora la calidad del *software* al garantizar que cada funcionalidad esté completamente probada y funcional, lo que reduce los errores y facilita el mantenimiento y las modificaciones futuras.

Clean Code

Continuando con enfoques que aportan valor al proyecto se hará uso de código limpio (Clean Code); como lo menciona Sánchez (2022) “Clean Code es una filosofía utilizada en el desarrollo de software cuyo objetivo es hacer más fácil la lectura y escritura de código” (párr.2).

Cuando se habla de Clean Code, se hace referencia a un conjunto de prácticas y convenciones para escribir código legible, claro y de fácil mantenimiento. Se enfoca en la simplicidad, la claridad y la eliminación de la duplicación de código. El código limpio es esencial

para un desarrollo ágil y efectivo, ya que facilita la colaboración entre los miembros del equipo, reduce la curva de aprendizaje y mejora la capacidad de respuesta a los cambios.

SOLID

Tomando en cuenta la creación de código limpio, modular, flexible y fácil de mantener y continuando con la adopción de buenas prácticas de desarrollo, se incluirá en el desarrollo del prototipo los cinco principios fundamentales de diseño *software*, los principios SOLID. De acuerdo con Martín, (2018) “SOLID es el acrónimo que acuñó Michael Feathers, basándose en los principios de la programación orientada a objetos que Robert C. Martin había recopilado en el año 2000 en su libro “Design Principles and Design Patterns” (párr.2).

SOLID es un acrónimo que representa cinco principios los cuales son: Responsabilidad Única, Abierto/Cerrado, Sustitución de Liskov, Segregación de Interfaces e Inversión de Dependencias. Estos principios promueven un diseño modular y cohesivo, evitan la dependencia excesiva y facilitan la extensibilidad del *software*. Al seguir estos principios, los equipos de desarrollo pueden construir sistemas flexibles, fáciles de entender y mantener.

SCRUM

Es un marco de trabajo ágil que se centra en la entrega iterativa e incremental de *software*. Como se menciona en proyectosagiles.org (2018)

Scrum es un proceso en el que se aplican de manera regular un conjunto de buenas prácticas para trabajar colaborativamente, en equipo, y obtener el mejor resultado posible de un proyecto. Estas prácticas se apoyan unas a otras y su selección tiene origen en un estudio de la manera de trabajar de equipos altamente productivos. (párr.1)

Se basa en la transparencia, la inspección y la adaptación continua. Los proyectos de desarrollo de *software* se dividen en ciclos de trabajo llamados "Sprint", que generalmente tienen una duración fija de una a dos semanas. Durante cada sprint, el equipo se compromete a completar un conjunto de tareas definidas en el backlog del producto. Al final de cada sprint, se realiza una revisión para obtener retroalimentación y realizar ajustes en función de los resultados obtenidos. Scrum permite una mayor visibilidad y control del progreso del proyecto, lo que facilita la toma de decisiones y la adaptación rápida a los cambios del mercado.

INVEST

Dentro de Scrum tenemos el principio INVEST que es un acrónimo que se utiliza para evaluar la calidad de las historias de usuario. Al seguir este principio o también conocido como una metodología, las historias de usuario se vuelven más claras, manejables y efectivas para el equipo de desarrollo. INVEST asegura que las historias de usuario sean independientes entre sí, lo que permite una planificación y una entrega más flexibles. También enfatiza la importancia de tener historias de usuario valiosas para el cliente, estimables en términos de esfuerzo, pequeñas en tamaño para una entrega más ágil y testeables para garantizar su correcta implementación.

El desarrollo de un *software* implica una gestión integral del proyecto, donde se combinan análisis, desarrollo y gestión utilizando buenas prácticas y principios como los SOLID. En este marco, la metodología ágil Scrum emerge como un enfoque eficaz para abordar los desafíos del ciclo de vida del *software*. Uno de los aspectos clave en Scrum es la identificación de las epopeyas, que son grandes unidades de trabajo que encapsulan un conjunto de funcionalidades relacionadas. Estas epopeyas se descomponen luego en características más pequeñas y detalladas, que se conocen como *features*, y finalmente se expresan como historias de usuario, que describen las necesidades y requerimientos del usuario desde su perspectiva.

Este proceso de análisis y desarrollo de *software* no solo contribuye al crecimiento profesional en ingeniería informática, sino que también aporta al ámbito gerencial. La administración de recursos humanos se vuelve fundamental para formar equipos cohesionados y altamente productivos. El enfoque en la administración de recursos en el contexto del proyecto de desarrollo de *software* para Zapatitos Lindos busca la optimización de los recursos disponibles para lograr eficiencia y maximizar el valor entregado. Aunque este proyecto es liderado por una sola persona, la administración eficaz de recursos sigue siendo crucial.

Se trata de aprovechar al máximo los recursos tecnológicos, financieros y temporales disponibles para garantizar el éxito del proyecto. Esto implica seleccionar las herramientas y tecnologías más adecuadas que permitan una implementación eficiente y rentable del *software*. Además, se debe gestionar el tiempo de manera efectiva, estableciendo prioridades y cumpliendo con los plazos establecidos.

En este sentido, el proyecto de desarrollo de *software* para Zapatitos Lindos busca entregar valor a la empresa al mejorar la eficiencia de sus operaciones. Esto puede implicar la automatización de procesos manuales, la optimización de flujos de trabajo y la mejora de la

productividad del personal. La administración de recursos se enfoca en identificar y asignar los recursos necesarios para alcanzar estos objetivos, ya sea en términos de tiempo, presupuesto o capacidades técnicas.

Además, se considera fundamental el monitoreo y la evaluación continua del uso de recursos a lo largo del proyecto, con el fin de identificar oportunidades de mejora y realizar ajustes, según sea necesario. Esto garantiza que el proyecto se mantenga en el camino correcto y que se alcancen los resultados esperados de manera eficiente y efectiva, para entregar, así, un valor tangible a Zapatitos Lindos y contribuir a su crecimiento y éxito en el mercado.

Por otro lado, el mercadeo estratégico y la estrategia empresarial se entrelazan con el desarrollo de *software* al identificar las necesidades del mercado y alinearlas con los objetivos de la empresa. Además, el presupuesto de la empresa juega un papel vital en la planificación y ejecución de proyectos de *software*, asegurando una asignación eficiente de recursos financieros.

La gestión de proyectos se convierte en una habilidad clave para coordinar todas estas actividades de manera efectiva, garantizando el cumplimiento de los plazos y la calidad del producto final. Y finalmente, la reingeniería del proceso impulsa la mejora continua, permitiendo adaptar y optimizar los procesos de desarrollo de *software* conforme evolucionan las necesidades del mercado y de los usuarios.

Para concluir, el desarrollo de *software* no solo implica la creación de productos tecnológicos, sino que también abarca una amplia gama de habilidades y conocimientos que, a nivel gerencial, en conjunto con la aplicación buenas prácticas y metodologías ágiles proporcionan una base sólida para la toma de decisiones, maximizando el éxito del proyecto y optimizando los recursos disponibles.

CAPITULO III: MARCO METODOLÓGICO

En este apartado se aborda la importancia de la investigación como un proceso fundamental para el avance científico y la generación de proyectos sólidos y bien fundamentados; como lo menciona Pereyra (2022):

La investigación empieza con algo sencillo como observar y preguntar, lo que a su vez permite adquirir nuevos conocimientos. Cuando una investigación sigue un proceso y un orden se puede hablar de una metodología; ésta implica una búsqueda ordenada y sistemática de información y representa un proceso que busca la solución de un problema. Su fin es descubrir o dar a conocer los hechos. (p.16)

La investigación se enfoca en ampliar ideas o conocimientos previamente documentados, centrándose en el desarrollo de estos dentro de un proyecto. Para lograr este propósito, se emplean diversos enfoques que permiten alcanzar los objetivos planteados de manera efectiva. Es así como este apartado presenta la metodología necesaria para abordar los procedimientos y técnicas por las cuales se guiará el desarrollo del prototipo.

De acuerdo con Pereyra (2022), “la metodología de la investigación se define como el estudio de los procedimientos que permiten alcanzar y obtener nuevos conocimientos en el campo de la investigación científica, así como el desarrollo de nuevas ideas.” (p.16)

Con el fin de garantizar una mejor comprensión, se proporciona una explicación detallada de cada uno de estos aspectos en este capítulo.

Enfoque De La Investigación

La investigación científica se sustenta en diferentes paradigmas que definen los enfoques utilizados para estudiar fenómenos y generar conocimiento. Dos paradigmas ampliamente reconocidos son el enfoque cuantitativo y el cualitativo, los cuales ofrecen perspectivas complementarias. Además de estos enfoques, existe el enfoque mixto, que combina elementos del enfoque cuantitativo y cualitativo en una sola investigación.

El enfoque cuantitativo se caracteriza por su énfasis en la recolección y análisis de datos numéricos y objetivos; busca la objetividad y la replicabilidad, utilizando encuestas, experimentos controlados y análisis estadísticos para obtener resultados precisos y cuantificables.

Por otro lado, el enfoque cualitativo se centra en la comprensión profunda y detallada de

fenómenos sociales y humanos desde una perspectiva subjetiva. Se basa en la premisa de que la realidad es construida socialmente y se enfoca en la interpretación y el significado.

Además de estos enfoques, existe el enfoque mixto, que combina elementos del enfoque cuantitativo y cualitativo en una sola investigación. Este enfoque utiliza tanto datos cuantitativos como cualitativos para obtener una comprensión más completa y profunda del fenómeno estudiado y, a su vez, permite abordar preguntas de investigación complejas y obtener una visión más holística y contextualizada.

Cada enfoque tiene sus propias fortalezas y limitaciones, y su elección depende de la naturaleza del problema de investigación y los objetivos planteados. A continuación, se detallarán el tipo de enfoque seleccionado para el presente estudio.

Enfoque Cualitativo

El enfoque cualitativo de investigación se caracteriza por su énfasis en la comprensión profunda de fenómenos y contextos desde una perspectiva subjetiva. Tal y como lo establecen Blasco y Pérez (2007):

En la investigación cualitativa, se estudia la realidad en su contexto natural tal y como sucede, sacando e interpretando los fenómenos de acuerdo con las personas implicadas. Utiliza variedad de instrumentos para recoger información como las entrevistas, imágenes, observaciones, historias de vida, en los que se describen las rutinas y las situaciones problemáticas, así como los significados en la vida de los participantes. (p. 25).

Este enfoque busca capturar la complejidad y los significados que rodean a los fenómenos estudiados y se fundamenta en la premisa de que la realidad es construida socialmente y que es crucial comprender en profundidad los significados, experiencias y contextos de los participantes para generar conocimiento relevante. Según Sampieri et al. (2014), “El enfoque cualitativo (también conocido como investigación naturalista, fenomenológica o interpretativa) es una especie de “paraguas” en el cual se incluye una variedad de concepciones, visiones, técnicas y estudios no cuantitativos” (p. 19). Por lo que este enfoque se centra en el proceso y permite una comprensión más completa de las experiencias y perspectivas de los participantes.

Enfoque De Investigación Seleccionado

La elección del enfoque cualitativo de investigación para el desarrollo del prototipo funcional para la gestión de ventas en la empresa Zapatitos Lindos S.A. se justifica por diversas razones, entre las que resaltan sus ventajas y beneficios.

En primer lugar, el enfoque cualitativo permite una comprensión profunda de las problemáticas y necesidades específicas de la empresa. A través de técnicas como la observación y las entrevistas no estructuradas, es posible obtener información detallada sobre los procesos internos, las interacciones entre los diferentes actores y las dinámicas de la gestión de ventas en la empresa. Este enfoque proporcionará un panorama completo y enriquecedor de la situación, lo que facilitará la identificación de los desafíos y oportunidades clave.

Adicionalmente, este enfoque permite una exploración más flexible y adaptable en comparación con el enfoque cuantitativo. Al centrarse en la comprensión de los significados, experiencias y perspectivas de los usuarios y empleados de la empresa, se pueden descubrir aspectos críticos que podrían haber pasado desapercibidos en un enfoque puramente cuantitativo. Esto implica una mayor capacidad para ajustar y adaptar el prototipo funcional a las necesidades reales de la empresa y sus usuarios y que se pueden abordar aspectos inesperados o emergentes durante el proceso de investigación, lo cual resulta especialmente útil en el desarrollo de un prototipo funcional de *software*, en donde es fundamental considerar múltiples perspectivas y necesidades.

Los datos cualitativos proporcionan ideas o entendimientos valiosos sobre las motivaciones, expectativas y barreras de los usuarios, lo que contribuye a un diseño más centrado en el usuario. Al ser menos estricto y más abierto a la exploración de nuevas perspectivas, el enfoque cualitativo se adapta de manera óptima para el desarrollo del prototipo funcional para la gestión de ventas de la empresa Zapatitos Lindos S.A., permitiendo una mayor comprensión, ajustes continuos y adaptación a las necesidades cambiantes de la empresa que lo vinculan de manera efectiva con la metodología Scrum que se utilizará para el desarrollo de este proyecto. En conjunto, estos enfoques fomentan un desarrollo más efectivo, eficiente y orientado al usuario del prototipo funcional.

Método De La Investigación

En el ámbito de la investigación existen diversos tipos de investigación los cuales se utilizan para abordar diferentes preguntas u objetivos de investigación. Entre los principales tipos de investigación se encuentran la investigación descriptiva, la exploratoria y la explicativa, que son ampliamente utilizadas en los ámbitos académico y científico.

La investigación descriptiva se enfoca en recopilar información detallada y precisa sobre un fenómeno o situación, con el fin de obtener una visión clara de sus características y comportamientos. La investigación exploratoria, por su parte, se utiliza cuando se busca explorar un tema o problema poco estudiado, con el objetivo de generar ideas, hipótesis o teorías iniciales que puedan guiar investigaciones futuras y la investigación explicativa se centra en establecer relaciones de causa y efecto entre variables, con el propósito de comprender las razones o mecanismos subyacentes que explican un fenómeno en particular.

Además de estos enfoques, existen otros tipos de investigación como la investigación histórica, la investigación-acción, la investigación longitudinal, entre otros, cada uno con sus características y metodologías específicas. La elección del enfoque de investigación dependerá de los objetivos de estudio, las preguntas de investigación y el contexto en el que se desarrolla la investigación.

Investigación Descriptiva

La investigación descriptiva es un tipo de investigación que se enfoca en describir y caracterizar fenómenos, situaciones o eventos de manera detallada. El objetivo principal de este tipo de investigación es proporcionar una representación precisa y comprensiva de la realidad estudiada. En la investigación descriptiva, se recopilan datos a través de diversos métodos, como encuestas, cuestionarios, análisis de registros o documentos, entre otros. Estos datos permiten capturar la información relevante sobre el fenómeno en estudio y generar una descripción sistemática de sus características.

El objetivo principal del investigador es realizar una descripción minuciosa y precisa de situaciones y eventos, enfocándose en comprender cómo se presenta y se manifiesta un fenómeno determinado, al mismo tiempo que se lleva a cabo la medición de los conceptos asociados.

Tipo De Investigación Seleccionado

La elección del tipo de investigación descriptivo en lugar del explicativo o exploratorio para el desarrollo de un prototipo de gestión de ventas en la empresa Zapatitos Lindos S.A. se basa en los objetivos del proyecto. Los objetivos planteados, como el análisis detallado de los requerimientos técnicos y funcionales, el diseño de un prototipo funcional y la programación de los módulos necesarios, indican la necesidad de comprender y describir en profundidad la problemática específica de la empresa.

La investigación descriptiva permite recopilar información detallada de los involucrados en el proceso de ventas, lo que es fundamental para comprender los requisitos y necesidades del sistema propuesto. Además, este enfoque permite obtener una visión clara y precisa de los elementos que deben considerarse en el prototipo, lo cual es esencial para garantizar su eficiencia y escalabilidad. Al centrarse en la descripción detallada de la problemática y los requerimientos, la investigación descriptiva proporciona una base sólida para el diseño, la programación y las pruebas exhaustivas del prototipo. Así, se asegura que el sistema de gestión de ventas cumpla con los objetivos establecidos y resuelva de manera efectiva los desafíos identificados en la empresa.

Adicionalmente la investigación descriptiva se complementa con la metodología ágil Scrum que se utilizará en el desarrollo del prototipo; para lo que se pueden aprovechar los principios de entrega incremental y desarrollo iterativo para trabajar de manera eficiente y adaptativa. Cada sprint en Scrum se enfoca en la entrega de funcionalidades y características específicas, lo que se alinea perfectamente con la naturaleza descriptiva de la investigación. Durante cada sprint, se tiene la oportunidad de recopilar datos y obtener retroalimentación sobre las funcionalidades implementadas. Esta retroalimentación puede utilizarse para validar y ajustar la descripción del prototipo para la de gestión de ventas, ya que se obtiene una comprensión más profunda de las necesidades y requisitos de los usuarios.

Fuentes De Información

En la elaboración del Prototipo Funcional para la Gestión de Ventas de la empresa Zapatitos Lindos S.A, se hace necesario contar con diversas fuentes de información para obtener los datos y conocimientos necesarios. Entre las fuentes de información más relevantes se encuentran las fuentes primarias, secundarias y terciarias.

Las fuentes primarias hacen referencia a la información original y directa obtenida de primera mano, y en este caso, los colaboradores de la empresa Zapatitos Lindos serán la fuente principal de información. Sus experiencias, conocimientos y perspectivas serán fundamentales para comprender con detalle los requerimientos técnicos y funcionales del prototipo. Las fuentes secundarias, por otro lado, incluyen documentos, estudios previos, informes o bases de datos que han sido generados por terceros y que contienen información relevante relacionada con la gestión de ventas y el desarrollo de *software* en general. Por último, las fuentes terciarias son recursos de información que recopilan y sintetizan datos de fuentes secundarias, como enciclopedias, resúmenes, manuales o catálogos, y pueden proporcionar una visión panorámica y contextualizada del tema en cuestión.

La combinación de estas diferentes fuentes permitirá obtener una base sólida de información para la planificación y desarrollo del prototipo funcional, garantizando la integración de diversos puntos de vista y conocimientos relevantes para el éxito del proyecto. A continuación, se describirán los diferentes tipos de fuentes de información utilizadas para obtener conocimientos y datos relevantes en diversos ámbitos:

Fuente De Información Primaria

Las fuentes de información primaria son aquellas que proporcionan datos originales y directos sobre un tema o fenómeno de estudio; según Ruiz et al., (2008) “Contienen información original, que ha sido publicada por primera vez y que no ha sido filtrada, interpretada o evaluada por nadie más. Son producto de una investigación o de una actividad eminentemente creativa.” (p.2). Estas fuentes son creadas por testigos presenciales, participantes o investigadores que han recopilado datos de primera mano relacionados con el objeto de investigación. Las fuentes de información primaria son esenciales para obtener datos originales y fundamentales en la investigación. Estas fuentes proporcionan una base sólida para el análisis y la generación de conocimiento, ya que ofrecen información directa y sin intermediarios. Al utilizar fuentes de

información primaria, los investigadores pueden acceder a datos y detalles específicos que son cruciales para comprender y abordar adecuadamente el tema de estudio.

Fuente De Información Secundaria

Las fuentes secundarias de información son recursos que se generan a partir del análisis y procesamiento de las fuentes primarias. Estas fuentes implican el trabajo de especialistas que recopilan, organizan y presentan la información contenida en los documentos originales. Su objetivo es facilitar el acceso y la búsqueda de información para los investigadores. De acuerdo con Ruiz et al., (2008) estas fuentes

Contienen información primaria, sintetizada y reorganizada. Están especialmente diseñadas para facilitar y maximizar el acceso a las fuentes primarias o a sus contenidos. Componen la colección de referencia de la biblioteca y facilitan el control y el acceso a las fuentes primarias. (p.3)

En conclusión, las fuentes secundarias de información son herramientas que recopilan, organizan y presentan la información contenida en las fuentes primarias, facilitando así el acceso y la búsqueda de información para los investigadores. Estas fuentes desempeñan un papel importante al proporcionar una visión general de la literatura disponible y ayudar a los investigadores a encontrar los recursos pertinentes para sus estudios.

Fuente De Información Terciaria

Las fuentes terciarias de información cumplen un papel esencial al identificar, recopilar e indizar una amplia variedad de fuentes relacionadas con una disciplina o tema específico. Estas fuentes son especialmente útiles para localizar tanto fuentes secundarias como primarias, y ofrecen información verificada y respaldada por expertos en el campo. Tanto en bibliotecas físicas como en portales de revistas abiertos y bibliotecas electrónicas, se pueden encontrar estas fuentes terciarias autorizadas. Tal y como lo exponen Ruiz et al., (2008):

Son guías físicas o virtuales que contienen información sobre las fuentes secundarias. Forman parte de la colección de referencia de una biblioteca. Facilitan el control y acceso a toda la gama de repertorios de referencia, como las guías de obras de referencia, o a un solo tipo, como las bibliografías. (p.4)

Estas fuentes autorizadas no solo facilitan la ubicación de fuentes secundarias y primarias,

sino que también proporcionan información respaldada por expertos y son útiles para la orientación y definición de conceptos clave en la investigación.

Fuente De información utilizada

A lo largo de la investigación para el desarrollo del prototipo funcional destinado a la gestión de ventas de la empresa Zapatitos Lindos S.A., se emplearán diferentes tipos de fuentes de información, entre las cuales se destacan las primarias, secundarias y terciarias. La principal fuente de información se basará en la interacción directa con los colaboradores de la empresa, especialmente con la administradora y la gerente. A través de la observación, entrevistas, reuniones y análisis de datos proporcionados por ellos, se recopilará información detallada y específica sobre los procesos de venta y gestión de la empresa.

Además, se utilizarán fuentes secundarias como libros, revistas y otras investigaciones relacionadas con el proyecto. Estas fuentes secundarias proporcionarán un marco teórico y antecedentes que respaldarán el diseño y desarrollo del prototipo. Se examinarán publicaciones relevantes en el campo de la gestión de ventas, tecnologías aplicadas y prácticas recomendadas en la industria.

Por último, se hará uso de fuentes terciarias, como artículos y sitios web especializados. Estas fuentes de información terciaria serán presentadas en el cuerpo de la investigación y en la elaboración del proyecto. Los artículos citados y referenciados se considerarán ejemplos de información que ha sido previamente recopilada y analizada por otras fuentes. Estos datos extraídos serán categorizados y utilizados para respaldar y enriquecer los hallazgos obtenidos en la investigación.

Variables o Unidades de Análisis

En el contexto de un estudio de investigación, se utilizan diferentes tipos de variables que desempeñan roles específicos en el análisis de datos. Estas variables incluyen las variables operacionales, conceptuales e instrumentales. Según Parra, (2020)

En términos generales, una variable representa un atributo medible que cambia a lo largo de un experimento comprobando los resultados. Estos atributos cuentan con diferentes medidas, dependiendo tanto de las variables, del contexto del estudio o de los límites que los investigadores consideren. (párr.4).

La medición de las variables puede realizarse de diversas formas, dependiendo del contexto y objetivo del estudio. Cada variable seleccionada desempeña un papel crucial en el análisis de los datos y permite obtener conclusiones y respuestas a las preguntas de investigación planteadas. La elección de las variables y los métodos de medición adecuados es fundamental para garantizar la validez y confiabilidad de los resultados obtenidos en el estudio.

Variable Conceptual

Una variable conceptual se refiere a un concepto abstracto o idea que se investiga en un estudio. Estas variables se basan en definiciones teóricas y se utilizan para comprender y analizar fenómenos o aspectos específicos de interés. A diferencia de las variables operacionales, las variables conceptuales no se miden directamente, sino que representan conceptos teóricos que requieren una interpretación y comprensión profunda. De acuerdo con Córdova et al., (2019) “La definición conceptual es aquella que definen teóricamente las variables, a través de la abstracción científica y que se expresa en la definición de los términos básicos del marco teórico” (p.6). Para definir una variable conceptual, los investigadores pueden recurrir a diccionarios especializados o libros que describan la esencia y las características del concepto en cuestión. Estas definiciones proporcionan una base teórica sólida y establecen los fundamentos para el estudio.

Variable Operacional

Las variables operacionales son aquellas que se definen y se miden de manera concreta y observable en un estudio de investigación. Una variable operacional es aquella que indica el procedimiento que se va a realizar para entender un tema. A diferencia de las variables conceptuales, que son constructos abstractos, las variables operacionales se refieren a las medidas específicas que se utilizan para capturar y cuantificar los aspectos concretos de interés en una investigación. Según Córdova et al., (2019) “La definición operacional de las variables es el proceso a través del cual se establecen los procedimientos empíricos que permiten la obtención de datos de la realidad, para verificar las hipótesis y/o solucionar el problema.” (p.6)

Estas variables se definen mediante indicadores o medidas tangibles que permiten una recolección de datos objetiva y verificable. Los indicadores operacionales establecen los criterios concretos para la medición de las variables y especifican cómo se observarán o medirán en el contexto del estudio. Estas variables desempeñan un papel fundamental al proporcionar

organización y claridad en la creación de los instrumentos utilizados para recolectar información en una investigación. Estas variables permiten establecer indicadores concretos que garantizan la consistencia y confiabilidad de los datos recopilados, lo que facilita, así, el análisis y la interpretación de los resultados.

Variable Instrumental

Una variable instrumental es aquella que se utiliza como una herramienta o medio para medir o influir en una variable de interés en un estudio de investigación. Se introduce con el propósito de obtener datos más precisos o controlar el entorno experimental. Según menciona Galindo (2018):

La definición instrumental de las variables es también conocido como la tercera variable, de modo que es aquello en la que se aclara como se estudiará la variable que se acaba de definir, los medios o instrumentos para recoger la información. En mérito de ello, se deben definirse y elaborarse los instrumentos y medios con que se recolectará la información. Los instrumentos nacen de las variables y de los objetivos. Nunca deberá elaborarse un instrumento sin tener definida la variable o variables.

Estas variables se utilizan en la investigación para medir, controlar o manipular variables de interés y proporcionan un medio para obtener datos precisos y facilitar el estudio de los fenómenos investigados.

Tabla 6.

Variables

Objetivo específico.	Variable	Variable Conceptual	Variable Operacional	Variable Instrumental
<p>Analizar en detalle los requerimientos técnicos y funcionales para el prototipo propuesto, utilizando la información recopilada de los involucrados en el proceso de ventas de la empresa.</p>	<ul style="list-style-type: none"> - Requerimientos técnicos -Requerimientos funcionales. 	<ul style="list-style-type: none"> - Requerimientos técnicos: Los requisitos técnicos son los aspectos técnicos que deben cumplirse para completar con éxito un proyecto. Se trata de aspectos como el rendimiento, la fiabilidad y la disponibilidad. (wrike.com, s.f, párr.1) - Requerimientos funcionales: Un requisito funcional es una declaración de cómo debe comportarse un sistema. (Visure, 2022, párr.2) 	<ul style="list-style-type: none"> - Entrevistas. - Observación. - Reuniones virtuales con los colaboradores de la empresa. 	<ul style="list-style-type: none"> - Guía de Entrevista. - Guía de observación. - Sistema existente.
<p>Diseñar un prototipo funcional para la gestión de ventas de la empresa, que cumpla con los requerimientos específicos definidos, garantizando un sistema eficiente y escalable.</p>	<ul style="list-style-type: none"> - Prototipo funcional. - Diseño de software. 	<ul style="list-style-type: none"> - Los prototipos funcionales se construyen para probar la funcionalidad de un producto o diseño. No están tan pulidos como el producto final, pero permiten probar el diseño. (Román, 2023, párr.14) - El diseño de <i>software</i> es el proceso de construcción para la planificación de una solución informática. Este proceso ayuda a los programadores para que puedan manejar la complejidad que significan la mayoría 	<ul style="list-style-type: none"> - Diagramas. - Documentos de análisis. 	<ul style="list-style-type: none"> - Casos de Uso. - Draw.io.

		de los programas informáticos; esto además ayuda a disminuir el riesgo de errores a la hora de ser puestos en marcha. (Melanie, 2022, párr.1)		
Programar los módulos necesarios solventando la problemática que se presenta en la empresa y cumpliendo con los requerimientos solicitados.	- Módulos.	- En el mundo de la informática, un módulo es una porción de un programa de ordenador. Por lo general, un módulo es el encargado de resolver una tarea o grupo de tareas muy específicas, lo cual facilita el desarrollo de un programa cuando este es muy amplio. Muchas veces, a la hora de resolver un problema lo que se hace es dividirlo en subproblemas más sencillos. Así, cada subproblema estará encargado de resolver una parte del problema principal. (Álvarez, 2021, párr.1)	- Diagrama E-R. - Casos de uso.	- Visual Studio Community 2022. - Microsoft SQL Server 2017 Express
Probar de manera exhaustiva y sistemática el prototipo de gestión de ventas de la empresa, evaluando y determinando su correcto funcionamiento.	- Pruebas unitarias. - Evaluación del funcionamiento.	- Las pruebas unitarias son de muy bajo nivel y se realizan cerca de la fuente de la aplicación. Consisten en probar métodos y funciones individuales de las clases, componentes o módulos que usa tu <i>software</i> . (Pittet, s.f., párr.5) - La evaluación del funcionamiento de <i>software</i> es el proceso de analizar y medir el desempeño, la eficacia y la calidad del <i>software</i> en relación con los requisitos y expectativas.	- Pruebas unitarias. - Pruebas funcionales.	- Visual Studio Community 2022.

Fuente: Elaboración propia (2024).

Instrumentos

En una investigación, los instrumentos de recolección de datos desempeñan un papel fundamental al permitir obtener información relevante y precisa. En el caso de la investigación para el desarrollo de un prototipo funcional para la gestión de ventas de la empresa Zapatitos Lindos S.A., se emplearán diferentes instrumentos para recopilar los datos necesarios de manera rigurosa y completa.

En primer lugar, se utilizarán las entrevistas presenciales y virtuales como instrumentos clave de recolección de datos. Estas entrevistas proporcionarán la oportunidad de interactuar directamente con las dos personas que corresponden a la población seleccionada y que están involucradas directamente en el proceso de ventas de la empresa. A través de preguntas estructuradas o semiestructuradas, se obtendrá información detallada sobre los requerimientos técnicos y funcionales del prototipo propuesto. Las entrevistas presenciales permitirán una comunicación más directa y una lectura de lenguaje corporal, mientras que las entrevistas virtuales brindarán la flexibilidad de acceder a participantes ubicados en diferentes lugares geográficos.

Además, para complementar las entrevistas y obtener una visión más completa, se recurrirá a la observación como instrumento adicional de recolección de datos. A través de la observación, se registrarán de manera sistemática los comportamientos, interacciones y procesos relacionados con la gestión de ventas de Zapatitos Lindos S.A. Esto permitirá capturar información valiosa, como el flujo de trabajo, las interacciones entre el personal y los desafíos que surgen en el proceso de ventas. Esto sentará las bases para el diseño y desarrollo de un prototipo funcional que garantice una gestión eficiente y efectiva de las ventas en la empresa.

La elección de entrevistas presenciales y virtuales, junto con la observación, como instrumentos de recolección de datos en la investigación del prototipo funcional para la gestión de ventas de Zapatitos Lindos S.A., se justifica por el enfoque de investigación seleccionado el cual es cualitativo y descriptivo. Estos instrumentos permiten obtener datos ricos, detallados y contextualmente relevantes, los cuales son fundamentales para comprender y describir en profundidad los requerimientos técnicos y funcionales del prototipo propuesto.

Procesos Para La Recolección Y Análisis De Datos

El proceso para la recolección y análisis de datos necesarios para el desarrollo y ejecución de este proyecto se realiza principalmente mediante entrevistas a dos figuras clave en la empresa, la administradora y la gerente. La selección de estas dos personas se justifica porque poseen un conocimiento profundo del negocio, además, están a cargo de la toma de decisiones en la empresa; lo cual hace fundamental su participación en las entrevistas con el objetivo de poder identificar las áreas críticas y las inconsistencias en los procesos de gestión de venta actuales. Para garantizar la integridad y riqueza de los datos recolectados, se programan dos citas con cada uno de las entrevistadas; y se abrió la posibilidad de ampliar la información recolectada mediante las entrevistas virtuales.

Además de las entrevistas, se emplea un enfoque complementario para obtener información valiosa el cual es la observación directa. De las cuatro visitas presenciales se dedica tiempo específico para observar cómo los usuarios interactúan con el sistema actual de gestión de ventas. Esta observación detallada permite capturar situaciones cotidianas y desafíos que pueden no ser evidentes solo mediante la comunicación verbal.

Durante el proceso de recolección de datos, también se busca obtener una visión más amplia del negocio. Se procederá a recopilar la información correspondiente a cada uno de los módulos necesarios para el prototipo. Se realizará un estudio exhaustivo del proceso utilizado para llevar a cabo las tareas de cada requisito, con el objetivo de diseñarlos de manera óptima, adecuándolos a las operaciones de la empresa de la mejor manera posible. Se indaga sobre aspectos específicos, como por ejemplo el mantenimiento de clientes, y cómo la empresa debe cumplir con requisitos específicos, como los necesarios para facturar a la Caja Costarricense del Seguro Social.

La privacidad y confidencialidad son valores fundamentales en todo el proceso. Por lo tanto, se solicita el permiso explícito de la gerente de Zapatitos Lindos S.A. antes de utilizar cualquier información obtenida durante la observación o las entrevistas.

El conjunto integral de datos recopilados en este proceso de investigación permite formar una base sólida para el desarrollo del prototipo funcional. A partir de estos datos, se puede diseñar una solución adaptada a las necesidades y particularidades de la empresa, abordando las áreas críticas identificadas y buscando optimizar la gestión de ventas de Zapatitos Lindos en su totalidad.

CAPITULO IV: ANÁLISIS DE RESULTADOS

En este capítulo, en el cual se abordan los procesos de recolección y análisis de datos, en los que se emplearon dos herramientas fundamentales: la entrevista y la observación. Estos instrumentos permitieron la obtención de información crucial para delinear con precisión los requerimientos funcionales y técnicos que moldearán el prototipo funcional. Esta estrategia proporcionó una visión completa de las necesidades intrínsecas de Zapaticos Lindos S.A. Siguiendo esta línea de pensamiento, la entrevista emerge como un pilar fundamental para lograr los objetivos del tema propuesto. Al expandir esta perspectiva y aplicarla a este proyecto; esta herramienta se convierte en esencial para comprender las necesidades genuinas, obtener una base esencial que aborde las distintas problemáticas concretamente y así, poder dar forma a un prototipo funcional que las resuelva de manera precisa.

No obstante, esta fase no se restringe únicamente a la realización exitosa de entrevistas, sino que también exige una perspicacia analítica y la capacidad de observación para captar y comprender las necesidades fundamentales de la empresa y sus colaboradores. Por esta razón, se incorpora el instrumento de recolección denominado observación. Los estudios observacionales poseen un significado relevante, ya que se centran en la observación pasiva de los individuos, sin requerir su participación activa. Esta perspectiva permite examinar sus acciones, pensamientos y emociones con detenimiento, posibilitando la extracción de conclusiones holísticas y comprensivas sobre sus necesidades, sin depender exclusivamente de su relato directo. Con base en esto se obtuvieron y analizaron los siguientes resultados.

Entrevista

La entrevista se enfoca en comprender el proceso actual de gestión de ventas y las operaciones asociadas, tales como el control de inventarios, el registro de productos en inventario, la administración de cuentas por cobrar y cuentas por pagar, así como el seguimiento de ingresos y gastos. Un aspecto distintivo de la empresa es su capacidad para llevar a cabo órdenes de trabajo, un proceso a través del cual se adaptan o modifican zapatos ortopédicos para cumplir con los requisitos médicos específicos del cliente. Estas órdenes de trabajo estarán asociadas al cliente Caja Costarricense del Seguro Social y a un paciente el cual será el cliente final.

Además, se busca detallar minuciosamente los desafíos actuales en diversos procesos y definir los requisitos funcionales destinados al prototipo web, con base en las necesidades identificadas en la empresa.

Con el objetivo de analizar en detalle los requerimientos técnicos y funcionales para el desarrollo del prototipo propuesto; se realizó la entrevista a la señora Kathleen Wilson Mora quien funge como gerente general de la empresa Zapatitos Lindos S.A. y a la señora Ana Cristina Chaves que ostenta el puesto de administradora. La entrevista aplicada fue del tipo semi estructurada, ya que en su enfoque se combinaron preguntas predefinidas con la flexibilidad de poder explorar los temas de manera más abierta durante la conversación. Las líneas de investigación resultantes de las preguntas no contenidas en la guía de la entrevista permitieron abordar y detallar los desafíos que enfrenta la empresa respecto a la gestión de ventas y el uso actual del sistema informático con el que la empresa cuenta. Además, permitió explorar áreas que no estaban previamente contempladas gracias al contexto que proporcionaron las entrevistadas y a la profundidad que se les dio a temas específicos; permitiendo que la conversación se desarrollara más allá de las preguntas predefinidas.

Posterior a la aplicación de las entrevistas se acordó con las encargadas una serie de reuniones virtuales con el objetivo de poder ampliar o aclarar dudas en torno al desarrollo del proyecto.

A continuación, se interpretan y muestran los resultados obtenidos en la entrevista:

La aplicación de la entrevista permitió conocer que actualmente la empresa opera con un sistema informático que, lamentablemente, no se encuentra completamente funcional debido a diversas inconsistencias que presenta. La empresa Zapatitos Lindos opera actualmente con dos sucursales ubicadas en diferentes lugares geográficos. A pesar de esta dispersión geográfica, ambas sucursales operan bajo la misma razón social y comparten información crucial; como detalles de productos, datos de clientes y proveedores, así como el proceso de adaptación y modificación de calzado. Ante esta necesidad la base de datos del sistema actual se trasladó a la nube sin embargo el sistema actual carece de las características de un sistema web lo que incremento los problemas y las inconsistencias en los datos. Con esta información se pudo inferir que la empresa Zapatitos Lindos requiere que el prototipo funcional a desarrollar sea un sistema del tipo web con la

capacidad de utilizarse desde cualquier dispositivo y desde cualquier ubicación geográfica y que además les permita compartir datos entre las sucursales.

Durante la entrevista la gerencia sugirió continuar con el proveedor de *hosting* actual CleverDot.com; sin embargo la decisión de no continuar con este servicio para la implementación del prototipo funcional se sustenta en consultas técnicas y que actualmente el proveedor no cuenta con el soporte necesario para la tecnología Asp Net Core. Además, cabe mencionar que su servicio de *hosting* solo es compatible con bases de datos MySQL y el prototipo funcional utilizará una base de datos relacional del tipo Microsoft SQL Server .

La segunda pregunta enmarco la problemática que aqueja a la empresa, ante la pregunta las dos personas entrevistadas coincidieron en que las inconsistencias de datos y errores del sistema actual tienen su origen en una serie de modificaciones realizadas para poder habilitar con la segunda sucursal de la empresa y adicional a esto también se realizaron cambios para cumplir con obligatoriedad de la emisión de facturas electrónicas en el país. Ambas personas relataron experiencias similares en cuando al uso del sistema e hicieron énfasis en la falta de confianza con respecto a los datos que genera el sistema informático actual. Se coincide en la calificación de “deficiente” para el proceso de facturación actual, señalando demoras en su ejecución, un aumento sustancial en la carga administrativa y la pérdida de recursos y tiempo valiosos al momento de facturar. El sistema actual carece de una arquitectura web y enfrenta desafíos en la conectividad y portabilidad; el ingreso a los módulos requiere de un tiempo considerable de espera y sumado a esto el tiempo de inactividad es sorprendentemente breve; un minuto sin estar utilizando el sistema provoca problemas de conexión, obligando a los usuarios a salir del sistema por completo y volver a ingresar. Este problema se agrava por la inestabilidad ocasional de la conexión a internet, lo que conlleva desconexiones frecuentes entre el sistema local y la base de datos en la nube.

En lo que respecta al proceso de facturación, se han detectado incongruencias en los cálculos de impuestos previos a la emisión de las facturas. Sin embargo, esto no se presenta con frecuencia debido a que la mayoría de los artículos están exentos de impuestos debido a la actividad económica principal de la empresa que se enfoca en la fabricación y comercialización de zapatos ortopédicos; únicamente la mano de obra en las adaptaciones la cual llaman “RP” está sujeta a una tarifa de impuesto del trece por ciento.

A nivel transaccional, el sistema también exhibe varias inconsistencias, la más evidente, son los problemas de conexión, se presentan con frecuencia y cuando ocurren en medio de una

operación transaccional; la base de datos no retrocede a un estado válido, y no se revierten las acciones realizadas previas al error; incumpliendo con el término ACID el cuál se refiere al conjunto de propiedades que garantizan la fiabilidad y la integridad de la transacciones en sistemas que utilizan bases de datos relacionales. Se garantizará que el prototipo funcional maneje correctamente los errores y las transacciones, devolviendo los datos a su estado original en caso de que ocurra un error.

En relación con el control de inventarios, tanto la administradora como la gerente de la empresa señalan que es difícil lograr que las cantidades físicas y las registradas en el sistema coincidan. Las discrepancias requieren tomas físicas regulares y ajustes constantes en el sistema. Estas diferencias se centran principalmente en el módulo de anulación de facturas; proceso que no reintegra las unidades al inventario de manera precisa y carece de la capacidad de realizar anulaciones parciales. Con base en esta información la propuesta de prototipo funcional ofrecerá la anulación completa de factura y la devolución parcial o por línea, y se implementaran acciones de control para prevenir y controlar las inconsistencias de inventario.

En el ámbito de la facturación, el sistema actual permite facturar tanto a crédito como al contado. La gestión de las cuentas por cobrar se realiza de manera manual, lo que resulta en pérdida de dinero y falta de visibilidad financiera. Adicionalmente, la empresa carece de un sistema que facilite la gestión de compras de materias primas a crédito, lo que lleva a procesos manuales. Esto conlleva una falta de visibilidad financiera y pérdida de tiempo en la generación de los insumos a requeridos por el contador y así poder llevar el control de ingresos y gastos de la empresa.

A manera de conclusión en lo que respecta al proceso de ventas, el 80% de estas involucran procesos de adaptación o modificación de calzado. Esta funcionalidad se implementó en el sistema actual en una sección que llaman “apartados” la cual no estaba destinada para dicho proceso, pero que logró solventar parcialmente la necesidad. La administración de la empresa espera que el prototipo funcional propuesto marque un cambio significativo en la eficiencia y los tiempos de los diversos módulos y procesos del sistema. Anhelan una solución ágil y precisa que aporte orden y control a la empresa, junto con los módulos y la funcionalidad necesarios para superar la problemática actual.

La gerencia y administran desean agregar un mayor control al proceso de facturación y anulación. Actualmente, para emitir la factura electrónica se utiliza un tercero, lo que no es funcional; cuando se emite una factura, deben ingresar al correo electrónico e imprimir el

documento electrónico generado por el tercero. Posteriormente anotan en el papel el número de factura del sistema local y la documentación se guarda en un archivo físico para poder establecer la relación entre las numeraciones, lo que se concluye en proceso verdaderamente tedioso, por lo que el prototipo realizará la emisión de comprobantes electrónicos sin necesidad de un tercero y podrá llevar un solo consecutivo que estará impreso en la facturas y que se podrá leer de manera ágil a través de la implementación del código QR que se mostrará en la representación gráfica de los comprobantes electrónicos; el consecutivo y la clave numérica de los documentos también se podrán visualizar en el archivo pdf.

A partir de la información recopilada, se concluye que la empresa requiere con urgencia un sistema que restaure la confianza en los datos, especialmente en lo que respecta al control de inventarios y transacciones en general. Es imperativo que todas las transacciones cumplan con las propiedades ACID ("Atomicidad, Consistencia, Aislamiento, Durabilidad"), ya que estas garantizan la integridad y coherencia de los datos. Esto asegura que las operaciones en la base de datos sean fiables, incluso en entornos con múltiples usuarios concurrentes y posibles fallos del sistema.

En cuanto a la funcionalidad con la que se cuenta, cabe resaltar que el sistema actual utilizado por Zapatitos Lindos S.A. proporciona una serie de funciones de mantenimiento esenciales para la gestión de información utilizada en los procesos de facturación, inventario y compras, los cuales formaran parte de la propuesta presentada. Estos procesos se citan a continuación

1. **Mantenimiento de áreas:** este módulo permite la creación de categorías para clasificar productos. Seguidamente, se ejemplifica una categorización posible: Zapatos ortopédicos, Sandalia ortopédica, Pre caminador formativo, Calzado formativo. Para efectos del prototipo propuesto en este proyecto este mantenimiento se le cambiara el nombre a Categorías a solicitud de la encargadas.
2. **Mantenimiento de inventarios:** este módulo permite la administración del catálogo de productos e incluye los siguientes campos que son indispensables: código de área, código del producto, código CABys, descripción del producto, precio, porcentaje de utilidad, costo, estado (Activo o Inactivo), unidad de medida, impuesto, tarifa de impuesto, control de existencias (sí/no), permite cambiar precios (sí/no). En este módulo, se ha expresado el interés en categorizar productos como servicios; ya que actualmente los ítems de servicio no se registran correctamente en las facturas electrónicas de venta. Otra de la funcionalidades solicitadas es la posibilidad de agregar a los productos un segundo y tercer precio, el cual se utilizaría como

precio final del producto para un cliente con un perfil específico, para ejemplificar el caso un zapato tendría asociado hasta tres precios, precio uno correspondería al precio de venta general para el cliente directo y el precio dos se le asocia al cliente con el perfil Caja Costarricense social y el precio tres a los clientes llamados distribuidores, aquellos que compran para revender. Por último, se desea incorporar un estado adicional que sirva para identificar cuando un producto o modelo ha sido descontinuado. Es crucial destacar que es de suma importancia que los productos en estado no activo no sean visibles en las pantallas principales utilizadas para los procesos de facturación, compras, inventario y otros módulos por desarrollar; como el de órdenes de trabajo. Para efectos el prototipo propuesta se le llamara “Mantenimiento Artículos”.

3. **Mantenimientos de tiendas:** actualmente el mantenimiento de tiendas permite la creación clientes distribuidores, clientes directos y el cliente Caja Costarricense del Seguro Social (CCSS). El tipo de cliente tiendas o “distribuidores” tiene la opción de facturar a crédito y, además, se beneficia de un descuento global del 10% en las facturas de venta. El cliente CCSS y cliente directo engloban a aquellas personas físicas o jurídicas con la única diferencia que al cliente directo no se le ingresa un campo denomina unidad ejecutora que se usa para diferenciar a los cliente CCSS. Para el desarrollo del prototipo funcional, se le cambian el nombre a mantenimiento de clientes, se mantienen los datos que se capturan en esta pantalla y se agregan otros los cuales son necesarios para la creación de los documentos electrónicos. Adicional se creará un módulo de perfil de clientes al cual tendrá acceso únicamente el usuario con el rol administrador con el objetivo de parametrizar cuales son los campos requeridos por cada tipo de cliente y también asociar una lista de precio específica, así como un descuento como es el caso de los distribuidores que les aplica un 10%. Los usuarios con el rol estándar podrán consultar y crear clientes siempre y cuando no sean de crédito.
4. **Mantenimientos de proveedores:** el presente módulo se encarga de gestionar la información relacionada con los proveedores. Actualmente, se caracteriza por su simplicidad al enfocarse principalmente en la recopilación de datos esenciales, como el nombre del proveedor y el nombre del contacto. Es importante destacar que este módulo se ha limitado en su funcionalidad, ya que la empresa no cuenta con un sistema de gestión de cuentas por pagar. Su uso se restringe únicamente al registro de compras y la generación de informes de compra para la tienda. En el prototipo funcional propuesto, se sugiere la captura de información adicional

del proveedor, como la cédula jurídica, la dirección, los datos de contacto, números de teléfono, correo electrónico, el tipo de proveedor ya sea "Crédito" o "Contado" y los plazos de pago. Esta ampliación de información sería de gran utilidad para facilitar la gestión de cuentas por pagar.

5. **Módulo de facturación:** La empresa Zapatitos Lindos S.A. dispone de dos tipos de facturación:
 - a. Facturación de crédito: Este método se utiliza para generar la facturación de las Tiendas "Distribuidores" y la CCSS.
 - b. Facturación de contado: Este proceso se emplea para generar la facturación de los clientes que realizan compras directamente en la tienda sin necesidad de crédito. Es importante destacar que el sistema actual con el que cuenta la empresa no incluye una funcionalidad de facturación electrónica dentro de la solución. Para abordar esta necesidad, implementaron una solución que utiliza el API de la empresa GTI. Este API es una herramienta que permite la comunicación entre el sistema actual de la empresa Zapatitos Lindos S.A y el sistema de facturación electrónica de GTI, lo que hace posible la generación de facturas electrónicas. Sin embargo, según lo indicado por las dos personas entrevistadas, esta integración con el API de GTI presenta algunas limitaciones.
6. **Módulo de Apartados:** este módulo está diseñado para simular el proceso de órdenes de trabajo, específicamente para realizar adaptaciones o modificaciones en el calzado. No obstante, es importante destacar que este módulo presenta varias limitaciones significativas por lo es necesario para mejorar el control de este proceso y abordar las deficiencias identificadas. Uno de los aspectos clave es la necesidad de controlar de manera más efectiva el inventario utilizado como insumo para las operaciones de modificación y adaptación del calzado. La revisión y mejora de este módulo son esenciales para garantizar una gestión más eficiente y precisa de las órdenes de trabajo, así como para optimizar el manejo de inventario en este contexto. En la propuesta presentada este módulo se conoce como el proceso para Generar órdenes de trabajo.
7. **Módulo de compras:** el sistema actual cuenta con un módulo de compras cuya única funcionalidad es el ingreso de existencias al inventario, el prototipo contempla un control de la existencia registrando los ingresos de mercadería en un Kardex o bitácora de transacciones, si la compra es de crédito se registra la cuenta por pagar y adicionalmente se incorpora el

asiento contable y el registro del costo promedio ponderado como método de valuación del inventario.

8. **Módulo de anulación:** este módulo actualmente facilita la generación de notas de crédito completas en relación con las facturas de venta. No obstante, presenta la limitación de no permitir anulaciones parciales de línea, lo que ha sido identificado como una de las principales causas de las inconsistencias en el inventario.

En el contexto del prototipo propuesto para el desarrollo, se plantea la creación de un módulo de anulaciones completamente reestructurado. Este módulo tendría la capacidad de generar notas de crédito electrónicas tanto completas como parciales. Además, estaría diseñado para integrarse de manera eficiente con los sistemas de inventario, cuentas por cobrar y el control de ingresos y gastos. La implementación de este nuevo módulo se prevé como una solución integral para abordar las dificultades asociadas con la anulación de transacciones y contribuir a la mejora de la precisión en el control de inventarios.

El análisis exhaustivo del sistema actual de Zapaticos Lindos S.A. ha identificado áreas críticas que requieren mejoras significativas en los módulos anteriormente listados los cuales contemplan mantenimientos, y procesos de facturación, inventario, compras y anulaciones.

Para una mayor comprensión de la gestión de ventas de la empresa las dos personas entrevistadas explicaron el proceso de ventas en Zapaticos Lindos S.A. de la siguiente manera: la venta se inicia cuando un cliente visita la zapatería, ya sea con o sin receta médica, en busca de calzado especializado para sus necesidades. Si el cliente es directo y la venta no requiere modificaciones ni adaptaciones, se procede a emitir una factura de venta en el sistema, utilizando la opción de facturación de contado. En este punto, el personal de ventas consulta al cliente sobre su preferencia de factura electrónica, y si no la requiere, se emite un tiquete electrónico. Además, se brinda la posibilidad de registrar datos básicos del cliente, como su cédula, nombre y teléfono, información que podría ser valiosa para mejorar la relación con el cliente, llevar a cabo campañas de mercadeo y proporcionar un seguimiento personalizado que fomente una experiencia de compra satisfactoria y la fidelización del cliente.

En caso de que los clientes presenten una receta médica y necesiten adaptaciones o modificaciones en el calzado, se procede a generar un apartado; concepto que se utiliza para simular la gestión de las órdenes de trabajo. Una vez completada la orden de trabajo, se emite la factura de venta correspondiente. El proceso de apartados se utiliza principalmente en las

transacciones con el cliente Caja Costarricense del Seguro Social (CCSS). Durante la gestión de apartados, se registra información relevante sobre las adaptaciones realizadas en la plantilla o suela del calzado, así como cualquier otra modificación necesaria. Sin embargo, se enfrenta a limitaciones importantes, como la falta de control sobre los insumos utilizados en las adaptaciones, lo que resulta en un desconocimiento de la cantidad de materia prima utilizada. Además, el inventario no se compromete adecuadamente, lo que significa que los productos pueden aparecer como disponibles para la venta en el sistema, a pesar de estar asignados a una orden de trabajo en proceso. Todas las órdenes de trabajo para la CCSS deben incluir información referente a la clínica u hospital a la cual va dirigida la factura electrónica de venta. También se requiere información del paciente y en el proceso de creación los usuarios pueden ingresar descripciones y precios manualmente. Las encargadas de la empresa Zapatitos han solicitado que el prototipo propuesto pueda obtener automáticamente los datos de la receta cuando se asocie el paciente ya que esta información se digita en el mantenimiento de pacientes. Además, se solicitó ampliar la cantidad de caracteres permitidos en los campos relacionados con la plantilla y la suela.

Por lo que la implementación del nuevo módulo propuesto tiene el potencial de mejorar significativamente este proceso al agilizarlo, proporcionar un mayor control sobre las adaptaciones, gestionar adecuadamente los insumos utilizados y corregir las limitaciones identificadas en el proceso actual.

En relación con el catálogo de artículos la entrevista realizada se proporcionó una visión general de los diferentes ítems que lo conforman; estos se detallan a continuación:

1. Código RP: corresponde a los servicios de reparación de calzados y artículos de cuero, para efectos de la factura electrónica se utiliza el código CABys 8721000000000. De acuerdo con el catálogo CABys este ítem es un servicio y se debe pagar el 13% IVA.
2. Productos Ortopédicos: el catálogo de productos comprende varios modelos de zapatos y otros tipos de productos que se engloban en la categoría de productos ortopédicos.

A continuación, algunos ejemplos:

- a. Calzado para diabéticos, producto exento, Código CAByS 4817203010000
- b. Faja sacrolumbar, de uso ortopédico o de rehabilitación, producto exento, Código CABys 4817105030100.
- c. Calzado para artrítico, producto exento, código CABys 4817203020000.

- d. Calzado y artículos similares, de uso médico u ortopédico, productos exentos, código CABys 4817203990000.
- e. Artículo diferencia de precio: este artículo se utiliza para cobrarle al cliente la diferencia de precio entre el monto que subsidia la CCSS y el monto final de la factura.
- f. Otros: a nivel de sistema se tienen otros códigos que no se facturan, no afectan el inventario y cuyo uso es únicamente es para la generación de facturas proformas.

Conocer los principales artículos gestionados en el sistema de Zapatitos Lindos S.A. resulta fundamental en el contexto del proyecto de desarrollo de un prototipo funcional para la gestión de ventas de la empresa. Esta información proporciona una visión detallada de la diversidad de productos que se manejan en la empresa, lo que permitirá una mejor organización y estructuración de la base de datos en el nuevo sistema. Además, el conocimiento de los códigos asociados a cada tipo de producto facilitará la correcta facturación electrónica y la gestión de impuestos, lo que contribuirá a la eficiencia en los procesos de ventas y a la adaptación de la empresa a las regulaciones fiscales. Por último, al identificar productos exentos de impuestos o sujetos a tasas específicas, el nuevo sistema podrá generar automáticamente los documentos fiscales adecuados, lo que simplificará el cumplimiento tributario y mejorará la precisión en la facturación. Toda esta información es esencial para diseñar un sistema de gestión de ventas efectivo y cumplir con las necesidades y regulaciones de Zapatitos Lindos S.A.

Durante el proceso de la entrevista surgió el tema del código de artículo el cual denominan diferencia de precio para abordar este tema las personas entrevistadas indicaron que el termino diferencia de precio es el costo que asume el cliente en el proceso de venta cuando la venta se le realiza al cliente Caja Costarricense del Seguro Social (CCSS) y al respecto se brindó el siguiente ejemplo: el par de zapatos para adultos tiene un precio de ₡76,000 colones, pero la CCSS solo subsidia ₡61,756.42 colones de ese precio. El cliente, en consecuencia, paga únicamente la diferencia entre el costo real del zapato y el monto subsidiado por la CCSS.

A partir de la información recolectada es esencial que el sistema a desarrollar cuente con controles sólidos para prevenir la facturación negativa y la facturación por debajo del costo de los productos. Esto garantiza que las transacciones se realicen de manera coherente con los principios comerciales y financieros de la empresa.

En cuanto al control de las cuentas por cobrar, el hecho de llevar un registro en Excel y comparar los pagos recibidos con los datos del sistema demuestra un compromiso con el

seguimiento y la conciliación de cuentas. Sin embargo, la complejidad de la identificación de las facturas y la falta de coincidencia entre los números de factura electrónica y el sistema subrayan la importancia de una comunicación eficaz y una integración más sólida entre los sistemas de facturación electrónica y el sistema de gestión interno.

El manejo efectivo de las cuentas por cobrar requiere un equilibrio entre mantener relaciones sólidas con los clientes, adaptarse a las prácticas de pago individuales y tener un sistema sólido de seguimiento y control. Además, la integración de sistemas y una comunicación clara son esenciales para garantizar una gestión eficiente y precisa de las cuentas por cobrar, lo que contribuye a la estabilidad financiera y al flujo de efectivo de la empresa.

En cuanto a la contabilidad, es relevante destacar que la empresa carece de un sistema que le permita gestionar sus ingresos y gastos de manera eficiente. Además, en lo que respecta a los servicios de contabilidad, estos han sido subcontratados, y según lo informado por las encargadas, la persona contratada lleva a cabo el control de manera manual utilizando archivos de Excel.

En el cierre de cada mes, la Administradora de la empresa recopila la información necesaria para el proceso contable y la envía por correo electrónico al contador. Actualmente la empresa no cuenta con la información actualizada de los estados financieros lo que no les da una visibilidad y se encuentran en el proceso de subcontratar otro contador.

El prototipo propuesto incluirá un módulo de control de ingresos y gastos que se integrara con los módulos que realizan procesos de ventas y gestión del inventario y permitirá manejar un catálogo contable, realizar asientos diarios y generar el reporte de asientos de diarios y la comprobación general de los asientos.

Observación

En el ámbito de la investigación, la observación se presenta como un recurso de gran valor que posibilita el análisis de la realidad que se desea estudiar. Este enfoque implica sumergirse en el entorno objeto de estudio, lo cual permite la contemplación de los fenómenos y procesos en su estado natural. Esta inmersión ofrece la oportunidad de registrar minuciosamente detalles, patrones de comportamiento y dinámicas que, en ocasiones, escapan a un análisis superficial. En consecuencia, la observación se posiciona como una herramienta esencial para descifrar la complejidad de la realidad, brindando una base sólida sobre la cual fundamentar análisis más profundos y respuestas respaldadas en el contexto de la investigación.

Con el objetivo de analizar exhaustivamente los requerimientos funcionales necesarios para la implementación exitosa del prototipo propuesto; se llevó a cabo un proceso de observación minuciosa y detallada, en conjunto con entrevistas a los actores clave involucrados en el proceso de ventas.

Durante el proceso de observación en el lugar de trabajo, se identificaron una serie de desafíos y problemas, estos proporcionaron una visión profunda de las áreas que requerían atención inmediata y mejora. Uno de los hallazgos más destacados fue la dificultad experimentada en la emisión de facturas a los clientes. Se detectaron problemas técnicos relacionados con la desconexión frecuente del sistema. Esta problemática no solo obligaba a los empleados a salir y reiniciar el sistema, sino que también prolongaba de manera innecesaria el tiempo de inactividad debido a la lentitud en el proceso de reconexión. Algunos empleados optaban por reiniciar por completo las máquinas, cuando el sistema no respondía, en lugar de emplear métodos más eficientes, como el acceso al administrador de tareas para cancelar procesos problemáticos.

En lo que respecta al control de inventario, la observación puso de manifiesto discrepancias notables entre los registros del sistema y la cantidad real de productos en *stock*. Esta falta de concordancia generaba ineficiencias significativas y dificultades en la gestión de inventario, lo que, a su vez, impactaba la capacidad de la empresa para atender las demandas de los clientes de manera eficiente.

Además, se observaron procesos manuales complejos en la gestión de la bodega y las requisiciones. Estos procesos requerían la aprobación de un administrador y se llevaban a cabo en papel, lo que resultaba en una pérdida de tiempo y en la desconfianza en la precisión de los registros.

La observación también reveló problemas relacionados con la conectividad a Internet. Cuando se producían interrupciones en la conexión, los empleados se veían obligados a compartir la conexión de Internet de sus teléfonos móviles como medida de contingencia. Sin embargo, el sistema actual no estaba diseñado para funcionar de manera eficiente con conexiones limitadas, de ahí las dificultades adicionales y pérdida de productividad.

En el proceso de emisión de facturas después de la preventa, se identificó un retraso de hasta cinco minutos, lo que tenía un impacto negativo en la eficiencia de las ventas y en la calidad del servicio al cliente.

Estos hallazgos desempeñarán un papel fundamental en el diseño del prototipo funcional para la gestión de ventas, por lo que se asegurará que el prototipo funcional resultante aborde de manera efectiva las deficiencias identificadas y cumpla con los requerimientos técnicos y funcionales específicos, a la vez que garantice un sistema eficiente y escalable para la empresa.

Análisis Integral

En el contexto de la investigación realizada en el capítulo mencionado, se aplicaron dos técnicas fundamentales para recopilar información vital destinada al diseño del prototipo de gestión de ventas de Zapatitos Lindos S.A. Estas técnicas fueron la entrevista y la observación, ambas contribuyeron de manera complementaria al logro de los objetivos planteados.

La entrevista, dirigida a la gerente general y a la administradora de la empresa, se llevó a cabo de manera semiestructurada. Este enfoque permitió no solo obtener respuestas a preguntas predefinidas, sino también explorar temas emergentes y detallar los desafíos operativos actuales. Se obtuvo información valiosa sobre los procesos de ventas, control de inventarios, gestión de cuentas y órdenes de trabajo, lo cual proporcionó una visión completa de las operaciones de la empresa y sus requisitos funcionales.

La observación, por su parte, brindó una visión directa y sin filtros de la realidad operativa de Zapatitos Lindos S.A. Este enfoque permitió identificar desafíos adicionales que podrían no haber sido evidentes en las entrevistas. Los problemas técnicos recurrentes en el sistema informático, las discrepancias en el control de inventario, los procesos manuales complejos y otros obstáculos operativos se identificaron mediante esta técnica. La observación permitió capturar detalles, patrones y dinámicas que contribuyeron significativamente a comprender la complejidad de la situación.

En conjunto, la información recopilada a través de la entrevista y la observación se convierte en un recurso integral y enriquecedor para el proyecto. Estos datos se alinean directamente con los objetivos del proyecto, que incluyen el análisis detallado de los requerimientos técnicos y funcionales, el diseño de un prototipo eficiente y escalable, la programación de módulos necesarios y pruebas exhaustivas para garantizar su correcto funcionamiento. Lo que será esencial para el diseño y desarrollo exitoso del prototipo funcional para la gestión de ventas de la empresa Zapatitos Lindos S.A., ubicada en San José.

CAPITULO V: CONCLUSIONES Y RECOMENDACIONES

Conclusiones

El proyecto dirigido al desarrollo de un prototipo funcional para la gestión de ventas de la empresa Zapatitos Lindos S.A., ubicada en San José, se ha desarrollado con éxito, arrojando resultados de gran importancia y relevancia. A través de la implementación de diversas estrategias de investigación y la utilización de dos instrumentos clave como la entrevista y la observación, se ha logrado avanzar significativamente hacia la consecución de los objetivos planteados.

El proceso de análisis de requerimientos técnicos y funcionales se enriqueció gracias a la colaboración activa y valiosa de la gerente general y la administradora de la empresa. El tipo de investigación cualitativa no solo permitió cumplir con los objetivos establecidos sino también descubrir desafíos previamente no identificados y aspectos de vital importancia.

El diseño del prototipo se basó en la creación de casos de uso, proporcionando una visión completa y minuciosa de la interacción de los actores con el sistema propuesto.

En la fase de programación, se utilizaron tecnologías de vanguardia, incluyendo Visual Studio 2022, ASP.NET Core MVC, C# y SQL Server, aprovechando la información detallada recopilada en etapas previas.

El resultado final de este proceso es un prototipo funcional altamente efectivo que aborda de manera integral los desafíos operativos de Zapatitos Lindos S.A. Cada uno de los módulos desarrollados ofrece soluciones concretas que elevan significativamente la eficiencia y calidad de la gestión de ventas en la empresa.

Además, las pruebas exhaustivas, garantizan la óptima operatividad del prototipo y su adhesión a los requerimientos predefinidos. Estas pruebas no solo validan las funcionalidades del sistema, sino que también anticipan cualquier posible impacto en futuras modificaciones.

Este proyecto representa un avance sustancial hacia la mejora de los procesos de ventas en Zapatitos Lindos S.A. La combinación de herramientas de investigación efectivas y enfoques de desarrollo sólidos ha culminado en un prototipo funcional altamente prometedor que tiene el potencial de abordar con éxito los desafíos identificados. Estas conclusiones subrayan el éxito alcanzado en la búsqueda de soluciones efectivas para la gestión de ventas de la empresa.

Recomendaciones

Basado en los resultados obtenidos de la investigación, se han identificado puntos clave para asegurar una implementación del prototipo funcional para la gestión de ventas de la empresa Zapatitos Lindos S.A.

A continuación, se detallan las recomendaciones para llevar a cabo este proceso de manera efectiva:

1. Se recomienda realizar pruebas funcionales del sistema. Para ello, es necesario contactar al contador de la empresa con el fin de obtener un usuario, una contraseña y el certificado digital de prueba. Estos datos se obtienen en la plataforma ATV del Ministerio de Hacienda de Costa Rica. La realización de estas pruebas permitirá certificar el correcto funcionamiento del prototipo y facilitará la familiarización con las funcionalidades de este.
 - Responsable: La administración de la zapatería.
 - Fecha de inicio sugerida: 01 de septiembre del 2024.
 - Fecha de finalización sugerida: 30 de septiembre del 2024.
2. Actualmente, el prototipo funcional se encuentra hospedado en el hosting SmarterASP.Net bajo el plan Premium y se encuentra pago hasta el 15 de septiembre del 2024, por lo que se recomienda extender el plan por un período mínimo de un año, lo cual garantiza la estabilidad y continuidad del servicio. El costo anual del plan de hosting es de 139.4 dólares según la página oficial del proveedor de hosting SmarterASP.Net al 15 de julio del 2024.
 - Responsable: La gerencia general de la zapatería.
 - Fecha de inicio sugerida: 01 de septiembre del 2024.
 - Fecha de finalización sugerida: 15 de septiembre del 2024.
3. Se recomienda contratar el plan de respaldos ofrecido por el proveedor de hosting, el cual tiene un costo de 35 dólares por año; costo tomado de la página oficial de SmarterASP.Net al 15 de julio del 2024. Esta acción garantizará la integridad y seguridad de los datos almacenados en el sistema, lo que es esencial para la continuidad de las operaciones de la empresa. En caso de pérdida de datos o problemas técnicos, la capacidad de recuperar la información respaldada minimizará el tiempo de inactividad, reducirá el riesgo de pérdida de información crítica y asegurará que las operaciones de la empresa puedan continuar sin interrupciones significativas.

- Responsable: La gerencia general de la zapatería.
 - Fecha de inicio sugerida: 01 de septiembre del 2024.
 - Fecha de finalización sugerida: 15 de septiembre del 2024.
4. Se recomienda no realizar carga de datos debido a que los datos actuales no son fiables. Ingresar la información desde cero en el prototipo funcional asegurará que la información en el nuevo sistema sea precisa, confiable y esté libre de errores históricos. Esta estrategia evita problemas de calidad de datos y reduce la necesidad de depuración y limpieza extensiva de datos antiguos, facilitando una transición más fluida y una mejor adaptación de los usuarios al nuevo sistema.
- Responsable: La administración de la zapatería.
 - Fecha de inicio sugerida: 15 de agosto del 2024.
 - Fecha de finalización sugerida: 01 de septiembre del 2024.
5. Se recomienda capacitar al personal de la zapatería en el uso del prototipo. Sin embargo, dado que dicha capacitación no forma parte del alcance del proyecto de investigación, debe considerarse como una actividad independiente. Una capacitación adecuada y bien planificada conducirá a una adopción más rápida y eficiente del nuevo sistema, mejorará significativamente la productividad, y reducirá los errores operativos derivados de la falta de familiaridad con el sistema.
- Responsable: La administración de la zapatería.
 - Fecha de inicio sugerida: 15 de septiembre del 2024.
 - Fecha de finalización sugerida: 20 de septiembre del 2024.
6. Se recomienda no extender el periodo de pruebas del sistema e iniciar la puesta en producción. Esto con el objetivo de empezar a ver los beneficios y valor entregado por el nuevo sistema. Para ejecutar esta acción se entrega un script de limpieza de tabla que elimina únicamente las transacciones de prueba manteniendo intacto los catálogos o datos maestros como clientes, productos, proveedores, pacientes etc. Adicionalmente deben editar los siguientes datos en el mantenimiento de compañías del proveedor: usuario hacienda, contraseña hacienda y certificado digital.

- Responsable: Soporte técnico de la zapatería.
 - Fecha de inicio sugerida: 01 de octubre del 2024.
 - Fecha de finalización sugerida: 01 de octubre del 2024.
7. Debido a que actualmente varios empleados comparten su identificador de usuario y contraseña se recomienda crear un usuario y contraseña único para toda aquella persona que vaya a hacer uso del sistema. No compartir usuarios en un sistema mejora la seguridad y la integridad de los datos al garantizar que cada persona tenga acceso solo a la información y funciones necesarias para su rol específico. Esto reduce el riesgo de errores o accesos no autorizados y asegura una mejor trazabilidad de acciones, lo que contribuye a un entorno de trabajo más seguro y eficiente.
- Responsable: La administración de la zapatería.
 - Fecha de inicio sugerida: 01 de septiembre del 2024.
 - Fecha de finalización sugerida: 15 de septiembre del 2024.
8. Finalmente, se recomienda enriquecer aún más este prototipo ampliando el alcance propuesto en cada uno de los módulos funcionales que fueron entregados. Algunas de las funcionalidades propuestas para enriquecer el prototipo son: aceptación de factura de compra y carga automática al sistema con el objetivo de no tener que digitar la información de las compras manualmente al sistema, calculo de importantes y más informes o reportes que apoyen la toma de decisiones en la empresa.
- Responsable: La gerencia general y soporte técnico de la zapatería.
 - Fecha de inicio sugerida: 01 de enero del 2025.
 - Fecha de finalización sugerida: 31 de diciembre del 2025.

Estas recomendaciones están diseñadas para optimizar la implementación del prototipo funcional y mejorar la eficiencia operativa de la empresa, brindando beneficios tangibles y sostenibles en términos de seguridad, costos, productividad y gestión de datos, asegurando así un crecimiento continuo y exitoso para Zapatitos Lindos S.A.

CAPITULO VI: PROPUESTA

En este capítulo convergen dos corrientes esenciales que definen el proyecto: la elaboración de un prototipo funcional destinado a potenciar la gestión de ventas en Zapatitos Lindos S.A., y la implementación de la metodología de desarrollo de software estratégicamente seleccionada. Aquí se revela con minuciosidad el camino completo, desde el análisis hasta las pruebas de ejecución, entrelazando diseño, programación y pruebas implementadas en una narrativa unificada.

Los elementos en esta etapa incluyen la presentación de casos de uso en forma de diagramas y tablas de descripción, un análisis detallado de los requerimientos de *hardware*, una exploración exhaustiva de las necesidades de telecomunicaciones, una descripción detallada de la estructura de la base de datos o del almacenamiento de la información y, no menos importante, una descripción minuciosa del personal requerido para operar y utilizar el sistema de manera eficaz.

En esta parte, los diagramas cobran vida, los análisis exploran *hardware* y comunicaciones, y las descripciones perfilan bases de datos. Conjuntamente dan forma a un prototipo que trasciende lo funcional, y cuya eficacia en cada detalle resalta.

Análisis

La definición precisa de los requerimientos analizados se vuelve fundamental mediante la utilización de casos de uso. Estos casos actúan como elementos determinantes para desentrañar y examinar las expectativas del usuario con respecto al sistema en desarrollo.

Análisis Detallado Del Software Desarrollado

El prototipo cuenta con 8 módulos que fueron definidos en el alcance del proyecto y que por medio del análisis de los resultados obtenidos de la entrevista y la observación, se les adicionaron y complementaron funcionalidades que permitirán solventar las inconsistencias y debilidades que la empresa enfrenta.

Seguidamente se explicará cada uno de ellos.

Módulo Generar Facturación

El módulo de facturación de la empresa Zapatitos Lindos S.A. se encargará de generar la facturación de contado y crédito. Permitirá emitir dos tipos de documentos electrónicos: factura y ticket; la diferencia principal entre ambos radica en que para emitir un ticket electrónico no es necesario indicar el cliente; ya que este tipo de comprobante no se puede utilizar como crédito fiscal, mientras que, para la factura electrónica, ya sea de contado o crédito, es obligatorio especificar el cliente al cual va dirigido. Desde el módulo de facturación se permitirá realizar una búsqueda de clientes y visualizar información básica como: el tipo de identificación, número de identificación, nombre, si el cliente cuenta con crédito, si es un cliente exonerado y el plazo en caso de ser un cliente de crédito. Desde la pantalla de búsqueda se pueden filtrar los datos por identificación o por nombre del cliente, este filtro permite una búsqueda más eficiente y solo se muestran clientes con el estado “Activo” en el sistema; el usuario podrá seleccionar un registro de la búsqueda y cargar la información de este en la pantalla de facturación en donde podrá visualizar los datos como: tipo de identificación, identificación, nombre, porcentaje de descuento global, plazo de crédito, teléfono, fax, provincia, cantón, distrito, barrio, dirección y los datos de exoneración en caso de que cuente con el perfil de cliente exonerado; estos datos son: tipo de documento de exoneración, el número de documento, la institución que emitió el documento, la fecha de exoneración y el porcentaje de exoneración. El prototipo propuesto no limita que el cliente se encuentre registrado para poder emitir una facturación de contado, se podría emitir una factura o ticket electrónicos seleccionado el cliente de contado y digitando la identificación y nombre, también se provee de un padrón de identificaciones para buscar el nombre del cliente por identificación sin tener que estar registrado en el sistema ya que utilizan un servicio de consulta de identificaciones provisto por el Ministerio de hacienda.

El módulo también permite consultar los artículos, visualizar su saldo en bodega y sus precios, lo que facilitará el proceso de facturación. Adicionalmente desde este módulo se podrán facturar artículos, servicios y conceptos. Los artículos son aquellos que afectan directamente el inventario; los servicios y conceptos no afectan existencias en bodegas. El término “concepto” se utiliza para aquellos casos en los que el ítem por facturar no se puede clasificar como un servicio y además no debe afectar el inventario; un ejemplo de este último es la diferencia de precio cobrada al cliente cuando la venta está subsidiada por la Caja Costarricense del Seguro Social.

Cada transacción que afecte el inventario creará en un registro a nivel de Kardex, lo que permitirá conocer la cantidad y el costo de un producto en un momento determinado; no se permitirá la facturación en negativo por lo que para facturar un artículo este deberá contar con saldo en bodega. Con relación a la facturación de servicios o conceptos se habilitará el campo precio para que al momento de la venta permita digitar el monto a cobrar.

La función más relevante de este módulo es la capacidad de generar documentos electrónicos, lo cual permite cumplir al 100% con los requisitos del Ministerio de Hacienda de Costa Rica. Esto incluye la creación de archivos XML firmados electrónicamente con el certificado proporcionado por el Ministerio a través del sistema ATV, así como el envío y la consulta de documentos a la plataforma de factura electrónica de Hacienda. El proceso de envío y consulta valida todos los cálculos relacionados con impuestos, totales, exoneraciones, entre otros; si se detecta algún error en el cálculo de la factura electrónica, esta será rechazada. Por lo tanto, antes de procesar los documentos electrónicos, el sistema corroborará todos los cálculos previos a la generación de la factura.

Una vez generado y firmado el documento XML, se procederá con el envío al Ministerio de Hacienda para su recepción y aceptación. Al obtener la respuesta de aceptación de Hacienda, se generará un correo electrónico al cliente, adjuntando los archivos XML asociados al documento electrónico, así como la representación gráfica en formato PDF. En resumen, este módulo proporciona una solución integral y automatizada para la gestión de facturación de la empresa, garantizando eficiencia, control y cumplimiento de la normativa fiscal.

La facturación de crédito únicamente podrá ser realizada por un usuario administrador y se contempla la integración con el módulo de órdenes de trabajo. El módulo de facturación también cuenta con integración con el módulo de control de ingresos y gastos.

Módulo Anular Facturación

El módulo anular facturación se encargará de generar las notas de crédito electrónicas asociadas a las facturas de venta, permitiendo así cumplir con la normativa fiscal que impide la anulación de documentos electrónicos. Este módulo permitirá la creación de notas de crédito completas o parciales; a las cuales se llamarán Devoluciones, todas las notas de crédito están debidamente vinculadas a una factura o tiquete electrónico y se contempla la integración con el

inventario; registrando cada movimiento en el Kardex y recalculando el costo mediante el método del costo promedio ponderado, además de realizar la reversión de la venta.

Cuando la nota de crédito esté asociada a una factura o tiquete electrónico con condición de venta a crédito, el módulo se integrará con el sistema de cuentas por cobrar, permitiendo así rebajar automáticamente el saldo pendiente del cliente. Esta capacidad de crear notas de crédito parciales es particularmente ventajosa, ya que no exige que todos los artículos regresen al inventario, evitando la necesidad de emitir una nueva factura. El usuario debe digitar o escanear el código QR de la factura que desea devolver, el sistema mostrará los ítems asociados al documento electrónico y permitirá anular las líneas que quedan en pantalla. De esta manera, se soluciona el problema actual de tener que ajustar manualmente el inventario y emitir nuevas facturas, lo que a menudo lleva a la pérdida de control y trazabilidad del inventario.

El módulo también se encargará de generar el documento XML de la nota de crédito, firmarlo electrónicamente y enviarlo al Ministerio de Hacienda para su recepción y aceptación. Además, tras recibir la aceptación del Ministerio, se enviará un correo electrónico al cliente, adjuntando los archivos XML y la representación gráfica de la nota de crédito en formato PDF.

Módulo Generar Ordenes De Trabajo

El módulo de control de órdenes de trabajo gestionará las órdenes de adaptación de los zapatos. Uno de los principales clientes de la zapatería es la Caja Costarricense del Seguro Social que subsidia la compra de zapatos o plantillas ortopédicas a los pacientes. Las ventas al cliente CCSS requieren adaptaciones las cuales se registran en el módulo de orden de trabajo.

En la creación de una orden de trabajo se solicitarán los siguientes datos: el cliente y el paciente; este último cargará los datos de plantilla y los datos de la suela. Estas secciones contienen los campos necesarios para las adaptaciones; tales como el arco longitudinal, el arco transversal, el virón, el espolón entre otros; estos datos se podrán modificar por orden de trabajo.

Las órdenes de trabajo se podrán clasificar con los siguientes estados: pendientes, en proceso de producción, finalizadas y facturadas. El registro de órdenes estará integrado con el módulo de inventario para comprometer los productos y controlar los insumos utilizados en la modificación o adaptación de los diferentes tipos de calzado.

Este módulo estará integrado con el módulo de facturación; dicha funcionalidad evitará la necesidad de volver a digitar información ingresada en la orden de trabajo para poder emitir la factura.

Al ingresar al módulo los usuarios podrán visualizar una lista de órdenes de trabajo que se encuentran en estado pendiente y proceso de producción; esta lista proporcionara información relevante como la identificación, el nombre del cliente, la fecha de creación de la orden, la fecha de entrega y el estado, para que se pueda priorizar las adaptaciones.

La gestión de órdenes de trabajo finaliza una vez que es facturada, sin embargo se podrá seguir consultando la información.

Módulo Gestionar Las Cuentas Por Cobrar

El módulo de cuentas por cobrar se encargará de gestionar de manera integral el control de los saldos de los clientes en la empresa. Este sistema registra los abonos realizados sobre los créditos de los clientes, establece las fechas de pago y asegura los ingresos de la empresa. Además, este módulo está estrechamente vinculado con el módulo de facturación, donde las ventas a crédito generan un aumento en la deuda actual del cliente. El módulo está equipado con un *Dashboard* que proporciona a la empresa una visión clara de las cuentas por cobrar. Esto incluye información sobre los montos pendientes de cobro y los vencimientos.

Módulo Gestionar Las Cuentas Por Pagar

El módulo de cuentas por pagar gestionará eficazmente las obligaciones financieras de la empresa relacionadas con los proveedores de zapatos y materias primas. Incluye la integración directa con el proceso de compras, automatizando el registro de facturas de compra cuando la condición de compra sea crédito. Además, permite el registro de facturas que no están directamente relacionadas con el inventario o materias primas, así como el manejo de pagos, notas de crédito y débito, tanto financieras como electrónicas.

El módulo contará con un *Dashboard* que ofrecerá información clave para la gestión financiera. Esto incluye proyecciones de flujo de efectivo, la identificación de proveedores clave junto con el monto total pendiente de pago y sus fechas de vencimiento, además de proporcionar información sobre posibles incumplimientos. Esta funcionalidad permite a la empresa mantener un control efectivo sobre sus compromisos financieros con los proveedores, de esta manera, optimiza la gestión de sus recursos y asegurando la puntualidad en los pagos.

Módulo Control De Ingresos y Gastos.

El módulo de contabilidad se encargará de administrar eficazmente los ingresos y gastos de la empresa, integrando la información de todos los módulos del sistema informático propuesto. Este sistema permite cumplir con los requisitos de contabilidad de manera directa y secuencial,

facilitando el registro de asientos contables y la gestión del catálogo contable para una verificación exhaustiva de los gastos e ingresos.

El módulo incluye el registro automático de asientos diarios provenientes de diferentes áreas como ventas, cuentas por cobrar y cuentas por pagar, garantizando un registro preciso y actualizado de todas las transacciones financieras. Este registro automático no solo proporciona un control detallado de los gastos e ingresos, sino que también ahorra tiempo a la empresa al eliminar la necesidad de registros manuales. Además, facilita la toma de decisiones estratégicas al ofrecer información financiera precisa y oportuna.

El módulo de contabilidad optimiza la gestión financiera al automatizar el registro de transacciones y proporcionar herramientas analíticas clave que respaldan una toma de decisiones informada y eficiente en la empresa.

Módulo De Mantenimientos.

Este módulo se encargará de realizar el borrado, inserción, modificación, actualización de datos de proveedores, clientes, pacientes, artículos, compañía, sucursal, categorías, usuarios entre otros

Módulo De Seguridad.

Este módulo se encargará de realizar la autenticación de contraseñas y la definición de perfiles.

Análisis Detallado Del Hardware Requerido

El presente análisis muestra de manera detallada el hardware necesario para la programación y la posterior puesta en producción del prototipo funcional. A continuación, se presenta una tabla que destaca las especificaciones técnicas mínimas y recomendadas.

Tabla 7.

Análisis detallado del Hardware requerido para la programación del prototipo

Recurso	Mínimo	Recomendado	Hardware utilizado		
			Recurso	Cantidad	Costo
Procesador	2 núcleos a 1.7 GHz o superior	4 núcleos a 2.5 GHz o superior.	11th Gen Intel(R) Core (TM) i7-1165G7 @ 2.80GHz	1	No se incurre en costos debido a que el equipo es propio.
Memoria RAM	4 GB de RAM	8 GB de RAM o más.	20 GB		
Almacenamiento (SDD)	50 GB, espacio disponible	100 GB espacio disponible o más	300 GB disponibles		
Resolución de Pantalla	1280x720	1920x1080 o superior	1366 x768		
Sistema Operativo	Windows 10	Windows 10 o 11	Windows 11		
MS SQL Server	SQL Express 2017	SQL 2017 o superior	SQL Express 2017		
Visual Studio	2019	2019,2022	2022		
Conexión a internet	5 Mbps (descarga), 1 Mbps (carga)	10 Mbps (descarga), 2 Mbps (carga) o más	50 Mbps simétrico		

Fuente: Elaboración propia (2024).

Cada elemento de hardware considerado en este análisis se adapta a los requerimientos específicos del proyecto, en este caso, el desarrollo de un sistema de gestión de ventas para la empresa "Zapatitos Lindos S.A.", ubicada en San José.

La fase de puesta en producción de este proyecto adquiere un papel central en la concreción de los objetivos. Dado que el software desarrollado es de naturaleza web, se ha optado por una solución de alojamiento acorde con las necesidades de la aplicación. En este contexto, se ha tomado la decisión de contratar los servicios de hosting proporcionados por SmarterASP. Esta

elección se alinea de manera precisa con las características inherentes de un software web, ofreciendo una infraestructura robusta y confiable para el despliegue de la aplicación. A través de este enfoque, se asegura la accesibilidad y disponibilidad del sistema en línea con las expectativas y requerimientos de los usuarios finales, de esta forma, se solidifica una experiencia de usuario óptima y eficiente.

Para la puesta en producción del prototipo funcional de la empresa Zapatitos Lindos S.A., que es de naturaleza web y utiliza una base de datos SQL junto con tecnología Microsoft, se hace necesario contratar un servicio de *hosting* que cumpla con ciertos requisitos específicos. Estos requisitos incluyen soporte para tecnología Microsoft, capacidad para alojar bases de datos SQL, garantía de seguridad y disponibilidad, así como un nivel adecuado de rendimiento para asegurar la óptima funcionalidad del prototipo durante su operación en línea. Es fundamental que el servicio de hosting seleccionado sea capaz de satisfacer estas necesidades para garantizar una implementación exitosa del prototipo en el entorno de producción.

A continuación, la tabla con el *hardware* requerido para poner en producción el prototipo:

Tabla 8.

Análisis detallado del Hardware requerido para la puesta en producción del Prototipo

Recurso	Detalle	Costo
Servidor	Windows server 2022	Plan .NET Premium 7.95 \$ por mes
Espacio Web	Ilimitado	
Grupo de aplicaciones dedicado	3 GB	
Espacio MS SQL	10 GB	
Número de sitios / Entradas IIS	1	
Cifrado SSL	Recomendado	
FTP	Ilimitado	
Cantidad de Base Datos	4	
ASP NET	6X	
MVC	6	
Seguridad	Cortafuegos, Antivirus y Antispam	
Soporte Técnico	24/7	

Fuente: Elaboración propia (2024).

Análisis Detallado De Telecomunicaciones

Por tratarse de un sistema web, es esencial contar con una conexión a internet para acceder a este. Por lo tanto, es esencial garantizar una conexión a Internet estable y de alta velocidad para el correcto funcionamiento del sistema.

En cuanto al uso de Internet, el sistema necesitará acceso continuo a la red para procesar solicitudes de usuarios, enviar y recibir datos, y realizar actualizaciones en tiempo real. Esto implica la necesidad de un ancho de banda adecuado para manejar el tráfico de datos esperado, así como la implementación de medidas de seguridad para proteger la integridad y la confidencialidad de la información transmitida a través de la red. Por lo que se recomienda una velocidad mínima de 1 Mbps, aunque lo ideal es disponer de 5 Mbps para una experiencia óptima. Se considera importante mencionar que la empresa Zapatitos Lindos ya dispone de una conexión a internet 10 Mbps, suficiente para asegurar una transferencia fluida de datos.

En cuanto al tipo de red, el sistema opera en una red de área amplia (WAN), ya que es accesible a través de internet desde cualquier dispositivo, lo cual permite un acceso remoto flexible. Para su desarrollo y puesta en producción, se emplea un servicio de *hosting* que garantiza la disponibilidad constante en línea. Esto asegura que la aplicación esté accesible las 24 horas del día, los 7 días de la semana.

Además, se emplean puertos estándar para comunicaciones web, como el puerto 80 para HTTP y el puerto 443 para HTTPS.

En términos de costos, es importante tener en cuenta los gastos de conexión a internet, que pueden variar según el proveedor y el plan seleccionado, así como los costos asociados al servicio de *hosting*, que dependerán del plan elegido en función del almacenamiento y ancho de banda requeridos. Es importante mencionar que podrían surgir costos adicionales asociados con el consumo de ancho de banda adicional, el uso de servicios adicionales ofrecidos por el proveedor de hospedaje, y la implementación de medidas de seguridad adicionales para proteger el sistema contra amenazas cibernéticas.

En conclusión, para su correcto funcionamiento, el prototipo funcional requiere una conexión a internet confiable y de velocidad adecuada.

Descripción Detallada De las Herramientas Técnicas Utilizadas

A continuación, una descripción detallada de las herramientas técnicas utilizadas en el desarrollo del prototipo funcional para la gestión de ventas de la empresa Zapatitos Lindos S.A., ubicada en San José.

Es importante hacer mención de que ninguna de estas herramientas tiene costos asociados y no requieren licenciamiento al ser herramientas de libre consumo.

Tabla 9.

Herramientas técnicas utilizadas en el desarrollo

Herramienta Técnica	Descripción
Visual Studio Community 2022	Visual Studio Community 2022 es un entorno de desarrollo integrado (IDE) gratuito y completo de Microsoft, diseñado para desarrolladores individuales y equipos pequeños. Este <i>software</i> proporciona soporte multiplataforma, que permite el desarrollo de aplicaciones para Windows, macOS, Linux, iOS y Android. Es compatible con una amplia gama de lenguajes de programación como C#, VB.NET, C++, F#, Python, JavaScript y TypeScript, entre otros. Incluye herramientas avanzadas de depuración, un editor de código con funciones de resaltado de sintaxis, finalización automática de código y refactorización. Visual Studio Community 2022 es ideal para el desarrollo de aplicaciones web con ASP.NET, HTML/CSS, JavaScript, Angular y React, así como para el desarrollo de aplicaciones móviles nativas e híbridas utilizando Xamarin y React Native. También ofrece capacidades para crear aplicaciones de escritorio con Windows Forms y WPF, convirtiéndolo en una herramienta versátil y potente para diversas necesidades de desarrollo de <i>software</i> . Para ejecutar Visual Studio Community 2022 de manera efectiva, se recomienda contar con un sistema que cumpla con ciertos requisitos técnicos mínimos. Idealmente, se necesita un procesador con una velocidad de al menos 1.8 GHz, aunque es beneficioso tener un procesador multi-core para un rendimiento más fluido. En cuanto a la memoria RAM, se sugieren al menos 2 GB, pero se recomienda tener 8 GB o más para un funcionamiento óptimo, especialmente al trabajar con proyectos grandes o varios entornos de

	<p>desarrollo simultáneamente. El espacio en disco requerido es de aproximadamente 20 GB para la instalación básica, aunque este requisito puede variar según los componentes adicionales que se instalen. Es importante contar con una tarjeta gráfica compatible que admita una resolución de al menos 720p para una visualización clara de la interfaz de usuario y la edición de código. Estos requisitos aseguran que Visual Studio pueda ejecutarse de manera eficiente y aprovechar al máximo sus capacidades de desarrollo de <i>software</i>.</p>
Visual Studio Code	<p>Visual Studio Code es un editor de código fuente ligero pero potente desarrollado por Microsoft, compatible con plataformas como Windows, macOS y Linux. Destaca por su versatilidad al admitir una amplia gama de lenguajes de programación, incluyendo JavaScript, TypeScript, Python, Java, C# y C++, entre otros. Su interfaz de usuario es minimalista y moderna, ofreciendo funciones como resaltado de sintaxis, finalización automática de código y una barra lateral para navegación y administración de archivos. Es altamente personalizable a través de extensiones disponibles en el Marketplace, que permiten agregar nuevas funcionalidades y soporte para diversas herramientas y servicios de desarrollo, como Git, Docker, Kubernetes, y servicios en la nube como Azure y AWS. Además, soporta depuración integrada para varios lenguajes y gestión de tareas configurables mediante archivos de configuración específicos. En cuanto a requisitos del sistema, Visual Studio Code es eficiente y puede ejecutarse de manera óptima en sistemas con al menos 1 GB de RAM y suficiente espacio en disco para la instalación y almacenamiento de proyectos.</p>
Microsoft® SQL Server® 2017 Express	<p>Microsoft SQL Server 2017 Express es una versión gratuita y ligera del sistema de gestión de bases de datos relacional de Microsoft, diseñada principalmente para aplicaciones pequeñas y medianas, así como para entornos de desarrollo y pruebas. Esta edición proporciona capacidades completas de SQL Server, incluyendo la capacidad de crear y administrar bases de datos relacionales con soporte para consultas complejas, procedimientos almacenados, disparadores, índices y restricciones de</p>

	<p>integridad. Sin embargo, la principal limitación de SQL Server 2017 Express radica en el tamaño máximo de la base de datos, que está limitado a 10 GB por instancia de SQL Server. Esto puede ser suficiente para muchos escenarios, pero puede resultar restrictivo para aplicaciones que requieren grandes volúmenes de datos. A pesar de estas limitaciones, SQL Server 2017 Express sigue siendo una opción popular y robusta para desarrolladores y pequeñas empresas que buscan una solución de base de datos gratuita y poderosa.</p>
<p>Management Studio 19.0.2</p>	<p>SQL Server Management Studio (SSMS) 19.0.2 es una herramienta de administración y desarrollo de bases de datos de Microsoft, diseñada para interactuar y administrar instancias de SQL Server de manera eficiente. A continuación, se detallan las características y funcionalidades principales de SSMS 19.0.2:</p> <p>SSMS 19.0.2 ofrece una interfaz gráfica intuitiva que facilita tareas como la creación y administración de bases de datos, la escritura y ejecución de consultas SQL, la configuración de la seguridad y la gestión de permisos de usuarios. Además, permite realizar tareas avanzadas como la optimización de consultas, el diseño de esquemas de bases de datos y la programación de tareas automatizadas mediante la creación de trabajos de SQL Server Agent. Esta versión de SQL Server Management Studio está integrada con herramientas de depuración y monitoreo que permiten a los administradores y desarrolladores analizar el rendimiento de las consultas y la actividad del servidor. También incluye soporte para la gestión de versiones y la comparación de esquemas de base de datos, lo que facilita la sincronización y actualización de bases de datos en diferentes entornos.</p> <p>SSMS 19.0.2 es compatible con todas las versiones de SQL Server, incluyendo SQL Server 2017 y versiones anteriores, así como con instancias de bases de datos en la nube como Azure SQL Database. Esta herramienta es fundamental para profesionales de bases de datos que necesitan gestionar y desarrollar soluciones de datos de manera efectiva y eficiente.</p>
<p>ASP.Net Core 6</p>	<p>ASP.NET Core 6 MVC es un <i>framework</i> de desarrollo web de código abierto</p>

MVC	de Microsoft, diseñado para la creación de aplicaciones web modernas y escalables. Esta versión introduce diversas mejoras y características nuevas que facilitan el desarrollo ágil y eficiente de aplicaciones. La arquitectura basada en <i>middleware</i> permite una configuración flexible de la <i>pipeline</i> de solicitud y respuesta, lo que facilita la implementación de características como la autenticación, autorización y <i>logging</i> .
------------	--

Fuente: Elaboración propia (2024).

Descripción Detallada Del Personal Requerido Para El Uso Del Sistema

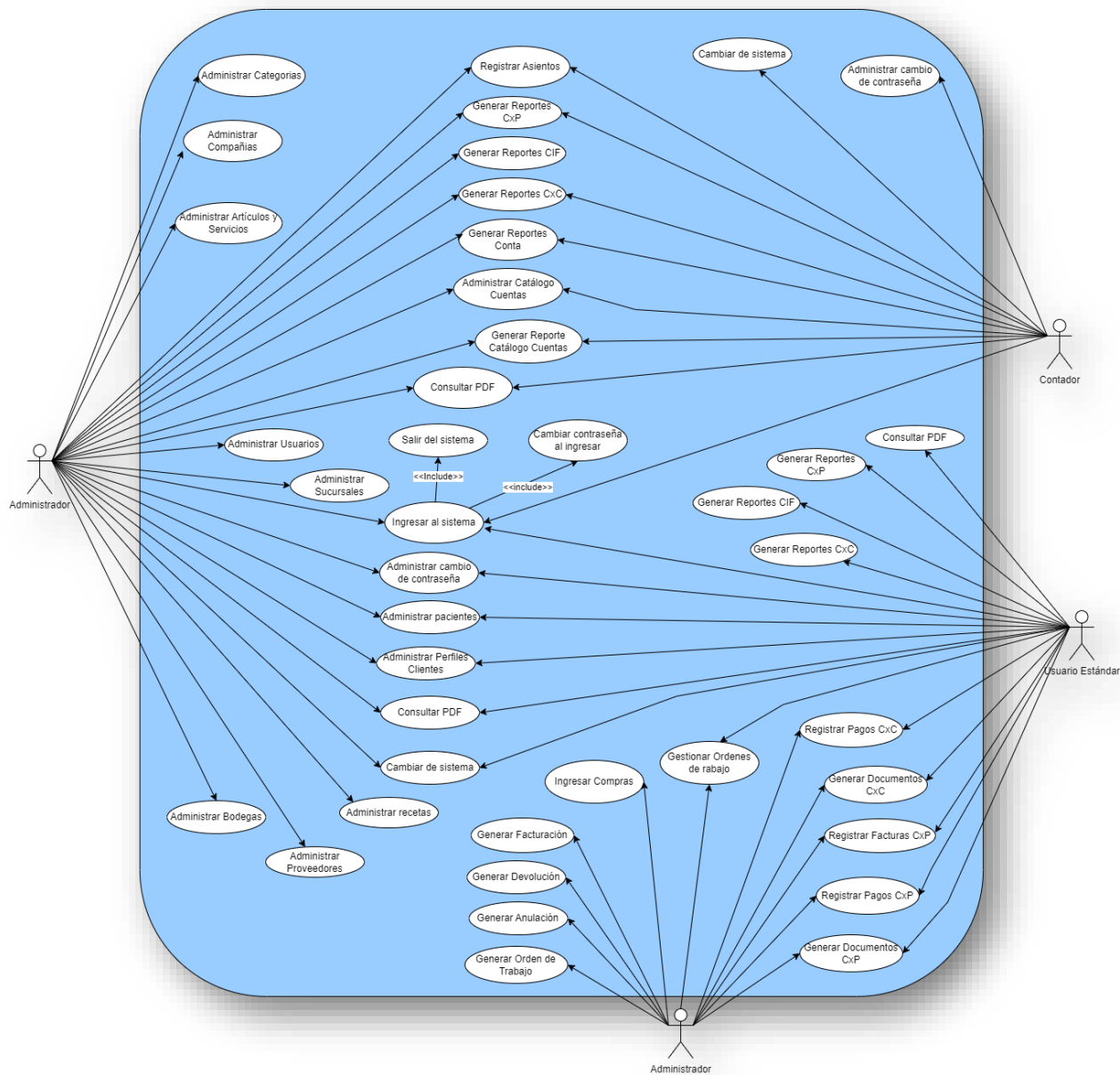
Para la implementación del prototipo funcional destinado a la gestión de ventas de Zapatitos Lindos S.A., ubicada en San José, se ha diseñado un sistema web con una interfaz intuitiva y fácil de usar para los usuarios. No se requieren conocimientos técnicos avanzados ni comprensión profunda de temas informáticos para su utilización; únicamente se necesitan habilidades básicas de computación, conocimientos sobre el uso de un navegador web y familiaridad con periféricos estándar como teclado, *mouse*, monitor e impresora. El sistema está diseñado para ser utilizado por al menos un administrador, quien tendrá acceso a funciones específicas como la gestión de catálogos tales como: clientes, proveedores, artículos, categorías, bodega, etc; así como a procesos que requieren un nivel más alto de responsabilidad, como lo son los movimientos de inventario, anulaciones y facturación a crédito. Además, se prevé al menos un usuario final que actuará como operador del sistema.

La capacitación no está contemplada en el alcance del proyecto de investigación, sin embargo, el prototipo se desarrollará de manera conjunta con la gerente general y la administradora, quienes serán responsables de transmitir el conocimiento necesario para el uso efectivo del sistema.

Casos De Uso

Figura 1.

Diagrama de caso de uso



Fuente: Elaboración propia (2024).

Tabla 10.

Caso de uso para ingresar al sistema

Prototipo: Prototipo funcional para la gestión de ventas de la empresa Zapatitos Lindos S.A., ubicada en San José.	
Número Caso de Uso: 1	Nombre del Caso de Uso: Ingresar al sistema.
Fecha elaboración:	27/03/2024
Descripción Caso de Uso:	Permitir el ingreso al sistema a los usuarios que se encuentren habilitados.
Autor caso de uso:	Daniel David Quesada Fallas.
Actores relacionados:	Administrador, Usuario Estándar, Contador.
Precondiciones:	<ol style="list-style-type: none"> 1. El actor debe estar registrado en la base de datos del sistema. 2. El actor debe contar con el estado “Activo” en la base de datos del sistema. 3. El actor debe contar con algunos de los siguientes roles: Administrador, Usuario Estándar, Contador. 4. El actor debe estar asociado a una compañía y sucursal en estado “Activo” y tener acceso al menos a un módulo del sistema.
Flujo Básico del caso de uso	
<ol style="list-style-type: none"> 1. Este caso de uso inicia cuando el actor ingresa al sistema web desde un navegador. 2. El sistema solicita la siguiente información: <ul style="list-style-type: none"> • Usuario. • Contraseña. • Compañía (Campo de selección). 3. El actor completa la información y continúa el proceso. 4. El sistema ejecuta las validaciones (SF-01, SF-02). 5. El sistema muestra y solicita la siguiente información: <ul style="list-style-type: none"> • Módulo (Selección). <ol style="list-style-type: none"> a) CIF (Compras, inventario, facturación, anulaciones, órdenes de trabajo). b) CXC (Cuentas por cobrar). c) CXP (Cuentas por pagar). d) CONTA (Gestión de ingresos y gastos). • Sucursal (Selección). 6. El actor completa la información y continúa el proceso. 7. El sistema ejecuta las validaciones (SF-03). 8. El sistema permite el ingreso. 9. El caso de uso finaliza. 	
Sub Flujos	
Sub Flujo Validaciones (SF-01)	1. El sistema verifica la información digitada. Si el actor no completo alguno de los siguientes campos: Usuario, Contraseña o Compañía, el sistema muestra el siguiente mensaje de error: “El campo “{Nombre Del Campo}” es requerido.”

	<p>2. El sistema verifica que el usuario exista en la base de datos. Si el usuario no existe, el sistema muestra el siguiente mensaje de error: “Las credenciales del usuario son incorrectas.”</p> <p>3. El sistema verifica que el usuario se encuentre en un estado valido.</p> <ul style="list-style-type: none"> • Si el estado es “Inactivo”, el sistema muestra el siguiente mensaje: “No podrá continuar el usuario no se encuentra habilitado para ingresar al sistema.” • Si el estado es “Bloqueado”, el sistema muestra el siguiente mensaje: “No podrá continuar, el usuario se encuentra bloqueado. Por favor contacte al administrador del sistema.” <p>4. El sistema verifica que la contraseña coincida con la que encuentra registrada en la base de datos. Si la contraseña no coincide, el sistema muestra el siguiente mensaje de error: “Las credenciales del usuario son incorrectas.”</p>
<p>Sub Flujo Intentos fallidos (SF-02)</p>	<p>1. Inicia cuando el actor ingresa el usuario y contraseña y continúa el proceso.</p> <p>2. El sistema ejecuta el Sub Flujo Validaciones (SF-01). Si las credenciales no son válidas, el sistema incrementa el contador de intentos fallidos.</p> <p>3. El sistema verifica el contador de intentos fallidos. Si el contador alcanzó un máximo de cinco intentos fallidos en un rango de una hora, el sistema bloquea al usuario por motivos de seguridad y muestra el siguiente mensaje de bloqueo: “No podrá continuar, el usuario se encuentra bloqueado. Por favor contacte al administrador del sistema”. Si el contador no ha alcanzado 5 intentos fallidos, el sistema muestra un mensaje de error y permite que el usuario intente iniciar sesión nuevamente.</p>
<p>Sub Flujo Selección de módulo y sucursal (SF-03)</p>	<p>1. El sistema verifica si el módulo seleccionado es válido: Si el módulo no lo es, el sistema muestra el siguiente mensaje: “No podrá continuar, debe seleccionar un módulo para poder ingresar al sistema.”</p> <p>2. El sistema verifica si la sucursal seleccionada es válida: Si la sucursal no lo es, se muestra el siguiente mensaje: “No podrá continuar, debe seleccionar una sucursal para poder ingresar al sistema.”</p>
<p>Flujos Alternos</p>	
<p>Flujo Alternativo No. 1</p>	<p>Si existe algún error en el proceso:</p> <p>1. El sistema registra el error.</p> <p>2. El sistema muestra el siguiente mensaje de error: “No se puede realizar la acción dado a que se presentó un</p>

	<p>problema.</p> <ol style="list-style-type: none"> a) Intente ejecutar nuevamente la acción. b) Si el problema persiste consulte al administrador del sistema. <p>Identificador del error: “{Identificador}”.</p> <ol style="list-style-type: none"> 3. El sistema registra el Log de error. 4. El caso de uso finaliza.
Flujo Alternativo No. 2	<p>Si existen errores de validación: SF-01, SF-03</p> <ol style="list-style-type: none"> 1. El sistema muestra el mensaje de error. 2. El caso de uso finaliza.
Flujo Alternativo No. 3	<p>Si el actor olvido la contraseña:</p> <ol style="list-style-type: none"> 1. El actor debe contactar al administrador del sistema para que le proporcione una nueva contraseña. 2. El primer ingreso posterior al cambio de contraseña el actor debe cambiarla para poder ingresar nuevamente. Se extiende al Caso de Uso “Cambiar contraseña al ingresar”. 3. El caso de uso finaliza.
Flujo Alternativo No. 4	<p>Si el actor ingresa una dirección de un módulo no permitido:</p> <ol style="list-style-type: none"> 1. El sistema muestra el siguiente mensaje: “Acceso Denegado!!! El usuario no cuenta con los requisitos para poder ingresar a esta opción.” 2. El caso de uso finaliza.
Flujo Alternativo No. 5	<p>Si existe una sola compañía configurada en el sistema:</p> <ol style="list-style-type: none"> 1. El sistema no muestra el campo de selección compañía. 2. El caso de uso finaliza.
Flujo Alternativo No. 6	<p>Si el actor que ingresa al sistema se encuentra asociado a una única sucursal y a un único módulo:</p> <ol style="list-style-type: none"> 1. El sistema omite los pasos 5, 6 y 7 del flujo base y no muestra la pantalla para seleccionar el módulo y la sucursal. 2. El caso de uso finaliza.
Requerimientos especiales	
<p>Acceso completo: El actor con el rol “Administrador” cuenta con acceso completo a todas las opciones del sistema.</p> <p>Acceso parcial: El actor con el rol “Usuario Estándar” puede ingresar solo a las opciones permitidas.</p> <p>Acceso solo lectura: El actor con el rol “Contador” cuenta con permisos de solo lectura en los módulos CIF, CXP, CXC para generar <u>reportes</u> y acceso completo a las opciones del módulo de contabilidad.</p>	
Post-Condiciones	
Una vez que el sistema validó la información del actor, se permite el ingreso y el uso de este de acuerdo con su rol.	

Fuente: Elaboración propia (2024).

Tabla 11.*Caso de uso para salir del sistema*

Prototipo: Prototipo funcional para la gestión de ventas de la empresa Zapatitos Lindos S.A., ubicada en San José.	
Número Caso de Uso: 2	Nombre del Caso de Uso: Salir del sistema.
Fecha elaboración:	27/03/2024
Descripción Caso de Uso:	Cerrar sesión de usuario.
Autor caso de uso:	Daniel David Quesada Fallas.
Actores relacionados:	Administrador, Usuario Estándar, Contador.
Precondiciones:	1. Es necesario que el actor haya ingresado previamente al sistema con sus credenciales.
Flujo Básico del caso de uso	
1. Este caso de uso inicia cuando el actor que ingresó con sus credenciales al sistema; decide salir y ejecuta la acción cerrar sesión. 2. El sistema cierra la sesión del actor y muestra la pantalla ingresó nuevamente. 3. El caso de uso finaliza.	
Sub Flujos	
Flujos Alternos	
Requerimientos especiales	
Post-Condicion	

Fuente: Elaboración propia (2024).**Tabla 12.***Caso de uso para cambiar contraseña al ingresar*

Prototipo: Prototipo funcional para la gestión de ventas de la empresa Zapatitos Lindos S.A., ubicada en San José.	
Número Caso de Uso: 3	Nombre del Caso de Uso: Cambiar contraseña al ingresar.
Fecha elaboración:	31/03/2024
Descripción Caso de Uso:	Permitir a los usuarios registrados en el sistema cambiar la contraseña que fue asignada o modificada previamente por un usuario administrador en el mantenimiento de usuarios.
Autor caso de uso:	Daniel David Quesada Fallas.
Actores relacionados:	Administrador, Usuario Estándar, Contador.

Precondiciones:	<ol style="list-style-type: none"> 1. El actor con el rol administrador crea un nuevo usuario. 2. Un actor con el rol administrador modifica un usuario existente. 3. Es el primer ingreso del usuario posterior a una modificación de contraseña realiza por un administrador. 4. Es el primer ingreso posterior a la creación del usuario.
Flujo Básico del caso de uso	
<ol style="list-style-type: none"> 1. Este caso de uso inicia cuando el actor ingresa las credenciales al sistema y ejecuta la acción “Ingresar” en la pantalla de “Ingreso al sistema”. 2. El sistema verifica; si el actor que está ingresando, debe cambiar la contraseña. 3. El sistema solicita la siguiente información. <ol style="list-style-type: none"> 1- Nueva contraseña (Requerido). 2- Confirmar contraseña (Requerido). 4. El actor completa la información. 5. El sistema ejecuta las validaciones (SF-01). 6. El sistema realiza el cambio de contraseña del actor. 7. El sistema continúa con el flujo de ingreso al sistema. 8. El caso de uso finaliza. 	
Sub Flujos	
Sub Flujo Validaciones (SF-01)	<ol style="list-style-type: none"> 1. El sub flujo inicia en el paso 5 del flujo base. 2. El sistema verifica que el actor completó la información de aquellos datos requeridos. Si actor no completó los datos de alguno de los campos que son requeridos, el sistema muestra el siguiente mensaje de error: “El campo “{NombreDelCampo} es requerido”. 3. El sistema verifica que la contraseña sea robusta. Debe tener como mínimo 5 caracteres que sean al menos un número, una letra mayúscula y una minúscula. Si la contraseña no cumple con los requisitos mínimos, el sistema muestra el siguiente mensaje de error: “La contraseña debe contar con mínimo 5 caracteres de los cuales debe existir al menos un número, una letra mayúscula y una minúscula”. 4. El sistema verifica que los campos Contraseña y Confirmar contraseña coincidan. Si la información digitada no coincide el sistema muestra el siguiente mensaje de error: “No podrá continuar. El campo "Contraseña" y "Confirmar Contraseña" deben coincidir. 5. El sistema verifica que la contraseña digitada sea diferente a la contraseña actual. Si la nueva contraseña es igual a la contraseña actual el sistema muestra el siguiente mensaje de error: “No podrá continuar, debe digitar una contraseña diferente a la actual”.
Sub Flujo Mostrar/Ocultar Contraseña (SF-02)	<ol style="list-style-type: none"> 1. El actor ejecuta la acción (Mostrar/Ocultar contraseña). 2. El sistema habilita o deshabilita la máscara (***) que protege los campos: Nueva contraseña y Confirmar contraseña; permitiendo o denegando la visibilidad del contenido en estos campos.
Flujos Alternos	

Flujo Alternativo No. 1	<p>Si existe algún error en el proceso:</p> <ol style="list-style-type: none"> 1. El sistema registra el error. 2. El sistema muestra el siguiente mensaje de error: “No se puede realizar la acción dado a que se presentó un problema”. <li style="margin-left: 40px;">a) Intente ejecutar nuevamente la acción. <li style="margin-left: 40px;">b) Si el problema persiste consulte al administrador del sistema. <p style="margin-left: 40px;">Identificador del error: “{Identificador}”</p> <ol style="list-style-type: none"> 3. El sistema registra el Log de error. 4. El caso de uso finaliza.
Flujo Alternativo No. 2	<p>Si existen errores de validación:</p> <ol style="list-style-type: none"> 1. El sistema muestra el mensaje de error. 2. El caso de uso finaliza.
Flujo Alternativo No. 3	<p>Si el actor ejecuta la acción “cancelar”:</p> <ol style="list-style-type: none"> 1. El sistema limpia de la información de los siguientes campos <ul style="list-style-type: none"> • Nueva contraseña. • Confirmar contraseña. • (Mostrar/Ocultar) contraseña. 2. El caso de uso finaliza.
Requerimientos especiales	
No aplica.	
Post-Condiciones	
Se permite el ingreso del usuario al sistema una vez que actualizó la contraseña temporal por la nueva.	

Fuente: Elaboración propia (2024).

Tabla 13.

Caso de uso para administrar cambio de contraseña

Prototipo: Prototipo funcional para la gestión de ventas de la empresa Zapatitos Lindos S.A., ubicada en San José.	
Número Caso de Uso: 4	Nombre del Caso de Uso: Administrar cambio de contraseña.
Fecha elaboración:	31/03/2024
Descripción Caso de Uso:	Permitir a los usuarios registrados en el sistema cambiar la contraseña de ingreso al sistema.
Autor caso de uso:	Daniel David Quesada Fallas.
Actores relacionados:	Administrador, Usuario Estándar, Contador.
Precondiciones:	1. El actor debe haber iniciado sesión en el sistema.
Flujo Básico del caso de uso	
<ol style="list-style-type: none"> 1. Este caso de uso inicia cuando el actor selecciona la opción del menú: “Cambiar Contraseña”. 2. El sistema muestra la siguiente información (sección no editable). <ol style="list-style-type: none"> 1- Usuario. 2- Nombre. 	

<p>3. El sistema solicita la siguiente información.</p> <ol style="list-style-type: none"> 1- Nueva contraseña. (Requerido). 2- Confirmar contraseña (Requerido). 3- (Mostrar/Ocultar) contraseña. <p>4. El actor completa la información.</p> <p>5. El sistema ejecuta las validaciones (SF-01).</p> <p>6. El sistema realiza el cambio de contraseña del actor.</p> <p>7. El caso de uso finaliza.</p>	
Sub Flujos	
Sub Flujo Validaciones (SF-01)	<ol style="list-style-type: none"> 1. El sub flujo inicia en el paso 5 del Flujo base. 2. El sistema verifica que el actor completó la información de aquellos datos requeridos. Si actor no completó los datos de alguno de los campos que son requeridos, el sistema muestra el siguiente mensaje de error: “El campo “{NombreDelCampo} es requerido.” 3. El sistema verifica que la contraseña sea robusta. Debe tener como mínimo 5 caracteres que sean al menos un número, una letra mayúscula y una minúscula. Si la contraseña no cumple con los requisitos mínimos, el sistema muestra el siguiente mensaje de error: “La contraseña debe contar con mínimo 5 caracteres de los cuales debe existir al menos un número, una letra mayúscula y una minúscula”. 4. El sistema verifica que los campos Contraseña y Confirman contraseña coincidan. Si la información digitada no coincide el sistema muestra el siguiente mensaje de error: “No podrá continuar. El campo “Contraseña” y “Confirmar Contraseña” debe coincidir.” 5. El sistema verifica que la contraseña digitada sea diferente a la contraseña actual. Si la nueva contraseña es igual a la contraseña actual el sistema muestra el siguiente mensaje de error: “No podrá continuar, debe digitar una contraseña diferente a la actual.”
Sub Flujo Mostrar/Ocultar Contraseña (SF-02)	<ol style="list-style-type: none"> 1. El actor ejecuta la acción (Mostrar/Ocultar contraseña) 2. El sistema habilita o deshabilita la máscara (***) que protege los campos: Nueva contraseña y Confirmar contraseña permitiendo o denegando la visibilidad del contenido en estos campos.
Flujos Alternos	
Flujo Alternativo No. 1	<p>Si existe algún error en el proceso:</p> <ol style="list-style-type: none"> 1. El sistema registra el error. 2. El sistema muestra el siguiente mensaje de error: “No se puede realizar la acción dado a que se presentó un problema. <ol style="list-style-type: none"> a) Intente ejecutar nuevamente la acción. b) Si el problema persiste consulte al administrador del sistema. Identificador del error: {Identificador}” 3. El sistema registra el Log de error.

	4. El caso de uso finaliza.
Flujo Alternativo No. 2	Si existen errores de validación: 1. El sistema muestra el mensaje de error. 2. El caso de uso finaliza.
Flujo Alternativo No. 3	Si el actor ejecuta la acción “cancelar” o “limpiar”: 1. El sistema limpia de la información de los siguiente campos <ul style="list-style-type: none"> • Nueva contraseña • Confirmar contraseña • (Mostrar/Ocultar) contraseña 2. El caso de uso finaliza.
Requerimientos especiales	
No aplica.	
Post-Condiciones	

Fuente: Elaboración propia (2024).

Tabla 14.

Caso de uso para administrar compañías

Prototipo: Prototipo funcional para la gestión de ventas de la empresa Zapatitos Lindos S.A., ubicada en San José.	
Número Caso de Uso: 5	Nombre del Caso de Uso: Administrar compañías.
Fecha elaboración:	31/03/2024
Descripción Caso de Uso:	Permitir la administración de compañías y la gestión de la información relacionada con el emisor de factura electrónica.
Autor caso de uso:	Daniel David Quesada Fallas.
Actores relacionados:	Administrador.
Precondiciones:	1. El actor debe iniciar sesión en el sistema y disponer del Rol "Administrador". 2. El actor debe estar inscrito ante la Administración Tributaria. 3. El actor debe poseer la llave criptográfica, así como un usuario y contraseña válidos, los cuales son proporcionados por el Ministerio de Hacienda y se generan en el sitio https://atv.hacienda.go.cr/ para el uso de la factura electrónica.
Flujo Básico del caso de uso	
1. Este caso de uso inicia cuando el actor selecciona del menú “ Mantenimientos ” > “ Sistema ” la opción “ Compañías ”.	
2. El sistema solicita la siguiente información: <ul style="list-style-type: none"> • Tipo identificación (Selección). • Identificación. • Nombre. • Nombre comercial. * • Código área teléfono (Selección). 	

- Teléfono. *
- Código área fax (Selección).
- Fax. *
- Correo.
- Provincia (Selección).
- Cantón (Selección).
- Distrito (Selección).
- Barrio (Selección).
- Dirección.
- Certificado firma digital.
- Clave del certificado.
- Fecha de vencimiento (solo lectura).
- Usuario de hacienda.
- Contraseña de hacienda.
- Estado (1-Inactiva, 2- Activa).

* Campos opcionales.

3. El actor completa la información y ejecuta la acción Grabar.
4. El sistema ejecuta las validaciones (SF-03).
5. El sistema registra o actualiza la información.
6. El caso de uso finaliza.

Sub Flujos

<p>Sub Flujo Buscar (SF-01)</p>	<ol style="list-style-type: none"> 1. El sub flujo inicia cuando el actor ejecuta la acción “Buscar”. 2. El sistema solicita la siguiente información. <ul style="list-style-type: none"> • Buscar por: Identificación, Nombre, Nombre Comercial. • Estado: “Todos”, “Activa”, “Inactiva” • Filtro opciones: Que inicie, Que contenga, Que finalice. • Llave de búsqueda. 3. El actor selecciona una opción del apartado “Buscar por”. 4. El actor selecciona una opción del apartado “Opciones”. 5. El actor ingresa el valor a buscar en el campo Llave búsqueda. 6. El actor selecciona el estado que desea filtrar. 7. El actor ejecuta la acción “Buscar”. 8. El sistema muestra un listado con las compañías que cumplan con el criterio de búsqueda. 9. El sistema muestra la siguiente opción: “Editar”.
<p>Sub Flujo Seleccionar para Editar (SF-02)</p>	<ol style="list-style-type: none"> 1. Inicia cuando el actor ejecuta la acción “Seleccionar para Editar” en el paso 9 del Sub Flujo Buscar (SF-01). 2. El sistema carga la información de la compañía seleccionada. 3. El sistema continúa en el paso 3 del Flujo Base.

<p>Sub Flujo Validaciones (SF-03)</p>	<ol style="list-style-type: none"> 1. El sub flujo inicia en el paso 4 del Flujo base. 2. El sistema verifica que el actor completó la información de aquellos datos requeridos. Si actor no completó los datos, el sistema muestra el siguiente mensaje de error: “El campo “{NombreDelCampo} es requerido”. 3.El sistema verifica el formato de la identificación cuando se tenga información en el campo “Tipo de identificación del emisor. Formato: La “Cédula física” debe de contener 9 dígitos, sin cero al inicio y sin guiones. La “cédula de personas Jurídicas” debe contener 10 dígitos y sin guiones. El “Documento de Identificación Migratorio para Extranjeros (DIMEX)” debe contener 11 o 12 dígitos, sin ceros al inicio y sin guiones. El “Documento de Identificación de la DGT (NITE)” debe contener 10 dígitos y sin guiones. Si la identificación no cumple con el formato el sistema muestra el siguiente mensaje de error: “La identificación no cumple con el siguiente formato {#####}”. 4. El sistema verifica que la dirección de correo digitada sea válida. Debe cumplir con la siguiente estructura: $\wedge \backslash s * (([\wedge \langle \rangle \backslash [\backslash] \backslash . , : \backslash s @ \backslash \backslash "] + (\backslash . [\wedge \langle \rangle \backslash [\backslash] \backslash . , : \backslash s @ \backslash \backslash "] +) *) ((\backslash \backslash " . + \backslash \backslash ") @ (([\wedge \langle \rangle \backslash [\backslash] \backslash . , : \backslash s @ \backslash \backslash "] + \backslash .) + [\wedge \langle \rangle \backslash [\backslash] \backslash . , : \backslash s @ \backslash \backslash "] \{ 0 , \}) \backslash s * \\$.$ <p>Si el correo es inválido el sistema muestra el siguiente mensaje de error: “El campo “Correo” no cumple con los requisitos mínimos de una dirección de correo electrónico.”</p> 5. El sistema verifica la clave del certificado. Si la clave del certificado digital no coincide, el sistema muestra el siguiente mensaje de error: “La clave del certificado digital es incorrecta”. 6. El sistema verifica el usuario y contraseña de hacienda, en caso ser inválidos el sistema muestra el siguiente mensaje de error: “No podrá continuar, el usuario o contraseña de hacienda son incorrectos”. 7. Si el actor no ha ejecutado el Sub Flujo (SF-02), el sistema llevará a cabo una validación para asegurarse de que no existan registros duplicados. En caso de que se detecte un registro previo con la misma información, el sistema mostrará un mensaje de error que indica: “No podrá continuar, ya existe un registro con la información ingresada”.
<p>Sub Flujo Cargar Certificado (SF-03)</p>	<ol style="list-style-type: none"> 1. El sistema abre el explorador de archivos y solo permite la carga de certificados digitales con la extensión de archivo “p12”.

	<ol style="list-style-type: none"> 2. El actor selecciona y carga el certificado digital. 3. El sistema muestra la fecha de vencimiento del certificado.
Flujos Alternos	
Flujo Alternativo No. 1	<p>Si existe algún error en el proceso:</p> <ol style="list-style-type: none"> 1. El sistema registra el error. 2. El sistema muestra el siguiente mensaje de error: “No se puede realizar la acción dado a que se presentó un problema”. <li style="padding-left: 40px;">c) Intente ejecutar nuevamente la acción. <li style="padding-left: 40px;">d) Si el problema persiste consulte al administrador del sistema. <p style="padding-left: 40px;">Identificador del error: “{Identificador}”</p> <ol style="list-style-type: none"> 3. El sistema registra el Log de error. 4. El caso de uso finaliza
Flujo Alternativo No. 2	<p>Si existen errores de validación:</p> <ol style="list-style-type: none"> 1. El sistema muestra el mensaje de error. 2. El caso de uso finaliza.
Flujo Alternativo No. 3	<p>Si el actor ejecuta la acción “cancelar” o “limpiar”:</p> <ol style="list-style-type: none"> 1. El sistema limpia la información completada. 2. El caso de uso finaliza.
Flujo Alternativo No. 4	<p>Si el actor no dispone del rol Administrador:</p> <ol style="list-style-type: none"> 1. El sistema muestra el siguiente mensaje: “Acceso Denegado!!! El usuario no cuenta con los requisitos para poder ingresar a esta opción.” 2. El caso de uso finaliza.
Flujo Alternativo No. 5	<p>Si ejecuta la acción “regresar” al buscar.</p> <ol style="list-style-type: none"> 1. El sistema continúa en el paso 2 del Flujo Base. 2. El caso de uso finaliza.
Requerimientos especiales	
No aplica.	
Post-Condiciones	

Fuente: Elaboración propia (2024).

Tabla 15.

Caso de uso para administrar categorías

Prototipo: Prototipo funcional para la gestión de ventas de la empresa Zapatitos Lindos S.A., ubicada en San José.	
Número Caso de Uso: 6	Nombre del Caso de Uso: Administrar categorías.
Fecha elaboración:	31/03/2024
Descripción Caso de Uso:	Permitir la administración de categorías que permitirán la clasificación de productos.
Autor caso de uso:	Daniel David Quesada Fallas.
Actores relacionados:	Usuario Administrador.
Precondiciones:	1. El actor debe ingresar al sistema y debe contar con el Rol "Administrador".
Flujo Básico del caso de uso	
<ol style="list-style-type: none"> 1. Este caso de uso inicia cuando el actor selecciona del menú “Mantenimientos” > “Inventario” la opción “Categorías”. 2. El sistema solicita la siguiente información: <ul style="list-style-type: none"> • Descripción (Requerido). • Estado (Activo, Inactivo) (Requerido). 3. El actor completa la información y ejecuta la acción Grabar. 4. El sistema ejecuta las validaciones (SF-03). 5. El sistema registra o actualiza la información. 6. El caso de uso finaliza. 	
Sub Flujos	
Sub Flujo Buscar (SF-01)	<ol style="list-style-type: none"> 1. El sub flujo inicia cuando el actor ejecuta la acción “Buscar”. 2. El sistema solicita la siguiente información. <ul style="list-style-type: none"> • Llave de búsqueda. • Estado. • Filtro opciones: Que inicie, Que contenga, Que finalice. 3. El actor ingresa un valor en el campo llave búsqueda. 4. El actor selecciona el filtro estado. 5. El actor selecciona una opción del filtro opciones. 6. El actor ejecuta la acción “Buscar”. 7. El sistema muestra un listado con las categorías que cumplan con el criterio de búsqueda seleccionado. 8. El sistema muestra la siguiente opción: “Editar”.
Sub Flujo Editar (SF-02)	<ol style="list-style-type: none"> 1. Inicia cuando el actor ejecuta la acción “Editar” en el paso 8 del Sub Flujo Buscar (SF-01). 2. El sistema carga la información de la categoría seleccionada. 3. El sistema continúa en el paso 3 del Flujo Base.
Sub Flujo Validaciones (SF-03)	<ol style="list-style-type: none"> 1. El sub flujo inicia en el paso 4 del Flujo base. 2. El sistema verifica que el actor completó la información de aquellos datos requeridos. Si actor no completó los datos, el sistema muestra el siguiente mensaje de error: “El campo

	<p>“{NombreDelCampo} es requerido”.</p> <p>3. El sistema verifica que no exista otra categoría registrada previamente con la misma descripción.</p> <p>4. Si el actor no ha ejecutado el Sub Flujo (SF-02), el sistema llevará a cabo una validación para asegurarse de que no existan registros duplicados. En caso de que se detecte un registro previo con la misma información, el sistema mostrará un mensaje de error que indica: “No puede continuar, ya existe un registro con la información ingresada.”</p>
Flujos Alternos	
Flujo Alternativo No. 1	<p>Si existe algún error en el proceso:</p> <p>1. El sistema registra el error.</p> <p>2. El sistema muestra el siguiente mensaje de error: “No se puede realizar la acción dado a que se presentó un problema”.</p> <p style="padding-left: 40px;">a) Intente ejecutar nuevamente la acción.</p> <p style="padding-left: 40px;">b) Si el problema persiste consulte al administrador del sistema.</p> <p style="padding-left: 40px;">Identificador del error: “{Identificador}”</p> <p>3. El sistema registra el Log de error.</p> <p>4. El caso de uso finaliza.</p>
Flujo Alternativo No. 2	<p>Si existen errores de validación:</p> <p>1. El sistema muestra el mensaje de error.</p> <p>2. El caso de uso finaliza.</p>
Flujo Alternativo No. 3	<p>Si el actor ejecuta la acción “cancelar” o “limpiar”:</p> <p>1. El sistema limpia la información completada.</p> <p>2. El caso de uso finaliza.</p>
Flujo Alternativo No. 4	<p>Si el actor no dispone del rol Administrador:</p> <p>1. El sistema muestra el siguiente mensaje: “Acceso Denegado!!! El usuario no cuenta con los requisitos para poder ingresar a esta opción”.</p> <p>2. El caso de uso finaliza.</p>
Flujo Alternativo No. 5	<p>Si ejecuta la acción “regresar” al buscar</p> <p>1. El sistema continúa en el paso 2 del Flujo Base.</p> <p>2. El caso de uso finaliza.</p>
Requerimientos especiales	
No aplica.	
Post-Condiciones	

Fuente: Elaboración propia (2024).

Tabla 16.

Caso de uso para administrar artículos y servicios

Prototipo: Prototipo funcional para la gestión de ventas de la empresa Zapatitos Lindos S.A., ubicada en San José.	
Número Caso de Uso: 7	Nombre del Caso de Uso: Administrar artículos y servicios.
Fecha elaboración:	16/04/2024
Descripción Caso de Uso:	Permitir la administración de bienes y servicios (Artículos)
Autor caso de uso:	Daniel David Quesada Fallas.
Actores relacionados:	Administrador.
Precondiciones:	1. El actor debe haber iniciado sesión en el sistema y disponer del Rol "Administrador".
Flujo Básico del caso de uso	
<p>1. Este caso de uso inicia cuando el actor selecciona del menú “Mantenimientos” > “Inventarios” la opción “Artículos”.</p> <p>2. El sistema solicita la siguiente información:</p> <p><u>Información General</u></p> <ul style="list-style-type: none"> • Tipo de código (Selección). • Código. • CAByS. • Descripción. • Proveedor. • Categoría. • Costo. • Margen 1. • Margen 2. • Margen 3. • Precio 1. • Precio 2. • Precio 3. • Estado. • Cantidad Mínima. • Cantidad Máxima. • Unidad Medida. • Unidad Medida Comercial. • Impuesto. • Tarifa. • Se Vende. • Es Servicio. • Afecta Inventario. • Permite Cambiar Precio. • Tipo Artículo (No aplica, Primario, Compuesto). <p><u>Asignación Bodegas</u></p>	

<ul style="list-style-type: none"> • Lista de Bodegas <ol style="list-style-type: none"> 3. El actor completa la información y ejecuta la acción Grabar. 4. El sistema ejecuta las validaciones. (SF-07). 5. El sistema registra o actualiza la información. 6. El caso de uso finaliza. 	
Sub Flujos	
Sub Flujo Buscar Artículo (SF-01)	<ol style="list-style-type: none"> 1. El sub flujo inicia cuando el actor ejecuta la acción “Buscar”. 2. El sistema solicita la siguiente información. <ul style="list-style-type: none"> • <u>Filtros:</u> (Código, Descripción, Estado: “Todos”, “Activo”, “Inactivo”, “Descontinuado”) • <u>Opciones:</u> (Que inicie, Que contenga, Que finalice) • <u>Llave de búsqueda</u> 3. El actor selecciona una opción del apartado “Filtros”. 4. El actor selecciona una opción del apartado “Opciones”. 5. El actor ejecuta la acción “Buscar”. 6. El sistema muestra un listado con los artículos que cumplan con el criterio de búsqueda. 7. El sistema muestra la siguiente opción: “Editar”.
Sub Flujo Editar (SF-02)	<ol style="list-style-type: none"> 1. Inicia cuando el actor ejecuta la acción “Editar” en el paso 7 del Sub Flujo Buscar (SF-01). 2. El sistema carga la información del artículo seleccionado. 3. El sistema continúa en el paso 3 del Flujo Base.
Sub Flujo Buscar Proveedor (SF-03)	<ol style="list-style-type: none"> 1. El sub flujo inicia cuando el actor ejecuta la acción “Buscar Proveedor”. 2. El sistema solicita la siguiente información. <ul style="list-style-type: none"> • <u>Opciones:</u> (Que inicie, Que contenga, Que finalice) • <u>Llave de búsqueda</u> 3. El actor selecciona una opción del apartado “Opciones”. 4. El actor ejecuta la acción “Buscar” 5. El sistema muestra un listado con los proveedores en estado “Activo” que cumplan con el criterio de búsqueda. El sistema muestra la siguiente opción: “Seleccionar”
Sub Flujo Seleccionar Proveedor (SF-04)	<ol style="list-style-type: none"> 1. Inicia cuando el actor ejecuta la acción “Seleccionar” en el paso 5 del Sub Flujo Buscar Proveedor (SF-03). 2. El sistema carga la información del proveedor seleccionado en el campo “Proveedor”.
Sub Flujo Buscar Categoría (SF-05)	<ol style="list-style-type: none"> 1. El sub flujo inicia cuando el actor ejecuta la acción “Buscar Categoría”. 2. El sistema solicita la siguiente información. <ul style="list-style-type: none"> • <u>Opciones:</u> (Que inicie, Que contenga, Que finalice) • <u>Llave de búsqueda.</u> 3. El actor selecciona una opción del apartado “Opciones”. 4. El actor ejecuta la acción “Buscar” 5. El sistema muestra un listado con las categorías en estado “Activo” que cumplan con el criterio de búsqueda.

	3. El sistema muestra la siguiente opción: “Seleccionar”.
Sub Flujo Seleccionar Categoria (SF-06)	<ol style="list-style-type: none"> 1. Inicia cuando el actor ejecuta la acción “Seleccionar” en el paso 5 del Sub Flujo Buscar Proveedor (SF-05). 2. El sistema carga la información de la categoría seleccionada en el campo “Proveedor”.
Sub Flujo Validaciones (SF-07)	<ol style="list-style-type: none"> 1. El sub flujo inicia en el paso 4 del flujo base. 2. El sistema verifica que el actor completo la información de aquellos datos requeridos. Si actor no completo los datos, el sistema muestra el siguiente mensaje de error: “El campo “{NombreDelCampo} es requerido”. 3.El sistema verifica que previamente no exista un registro con los datos ingresados. Si existe un registro previo el sistema muestra el siguiente mensaje de error: “No puede continuar, ya existe un registro con la información ingresada”. 4. Al editar, el sistema verifica que no puede desasociar artículos de bodega con existencias. Si el artículo cuenta con saldo en la bodega que se desea desasociar, el sistema muestra el siguiente mensaje de error: “No podrá desmarcar la opción porque hay bodegas con saldo mayor a cero.” 5.El sistema verifica que la información digitada en los campos alfanuméricos no exceda la cantidad máxima de caracteres. En caso de exceder el límite el sistema muestra el siguiente mensaje de error: “No podrá continuar, el campo “{NombreDelCampo}” supera los {CantMáxima} caracteres permitidos.” 6. El sistema verifica que la información digitada en los campos alfanuméricos no contenga caracteres especiales no permitidos. En caso de contener caracteres especiales o restringidos el sistema muestra el siguiente mensaje de error: “No podrá continuar, el campo {campo} contiene caracteres no permitidos”. 7. El sistema valide que no se puedan asociar articulo “Compuestos” a la bodegas. Si el usuario selecciona la bodega para un artículo compuesto el sistema mostrará el siguiente mensaje: “No podrá continuar, los artículos compuestos no se pueden relacionar directamente a una bodega.”
Flujos Alternos	
Flujo Alternativo No. 1	<p>Si existe algún error en el proceso:</p> <ol style="list-style-type: none"> 1. El sistema registra el error. 2. El sistema muestra el siguiente mensaje de error: “No se puede realizar la acción dado a que se presentó un problema. <ol style="list-style-type: none"> a) Intente ejecutar nuevamente la acción.

	<p>b) Si el problema persiste consulte al administrador del sistema.</p> <p>Identificador del error: {Identificador}”</p> <p>3. El sistema registra el Log de error.</p> <p>4. El caso de uso finaliza.</p>
Flujo Alternativo No. 2	<p>Si existente errores de validación:</p> <p>1. El sistema muestra el mensaje de error.</p> <p>2. El caso de uso finaliza.</p>
Flujo Alternativo No. 3	<p>Si el actor ejecuta la acción “cancelar” o “limpiar”:</p> <p>1. El sistema limpia la información completada.</p> <p>2. El caso de uso finaliza.</p>
Flujo Alternativo No. 4	<p>Si el actor no dispone del rol Administrador:</p> <p>1. El sistema muestra el siguiente mensaje: “Acceso Denegado!!! El usuario no cuenta con los requisitos para poder ingresar a esta opción.”</p> <p>2. El caso de uso finaliza.</p>
Flujo Alternativo No. 5	<p>Si ejecuta la acción “regresar” al buscar</p> <p>1. El sistema continua en el paso 2 del Flujo Base.</p> <p>2. El caso de uso finaliza.</p>
Flujo Alternativo No. 6	<p>Si el actor selecciona la Unidad de Medida “Otros”</p> <p>1. El sistema habilita el campo Unidad Medida Comercial.</p> <p>2. El caso de uso finaliza.</p>
Flujo Alternativo No. 7	<p>Si el actor selecciona el campo “Es Servicio”</p> <p>1. El sistema deshabilita la opción Afecta Inventario.</p> <p>2. El sistema deshabilita y desmarca la asignación de Bodegas.</p> <p>3. El caso de uso finaliza.</p>
Flujo Alternativo No. 8	<p>Si el actor selecciona el campo “Afecta inventario”</p> <p>1. El sistema deshabilita la opción Es Servicio.</p> <p>2. El sistema habilita la asignación de Bodegas.</p> <p>3. El caso de uso finaliza.</p>
Requerimientos especiales	
No aplica.	
Post-Condiciones	

Fuente: Elaboración propia (2024).

Tabla 17.

Caso de uso para administrar usuarios

Prototipo: Prototipo funcional para la gestión de ventas de la empresa Zapatitos Lindos S.A., ubicada en San José.	
Número Caso de Uso: 8	Nombre del Caso de Uso: Administrar usuarios.
Fecha elaboración:	17/04/2024
Descripción Caso de Uso:	Permitir la administración de usuarios, asignar módulos de uso, asignar sucursales y definir permisos, así como establecer el rol que contiene los accesos correspondientes.
Autor caso de uso:	Daniel David Quesada Fallas.
Actores relacionados:	Administrador.
Precondiciones:	El actor debe haber iniciado sesión en el sistema y disponer del Rol "Administrador".
Flujo Básico del caso de uso	
<p>1. Este caso de uso inicia cuando el actor selecciona del menú “Seguridad” la opción Usuarios.</p> <p>2. El sistema solicita la siguiente información:</p> <p><u>Datos de usuario</u></p> <ul style="list-style-type: none"> • Usuario. • Nombre. • Apellido. • Correo. • Contraseña. • Confirmar contraseña. • Rol (Administrador, Usuario Estándar, Contador). • Estado (Activo, Inactivo, Bloqueado). <p><u>Asignación de módulos</u></p> <ul style="list-style-type: none"> • Módulos (Selección múltiple). <p><u>Asignación de sucursales</u></p> <ul style="list-style-type: none"> • Sucursales (Selección múltiple). <p><u>Asignación de permisos</u> (Opcional).</p> <ul style="list-style-type: none"> • Permisos (Selección múltiple). <p>3. El actor completa la información y ejecuta la acción Grabar.</p> <p>4. El sistema ejecuta las validaciones (SF-04).</p> <p>5. El sistema registra o actualiza la información.</p> <p>6. El caso de uso finaliza.</p>	
Sub Flujos	
Sub Flujo Buscar (SF-01)	<p>1. El sub flujo inicia cuando el actor ejecuta la acción “Buscar”.</p> <p>2. El sistema solicita la siguiente información.</p> <ul style="list-style-type: none"> • <u>Filtros</u>: Usuario, Nombre completo • <u>Estado</u>: Todos, Activo, Inactivo, Bloqueado • <u>Opciones</u>: Que inicie, Que contenga, Que finalice • <u>Llave de búsqueda</u>

	<ol style="list-style-type: none"> 3. El actor selecciona una opción del apartado “Filtros”. 4. El actor selecciona una opción del apartado “Opciones”. 5. El actor ejecuta la acción “Buscar”. 6. El sistema muestra un listado con los usuarios que cumplan con el criterio de búsqueda. 7. El sistema muestra las siguientes opciones: “Editar”, “Bloquear” o “Desbloquear”.
Sub Flujo Editar registro (SF-02)	<ol style="list-style-type: none"> 1. Inicia cuando el actor ejecuta la acción “Editar” en el paso 7 del Sub Flujo Buscar (SF-01) 2. El sistema carga la información del usuario seleccionado. 3. El sistema continúa en el paso 3 del flujo case.
Sub Flujo Bloquear o Desbloquear usuario (SF-03)	<ol style="list-style-type: none"> 1. El sub flujo inicia cuando el actor ejecuta la acción “Bloquear” o “desbloquear” en el paso 7 del Sub Flujo Buscar (SF-01). 2. El sistema solicita confirmación de la acción. 3. El actor confirma la acción. 4. El caso de uso finaliza.
Sub Flujo Validaciones (SF-04)	<ol style="list-style-type: none"> 1. El sub flujo inicia en el paso 4 del flujo base. 2. El sistema verifica que el actor completó la información en la sección “Datos de usuario”. Si actor no completó los datos, el sistema muestra el siguiente mensaje de error: “El campo “{NombreDelCampo} es requerido”. 3. El sistema verifica que la contraseña sea robusta. Debe tener como mínimo 5 caracteres que sean al menos un número, una letra mayúscula y una minúscula. Si la contraseña no cumple con requisitos mínimos el sistema muestra el siguiente mensaje de error: “La contraseña debe contar con mínimo 5 caracteres de los cuales debe existir al menos un número, una letra mayúscula y una minúscula”. 4. El sistema verifica que los campos Contraseña y Confirman contraseña coincidan. Si la información digitada no coincide el sistema muestra el siguiente mensaje de error: “El campo “Contraseña” y “Confirmar Contraseña” deben coincidir.” 5. El sistema verifica que el actor haya seleccionado al menos un módulo. Si no seleccionó módulos y el estado es activo, el sistema muestra el siguiente mensaje de error: “No podrá continuar, debe seleccionar al menos un módulo.” 6. El sistema verifica que el actor haya seleccionado al menos una sucursal. Si no seleccionó sucursales y el estado es activo, el sistema muestra el siguiente mensaje de error: “No podrá continuar, debe seleccionar al menos una sucursal.” 7. El sistema verifica que la dirección de correo digitada sea válida. Si es inválida el sistema muestra el siguiente mensaje de error: “El campo “Correo” no cumple con los requisitos mínimos de una dirección de correo electrónico.” 8. Si el actor no ha ejecutado el Sub Flujo (SF-02), el sistema llevará a cabo una validación para asegurarse de que no existan

	registros duplicados. En caso de que se detecte un registro previo con la misma información, el sistema mostrará un mensaje de error que indica: "No puede continuar, ya existe un registro con la información ingresada; si desea agregar uno nuevo, modifique los datos del usuario."
Flujos Alternos	
Flujo Alternativo No. 1	<p>Si existe algún error en el proceso:</p> <ol style="list-style-type: none"> 1. El sistema registra el error. 2. El sistema muestra el siguiente mensaje de error: No se puede realizar la acción dado a que se presentó un problema. <ol style="list-style-type: none"> a) Intente ejecutar nuevamente la acción. b) Si el problema persiste consulte al administrador del sistema. Identificador del error: {Identificador} 3. El sistema registra el Log de error. 4. El caso de uso finaliza
Flujo Alternativo No. 2	<p>Si existen errores de validación:</p> <ol style="list-style-type: none"> 1. El sistema muestra el mensaje de error. 2. El caso de uso finaliza.
Flujo Alternativo No. 3	<p>Si el actor ejecuta la acción cancelar:</p> <ol style="list-style-type: none"> 1. El sistema limpia la información completada. 2. El caso de uso finaliza.
Flujo Alternativo No. 4	<p>Si el actor no dispone del rol Administrador:</p> <ol style="list-style-type: none"> 1. El sistema muestra el siguiente mensaje: "Acceso Denegado!!! El usuario no cuenta con los requisitos para poder ingresar a esta opción." 2. El caso de uso finaliza.
Flujo Alternativo No. 5	<p>Si no existen sucursales registradas en estado "Activo"</p> <ol style="list-style-type: none"> 1. El sistema muestra el siguiente mensaje: "No existen sucursales para mostrar". 2. El sistema no permite ejecutar la acción "Grabar". 3. El caso de uso finaliza.
Flujo Alternativo No. 6	<p>Si no existen módulos en estado "Activo"</p> <ol style="list-style-type: none"> 1. El sistema muestra el siguiente mensaje: No existen sucursales para mostrar. 2. El sistema no permite ejecutar la acción "Grabar" 3. El caso de uso finaliza.
Flujo Alternativo No. 7	<p>Si no existen permisos para mostrar</p> <ol style="list-style-type: none"> 1. El sistema muestra el siguiente mensaje: "No existen permisos para mostrar". 2. El sistema permite ejecutar la acción "Grabar" 3. El caso de uso finaliza.
Flujo Alternativo No. 8	<p>Si ejecuta la acción "regresar" al buscar</p> <ol style="list-style-type: none"> 3. El sistema

	4. El sistema permite ejecutar la acción “Grabar” 5. El caso de uso finaliza.
Requerimientos especiales	
No aplica.	
Post-Condiciones	
Una vez que se haya registrado o actualizado la información y el estado esté configurado como activo, el usuario registrado o modificado podrá acceder al sistema y utilizarlo de acuerdo con su rol, los módulos y sucursales seleccionadas. Sin embargo, si el actor bloquea o desactiva al usuario, este no tendrá acceso al sistema.	

Fuente: Elaboración propia (2024).

Tabla 18.

Caso de uso para administrar sucursales

Prototipo: Prototipo funcional para la gestión de ventas de la empresa Zapaticos Lindos S.A., ubicada en San José.	
Número Caso de Uso: 9	Nombre del Caso de Uso: Administrar Sucursales.
Fecha elaboración:	17/04/2024
Descripción Caso de Uso:	Permitir la administración de sucursales.
Autor caso de uso:	Daniel David Quesada Fallas.
Actores relacionados:	Administrador.
Precondiciones:	1. El actor debe haber iniciado sesión en el sistema y disponer del Rol "Administrador". 2. El actor debe estar asociado previamente a una compañía y una sucursal.
Flujo Básico del caso de uso	
<ol style="list-style-type: none"> 1. Este caso de uso inicia cuando el actor selecciona del menú “Mantenimientos” > “Sistema” la opción “Sucursal”. 2. El sistema solicita la siguiente información: <ul style="list-style-type: none"> • Nombre. * • Correo. • Dirección. • Código de área teléfono. • Teléfono. • Código de área teléfono 2. • Teléfono 2. • Estado (1-Inactiva, 2- Activa). * <p>* <u>Campos requeridos.</u></p> 3. El actor completa la información y ejecuta la acción Grabar. 4. El sistema ejecuta las validaciones (SF-03). 5. El sistema registra o actualiza la información. 6. El caso de uso finaliza. 	
Sub Flujos	

	4. El caso de uso finaliza
Flujo Alternativo No. 2	Si existen errores de validación: 1. El sistema muestra el mensaje de error. 2. El caso de uso finaliza.
Flujo Alternativo No. 3	Si el actor ejecuta la acción “cancelar” o “limpiar”: 1. El sistema limpia la información completada. 2. El caso de uso finaliza.
Flujo Alternativo No. 4	Si el actor no dispone del rol Administrador: 1. El sistema muestra el siguiente mensaje: “Acceso Denegado!!! El usuario no cuenta con los requisitos para poder ingresar a esta opción.” 2. El caso de uso finaliza.
Flujo Alternativo No. 5	Si ejecuta la acción “regresar” al buscar 1. El sistema continúa en el paso 2 del flujo base. 2. El caso de uso finaliza.
Requerimientos especiales	
No aplica.	
Post-Condiciones	
Se habilita el sistema para emitir documentos electrónicos.	

Fuente: Elaboración propia (2024).

Tabla 19.

Caso de uso para administrar perfiles cliente

Prototipo: Prototipo funcional para la gestión de ventas de la empresa Zapatitos Lindos S.A., ubicada en San José.	
Número Caso de Uso: 10	Nombre del Caso de Uso: Administrar perfiles cliente.
Fecha elaboración:	18/04/2024
Descripción Caso de Uso:	Permitir la administración de perfiles de clientes. El actor podrá indicar que lista de precio utilizará el tipo de cliente, si desea habilitar el crédito, si el cliente es Exonerado, si el cliente es la Caja Costarricense Del Seguro Social entre otras opciones.
Autor caso de uso:	Daniel David Quesada Fallas.
Actores relacionados:	Administrador.
Precondiciones:	2. El actor debe haber iniciado sesión en el sistema y disponer del Rol "Administrador".
Flujo Básico del caso de uso	
1. Este caso de uso inicia cuando el actor selecciona del menú “ Mantenimientos ” > “ Facturación ” la opción “ Perfiles Clientes ”. 2. El sistema solicita la siguiente información: <ul style="list-style-type: none"> • Descripción. • Lista precio: Precio1, Precio2, Precio 3. • Habilitar Crédito. 	

<ul style="list-style-type: none"> • Es Exonerado. • Es CCSS. • Descuento permanente. • Descuento. • Rango de fechas del descuento. • Estado: Activo, Inactivo. <p>3. El actor completa la información y ejecuta la acción Grabar.</p> <p>4. El sistema ejecuta las validaciones (SF-03).</p> <p>5. El sistema registra o actualiza la información.</p> <p>6. El caso de uso finaliza.</p>	
Sub Flujos	
Sub Flujo Buscar Artículo (SF-01)	<ol style="list-style-type: none"> 1. El sub flujo inicia cuando el actor ejecuta la acción “Buscar”. 2. El sistema solicita la siguiente información. <ul style="list-style-type: none"> • <u>Estado</u>: Todas, Activa, Inactiva • <u>Opciones</u>: Que inicie, Que contenga, Que finalice • <u>Llave de búsqueda</u> 3. El actor selecciona el Estado. 4. El actor selecciona una opción del apartado “Opciones”. 5. El actor ejecuta la acción “Buscar” 6. El sistema muestra un listado con los artículos que cumplan con el criterio de búsqueda. 7. El sistema muestra la siguiente opción: “Editar”.
Sub Flujo Editar (SF-02)	<ol style="list-style-type: none"> 1. Inicia cuando el actor ejecuta la acción “Editar” en el paso 7 del Sub Flujo Buscar (SF-01). 2. El sistema carga la información del perfil de cliente seleccionado. 3. El sistema continúa en el paso 3 del flujo base.
Sub Flujo Validaciones (SF-03)	<ol style="list-style-type: none"> 1. El sub flujo inicia en el paso 4 del flujo base. 2. El sistema verifica que el actor completó la información de aquellos datos requeridos. Si actor no completó los datos, el sistema muestra el siguiente mensaje de error: “El campo “{NombreDelCampo} es requerido.” 3. El sistema verifica que previamente no exista un registro con los datos ingresados. Si existe un registro previo el sistema muestra el siguiente mensaje de error: “No puede continuar, ya existe un registro con la información ingresada.” 4. El sistema verifica que el porcentaje de descuento no sea menor a cero o mayor igual a 100%; si no se cumple con el criterio el sistema muestra el siguiente error: El campo “Porcentaje” es inválido.
Flujos Alternos	
Flujo Alternativo No. 1	<p>Si existe algún error en el proceso:</p> <ol style="list-style-type: none"> 1. El sistema registra el error. 2. El sistema muestra el siguiente mensaje de error: No se puede realizar la acción dado a que se presentó un

	<p>problema.</p> <ol style="list-style-type: none"> a) Intente ejecutar nuevamente la acción. b) Si el problema persiste consulte al administrador del sistema. <p>Identificador del error: {Identificador}</p> <ol style="list-style-type: none"> 3. El sistema registra el Log de error. 4. El caso de uso finaliza
Flujo Alternativo No. 2	<p>Si existen errores de validación:</p> <ol style="list-style-type: none"> 1. El sistema muestra el mensaje de error. 2. El caso de uso finaliza.
Flujo Alternativo No. 3	<p>Si el actor ejecuta la acción “cancelar”:</p> <ol style="list-style-type: none"> 1. El sistema limpia la información completada. 2. El caso de uso finaliza.
Flujo Alternativo No. 4	<p>Si el actor no dispone del rol Administrador:</p> <ol style="list-style-type: none"> 1. El sistema muestra el siguiente mensaje: “Acceso Denegado!!! El usuario no cuenta con los requisitos para poder ingresar a esta opción.” 2. El caso de uso finaliza.
Flujo Alternativo No. 5	<p>Si ejecuta la acción “regresar” al buscar</p> <ol style="list-style-type: none"> 1. El sistema continúa en el paso 2 del flujo base. 2. El caso de uso finaliza.
Requerimientos especiales	
No aplica.	
Post-Condiciones	

Fuente: Elaboración propia (2024).

Tabla 20.

Caso de uso para administrar clientes

Prototipo: Prototipo funcional para la gestión de ventas de la empresa Zapaticos Lindos S.A., ubicada en San José.	
Número Caso de Uso: 11	Nombre del Caso de Uso: Administrar clientes.
Fecha elaboración:	18/04/2024
Descripción Caso de Uso:	Permitir la administración de clientes y la gestión de la información relacionada con los receptores de factura electrónica.
Autor caso de uso:	Daniel David Quesada Fallas.
Actores relacionados:	Administrador, Usuario Estándar.
Precondiciones:	<ol style="list-style-type: none"> 1. El actor debe haber iniciado sesión en el sistema y disponer de alguno de los siguientes roles: “Administrador”, “Usuario Estándar”. 2. Debe existir al menos un perfil de cliente.
Flujo Básico del caso de uso	
1. Este caso de uso inicia cuando el actor selecciona del menú “ Mantenimientos ” >	

“Facturación” la opción “Clientes”.

2. El sistema solicita la siguiente información:

Datos del Cliente

- Perfil Cliente.
- Tipo Identificación.
- Identificación.
- Nombre.
- Correo. *
- Correo Facturación Electrónica.
- Código de área teléfono.
- Teléfono.
- Código área fax.
- Fax.
- Provincia.
- Cantón.
- Distrito.
- Barrio.
- Dirección (otras señas).
- Límite de crédito.
- Plazo.
- Unidad ejecutora.
- Representante Legal.
- Contacto.
- Contacto Cobro.
- Estado (activo, inactivo).

Datos Exoneración

- Tipo documento de exoneración.
- Número documento.
- Institución.
- Fecha de exoneración.
- Porcentaje de exoneración.

3. El actor completa la información y ejecuta la acción Grabar.

4. El sistema ejecuta las validaciones. (SF-03)

5. El sistema registra o actualiza la información.

6. El caso de uso finaliza.

Sub Flujos

**Sub Flujo Buscar
Artículo.(SF-01)**

1. El sub flujo inicia cuando el actor ejecuta la acción “Buscar”.
2. El sistema solicita la siguiente información.
 Buscar por: Identificación, Nombre
 Llave de búsqueda
 Estado: “Todas”, “Activa”, “Inactiva”
 Filtro opciones: Que inicie, Que contenga, Que finalice
- 3.El actor selecciona e ingresa la información solicitada.
4. El actor ejecuta la acción “Buscar”

	<ol style="list-style-type: none"> 5. El sistema muestra un listado con los clientes que cumplan con el criterio de búsqueda. 6. El sistema muestra la siguiente opción: “Editar”
Sub Flujo Editar (SF-02)	<ol style="list-style-type: none"> 1. Inicia cuando el actor ejecuta la acción “Editar” en el paso 7 del Sub Flujo Buscar (SF-01). 2. El sistema carga la información del cliente seleccionado. 3. El sistema continúa en el paso 3 del flujo base.
Sub Flujo Validaciones (SF-03)	<ol style="list-style-type: none"> 1. El sub flujo inicia en el paso 4 del flujo base. 2. El sistema verifica que el actor completo la información de aquellos datos requeridos. Si actor no completo los datos, el sistema muestra el siguiente mensaje de error: “El campo “{NombreDelCampo} es requerido.” 3. El sistema verifica que previamente no exista un registro con los datos ingresados. Si existe un registro previo el sistema muestra el siguiente mensaje de error: “No puede continuar, ya existe un registro con la información ingresada.” 4. Si el actor está registrando o modificando la información de un cliente de crédito, el sistema verifica que cuente con el rol “Administrador”, en caso de contar con el rol el sistema mostrará el siguiente mensaje de error: “No podrá continuar el usuario no cuenta con el rol autorizado para crear clientes de crédito.” 5. Si el actor está registrando o modificando la información de un cliente que tenga un descuento global habilitado por medio del perfil del cliente, el sistema verifica que el usuario cuente con el rol “Administrado”, en caso de contar con rol requerido el sistema mostrará el siguiente mensaje: “No podrá continuar el usuario no cuenta con el rol autorizado para crear clientes que tengan habilitado un descuento global de acuerdo con el perfil del cliente”. 6. Si el actor está registrando o modificando la información de un cliente exonerado según el perfil de cliente, el sistema verifica que la sección Datos de Exoneración contengan datos. Si actor no completo los datos, el sistema muestra el siguiente mensaje de error: “El campo “{NombreDelCampo} es requerido.” 7. Si el tipo de identificación es “Cliente de Contado”, el sistema verifica que no se digite la identificación.
Flujos Alternos	
Flujo Alternativo No. 1	<p>Si existe algún error en el proceso:</p> <ol style="list-style-type: none"> 1. El sistema registra el error. 2. El sistema muestra el siguiente mensaje de error: No se puede realizar la acción dado a que se presentó un problema”. <ol style="list-style-type: none"> a) Intente ejecutar nuevamente la acción. b) Si el problema persiste consulte al administrador del

	<p>sistema.</p> <p>Identificador del error: {Identificador}</p> <p>3. El sistema registra el Log de error.</p> <p>4. El caso de uso finaliza</p>
Flujo Alternativo No. 2	<p>Si existen errores de validación:</p> <p>1. El sistema muestra el mensaje de error.</p> <p>2. El caso de uso finaliza.</p>
Flujo Alternativo No. 3	<p>Si el actor ejecuta la acción “cancelar”:</p> <p>1. El sistema limpia la información completada.</p> <p>2. El caso de uso finaliza.</p>
Flujo Alternativo No. 4	<p>Si el actor no dispone del rol Administrador o Usuario Estándar:</p> <p>1. El sistema muestra el siguiente mensaje: “Acceso Denegado!!! El usuario no cuenta con los requisitos para poder ingresar a esta opción.”</p> <p>2. El caso de uso finaliza.</p>
Flujo Alternativo No. 5	<p>Si ejecuta la acción “regresar” al buscar</p> <p>1. El sistema continúa en el paso 2 del flujo base.</p> <p>2. El caso de uso finaliza.</p>
Requerimientos especiales	
El Usuario Estándar únicamente podrá registrar o modificar información de clientes que no tengan crédito habilitado y que no cuenten con un descuento global.	
Post-Condiciones	

Fuente: Elaboración propia (2024).

Tabla 21.

Caso de uso para administrar pacientes

Prototipo: Prototipo funcional para la gestión de ventas de la empresa Zapatitos Lindos S.A., ubicada en San José.	
Número Caso de Uso: 12	Nombre del Caso de Uso: Administrar pacientes.
Fecha elaboración:	18/04/2024
Descripción Caso de Uso:	Permitir la administración de pacientes y la gestión de los datos del paciente requeridos para realizar adaptaciones al calzado.
Autor caso de uso:	Daniel David Quesada Fallas.
Actores relacionados:	Administrador, Usuario Estándar.
Precondiciones:	1. El actor debe haber iniciado sesión en el sistema y disponer de alguno de los siguientes roles: “Administrador”, “Usuario Estándar”.
Flujo Básico del caso de uso	
<p>1. Este caso de uso inicia cuando el actor selecciona del menú “Mantenimientos” > “Facturación” la opción “Pacientes”.</p> <p>2. El sistema solicita la siguiente información:</p>	

<p><u>Datos del Paciente</u></p> <ul style="list-style-type: none"> • Tipo Identificación. • Identificación. • Nombre. • Correo. • Teléfono. • Estado (Activo, Inactivo). • Observación. <p><u>Datos Plantilla</u></p> <ul style="list-style-type: none"> • Arco Longitudinal. • Arco Transverso. • Virón. • Espolón. • Soporte Ant. • Otros. <p><u>Datos Suela</u></p> <ul style="list-style-type: none"> • Thomas. • Tacón corriente. • Virón. • Otros. <p>3. El actor completa la información y ejecuta la acción Grabar. 4. El sistema ejecuta las validaciones (SF-03). 5. El sistema registra o actualiza la información. 6. El caso de uso finaliza.</p>	
Sub Flujos	
<p>Sub Flujo Buscar Artículo (SF-01)</p>	<ol style="list-style-type: none"> 1. El sub flujo inicia cuando el actor ejecuta la acción “Buscar”. 2. El sistema solicita la siguiente información. Buscar por: Identificación, Nombre. Llave de búsqueda. Estado: “Todas”, “Activa”, “Inactiva”. Filtro opciones: Que inicie, Que contenga, Que finalice. 3. El actor ingresa la información solicitada. 4. El actor ejecuta la acción “Buscar” 5. El sistema muestra un listado con los pacientes que cumplan con el criterio de búsqueda. 6. El sistema muestra la siguiente opción: “Editar”.
<p>Sub Flujo Editar (SF-02)</p>	<ol style="list-style-type: none"> 1. Inicia cuando el actor ejecuta la acción “Editar” en el paso 6 del Sub Flujo Buscar (SF-01). 2. El sistema carga la información del paciente seleccionado. 3. El sistema continua en el paso 3 del flujo base.
<p>Sub Flujo Validaciones (SF-03)</p>	<ol style="list-style-type: none"> 1. El sub flujo inicia en el paso 4 del flujo base. 2. El sistema verifica que el actor completó la información de aquellos datos requeridos. Si actor no completó los datos, el sistema muestra el siguiente mensaje de error: “El campo “{NombreDelCampo} es requerido.”

	<p>3. El sistema verifica que previamente no exista un registro con los datos ingresados. Si existe un registro previo el sistema muestra el siguiente mensaje de error: “No puede continuar, ya existe un registro con la información ingresada.”</p> <p>4. El sistema verifica el formato de la identificación cuando se posea información en el campo “Tipo de identificación del emisor. Formato: La “Cédula física” debe de contener 9 dígitos, sin cero al inicio y sin guiones. La “cédula de personas Jurídicas” debe contener 10 dígitos y sin guiones. El “Documento de Identificación Migratorio para Extranjeros (DIMEX)” debe contener 11 o 12 dígitos, sin ceros al inicio y sin guiones. El “Documento de Identificación de la DGT (NITE)” debe contener 10 dígitos y sin guiones. Si la identificación no cumple con el formato el sistema muestra el siguiente mensaje de error: “La identificación no cumple con el siguiente formato {#####}”.</p> <p>5. El sistema verifica que la dirección de correo digitada sea válida. Debe cumplir con la siguiente estructura: $^\\ s*(([^<O\\[\\ \\.,:;\\ s@\\ \\ "]+)(\\.[^<O\\[\\ \\.,:;\\ s@\\ \\ "]+)* (\\ \\ ".+\\ \\ ")) @(([^<O\\[\\ \\.,:;\\ s@\\ \\ "]+\\.)+[<O\\[\\ \\.,:;\\ s@\\ \\ "]{0,})\\ s*\$.$ </p> <p>6. Si el correo es inválido el sistema muestra el siguiente mensaje de error: “El campo “Correo” no cumple con los requisitos mínimos de una dirección de correo electrónico.”</p>
Flujos Alternos	
Flujo Alternativo No. 1	<p>Si existe algún error en el proceso:</p> <ol style="list-style-type: none"> 1. El sistema registra el error. 2. El sistema muestra el siguiente mensaje de error: “No se puede realizar la acción dado a que se presentó un problema. <ol style="list-style-type: none"> a) Intente ejecutar nuevamente la acción. b) Si el problema persiste consulte al administrador del sistema. Identificador del error: {Identificador} 3. El sistema registra el Log de error. 4. El caso de uso finaliza
Flujo Alternativo No. 2	<p>Si existen errores de validación:</p> <ol style="list-style-type: none"> 1. El sistema muestra el mensaje de error. 2. El caso de uso finaliza.
Flujo Alternativo No. 3	<p>Si el actor ejecuta la acción “cancelar”:</p> <ol style="list-style-type: none"> 1. El sistema limpia la información completada. 2. El caso de uso finaliza.
Flujo Alternativo No. 4	<p>Si el actor no dispone del rol Administrador o Usuario Estándar:</p> <ol style="list-style-type: none"> 1. El sistema muestra el siguiente mensaje: “Acceso Denegado!!!”

	El usuario no cuenta con los requisitos para poder ingresar a esta opción.” 2. El caso de uso finaliza.
Flujo Alternativo No. 5	Si ejecuta la acción “regresar” al buscar 1. El sistema continúa en el paso 2 del flujo base. 2. El caso de uso finaliza.
Requerimientos especiales	
Post-Condiciones	

Fuente: Elaboración propia (2024).

Tabla 22.

Caso de uso para administrar bodegas

Prototipo: Prototipo funcional para la gestión de ventas de la empresa Zapatitos Lindos S.A., ubicada en San José.	
Número Caso de Uso: 12	Nombre del Caso de Uso: Administrar bodegas.
Fecha elaboración:	18/04/2024
Descripción Caso de Uso:	Permitir la administración de bodegas para el almacenamiento del inventario.
Autor caso de uso:	Daniel David Quesada Fallas.
Actores relacionados:	Administrador.
Precondiciones:	1. El actor debe haber iniciado sesión en el sistema y disponer de alguno de los siguientes roles: “Administrador”.
Flujo Básico del caso de uso	
<ol style="list-style-type: none"> 1. Este caso de uso inicia cuando el actor selecciona del menú “Mantenimientos” > “Inventario” la opción “Bodegas”. 2. El sistema solicita la siguiente información: <ul style="list-style-type: none"> • Descripción. • Factura. • Es Principal. • Estado (Activo, Inactivo). 3. El actor completa la información y ejecuta la acción Grabar. 4. El sistema ejecuta las validaciones (SF-03). 5. El sistema registra o actualiza la información. 6. El caso de uso finaliza. 	
Sub Flujos	
Sub Flujo Buscar Artículo (SF-01)	<ol style="list-style-type: none"> 1. El sub flujo inicia cuando el actor ejecuta la acción “Buscar”. 2. El sistema solicita la siguiente información. Llave de búsqueda Estado: “Todas”, “Activa”, “Inactiva”. Filtro opciones: Que inicie, Que contenga, Que finalice.

	<p>3.El actor ingresa la información solicitada.</p> <p>4. El actor ejecuta la acción “Buscar”.</p> <p>5. El sistema muestra un listado con las bodegas que cumplan con el criterio de búsqueda.</p> <p>6. El sistema muestra la siguiente opción: “Editar”</p>
Sub Flujo Editar (SF-02)	<p>1. Inicia cuando el actor ejecuta la acción “Editar” en el paso 6 del Sub Flujo Buscar (SF-01).</p> <p>2. El sistema carga la información de la bodega seleccionado.</p> <p>3. El sistema continúa en el paso 3 del flujo base.</p>
Sub Flujo Validaciones (SF-03)	<p>1. El sub flujo inicia en el paso 4 del flujo base.</p> <p>2. El sistema verifica que el actor completó la información de aquellos datos requeridos. Si actor no completó los datos, el sistema muestra el siguiente mensaje de error: “El campo “{NombreDelCampo} es requerido.”</p> <p>1. El sistema verifica que previamente no exista un registro con los datos ingresados. Si existe un registro previo el sistema muestra el siguiente mensaje de error: “No puede continuar, ya existe un registro con la información ingresada.”</p> <p>2. El sistema verifica que solo exista una bodega principal, en caso de que exista una bodega como principal el sistema mostrará la siguiente mensaje de error: “No podrá continuar, ya existe una bodega como principal.”</p> <p>3. Si al modificar el actor desea inactivar el registro, el sistema valida que no existan existencias en la bodega; en caso de que existan el sistema mostrará el siguiente mensaje de error: “No podrá continuar, la bodega no se puede inactivar, verifique que todos los artículos relacionados a la bodega tengan saldo igual a 0”.</p>
Flujos Alternos	
Flujo Alternativo No. 1	<p>Si existe algún error en el proceso:</p> <p>1. El sistema registra el error.</p> <p>2. El sistema muestra el siguiente mensaje de error: “No se puede realizar la acción dado a que se presentó un problema”.</p> <p style="padding-left: 40px;">a) Intente ejecutar nuevamente la acción.</p> <p style="padding-left: 40px;">b) Si el problema persiste consulte al administrador del sistema.</p> <p style="padding-left: 40px;">Identificador del error: {Identificador}”</p> <p>3. El sistema registra el Log de error.</p> <p>4. El caso de uso finaliza</p>
Flujo Alternativo No. 2	<p>Si existen errores de validación:</p> <p>1. El sistema muestra el mensaje de error.</p> <p>2. El caso de uso finaliza.</p>
Flujo Alternativo No. 3	<p>Si el actor ejecuta la acción “cancelar”:</p> <p>1. El sistema limpia la información completada.</p>

	2. El caso de uso finaliza.
Flujo Alternativo No. 4	Si el actor no dispone del rol Administrador: 1. El sistema muestra el siguiente mensaje: “Acceso Denegado!!! El usuario no cuenta con los requisitos para poder ingresar a esta opción.” 2. El caso de uso finaliza.
Flujo Alternativo No. 5	Si ejecuta la acción “regresar” al buscar 1. El sistema continúa en el paso 2 del flujo base. 2. El caso de uso finaliza.
Requerimientos especiales	
Post-Condiciones	

Fuente: Elaboración propia (2024).

Tabla 23.

Caso de uso para administrar proveedores

Prototipo: Prototipo funcional para la gestión de ventas de la empresa Zapatitos Lindos S.A., ubicada en San José.	
Número Caso de Uso: 13	Nombre del Caso de Uso: Administrar Proveedores.
Fecha elaboración:	19/04/2024
Descripción Caso de Uso:	Permitir la administración de proveedores y la gestión de datos de proveedores de crédito o contado.
Autor caso de uso:	Daniel David Quesada Fallas.
Actores relacionados:	Administrador.
Precondiciones:	1. El actor debe haber iniciado sesión en el sistema. 2. El actor debe contar con acceso al módulo de cuentas por pagar. 3. El actor debe contar con el rol “Administrador”.
Flujo Básico del caso de uso	
<p>1. Este caso de uso inicia cuando el actor desde el módulo de cuentas de por pagar selecciona del menú “Mantenimientos” la opción “Proveedores”.</p> <p>2. El sistema solicita la siguiente información:</p> <ul style="list-style-type: none"> • Tipo identificación. • Identificación. • Nombre. • Dirección. • Contacto. • Código área teléfono. • Teléfono. • Tiene Crédito. • Plazo. • Correo. • Otros. • Estado (Activo, Inactivo). 	

<p>3. El actor completa la información y ejecuta la acción Grabar.</p> <p>4. El sistema ejecuta las validaciones (SF-03).</p> <p>5. El sistema registra o actualiza la información.</p> <p>6. El caso de uso finaliza.</p>	
Sub Flujos	
<p>Sub Flujo Buscar Artículo (SF-01)</p>	<p>1. El sub flujo inicia cuando el actor ejecuta la acción “Buscar”.</p> <p>2. El sistema solicita la siguiente información.</p> <ul style="list-style-type: none"> • <u>Buscar por:</u> Identificación, Nombre. • <u>Estado:</u> “Todas”, “Activa”, “Inactiva” • <u>Llave de búsqueda.</u> • <u>Filtro opciones:</u> Que inicie, Que contenga, Que finalice. <p>3. El actor ingresa la información solicitada.</p> <p>4. El actor ejecuta la acción “Buscar”</p> <p>5. El sistema muestra un listado con los proveedores que cumplan con el criterio de búsqueda.</p> <p>6. El sistema muestra la siguiente opción: “Editar”</p>
<p>Sub Flujo Editar (SF-02)</p>	<p>1. Inicia cuando el actor ejecuta la acción “Editar” en el paso 6 del Sub Flujo Buscar (SF-01).</p> <p>2. El sistema carga la información del proveedor seleccionado.</p> <p>3. El sistema continúa en el paso 3 del flujo base.</p>
<p>Sub Flujo Validaciones (SF-03)</p>	<p>1. El sub flujo inicia en el paso 4 del flujo base.</p> <p>2. El sistema verifica que el actor completó la información de aquellos datos requeridos. Si actor no completó los datos, el sistema muestra el siguiente mensaje de error: “El campo “{NombreDelCampo} es requerido.”</p> <p>3. El sistema verifica que previamente no exista un registro con los datos ingresados. Si existe un registro previo el sistema muestra el siguiente mensaje de error: “No puede continuar, ya existe un registro con la información ingresada.”</p> <p>4. El sistema verifica el formato de la identificación cuando se posea información en el campo “Tipo de identificación del emisor. Formato: La “Cédula física” debe de contener 9 dígitos, sin cero al inicio y sin guiones. La “cédula de personas Jurídicas” debe contener 10 dígitos y sin guiones. El “Documento de Identificación Migratorio para Extranjeros (DIMEX)” debe contener 11 o 12 dígitos, sin ceros al inicio y sin guiones. El “Documento de Identificación de la DGT (NITE)” debe contener 10 dígitos y sin guiones. Si la identificación no cumple con el formato el sistema muestra el siguiente mensaje de error: “La identificación no cumple con el siguiente formato {#####}.”</p> <p>5. El sistema verifica que la dirección de correo digitada sea válida. Debe cumplir con la siguiente estructura: $\wedge s*((\wedge < O \wedge [\wedge] \wedge .,; : \wedge s @ \wedge \wedge \wedge] + (\wedge . [\wedge < O \wedge [\wedge] \wedge .,; : \wedge s @ \wedge \wedge \wedge] + *) (\wedge \wedge . + \wedge \wedge \wedge \wedge))$ </p>

	<p>@((([^<>()\\[\]\/\.,;:\s@\\"]+\\.)+[<>()\\[\]\/\.,;:\s@\\"]{0,})\s*\$.</p> <p>Si el correo es invalido el sistema muestra el siguiente mensaje de error: “El campo “Correo” no cumple con los requisitos mínimos de una dirección de correo electrónico.”</p>
Flujos Alternos	
Flujo Alternativo No. 1	<p>Si existe algún error en el proceso:</p> <ol style="list-style-type: none"> 1. El sistema registra el error. 2. El sistema muestra el siguiente mensaje de error: “No se puede realizar la acción dado a que se presentó un problema. <ol style="list-style-type: none"> a) Intente ejecutar nuevamente la acción. b) Si el problema persiste consulte al administrador del sistema. <p>Identificador del error: “{Identificador}”</p> <ol style="list-style-type: none"> 3. El sistema registra el Log de error. 4. El caso de uso finaliza.
Flujo Alternativo No. 2	<p>Si existen errores de validación:</p> <ol style="list-style-type: none"> 1. El sistema muestra el mensaje de error. 2. El caso de uso finaliza.
Flujo Alternativo No. 3	<p>Si el actor ejecuta la acción “cancelar”:</p> <ol style="list-style-type: none"> 1. El sistema limpia la información completada. 2. El caso de uso finaliza.
Flujo Alternativo No. 4	<p>Si el actor no dispone del rol Administrador:</p> <ol style="list-style-type: none"> 1. El sistema muestra el siguiente mensaje: “Acceso Denegado!!! El usuario no cuenta con los requisitos para poder ingresar a esta opción.” 2. El caso de uso finaliza.
Flujo Alternativo No. 5	<p>Si ejecuta la acción “regresar” al buscar</p> <ol style="list-style-type: none"> 1. El sistema continúa en el paso 2 del flujo base. 2. El caso de uso finaliza.
Requerimientos especiales	
Post-Condiciones	

Fuente: Elaboración propia (2024).

Tabla 24.

Caso de uso: Administrar catálogo contable

Prototipo: Prototipo funcional para la gestión de ventas de la empresa Zapatitos Lindos S.A., ubicada en San José.	
Número Caso de Uso: 14	Nombre del Caso de Uso: Administrar catálogo de cuentas.
Fecha elaboración:	19/04/2024
Descripción Caso de Uso:	Permitir la administración del catálogo contable.
Autor caso de uso:	Daniel David Quesada Fallas.
Actores relacionados:	Administrador, Contador.
Precondiciones:	<ol style="list-style-type: none"> 1. El actor debe haber iniciado sesión en el sistema. 2. El actor debe contar con acceso al módulo de contabilidad. 3. El actor debe contar con alguno de los siguientes roles: “Administrador”, “Contador”
Flujo Básico del caso de uso	
<ol style="list-style-type: none"> 1. Este caso de uso inicia cuando el actor desde el módulo de contabilidad selecciona del menú “Mantenimientos” la opción “Catálogo Cuentas”. 2. El sistema solicita la siguiente información: <ul style="list-style-type: none"> • Tipo de cuenta. • Cuenta. • Descripción. • Es Cuenta Mayor. • Utilizar cuenta para gastos CxP. • Impuesto de renta por pagar. • Utilidad / Pérdida del período. • Acumulación de utilidad del período. • Estado (Activo, Inactivo). 3. El actor completa la información y ejecuta la acción Grabar. 4. El sistema ejecuta las validaciones (SF-03). 5. El sistema registra o actualiza la información. 6. El caso de uso finaliza. 	
Sub Flujos	
Sub Flujo Buscar Artículo (SF-01)	<ol style="list-style-type: none"> 1. El sub flujo inicia cuando el actor ejecuta la acción “Buscar”. 2. El sistema solicita la siguiente información. <ul style="list-style-type: none"> • <u>Buscar por:</u> Cuenta, Nombre. • <u>Estado:</u> Todas, Activa, Inactiva • <u>Llave de búsqueda</u> • <u>Filtro opciones:</u> Que inicie, Que contenga, Que finalice. 3. El actor ingresa la información solicitada. 4. El actor ejecuta la acción “Buscar”. 5. El sistema muestra un listado con las cuentas contables que cumplan con el criterio de búsqueda. 6. El sistema muestra la siguiente opción: “Editar”.
Sub Flujo Editar	1. Inicia cuando el actor ejecuta la acción “Editar” en el paso 6

(SF-02)	<p>del Sub Flujo Buscar (SF-01)</p> <ol style="list-style-type: none"> 2. El sistema carga la información de la cuenta contable seleccionada. 3. El sistema continúa en el paso 3 del flujo base.
Sub Flujo Validaciones (SF-03)	<ol style="list-style-type: none"> 1. El sub flujo inicia en el paso 4 del flujo base. 2. El sistema verifica que el actor completó la información de aquellos datos requeridos. Si actor no completó los datos, el sistema muestra el siguiente mensaje de error: “El campo “{NombreDelCampo} es requerido.” 3. El sistema verifica que previamente no exista un registro con los datos ingresados. Si existe un registro previo el sistema muestra el siguiente mensaje de error: “No podrá continuar, ya existe un registro con la información ingresada; si desea agregar uno nuevo, modifique los datos de la cuenta. 4. El sistema verifica que los campos no excedan la cantidad de caracteres permitidos. Si esto ocurre el sistema muestra el siguiente mensaje: “No podrá continuar, el campo {NombreDelCampo} supera los {cantidad} caracteres permitidos”. 5. El sistema verifica que los campos no contengan caracteres no permitidos “[<>'"'\{\}\[\]]”. Si esto ocurre el sistema mostrará el siguiente mensaje: “No podrá continuar, el campo {campo} contiene caracteres no permitidos”.
Flujos Alternos	
Flujo Alternativo No. 1	<p>Si existe algún error en el proceso:</p> <ol style="list-style-type: none"> 1. El sistema registra el error. 2. El sistema muestra el siguiente mensaje de error: “No se puede realizar la acción dado a que se presentó un problema”. <ol style="list-style-type: none"> a) Intente ejecutar nuevamente la acción. b) Si el problema persiste consulte al administrador del sistema. <p>Identificador del error: “{Identificador}”</p> 3. El sistema registra el Log de error. 4. El caso de uso finaliza.
Flujo Alternativo No. 2	<p>Si existen errores de validación:</p> <ol style="list-style-type: none"> 1. El sistema muestra el mensaje de error. 2. El caso de uso finaliza.
Flujo Alternativo No. 3	<p>Si el actor ejecuta la acción “cancelar”:</p> <ol style="list-style-type: none"> 1. El sistema limpia la información completada. 2. El caso de uso finaliza.
Flujo Alternativo No. 4	<p>Si el actor no dispone del rol Administrador o Contador:</p> <ol style="list-style-type: none"> 1. El sistema muestra el siguiente mensaje: “Acceso Denegado!!! El usuario no cuenta con los requisitos para poder ingresar a esta opción.”

	2. El caso de uso finaliza.
Flujo Alternativo No. 5	Si ejecuta la acción “regresar” al buscar 1. El sistema continúa en el paso 2 del flujo base. 2. El caso de uso finaliza.
Requerimientos especiales	
Post-Condiciones	

Fuente: Elaboración propia (2024).

Tabla 25.

Caso de uso para cambiar de sistema

Prototipo: Prototipo funcional para la gestión de ventas de la empresa Zapatitos Lindos S.A., ubicada en San José.	
Número Caso de Uso: 15	Nombre del Caso de Uso: Cambiar de sistema.
Fecha elaboración:	19/04/2024
Descripción Caso de Uso:	Permitir la navegación entre los diferentes módulos y sucursales del sistema.
Autor caso de uso:	Daniel David Quesada Fallas.
Actores relacionados:	Administrador, Usuario estándar, Contador.
Precondiciones:	1. El actor debe haber ingresado al sistema previamente con sus credenciales. 2. El actor debe tener asociado más de un módulo o más de una sucursal.
Flujo Básico del caso de uso	
10. Este caso de uso inicia cuando el actor que ingresó con sus credenciales al sistema; decide cambiar de módulo o cambiar de sucursal y ejecuta la acción cambiar de sistema.	
11. El sistema verifica que el usuario cuenta con acceso a más de un módulo o sucursal.	
12. El sistema muestra y solicita la siguiente información:	
<ul style="list-style-type: none"> • Módulo (Selección). <ul style="list-style-type: none"> e) CIF (compras, inventario, facturación, anulaciones, órdenes de trabajo). f) CXC (Cuentas por cobrar). g) CXP (Cuentas por Pagar). h) CONTA (Gestión de ingresos y gastos). • Sucursal (Selección). 	
13. El actor completa la información y continúa el proceso.	
14. El caso de uso finaliza.	
Sub Flujos	
Flujos Alternos	
Flujo Alternativo No. 1	Si existe algún error en el proceso: 1. El sistema registra el error. 2. El sistema muestra el siguiente mensaje de error: “No se puede realizar la acción dado a que se presentó un problema”.

	a) Intente ejecutar nuevamente la acción. b) Si el problema persiste consulte al administrador del sistema. Identificador del error: “{Identificador}” 3. El sistema registra el Log de error. 4. El caso de uso finaliza
Requerimientos especiales	
Post-Condiciones	

Fuente: Elaboración propia (2024).

Tabla 26.

Caso de uso para administrar recetas

Prototipo: Prototipo funcional para la gestión de ventas de la empresa Zapatitos Lindos S.A, ubicada en San José.	
Número Caso de Uso: 16	Nombre del Caso de Uso: Administrar recetas.
Fecha elaboración:	19/04/2024
Descripción Caso de Uso:	Permitir la configuración y administración de los artículos compuestos, los cuales se utilizan como insumo en las órdenes de trabajo .
Autor caso de uso:	Daniel David Quesada Fallas.
Actores relacionados:	Administrador.
Precondiciones:	1. El actor debe haber iniciado sesión en el sistema. 2. El actor debe contar con acceso al módulo de compras, inventario y facturación. 3. El actor debe contar con el rol “Administrador”. 4. Deben existir en el catálogo de artículos, ítems con la clasificación “Primario” los cuales son los artículos que conformaran el artículo compuesto. 5. Deben existir en el catálogo de artículos, ítems con la clasificación “Compuesto”.
Flujo Básico del caso de uso	
1. Este caso de uso inicia cuando el actor selecciona del menú “ Mantenimientos ” > “ Inventario ” la opción “ Recetas ”. 2. El sistema muestra y solicita la siguiente información: <ul style="list-style-type: none"> • Código artículo compuesto. • Descripción artículo compuesto (No editable). • Código artículo primario. • Descripción artículo primario (No editable). • Bodega. • Costo (No editable). • Cantidad. • Detalle (Tabla) 	

<ul style="list-style-type: none"> ▪ N° Línea ▪ Artículo ▪ Descripción ▪ Costo ▪ Costo Total <p>3. El actor digita el código del artículo compuesto.</p> <p>4. El sistema ejecuta las validaciones.</p> <p>5. El sistema carga la descripción del artículo compuesto.</p> <p>6. Si el código del artículo compuesto ya cuenta con artículos primarios relacionados, el sistema carga el detalle.</p> <p>7. El actor digita el código de artículo primario.</p> <p>8. El sistema ejecuta las validaciones.</p> <p>9. El sistema carga la descripción y el costo del artículo primario.</p> <p>10. El sistema carga la lista de bodegas relacionadas con el artículo primario.</p> <p>11. El actor selecciona la bodega y digita la cantidad.</p> <p>12. El actor ejecuta la acción Agregar.</p> <p>13. El sistema ejecuta las validaciones.</p> <p>14. El sistema registra o actualiza la información.</p> <p>15. El caso de uso finaliza.</p>	
Sub Flujos	
<p>Sub Flujo Buscar artículo compuesto (SF-01)</p>	<p>1. El sub flujo inicia cuando el actor ejecuta la acción “Buscar” del código de artículo compuesto.</p> <p>2. El sistema solicita la siguiente información.</p> <ul style="list-style-type: none"> • <u>Llave de búsqueda</u> • <u>Buscar por:</u> Código, Descripción. • <u>Filtro opciones:</u> Que inicie, Que contenga, Que finalice. <p>3. El actor ingresa la información solicitada.</p> <p>4. El actor ejecuta la acción “Buscar”.</p> <p>5. El sistema muestra un listado con los artículos compuestos con estado activo que cumplan con el criterio de búsqueda.</p> <p>6. El sistema muestra la siguiente opción: “Seleccionar”</p>
<p>Sub Flujo Buscar artículo primario (SF-02)</p>	<p>1. El sub flujo inicia cuando el actor ejecuta la acción “Buscar” del código de artículo primario.</p> <p>2. El sistema solicita la siguiente información.</p> <ul style="list-style-type: none"> • Llave de búsqueda. • Buscar por: Código, Descripción. • Filtro opciones: Que inicie, Que contenga, Que finalice. <p>3. El actor ingresa la información solicitada.</p> <p>4. El actor ejecuta la acción “Buscar”</p> <p>5. El sistema muestra un listado con los artículos primarios con el estado activo que cumplan con el criterio de búsqueda.</p> <p>6. El sistema muestra la siguiente opción: “Seleccionar”</p>
<p>Sub Flujo Seleccionar artículo compuesto (SF-03)</p>	<p>1. Inicia cuando el actor ejecuta la acción “Seleccionar” en el paso 6 del Sub Flujo Buscar artículo compuesto (SF-01).</p> <p>2. El sistema obtiene la información del artículo compuesto.</p>

	3. El caso de uso continua en el paso 2 del flujo base.
Sub Flujo Seleccionar artículo primario (SF-04)	<ol style="list-style-type: none"> 1. Inicia cuando el actor ejecuta la acción “Seleccionar” en el paso 6 del Sub Flujo Buscar artículo primario (SF-02). 2. El sistema obtiene la información del artículo primario. 3. El caso de uso continua en el paso 9 del flujo base.
Sub Flujo Agregar artículo primario (SF-05)	<ol style="list-style-type: none"> 1. El sub flujo inicia cuando el actor ejecuta la acción “Agregar” línea al detalle de la receta. 2. El sistema ejecuta las validaciones. 3. El sistema agrega o modifica la línea al detalle de receta. 4. El caso de uso continúa en el paso 7 del flujo base.
Sub Flujo Eliminar artículo primario del detalle (SF-06)	<ol style="list-style-type: none"> 1. El sub flujo inicia cuando el actor ejecuta la acción “Eliminar” en el detalle de la receta. 2. El sistema elimina la línea del detalle.
Sub Flujo Validaciones (SF-07)	<ol style="list-style-type: none"> 1. El sistema verifica que el código de artículo compuesto exista y se encuentre en estado activo. En caso de cumplir con la regla el sistema mostrará el siguiente mensaje de error: “No podrá continuar, verifique que el código de artículo digitado exista, se encuentre activo y esté categorizado como artículo compuesto.” 2. El sistema verifica que el código de artículo primario exista, se encuentre en estado activo y este categorizado como artículo primario. En caso de cumplir con la regla el sistema mostrará el siguiente mensaje de error: “No podrá continuar, verifique que el código de artículo digitado exista y se encuentre activo.” 3. Al agregar el sistema verifica que el actor haya seleccionado la bodega. Si no se cumple la regla el sistema muestra el siguiente mensaje: “El campo “Bodega”, es requerido. 4. Al agregar el sistema verifica que el actor haya ingresado datos en el campo cantidad. Si no se cumple la regla el sistema muestra el siguiente mensaje: “El campo “Cantidad” es requerido.” 5. Al agregar el sistema verifica que la cantidad sea un valor numérico mayor a cero. Si no se cumple la regla el sistema muestra el siguiente mensaje: “El campo “Cantidad”, debe ser un valor numérico mayor a cero.” 6. Al registrar o actualizar la información de la receta el sistema verifica que se haya digitado el código del artículo compuesto. Si no cumple la regla el sistema muestra el siguiente mensaje: “El campo “Cod Artículo Compuesto” es requerido.” 7. Al registrar o actualizar la información de la receta el sistema verifica que se haya digitado el código del artículo primario. Si no cumple la regla el sistema muestra el siguiente mensaje: “El campo “Cod Artículo Primario” es requerido.” 8. Al registrar o actualizar la información de la receta el sistema verifica que se haya agregado al menos un artículo primario al detalle de la receta. Si no se cumple la regla el artículo, el sistema muestra el siguiente mensaje: “No podrá continuar,

	debe agregar al menos un artículo primario al detalle de la receta”.
Flujos Alternos	
Flujo Alternativo No. 1	<p>Si existe algún error en el proceso:</p> <ol style="list-style-type: none"> 1. El sistema registra el error. 2. El sistema muestra el siguiente mensaje de error: “No se puede realizar la acción dado a que se presentó un problema”. <ol style="list-style-type: none"> a) Intente ejecutar nuevamente la acción. b) Si el problema persiste consulte al administrador del sistema. <p>Identificador del error: “{Identificador}”</p> <ol style="list-style-type: none"> 3. El sistema registra el Log de error. 4. El caso de uso finaliza
Flujo Alternativo No. 2	<p>Si existen errores de validación:</p> <ol style="list-style-type: none"> 1. El sistema muestra el mensaje de error. 2. El caso de uso finaliza.
Flujo Alternativo No. 3	<p>Si el actor ejecuta la acción “cancelar”:</p> <ol style="list-style-type: none"> 1. El sistema limpia la información completada. 2. El caso de uso finaliza.
Flujo Alternativo No. 4	<p>Si el actor no dispone del rol Administrador:</p> <ol style="list-style-type: none"> 1. El sistema muestra el siguiente mensaje: “Acceso Denegado!!! El usuario no cuenta con los requisitos para poder ingresar a esta opción”. 2. El caso de uso finaliza.
Flujo Alternativo No. 5	<p>Si ejecuta la acción “cerrar” al buscar artículos compuestos y primarios.</p> <ol style="list-style-type: none"> 1. El sistema cierra la pantalla de búsqueda. 2. El caso de uso finaliza.
Requerimientos especiales	
Post-Condiciones	

Fuente: Elaboración propia (2024).

Tabla 27.

Caso de uso para generar reporte asientos de diario

Prototipo: Prototipo funcional para la gestión de ventas de la empresa Zapatitos Lindos S.A, ubicada en San José.	
Número Caso de Uso: 17	Nombre del Caso de Uso: Generar reporte asientos de diario.
Fecha elaboración:	20/04/2024
Descripción Caso de Uso:	Permitir la visualización de los asientos de diarios generados automáticamente por los módulos de facturación, anulación, devolución, compras, notas de crédito y débito de cuentas por pagar y de cuentas por cobrar, las facturas manuales de cuentas por pagar y los asientos ingresados manualmente desde el módulo de contabilidad.
Autor caso de uso:	Daniel David Quesada Fallas.
Actores relacionados:	Administrador, Contador.
Precondiciones:	<ol style="list-style-type: none"> 3. El actor debe haber iniciado sesión en el sistema. 1. El actor debe contar con acceso al módulo de contabilidad. 2. El actor debe contar con alguno de los siguientes roles: “Administrador”, “Contador”.
Flujo Básico del caso de uso	
<ol style="list-style-type: none"> 1. Este caso de uso inicia cuando el actor ingresa al módulo de contabilidad y selecciona del menú “Reportes” la opción “Asiento de diario”. 2. El sistema muestra y solicita la siguiente información: <ul style="list-style-type: none"> • Rango de fechas. • Tipo de asiento. 3. El actor completa la información y ejecuta la acción “Generar”. 4. El sistema obtiene los asientos ingresados por rango de fecha y tipo de asiento. 5. El sistema muestra los resultados. <ul style="list-style-type: none"> • N° Asiento. • Tipo • Año. • Mes. • Observación. • Fecha. • Cuenta. • Descripción. • Debe • Haber. • Número de documento. • Detalle • Total, Debe. • Total, Haber. 6. El caso de uso finaliza. 	
Sub Flujos	

Sub Flujo Imprimir (SF-01)	<ol style="list-style-type: none"> 1. El sub flujo inicia cuando el actor ejecuta la acción “Imprimir”. 2. El sistema imprime en pantalla el resultado. 3. El caso de uso finaliza.
Sub Flujo Generar PDF (SF-02)	<ol style="list-style-type: none"> 1. El sub flujo inicia cuando el actor ejecuta la acción “PDF”. 2. El sistema genera en formato PDF los resultados. 3. El caso de uso finaliza.
Sub Flujo Exportar Excel (SF-03)	<ol style="list-style-type: none"> 1. Inicia cuando el actor ejecuta la acción “Excel”. 2. El sistema exporta los resultados a un archivo de Excel. 3. El caso de uso finaliza.
Flujos Alternos	
Flujo Alternativo No. 1	<p>Si existe algún error en el proceso:</p> <ol style="list-style-type: none"> 1. El sistema registra el error. 2. El sistema muestra el siguiente mensaje de error: “No se puede realizar la acción dado a que se presentó un problema”. <ol style="list-style-type: none"> a) Intente ejecutar nuevamente la acción. b) Si el problema persiste consulte al administrador del sistema. <p>Identificador del error: “{Identificador}”</p> 3. El sistema registra el Log de error. 4. El caso de uso finaliza
Flujo Alternativo No. 2	<p>Si el actor no dispone de alguno de los siguientes roles: Administrador, Contador</p> <ol style="list-style-type: none"> 1. El sistema muestra el siguiente mensaje: “Acceso Denegado!!! El usuario no cuenta con los requisitos para poder ingresar a esta opción.” 2. El caso de uso finaliza.
Flujo Alternativo No. 3	<p>Si al generar el reporte no se encuentra datos.</p> <ol style="list-style-type: none"> 1. El sistema muestra en los resultados la leyenda: “No se encontraron resultados”. 2. El sistema muestra el siguiente mensaje: “No se encontraron datos con los parámetros seleccionados”.
Requerimientos especiales	
Post-Condiciones	

Fuente: Elaboración propia (2024).

Tabla 28.

Caso de uso para generar comprobación general

Prototipo: Prototipo funcional para la gestión de ventas de la empresa Zapatitos Lindos S.A., ubicada en San José.	
Número Caso de Uso: 18	Nombre del Caso de Uso: Generar reporte comprobación general.
Fecha elaboración:	20/04/2024
Descripción Caso de Uso:	Permitir la comprobación general por período de las cuentas contables y así poder verificar que los asientos contables estén balanceados, dicha comprobación se puede realizar por nivel, en donde el nivel 1 es la cuenta maestra y el último nivel es el detalle de la cuenta que se afectó a nivel del asiento.
Autor caso de uso:	Daniel David Quesada Fallas.
Actores relacionados:	Administrador, Contador.
Precondiciones:	<ol style="list-style-type: none"> 1. El actor debe haber iniciado sesión en el sistema. 2. El actor debe contar con acceso al módulo de contabilidad. 3. El actor debe contar con alguno de los siguientes roles: “Administrador”, “Contador”.
Flujo Básico del caso de uso	
<ol style="list-style-type: none"> 1. Este caso de uso inicia cuando el actor ingresa al módulo de contabilidad y selecciona del menú “Reportes” la opción “Comprobación General”. 2. El sistema muestra y solicita la siguiente información: <ul style="list-style-type: none"> • Año. • Mes. • Nivel. 3. El actor completa la información y ejecuta la acción “Generar”. 4. El sistema obtiene el resultado del reporte y los muestra en formato tabla. <ul style="list-style-type: none"> • Cuenta. • Descripción. • Debe. • Haber. • Total, Debe. • Total, Haber. 5. El caso de uso finaliza. 	
Sub Flujos	
Sub Flujo Imprimir (SF-01)	<ol style="list-style-type: none"> 1. El sub flujo inicia cuando el actor ejecuta la acción “Imprimir”. 2. El sistema imprime en pantalla el resultado. 3. El caso de uso finaliza.
Sub Flujo Generar PDF (SF-02)	<ol style="list-style-type: none"> 1. El sub flujo inicia cuando el actor ejecuta la acción “PDF”. 2. El sistema genera en formato PDF los resultados. 3. El caso de uso finaliza.
Sub Flujo Exportar	<ol style="list-style-type: none"> 1. Inicia cuando el actor ejecuta la acción “Excel”.

Excel (SF-03)	<ol style="list-style-type: none"> 2. El sistema exporta los resultados a un archivo de Excel. 3. El caso de uso finaliza.
Flujos Alternos	
Flujo Alternativo No. 1	<p>Si existe algún error en el proceso:</p> <ol style="list-style-type: none"> 1. El sistema registra el error. 2. El sistema muestra el siguiente mensaje de error: “No se puede realizar la acción dado a que se presentó un problema. <ol style="list-style-type: none"> c) Intente ejecutar nuevamente la acción. d) Si el problema persiste consulte al administrador del sistema. <p>Identificador del error: “{Identificador}”</p> <ol style="list-style-type: none"> 3. El sistema registra el Log de error. 4. El caso de uso finaliza
Flujo Alternativo No. 2	<p>Si el actor no dispone de alguno de los siguientes roles: Administrador, Contador</p> <ol style="list-style-type: none"> 1. El sistema muestra el siguiente mensaje: “Acceso Denegado!!! El usuario no cuenta con los requisitos para poder ingresar a esta opción.” 2. El caso de uso finaliza.
Flujo Alternativo No. 3	<p>Si al generar el reporte no se encuentra datos.</p> <ol style="list-style-type: none"> 1. El sistema muestra en los resultados la leyenda: No se encontraron resultados. 2. El sistema muestra el siguiente mensaje: “No se encontraron datos con los parámetros seleccionados”.
Requerimientos especiales	
Post-Condiciones	

Fuente: Elaboración propia (2024).

Tabla 29.

Caso de uso para generar reporte catálogo de cuentas

Prototipo: Prototipo funcional para la gestión de ventas de la empresa Zapatitos Lindos S.A., ubicada en San José.	
Número Caso de Uso: 19	Nombre del Caso de Uso: Generar reporte catálogo de cuentas.
Fecha elaboración:	20/04/2024
Descripción Caso de Uso:	Permitir la visualización completa del catálogo de cuentas por tipo de cuenta.
Autor caso de uso:	Daniel David Quesada Fallas.
Actores relacionados:	Administrador, Contador.
Precondiciones:	<ol style="list-style-type: none"> 1. El actor debe haber iniciado sesión en el sistema. 2. El actor debe contar con acceso al módulo de contabilidad. 3. El actor debe contar con alguno de los siguientes roles:

	“Administrador”, “Contador”.
Flujo Básico del caso de uso	
<ol style="list-style-type: none"> 1. Este caso de uso inicia cuando el actor ingresa al módulo de contabilidad y selecciona del menú “Reportes” la opción “Catálogo de cuentas”. 2. El sistema muestra y solicita la siguiente información: <ul style="list-style-type: none"> • Tipo cuenta. • Estado. 3. El actor completa la información y ejecuta la acción “Generar”. 4. El sistema obtiene el resultado del reporte y muestra la información en formato tabla: <ul style="list-style-type: none"> ▪ Cuenta. ▪ Tipo Cuenta. ▪ Debe. ▪ Haber. ▪ Total, Debe. ▪ Total, Haber. 5. El caso de uso finaliza. 	
Sub Flujos	
Sub Flujo Imprimir (SF-01)	<ol style="list-style-type: none"> 1. El sub flujo inicia cuando el actor ejecuta la acción “Imprimir”. 2. El sistema imprime en pantalla el resultado. 3. El caso de uso finaliza.
Sub Flujo Generar PDF (SF-02)	<ol style="list-style-type: none"> 1. El sub flujo inicia cuando el actor ejecuta la acción “PDF”. 2. El sistema genera en formato PDF los resultados. 3. El caso de uso finaliza.
Sub Flujo Exportar Excel (SF-03)	<ol style="list-style-type: none"> 1. Inicia cuando el actor ejecuta la acción “Excel”. 2. El sistema exporta los resultados a un archivo de Excel. 3. El caso de uso finaliza.
Flujos Alternos	
Flujo Alternativo No. 1	<p>Si existe algún error en el proceso:</p> <ol style="list-style-type: none"> 1. El sistema registra el error. 2. El sistema muestra el siguiente mensaje de error: “No se puede realizar la acción dado a que se presentó un problema”. <li style="margin-left: 20px;">a) Intente ejecutar nuevamente la acción. <li style="margin-left: 20px;">b) Si el problema persiste consulte al administrador del sistema. <p style="margin-left: 20px;">Identificador del error: “{Identificador}”</p> <ol style="list-style-type: none"> 3. El sistema registra el Log de error. 4. El caso de uso finaliza.
Flujo Alternativo No. 2	Si el actor no dispone de alguno de los siguientes roles:

	Administrador, Contador 1. El sistema muestra el siguiente mensaje: “Acceso Denegado!!! El usuario no cuenta con los requisitos para poder ingresar a esta opción.” 2. El caso de uso finaliza.
Flujo Alternativo No. 3	Si al generar el reporte no se encuentra datos. 1. El sistema muestra en los resultados la leyenda: “No se encontraron resultados”. 2. El sistema muestra el siguiente mensaje: “No se encontraron datos con los parámetros seleccionados”.
Requerimientos especiales	
Post-Condiciones	

Fuente: Elaboración propia (2024).

Tabla 30.

Caso de uso para generar reporte estado de cuenta CxP

Prototipo: Prototipo funcional para la gestión de ventas de la empresa Zapatitos Lindos S.A., ubicada en San José.	
Número Caso de Uso: 20	Nombre del Caso de Uso: Generar reporte estado de cuenta CxP.
Fecha elaboración:	20/04/2024
Descripción Caso de Uso:	Permitir la visualización de los distintos tipos de documentos del módulo de cuentas por pagar que tienen un saldo pendiente.
Autor caso de uso:	Daniel David Quesada Fallas.
Actores relacionados:	Administrador, Usuario Estándar, Contador.
Precondiciones:	1. El actor debe haber iniciado sesión en el sistema. 2. El actor debe contar con acceso al módulo de cuentas por pagar. 3. El actor debe contar con alguno de los siguientes roles: “Administrador”, “Usuario Estándar”, “Contador”.
Flujo Básico del caso de uso	
1. Este caso de uso inicia cuando el actor ingresa al módulo de cuentas por pagar y selecciona del menú “ Reportes ” la opción “ Estado de cuenta ” o desde el Dashboard de la pantalla principal ejecuta la acción “ Reporte Estado de Cuenta ” > Ir al reporte. 2. El sistema muestra y solicita la siguiente información: <ul style="list-style-type: none"> • Proveedor. • Tipo de documento: (selección múltiple). <ul style="list-style-type: none"> ✓ Factura de compra. ✓ Factura de gasto. ✓ Factura manual. ✓ Nota de crédito. ✓ Nota de débito. 3. El actor completa la información y ejecuta la acción “Generar”.	

<p>4. El sistema obtiene el resultado del reporte y muestra la información en formato tabla:</p> <ul style="list-style-type: none"> • Tipo Documento. • N° Documento. • Identificación. • Nombre Proveedor. • Fecha Hora Emisión. • Fecha Vencimiento. • Total. • Saldo. <p>5. El caso de uso finaliza.</p>	
Sub Flujos	
Sub Flujo Imprimir (SF-01)	<ol style="list-style-type: none"> 1. El sub flujo inicia cuando el actor ejecuta la acción “Imprimir”. 2. El sistema imprime en pantalla el resultado. 3. El caso de uso finaliza.
Sub Flujo Generar PDF (SF-02)	<ol style="list-style-type: none"> 1. El sub flujo inicia cuando el actor ejecuta la acción “PDF”. 2. El sistema genera en formato PDF los resultados. 3. El caso de uso finaliza.
Sub Flujo Exportar Excel (SF-03)	<ol style="list-style-type: none"> 1. Inicia cuando el actor ejecuta la acción “Excel”. 2. El sistema exporta los resultados a un archivo de Excel. 3. El caso de uso finaliza.
Flujos Alternos	
Flujo Alternativo No. 1	<p>Si existe algún error en el proceso:</p> <ol style="list-style-type: none"> 1. El sistema registra el error. 2. El sistema muestra el siguiente mensaje de error: “No se puede realizar la acción dado a que se presentó un problema”. <ol style="list-style-type: none"> a) Intente ejecutar nuevamente la acción. b) Si el problema persiste consulte al administrador del sistema. Identificador del error: {Identificador}” 3. El sistema registra el Log de error. 4. El caso de uso finaliza
Flujo Alternativo No. 2	<p>Si el actor no dispone de alguno de los siguientes roles: Administrador, Contador</p> <ol style="list-style-type: none"> 1. El sistema muestra el siguiente mensaje: “Acceso Denegado!!! El usuario no cuenta con los requisitos para poder ingresar a esta opción.” 2. El caso de uso finaliza.
Flujo Alternativo No. 3	<p>Si al generar el reporte no se encuentra datos.</p> <ol style="list-style-type: none"> 1. El sistema muestra en los resultados la leyenda: “No se encontraron resultados”. 2. El sistema muestra el siguiente mensaje: “No se encontraron datos con los parámetros seleccionados”.

Requerimientos especiales
Post-Condiciones

Fuente: Elaboración propia (2024).

Tabla 31.

Caso de uso para generar reporte estado de cuenta CxC

Prototipo: Prototipo funcional para la gestión de ventas de la empresa Zapatitos Lindos S.A., ubicada en San José.	
Número Caso de Uso: 21	Nombre del Caso de Uso: Generar reporte estado de cuenta CxC.
Fecha elaboración:	21/04/2024
Descripción Caso de Uso:	Permitir la visualización de los distintos tipos de documentos del módulo de cuentas por cobrar que tienen un saldo pendiente.
Autor caso de uso:	Daniel David Quesada Fallas.
Actores relacionados:	Administrador, Usuario Estándar, Contador.
Precondiciones:	<ol style="list-style-type: none"> 1. El actor debe haber iniciado sesión en el sistema. 2. El actor debe contar con acceso al módulo de cuentas por cobrar. 3. El actor debe contar con alguno de los siguientes roles: “Administrador”, “Usuario Estándar”, “Contador”.
Flujo Básico del caso de uso	
<ol style="list-style-type: none"> 1. Este caso de uso inicia cuando el actor ingresa al módulo de cuentas por cobrar y selecciona del menú “Reportes” la opción “Estado de cuenta” o desde el Dashboard de la pantalla principal ejecuta la acción “Reporte Estado de Cuenta” > Ir al reporte. 2. El sistema muestra y solicita la siguiente información: <ul style="list-style-type: none"> • Cliente. • Tipo de documento: (selección múltiple) <ul style="list-style-type: none"> ✓ Factura electrónica. ✓ Tiquete electrónico. ✓ Notas de crédito electrónica. ✓ Notas de crédito financiera. ✓ Notas de débito electrónica. ✓ Notas de débito financiera. 3. El actor completa la información y ejecuta la acción “Generar”. 4. El sistema obtiene el resultado del reporte y muestra la información en formato tabla: <ul style="list-style-type: none"> • Tipo. • N° Documento. • Identificación. • Nombre Cliente. • Fecha Hora Emisión. • Fecha Vencimiento. • Total. 	

<ul style="list-style-type: none"> • Saldo. <p>5. El caso de uso finaliza.</p>	
Sub Flujos	
Sub Flujo Imprimir (SF-01)	<ol style="list-style-type: none"> 1. El sub flujo inicia cuando el actor ejecuta la acción “Imprimir”. 2. El sistema imprime en pantalla el resultado. 3. El caso de uso finaliza.
Sub Flujo Generar PDF (SF-02)	<ol style="list-style-type: none"> 1. El sub flujo inicia cuando el actor ejecuta la acción “PDF”. 2. El sistema genera en formato PDF los resultados. 3. El caso de uso finaliza.
Sub Flujo Exportar Excel (SF-03)	<ol style="list-style-type: none"> 1. Inicia cuando el actor ejecuta la acción “Excel”. 2. El sistema exporta los resultados a un archivo de Excel. 3. El caso de uso finaliza.
Flujos Alternos	
Flujo Alternativo No. 1	<p>Si existe algún error en el proceso:</p> <ol style="list-style-type: none"> 1. El sistema registra el error. 2. El sistema muestra el siguiente mensaje de error: “No se puede realizar la acción dado a que se presentó un problema”. <ol style="list-style-type: none"> a) Intente ejecutar nuevamente la acción. b) Si el problema persiste consulte al administrador del sistema. <p>Identificador del error: {Identificador}”</p> <ol style="list-style-type: none"> 3. El sistema registra el Log de error. 4. El caso de uso finaliza.
Flujo Alternativo No. 2	<p>Si el actor no dispone de alguno de los siguientes roles: Administrador, Contador</p> <ol style="list-style-type: none"> 1. El sistema muestra el siguiente mensaje: “Acceso Denegado!!! El usuario no cuenta con los requisitos para poder ingresar a esta opción.” 2. El caso de uso finaliza.
Flujo Alternativo No. 3	<p>Si al generar el reporte no se encuentra datos.</p> <ol style="list-style-type: none"> 1. El sistema muestra en los resultados la leyenda: “No se encontraron resultados”. 2. El sistema muestra el siguiente mensaje: “No se encontraron datos con los parámetros seleccionados”.
Requerimientos especiales	
Post-Condiciones	

Fuente: Elaboración propia (2024).

Tabla 32.

Caso de uso para generar reporte antigüedad de saldos CxP

Prototipo: Prototipo funcional para la gestión de ventas de la empresa Zapatitos Lindos S.A, ubicada en San José.	
Número Caso de Uso: 22	Nombre del Caso de Uso: Generar reporte antigüedad de saldos CxP.
Fecha elaboración:	22/04/2024
Descripción Caso de Uso:	Permitir la visualización tabulada de los saldos de las facturas de cuentas por pagar. La tabulación se realiza en por monto no vencido, vencimientos de 1 a 30 días, vencimientos de 30 a 60 días, vencimientos de 60 a 90 días y saldos vencidos mayo a 90 días.
Autor caso de uso:	Daniel David Quesada Fallas.
Actores relacionados:	Administrador, Usuario Estándar, Contador.
Precondiciones:	4. El actor debe haber iniciado sesión en el sistema. 5. El actor debe contar con acceso al módulo de cuentas por pagar. 6. El actor debe contar con alguno de los siguientes roles: “Administrador”, “Usuario Estándar”, “Contador”.
Flujo Básico del caso de uso	
<p>1. Este caso de uso inicia cuando el actor ingresa al módulo de cuentas por pagar y selecciona del menú “Reportes” la opción “Antigüedad saldos” o desde el Dashboard de la pantalla principal ejecuta la acción “Reporte Antigüedad de saldos” > Ir al reporte.</p> <p>2. El sistema muestra y solicita la siguiente información:</p> <ul style="list-style-type: none"> • Fecha de corte. • Proveedor. • Mostrar vencimientos. Sí/No. <p>3. El actor completa la información y ejecuta la acción “Generar”.</p> <p>4. El sistema obtiene el resultado del reporte y muestra la información en formato tabla:</p> <ul style="list-style-type: none"> • N° Factura. • Identificación. • Nombre Proveedor. • Fecha Hora Emisión. • Total. • Saldo No Vencido. • Vencimiento de 1 a 30. • Vencimiento de 30 a 60. • Vencimiento de 60 a 90. • Vencimiento mayor a 90. <p>5. El caso de uso finaliza.</p>	
Sub Flujos	
Sub Flujo Imprimir (SF-01)	<p>1. El sub flujo inicia cuando el actor ejecuta la acción “Imprimir”.</p> <p>2. El sistema imprime en pantalla el resultado.</p>

	3. El caso de uso finaliza.
Sub Flujo Generar PDF (SF-02)	4. El sub flujo inicia cuando el actor ejecuta la acción “PDF”. 5. El sistema genera en formato PDF los resultados. 6. El caso de uso finaliza.
Sub Flujo Exportar Excel (SF-03)	4. Inicia cuando el actor ejecuta la acción “Excel”. 5. El sistema exporta los resultados a un archivo de Excel. 6. El caso de uso finaliza.
Flujos Alternos	
Flujo Alternativo No. 1	Si existe algún error en el proceso: 1. El sistema registra el error. 2. El sistema muestra el siguiente mensaje de error: “No se puede realizar la acción dado a que se presentó un problema”. a) Intente ejecutar nuevamente la acción. b) Si el problema persiste consulte al administrador del sistema. Identificador del error: {Identificador}” 3. El sistema registra el Log de error. 4. El caso de uso finaliza
Flujo Alternativo No. 2	Si el actor no dispone de alguno de los siguientes roles: Administrador, Contador 1. El sistema muestra el siguiente mensaje: “Acceso Denegado!!! El usuario no cuenta con los requisitos para poder ingresar a esta opción.” 2. El caso de uso finaliza.
Flujo Alternativo No. 3	Si al generar el reporte no se encuentra datos. 1. El sistema muestra en los resultados la leyenda: “No se encontraron resultados”. 2. El sistema muestra el siguiente mensaje: “No se encontraron datos con los parámetros seleccionados”.
Requerimientos especiales	
Post-Condiciones	

Fuente: Elaboración propia (2024).

Tabla 33.

Caso de uso para generar reporte antigüedad de saldo CxC

Prototipo: Prototipo funcional para la gestión de ventas de la empresa Zapatitos Lindos S.A, ubicada en San José.	
Número Caso de Uso: 23	Nombre del Caso de Uso: Generar reporte antigüedad de saldos CxC.
Fecha elaboración:	22/04/2024
Descripción Caso de Uso:	Permitir la visualización tabulada de los saldos de las facturas de cuentas por cobrar. La tabulación se realiza en por monto no vencido, vencimientos de 1 a 30 días, vencimientos de 30 a 60 días, vencimientos de 60 a 90 días y saldos vencidos mayo a 90 días.
Autor caso de uso:	Daniel David Quesada Fallas.
Actores relacionados:	Administrador, Usuario Estándar, Contador.
Precondiciones:	<ol style="list-style-type: none"> 1. El actor debe haber iniciado sesión en el sistema. 2. El actor debe contar con acceso al módulo de cuentas por cobrar. 3. El actor debe contar con alguno de los siguientes roles: “Administrador”, “Usuario Estándar”, “Contador”.
Flujo Básico del caso de uso	
<ol style="list-style-type: none"> 1. Este caso de uso inicia cuando el actor ingresa al módulo de cuentas por cobrar y selecciona del menú “Reportes” la opción “Antigüedad saldos” o desde el Dashboard de la pantalla principal ejecuta la acción “Reporte Antigüedad de saldos” > Ir al reporte. 2. El sistema muestra y solicita la siguiente información: <ul style="list-style-type: none"> • Fecha de corte. • Cliente. • Mostrar vencimientos. Sí/No. 3. El actor completa la información y ejecuta la acción “Generar”. 4. El sistema obtiene el resultado del reporte y muestra la información en formato tabla: <ul style="list-style-type: none"> • N° Factura. • Identificación. • Nombre Cliente. • Fecha Hora Emisión. • Fecha Vencimiento. • Total. • Saldo No Vencido. • Vencimiento de 1 a 30. • Vencimiento de 30 a 60. • Vencimiento de 60 a 90. • Vencimiento mayor a 90. 5. El caso de uso finaliza. 	
Sub Flujos	

Sub Flujo Imprimir (SF-01)	<ol style="list-style-type: none"> 1. El sub flujo inicia cuando el actor ejecuta la acción “Imprimir”. 2. El sistema imprime en pantalla el resultado. 3. El caso de uso finaliza.
Sub Flujo Generar PDF (SF-02)	<ol style="list-style-type: none"> 1. El sub flujo inicia cuando el actor ejecuta la acción “PDF”. 2. El sistema genera en formato PDF los resultados. 3. El caso de uso finaliza.
Sub Flujo Exportar Excel (SF-03)	<ol style="list-style-type: none"> 1. Inicia cuando el actor ejecuta la acción “Excel”. 2. El sistema exporta los resultados a un archivo de Excel. 3. El caso de uso finaliza.
Flujos Alternos	
Flujo Alternativo No. 1	<p>Si existe algún error en el proceso:</p> <ol style="list-style-type: none"> 1. El sistema registra el error. 2. El sistema muestra el siguiente mensaje de error: “No se puede realizar la acción dado a que se presentó un problema”. <ol style="list-style-type: none"> a) Intente ejecutar nuevamente la acción. b) Si el problema persiste consulte al administrador del sistema. Identificador del error: {Identificador}” 3. El sistema registra el Log de error. 4. El caso de uso finaliza
Flujo Alternativo No. 2	<p>Si el actor no dispone de alguno de los siguientes roles: Administrador, Contador</p> <ol style="list-style-type: none"> 1. El sistema muestra el siguiente mensaje: “Acceso Denegado!!! El usuario no cuenta con los requisitos para poder ingresar a esta opción.” 2. El caso de uso finaliza.
Flujo Alternativo No. 3	<p>Si al generar el reporte no se encuentra datos.</p> <ol style="list-style-type: none"> 1. El sistema muestra en los resultados la leyenda: “No se encontraron resultados”. 2. El sistema muestra el siguiente mensaje: “No se encontraron datos con los parámetros seleccionados”.
Requerimientos especiales	
Post-Condiciones	

Fuente: Elaboración propia (2024).

Tabla 34.

Caso de uso para generar reporte histórico pagos CxP

Prototipo: Prototipo funcional para la gestión de ventas de la empresa Zapatitos Lindos S.A., ubicada en San José.	
Número Caso de Uso: 24	Nombre del Caso de Uso: Generar reporte histórico pagos CxP
Fecha elaboración:	23/04/2024
Descripción Caso de Uso:	Permitir la visualización del histórico de pagos realizados a facturas del módulo cuentas por pagar en un rango de fecha previamente establecido.
Autor caso de uso:	Daniel David Quesada Fallas.
Actores relacionados:	Administrador, Usuario Estándar, Contador.
Precondiciones:	<ol style="list-style-type: none"> 1. El actor debe haber iniciado sesión en el sistema. 2. El actor debe contar con acceso al módulo de cuentas por pagar. 3. El actor debe contar con alguno de los siguientes roles: “Administrador”, “Usuario Estándar”, “Contador”.
Flujo Básico del caso de uso	
<ol style="list-style-type: none"> 1. Este caso de uso inicia cuando el actor ingresa al módulo de cuentas por pagar y selecciona del menú “Reportes” la opción “Histórico de pagos” o desde el Dashboard de la pantalla principal ejecuta la acción “Reporte Histórico de pagos” > Ir al reporte. 2. El sistema muestra y solicita la siguiente información: <ul style="list-style-type: none"> • Rango de fechas. • Proveedor. 3. El actor completa la información y ejecuta la acción “Generar”. 4. El sistema obtiene el resultado del reporte y muestra la información en formato tabla: <ul style="list-style-type: none"> • Identificación. • Nombre proveedor. • Fecha hora emisión. • Tipo pago. • N° factura afectada. • Total. • Saldo actual. • Monto pagado. • Saldo al aplicar el pago. 5. El caso de uso finaliza. 	
Sub Flujos	
Sub Flujo Imprimir (SF-01)	<ol style="list-style-type: none"> 1. El sub flujo inicia cuando el actor ejecuta la acción “Imprimir”. 2. El sistema imprime en pantalla el resultado. 3. El caso de uso finaliza.

Sub Flujo Generar PDF (SF-02)	<ol style="list-style-type: none"> 1. El sub flujo inicia cuando el actor ejecuta la acción “PDF”. 2. El sistema genera en formato PDF los resultados. 3. El caso de uso finaliza.
Sub Flujo Exportar Excel (SF-03)	<ol style="list-style-type: none"> 1. Inicia cuando el actor ejecuta la acción “Excel”. 2. El sistema exporta los resultados a un archivo de Excel. 3. El caso de uso finaliza.
Flujos Alternos	
Flujo Alternativo No. 1	<p>Si existe algún error en el proceso:</p> <ol style="list-style-type: none"> 1. El sistema registra el error. 2. El sistema muestra el siguiente mensaje de error: “No se puede realizar la acción dado a que se presentó un problema”. <ol style="list-style-type: none"> a) Intente ejecutar nuevamente la acción. b) Si el problema persiste consulte al administrador del sistema. <p>Identificador del error: {Identificador}”</p> 3. El sistema registra el Log de error. 4. El caso de uso finaliza
Flujo Alternativo No. 2	<p>Si el actor no dispone de alguno de los siguientes roles: Administrador, Contador</p> <ol style="list-style-type: none"> 1. El sistema muestra el siguiente mensaje: “Acceso Denegado!!! El usuario no cuenta con los requisitos para poder ingresar a esta opción.” 2. El caso de uso finaliza.
Flujo Alternativo No. 3	<p>Si al generar el reporte no se encuentra datos.</p> <ol style="list-style-type: none"> 1. El sistema muestra en los resultados la leyenda: No se encontraron resultados. 2. El sistema muestra el siguiente mensaje: “No se encontraron datos con los parámetros seleccionados”.
Requerimientos especiales	
Post-Condiciones	

Fuente: Elaboración propia (2024).

Tabla 35.

Caso de uso para generar reporte histórico de pagos CxC

Prototipo: Prototipo funcional para la gestión de ventas de la empresa Zapatitos Lindos S.A, ubicada en San José.	
Número Caso de Uso: 25	Nombre del Caso de Uso: Generar reporte histórico pagos CxC
Fecha elaboración:	24/04/2024
Descripción Caso de Uso:	Permitir la visualización del histórico de pagos realizados a facturas del módulo cuentas por cobrar en un rango de fecha previamente establecido.
Autor caso de uso:	Daniel David Quesada Fallas.
Actores relacionados:	Administrador, Usuario Estándar, Contador.
Precondiciones:	<ol style="list-style-type: none"> 1. El actor debe haber iniciado sesión en el sistema. 2. El actor debe contar con acceso al módulo de cuentas por cobrar. 3. El actor debe contar con alguno de los siguientes roles: “Administrador”, “Usuario Estándar”, “Contador”.
Flujo Básico del caso de uso	
<ol style="list-style-type: none"> 1. Este caso de uso inicia cuando el actor ingresa al módulo de cuentas por cobrar y selecciona del menú “Reportes” la opción “Histórico de pagos” o desde el Dashboard de la pantalla principal ejecuta la acción “Reporte Histórico de pagos” > Ir al reporte. 2. El sistema muestra y solicita la siguiente información: <ul style="list-style-type: none"> • Rango de fechas. • Cliente. 3. El actor completa la información y ejecuta la acción “Generar”. 4. El sistema obtiene el resultado del reporte y muestra la información en formato tabla: <ul style="list-style-type: none"> • Identificación. • Nombre Cliente. • Fecha Hora Emisión. • Tipo Pago. • N° Factura Afectada. • Total. • Saldo Actual. • Monto pagado. 5. El caso de uso finaliza. 	
Sub Flujos	
Sub Flujo Imprimir (SF-01)	<ol style="list-style-type: none"> 1. El sub flujo inicia cuando el actor ejecuta la acción “Imprimir”. 2. El sistema imprime en pantalla el resultado. 3. El caso de uso finaliza.
Sub Flujo Generar PDF (SF-02)	<ol style="list-style-type: none"> 1. El sub flujo inicia cuando el actor ejecuta la acción “PDF”. 2. El sistema genera en formato PDF los resultados. 3. El caso de uso finaliza.
Sub Flujo Exportar	<ol style="list-style-type: none"> 1. Inicia cuando el actor ejecuta la acción “Excel”.

Excel (SF-03)	<ol style="list-style-type: none"> 2. El sistema exporta los resultados a un archivo de Excel. 3. El caso de uso finaliza.
Flujos Alternos	
Flujo Alternativo No. 1	<p>Si existe algún error en el proceso:</p> <ol style="list-style-type: none"> 1. El sistema registra el error. 2. El sistema muestra el siguiente mensaje de error: “No se puede realizar la acción dado a que se presentó un problema”. <ol style="list-style-type: none"> a) Intente ejecutar nuevamente la acción. b) Si el problema persiste consulte al administrador del sistema. <p>Identificador del error: “{Identificador}”</p> <ol style="list-style-type: none"> 3. El sistema registra el Log de error. 4. El caso de uso finaliza.
Flujo Alternativo No. 2	<p>Si el actor no dispone de alguno de los siguientes roles: Administrador, Contador</p> <ol style="list-style-type: none"> 1. El sistema muestra el siguiente mensaje: “Acceso Denegado!!! El usuario no cuenta con los requisitos para poder ingresar a esta opción.” 2. El caso de uso finaliza.
Flujo Alternativo No. 3	<p>Si al generar el reporte no se encuentra datos.</p> <ol style="list-style-type: none"> 1. El sistema muestra en los resultados la leyenda: “No se encontraron resultados”. 2. El sistema muestra el siguiente mensaje: “No se encontraron datos con los parámetros seleccionados”.
Requerimientos especiales	
Post-Condiciones	

Fuente: Elaboración propia (2024).

Tabla 36.

Caso de uso para generar facturación

Prototipo: Prototipo funcional para la gestión de ventas de la empresa Zapatitos Lindos S.A, ubicada en San José.	
Número Caso de Uso: 26	Nombre del Caso de Uso: Generar facturación.
Fecha elaboración:	24/04/2024
Descripción Caso de Uso:	Permitir la generación de tiquetes y facturas electrónicas.
Autor caso de uso:	Daniel David Quesada Fallas.
Actores relacionados:	Usuario Administrador.
Precondiciones:	<ol style="list-style-type: none"> 1. El actor debe haber iniciado sesión en el sistema y disponer de alguno de los siguientes roles “Administrador”, “Usuario Estándar”. 2. El actor debe haber completado la información del mantenimiento de compañías. 3. Los artículos que se desean facturar deben existir previamente, en caso de que el artículo afecte el inventario este debe contar con saldo. 4. Si la factura electrónica es de crédito, el cliente debe estar registrado desde el mantenimiento de clientes. 5. Si el cliente a facturar es un cliente Exonerado y se le deben facturar artículos con impuesto de venta, el cliente debe contar el perfil exonerado y se deben completar los datos de exoneración desde el mantenimiento de cliente.
Flujo Básico del caso de uso	
<ol style="list-style-type: none"> 1. Este caso de uso inicia cuando el actor selecciona del menú “Facturación” > “Procesos” la opción “Facturar”. 2. El sistema muestra y solicita la siguiente información: <ul style="list-style-type: none"> <u>Encabezado</u> <u>Datos Generales</u> <ul style="list-style-type: none"> • Tipo Documento (Factura electrónica, Tiquete electrónico). • Condición de venta (Contado, Crédito). <ul style="list-style-type: none"> * Por defecto la condición de venta es de contado. Se habilita únicamente si el cliente está registrado y cuenta con crédito. • Forma de Pago (Efectivo, Tarjeta, Cheque, Transferencia-Depósito Bancario, Recaudado por terceros, Otros). • Tipo Identificación (Cliente contado, Cédula física, Cédula Jurídica, DIMEX, NITE, Identificación de extranjero). • Identificación. <ul style="list-style-type: none"> * Este campo se habilita cuando el tipo de identificación es diferente a cliente contado. • Nombre de Cliente. <u>Datos del cliente</u> <ul style="list-style-type: none"> • % Descuento Global (Campo no editable). 	

- Plazo (Campo no editable).
- Vencimiento. (Campo no editable).
- Correo.
- Cod Área Teléfono.
- Teléfono.
- Cod Área Fax.
- Fax.
- Provincia.
- Cantón.
- Distrito.
- Barrio.
- Dirección.

Otros Datos

- N° Pedido
- N° Orden trabajo.
- N° Orden de compra.
- Paciente.
- Bodega.
- Moneda.
- Actividad Comercial.
- Datos de exoneración (sección no editable).
- Tipo documento de exoneración.
- Número de documento.
- Institución.
- Fecha de exoneración.
- Porcentaje de exoneración.

Detalle

- Cod Artículo.
- Descripción.
- Es Servicio (campo no editable).
- Saldo (campo no editable).
- Unidad de medida (campo no editable)
- Precio.
 - * Solo se podrán modificar precios de aquellos artículos que no afecten el inventario (Servicios, conceptos).
- Porcentaje de descuento.
- Cantidad.
- Lista de artículos a facturar en formato tabla.
 - N° Línea.
 - Artículo.
 - Descripción.
 - CABys.
 - Cantidad.
 - Precio.

- % Desc 1.
- % Desc 2.
- IVA.
- Total, Línea.

Totales

- Observación.
 - Subtotal (campo no editable).
 - Descuento (campo no editable).
 - IVA (campo no editable).
 - Total (campo no editable)
3. El actor selecciona el tipo de documento que se desea emitir.
 4. El actor selecciona la forma de pago.
 5. El actor selecciona la bodega de la cual quiere facturar.
 6. El actor selecciona el tipo de identificación del cliente.
 7. El actor digita el número de identificación del cliente.
 8. El sistema obtiene y muestra la información del cliente; si el cliente registrado es de contando se bloquea el campo condición de venta para que no se le pueda facturar a crédito.
 9. El actor digita el código de artículo o servicio a vender.
 10. El sistema consulta si el código de artículo o servicio se encuentra registrado y carga la información.
 11. El actor digita la cantidad que desea facturar y ejecuta la acción agregar.
 12. El sistema realiza las validaciones del artículo.
 13. El sistema agrega el artículo al detalle y realiza los cálculos subtotal, descuento, impuesto de ventas valor agregado y total.
 14. El actor puede agregar otro artículo, regresando al flujo base número 9.
 15. El actor ejecute la acción “facturar”.
 16. El sistema solicita la confirmación de la acción.
 17. El actor confirma acción.
 18. El sistema realiza las validaciones.
 19. El sistema rebaja las existencias en bodega de los artículos que afectan el inventario.
 20. El sistema registra la bitácora Kardex de inventario para los artículos que afectan el inventario.
 21. El sistema registra la cuenta por cobrar si la condición de venta es “Crédito”.
 22. El sistema registra el asiento contable del documento.
 23. El sistema ejecuta el Sub flujo emitir comprobante electrónico.
 24. El sistema muestra en pantalla el comprobante electrónico en formato pdf.
 25. El caso de uso finaliza.

Sub Flujos

Sub Flujo Generar documento electrónico sin cliente. (SF-01)

- 1.El sub flujo inicia cuando el actor confirma acción “Facturar”.
2. El sistema genera el comprobante electrónico.
3. El sistema firma electrónicamente el comprobante.
4. El sistema establece comunicación con el ministerio de hacienda.
- 5.El sistema consulta el último token de autenticación obtenido de la plataforma del Ministerio de Hacienda. Si el

	<p>Token es válido lo utiliza de lo contrario obtiene uno nuevo.</p> <p>6. El sistema envía el documento para su respectiva validación.</p> <p>7. El sistema consulta el estado de comprobante electrónico. En caso que el comprobante electrónico fue recibido, pero no ha sido procesado por la plataforma del Ministerio de Hacienda el sistema continua en el flujo base 24.</p> <p>8. El sistema crea una tarea asincrónica que validara el estado de documento 5 minutos después para tratar de obtener la respuesta. Si se logra obtener la respuesta del ministerio de hacienda se le envía un correo electrónico al cliente con los documentos XML y con la representación gráfica.</p> <p>9. El caso de uso finaliza.</p>
<p>Sub Flujo Buscar cliente. (SF-02)</p>	<p>1. El sub flujo inicia cuando el actor ejecuta la acción buscar cliente.</p> <p>2. El sistema muestra y solicita la siguiente información: Llave de búsqueda. Buscar por: Identificación del cliente, Nombre del cliente. Filtro opciones: Que inicie, Que contenga, Que finalice.</p> <p>3. El actor completa la información solicitada y ejecuta la acción buscar.</p> <p>4. El sistema muestra el resultado en forma de tabla:</p> <ul style="list-style-type: none"> • Tipo Identificación. • Identificación. • Nombre. • Tiene Crédito. • Es Exonerado. • Plazo. <p>5. El actor ejecuta la acción “Seleccionar”.</p> <p>6. El sistema carga la información del cliente, si es un cliente de crédito se habilita el campo condición de venta.</p> <p>7. El caso de uso continua en el flujo base 9.</p>
<p>Sub Flujo Buscar articulo o servicio. (SF-03)</p>	<p>1. El sub flujo inicia cuando el actor ejecuta la acción buscar artículo.</p> <p>2. El sistema muestra y solicita la siguiente información:</p> <ul style="list-style-type: none"> • Llave de búsqueda. • Buscar por: Código, Descripción. • Filtro opciones: Que inicie, Que contenga, Que finalice. <p>3. El actor ejecuta la acción buscar.</p> <p>4. El sistema muestra el resultado en forma de tabla:</p> <ul style="list-style-type: none"> • Código. • Descripción. • Precio 1. • Precio 2. • Precio 3.

	<ul style="list-style-type: none"> • Es servicio. • Afecta Inventario. • Tipo de impuesto. • Tarifa de impuesto. • Unidad medida. • Saldo. • Comprometido. <p>5. El actor ejecuta la acción seleccionar.</p> <p>6. El sistema carga la información del artículo. Si el artículo no afecta inventario el sistema habilita el campo Precio para que se pueda digitar.</p> <p>7. El caso de uso continua en flujo base 11.</p>
Sub Flujo Grabar pedido (SF-04)	<p>1. El sub flujo inicia cuando el actor ejecuta y confirma la acción grabar pedido.</p> <p>2. El sistema graba la información de la factura para que pueda cargarse posteriormente y facturarse si necesidad de volver a ingresar la información.</p> <p>3. El caso de uso finaliza.</p>
Sub Flujo Cargar pedido (SF-05)	<p>1. El sub flujo inicia cuando el actor ejecuta la acción buscar número de pedido desde la sección Otros Datos.</p> <p>2. El sistema muestra la lista de pedidos pendientes de facturar en formato tabla;</p> <ul style="list-style-type: none"> ▪ Número Pedido. ▪ Fecha Hora. ▪ Identificación. ▪ Nombre cliente. <p>3. El actor ejecuta la acción “seleccionar”.</p> <p>4. El sistema obtiene y carga la información del pedido.</p> <p>5. El sistema verifica que todas la líneas cuenten con el saldo suficiente para poder generar la factura, en caso de contar con el saldo suficiente el sistema omitirá cargar la línea y mostrará el siguiente mensaje: Algunas líneas no se cargaron debido a que no cuentan con saldo suficiente, por lo que no se podrán facturar.</p> <p>6. El caso de uso continua en el flujo base 9.</p>
Sub Flujo Facturar orden de trabajo. (SF-06)	<p>1. El sub flujo inicia cuando el actor ejecuta la acción buscar en el N° de orden de trabajo en la sección “Otros Datos”.</p> <p>2. El sistema muestra la lista de órdenes de trabajo finalizadas en formato tabla:</p> <ul style="list-style-type: none"> • N° Orden. • Identificación el cliente. • Nombre del cliente. • Identificación del paciente. • Nombre del paciente. • Fecha de creación.

	<ul style="list-style-type: none"> • Fecha de entrega. <ol style="list-style-type: none"> 3. El actor ejecuta la acción “Seleccionar” 4. El sistema carga la información de la orden de trabajo. 5. El actor puede agregar líneas, más no puede editar ni editar ya que el control de la orden de trabajo se realiza desde el módulo de órdenes de trabajo. 6. El actor ejecuta y confirma la acción “Facturar”. 7. El sistema descompromete las unidades del inventario para que se pueden rebajar las existencias. 8. El sistema verifica si existen insumos asociados a la orden de trabajo y rebaja las existencias del inventario y registra la bitácora Kardex. 9. El caso de uso continuar en el flujo base 19.
Sub Flujo Asociar paciente (SF-07)	<ol style="list-style-type: none"> 1. El sub flujo inicia cuando el actor ejecuta la acción buscar en el campo paciente en la sección “Otros datos.” 2. El sistema muestra y solicita la siguiente información: Llave de búsqueda. Buscar por: Identificación del paciente, Nombre del paciente. Filtro opciones: Que inicie, Que contenga, Que finalice. 3. El actor completa la información solicitada y ejecuta la acción buscar. 4. El sistema muestra el resultado en forma de tabla: <ul style="list-style-type: none"> ▪ Tipo Identificación. ▪ Identificación. ▪ Nombre. 5. El actor ejecuta la acción seleccionar. 6. El sistema asocia el cliente. 7. El caso de uso continuar en el flujo base 9.
Sub Flujo Validaciones (SF-08)	<ol style="list-style-type: none"> 1. Si el tipo de documento es factura electrónica el sistema valida los siguiente datos: Tipo identificación, Identificación y Nombre. Si los campos no contienen datos el sistema muestra el siguiente mensaje: No podrá continuar el campo “{Nombre}” es requerido. 2. Si la identificación contiene dato el sistema valida que la identificación cumpla el formato, en caso de cumplir se muestra el siguiente mensaje: “La identificación no cumple con el siguiente formato {Formato}.” 3. Si no agregaron articulos al detalle, el sistema muestra el siguiente mensaje: “No podrá continuar debe agregar al menos un artículo”. 4. Si una línea del detalle no cuenta con suficiente saldo para emitir la factura. El sistema muestra el siguiente mensaje: “No podrá continuar; el artículo no cuenta con el saldo suficiente.” 5. El sistema valida que la que el descuento global del cliente

	<p>y el descuento por línea no sumen 100 o más. El sistema muestra el siguiente mensaje: “No podrá continuar; la suma del descuento global del cliente y el descuento por línea no puede ser mayor a 100”.</p> <p>6.El sistema valida que el nombre del cliente no exceda el límite. El sistema muestra el siguiente mensaje: “No podrá continuar, el campo "Nombre" supera los {Cantidad} caracteres permitidos.”</p> <p>7. El sistema valida que ninguna entrada de información al sistema contenga caracteres especiales no permitidos [& < " " " \ { } \ []]. El sistema muestra el siguiente mensaje: “No podrá continuar, el campo {campo} contiene caracteres no permitidos”.</p> <p>8. El sistema valida que el correo electrónico sea válido. El sistema muestra el siguiente mensaje: “No podrá continuar, el "Correo" no cumple con los requisitos mínimos de una dirección de correo electrónico.”.</p> <p>9. Para los artículos que afectan inventario el sistema valida que el precio digitado no sea menor al costo. El sistema muestra el siguiente mensaje: “No podrá continuar, el precio del artículo no puede ser menor al costo.”</p>
Flujos Alternos	
Flujo Alternativo No. 1	<p>Si existe algún error en el proceso:</p> <ol style="list-style-type: none"> 1. El sistema registra el error. 2. El sistema muestra el siguiente mensaje de error: “No se puede realizar la acción dado a que se presentó un problema. <ol style="list-style-type: none"> a) Intente ejecutar nuevamente la acción. b) Si el problema persiste consulte al administrador del sistema. <p>Identificador del error: {Identificador}”</p> <ol style="list-style-type: none"> 3. El sistema registra el Log de error. 4. El caso de uso finaliza
Flujo Alternativo No. 2	<p>Si existente errores de validación:</p> <ol style="list-style-type: none"> 1.El sistema muestra el mensaje de error. 2.El caso de uso finaliza.
Flujo Alternativo No. 3	<p>Si el actor ejecuta la acción “cancelar”:</p> <ol style="list-style-type: none"> 1.El sistema limpia la información completada. 2.El caso de uso finaliza.
Flujo Alternativo No. 4	<p>Si el actor no dispone de alguno de los siguiente roles: Administrador o Usuario Estándar:</p> <ol style="list-style-type: none"> 1. El sistema muestra el siguiente mensaje: “Acceso Denegado!!! El usuario no cuenta con los requisitos para poder ingresar a esta opción.” 2. El caso de uso finaliza.

Flujo Alternativo No. 5	<p>Si ejecuta la acción “cerrar” en cualquiera de la pantallas emergentes del sistema:</p> <ol style="list-style-type: none"> 1. El sistema cierra la pantalla. 2. El caso de uso finaliza.
Flujo Alternativo No. 6	<p>Si el cliente no está de acuerdo en proporcionar sus datos personales o no necesita una factura electrónica:</p> <ol style="list-style-type: none"> 1. El actor selecciona el tipo de documento “Tiquete Electrónico”. 2. El actor omite ingresar la información del cliente. 3. El caso de uso continua en el flujo base 9.
Flujo Alternativo No. 7	<p>Si la condición de venta es de contado y no se requiere que el cliente este registrado en el mantenimiento de clientes:</p> <ol style="list-style-type: none"> 1. El actor selecciona el tipo de documento “Tiquete Electrónico” o “Factura Electrónica”. La elección del tipo de documento depende si el cliente desea utilizar el comprobante electrónico como crédito fiscal. Los tiquetes electrónicos no son válidos para respaldar un crédito fiscal. 2. El actor selecciona un tipo de identificación diferente a “Cliente de contado”. 3. El actor ingresa la identificación del cliente y ejecuta la acción “Buscar”. 4. El sistema obtiene el nombre del cliente del padrón de identificaciones. 5. El actor puede ingresar el nombre del cliente de manera manual en caso de que no se encuentre información de la identificación en el padrón. 6. El actor puede modificar el nombre del cliente. 7. El caso de uso continua en el flujo base 9.
Flujo Alternativo No. 7	<p>Si al digitar la identificación del cliente se encuentra más de un registro asociado al número de identificación:</p> <p>(* Este flujo alterno es funcional para el cliente CCSS, todos los hospitales se encuentran bajo una misma cédula jurídica, lo que los identifica es la unidad ejecutora, por lo tanto, se considera valido que dos o más clientes compartan una identificación.)</p> <ol style="list-style-type: none"> 1. El sistema muestra los clientes asociados a la identificación en formato tabla, mostrando el tipo de identificación, la identificación, el nombre, si tiene crédito, si es exonerado y el plazo. 2. El actor selecciona un registro. 3. El caso de uso continua el flujo base 9.
Flujo Alternativo No. 8	<p>Si se obtuvo un Token de la plataforma del Ministerio de hacienda y se cierra la pantalla de facturación:</p>

	<ol style="list-style-type: none"> 1. El sistema cierra la conexión con el servicio de autenticación del Ministerio de Hacienda. 2. El caso de uso finaliza.
Requerimientos especiales	
<ol style="list-style-type: none"> 1. El actor con el perfil Usuario Estándar no podrá emitir facturas de crédito. 2. No se permite la facturación bajo saldo por lo que los artículos que afecten el inventario deben contar con existencias en bodega. 3. El actor con el perfil Usuario Estándar no podrá otorgar descuentos por línea. 4. Si el documento electrónico corresponde se debe exonerar y no se cuenta con la información, se debe realizar la gestión de ingreso de información desde el mantenimiento de clientes. 	
Post-Condiciones	
Se envía un correo electrónico al cliente y al emisor del documento electrónico que contiene los siguiente archivos: documento electrónico en formato XML, respuesta de hacienda en formato XML y la representación gráfica del documento en formato pdf.	

Fuente: Elaboración propia (2024).

Tabla 37.

Caso de uso para generar anulación

Prototipo: Prototipo funcional para la gestión de ventas de la empresa Zapatitos Lindos S.A., ubicada en San José.	
Número Caso de Uso: 27	Nombre del Caso de Uso: Generar Anulación.
Fecha elaboración:	26/04/2024
Descripción Caso de Uso:	Permitir anulación de comprobantes electrónicos. En facturación electrónica el concepto de anulación no existe, por lo que al anular el comprobante lo que se genera es una nota de crédito asociado a la factura o tiquete electrónico.
Autor caso de uso:	Daniel David Quesada Fallas.
Actores relacionados:	Administrador
Precondiciones:	<ol style="list-style-type: none"> 1. El actor debe haber iniciado sesión en el sistema. 2. El actor debe contar con el rol: “Administrador”. 3. El actor debe conocer la clave del comprobante electrónico al cual quiere asociar la nota de crédito.
Flujo Básico del caso de uso	
<ol style="list-style-type: none"> 1. Este caso de uso inicia cuando el actor selecciona del menú “Facturación” > “Procesos” la opción “Anulación”. 2. El sistema muestra y solicita la siguiente información: <ul style="list-style-type: none"> • Clave Numérica. • Tipo documento (No editable). • Documento exonerado (Sí/No). <u>Datos del cliente</u> (No editables). <ul style="list-style-type: none"> • Tipo Identificación. • Identificación. 	

- Nombre del cliente.

Detalle (Formato tabla)

- N° Línea.
- Artículo.
- Descripción.
- CABys.
- Precio.
- % Desc 1.
- % Desc 2.
- IVA.
- Total Línea.

Totales

- Subtotal.
- Descuento.
- IVA.
- Total.
- Código de referencia.
- Observación | Razón de referencia.

3. El actor ingresa la clave numérica y ejecuta la acción “Cargar Documento”.
4. El sistema obtiene la información del comprobante electrónico y carga la información.
5. El sistema ejecuta validaciones.
6. El actor selecciona el código de referencia.
7. El actor digita la observación o razón de referencia.
8. El actor ejecuta y confirma la acción Anular.
9. El sistema realiza validaciones.
10. El sistema devuelve retorna las existencias de los artículos que afectan el inventario.
11. El sistema registra en la bitácora Kardex los artículos que afectan el inventario.
12. El sistema realiza el cálculo costo promedio ponderado para los artículos que afectan el inventario.
13. Si el documento electrónico es de crédito y cuenta con saldo se liquida.
14. El sistema registra la nota de crédito.
15. El sistema emita el comprobante electrónico.
16. El sistema envía el comprobante electrónico al Ministerio de Hacienda.
17. El sistema consulta el estado del comprobante electrónico.
18. El sistema muestra en pantalla la representación gráfica del comprobante electrónico.
19. El sistema envía el correo electrónico a las direcciones registradas en el comprobante de referencia.
20. El caso de uso finaliza.

Sub Flujos

**Sub Flujo Validaciones
(SF-01)**

1. El sistema verifica que la clave del documento numérico sea válida. Si es inválida le sistema muestra el siguiente mensaje: “No se encontró información con la clave numérica {clave}”.
2. El sistema verifica que el documento electrónico no este anulado. El sistema muestra el siguiente mensaje: “El Documento ya se encuentra anulado”.

	<p>3. El sistema verifica que el documento electrónico no tenga una devolución asociada. El sistema muestra el siguiente mensaje “El Documento no se puede anular, ya que se procesó como devolución (Anulación Parcial), si desea aplicar otra Nota de Crédito debe procesarlo desde la pantalla de devoluciones.”</p> <p>4. El sistema verifica que se haya cargado el documento electrónico a anular. El sistema muestra el siguiente mensaje de error: “No podrá continuar, deberá cargar el documento previamente.”</p> <p>5. El sistema verifica que se haya seleccionado el tipo de referencia. El sistema muestra el siguiente mensaje: “El campo "Tipo de referencia" es requerido.”</p> <p>6. El sistema verifica que se haya ingresado una observación de la razón de referencia por la cual se genera la nota de crédito. El sistema muestra el siguiente mensaje: “El campo "Observación" es requerido.”</p>
Flujos Alternos	
Flujo Alternativo No. 1	<p>Si existe algún error en el proceso:</p> <p>1. El sistema registra el error.</p> <p>2. El sistema muestra el siguiente mensaje de error: “No se puede realizar la acción dado a que se presentó un problema”.</p> <p style="padding-left: 40px;">a) Intente ejecutar nuevamente la acción.</p> <p style="padding-left: 40px;">b) Si el problema persiste consulte al administrador del sistema.</p> <p>Identificador del error: “{Identificador}”</p> <p>3. El sistema registra el Log de error.</p> <p>4. El caso de uso finaliza</p>
Flujo Alternativo No. 2	<p>Si el actor no dispone de alguno de los siguientes roles: Administrador</p> <p>1. El sistema muestra el siguiente mensaje: “Acceso Denegado!!! El usuario no cuenta con los requisitos para poder ingresar a esta opción.”</p> <p>2. El caso de uso finaliza.</p>
Requerimientos especiales	
Post-Condiciones	

Fuente: Elaboración propia (2024).

Tabla 38.

Caso de uso para generar la devolución

Prototipo: Prototipo funcional para la gestión de ventas de la empresa Zapatitos Lindos S.A, ubicada en San José.	
Número Caso de Uso: 28	Nombre del Caso de Uso: Generar devolución.
Fecha elaboración:	27/04/2024
Descripción Caso de Uso:	Permitir anulación parcial de comprobantes electrónicos. En facturación electrónica el concepto de anulación no existe, por lo que al anular el comprobante lo que se genera es una nota de crédito asociado a la factura o tiquete electrónico.
Autor caso de uso:	Daniel David Quesada Fallas.
Actores relacionados:	Administrador
Precondiciones:	<ol style="list-style-type: none"> 1. El actor debe haber iniciado sesión en el sistema. 2. El actor debe contar con el rol: “Administrador”. 3. El actor debe conocer la clave del comprobante electrónico al cual quiere asociar la nota de crédito.
Flujo Básico del caso de uso	
<ol style="list-style-type: none"> 1. Este caso de uso inicia cuando el actor selecciona del menú “Facturación” > “Procesos” la opción “Devolución”. 2. El sistema muestra y solicita la siguiente información: <ul style="list-style-type: none"> • Clave Numérica. • Tipo documento (No editable). • Documento exonerado (Sí/No). <u>Datos del cliente</u> (No editables). <ul style="list-style-type: none"> • Tipo Identificación. • Identificación. • Nombre del cliente. <u>Detalle</u> (Formato tabla) <ul style="list-style-type: none"> • N° Línea. • Artículo. • Descripción. • CABys. • Precio. • % Desc 1. • % Desc 2. • IVA. • Total Línea. <u>Totales</u> <ul style="list-style-type: none"> • Subtotal. • Descuento. • IVA. • Total. • Código de referencia. 	

<ul style="list-style-type: none"> • Observación Razón de referencia. <ol style="list-style-type: none"> 3. El actor ingresa la clave numérica y ejecuta la acción “Cargar Documento”. 4. El sistema obtiene la información del comprobante electrónico y carga la información de las líneas que no se han anulado. 5. El sistema ejecuta validaciones. 6. El actor elimina las líneas que no sea devolver. 7. El actor selecciona el código de referencia. 8. El actor digita la observación o razón de referencia. 9. El actor ejecuta y confirma la acción devolución. 10. El sistema realiza validaciones. 11. El sistema devuelve retorna las existencias de los artículos que afectan el inventario. 12. El sistema registra en la bitácora Kardex los artículos que afectan el inventario. 13. El sistema realiza el cálculo costo promedio ponderado para los artículos que afectan el inventario. 14. Si el documento electrónico es de crédito y cuenta con saldo, el sistema lo liquida. 15. El sistema registra la nota de crédito. 16. El sistema emita el comprobante electrónico. 17. El sistema envía el comprobante electrónico al Ministerio de Hacienda. 18. El sistema consulta el estado del comprobante electrónico. 19. El sistema muestra en pantalla la representación gráfica del comprobante electrónico. 20. El sistema envía el correo electrónico a las direcciones registradas en el comprobante de referencia. 21. El caso de uso finaliza. 	
Sub Flujos	
Sub Flujo Validaciones (SF-01)	<ol style="list-style-type: none"> 1. El sistema verifica que la clave del documento numérico sea válida. Si es inválida le sistema muestra el siguiente mensaje: “No se encontró información con la clave numérica {clave}”. 2. El sistema verifica que el documento electrónico no este anulado. El sistema muestra el siguiente mensaje: “El Documento ya se encuentra anulado”. 3. El sistema verifica que el documento electrónico no tenga una devolución asociada. El sistema muestra el siguiente mensaje “El Documento no se puede anular, ya que se procesó como devolución (Anulación Parcial), si desea aplicar otra Nota de Crédito debe procesarlo desde la pantalla de devoluciones.” 4. El sistema verifica que se haya cargado el documento electrónico a anular. El sistema muestra el siguiente mensaje de error: “No podrá continuar, deberá cargar el documento previamente.” 5. El sistema verifica que se haya seleccionado el tipo de referencia. El sistema muestra el siguiente mensaje: “El campo "Tipo de referencia" es requerido.” 6. El sistema verifica que se haya ingresado una observación de la razón de referencia por la que se genera la nota de crédito. El sistema muestra el siguiente mensaje: “El campo "Observación" es requerido.”

	7.El sistema verifica que si exista una sola línea no se puede eliminar. El sistema muestra el siguiente mensaje: “No podrá continuar, para realizar la devolución debe tener al menos una línea disponible.”
Flujos Alternos	
Flujo Alternativo No. 1	<p>Si existe algún error en el proceso:</p> <ol style="list-style-type: none"> 1. El sistema registra el error. 2. El sistema muestra el siguiente mensaje de error: “No se puede realizar la acción dado a que se presentó un problema”. <ol style="list-style-type: none"> a) Intente ejecutar nuevamente la acción. b) Si el problema persiste consulte al administrador del sistema. <p>Identificador del error: {Identificador}”</p> <ol style="list-style-type: none"> 3. El sistema registra el Log de error. 4. El caso de uso finaliza.
Flujo Alternativo No. 2	<p>Si el actor no dispone de alguno de los siguientes roles: Administrador</p> <ol style="list-style-type: none"> 1. El sistema muestra el siguiente mensaje: “Acceso Denegado!!! El usuario no cuenta con los requisitos para poder ingresar a esta opción.” 2. El caso de uso finaliza.
Requerimientos especiales	
Post-Condiciones	

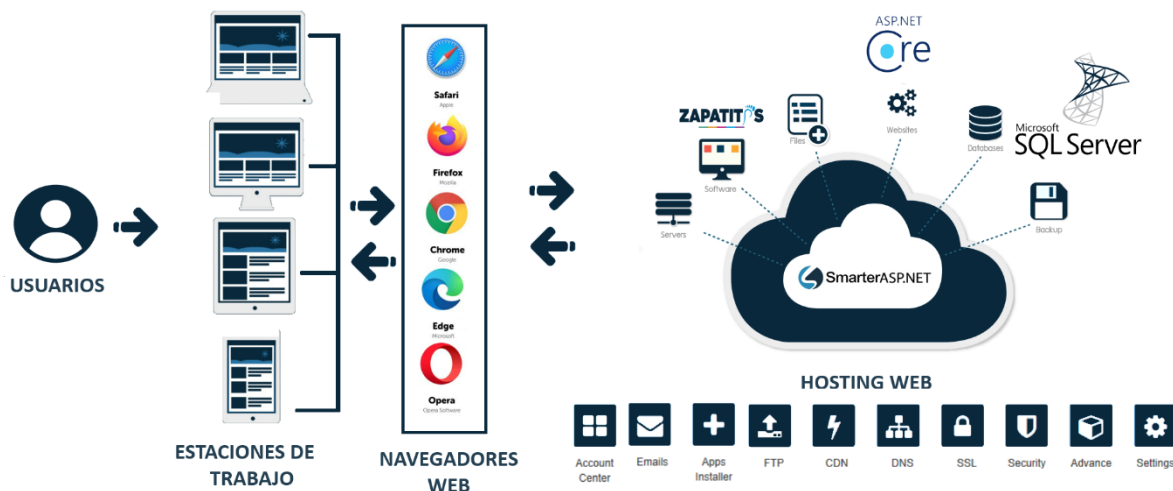
Fuente: Elaboración propia (2024).

Diseño

Arquitectura Del Sistema

Figura 2.

Arquitectura del sistema



Fuente: Elaboración propia (2024).

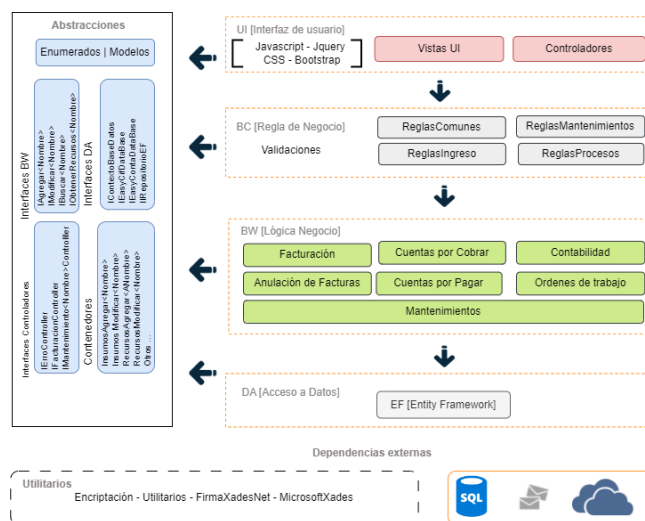
En lo que respecta a la arquitectura del sistema, se permitirá el acceso desde cualquier dispositivo electrónico con capacidad de conexión a internet, utilizando un navegador web. Por medio del navegador se establece la comunicación con la infraestructura y los servicios alojados en el *hosting* SmarterAsp.Net. Este proveedor es responsable de gestionar una serie de recursos críticos, como almacenamiento, ancho de banda y seguridad. Además, se encargará de la instalación de certificados de seguridad y otras características esenciales para la operatividad del sistema.

Para la arquitectura de un sistema web desarrollado en .NET Core 6 con ASP.NET MVC, es esencial que el servicio de hosting cumpla con ciertos requisitos. El servidor debe ser compatible con .NET Core 6, y se recomienda el uso de servidores web como Internet Information Services para entornos *Windows*, o *Nginx* o *Apache* para entornos *Linux*. Además, dado que la base de datos utilizada es SQL Server, el servicio de hosting debe ofrecer compatibilidad y soporte para SQL Server, ya sea en su versión completa o SQL Server Express. Asimismo, el *hosting* debe proporcionar suficientes recursos de CPU, memoria y almacenamiento para garantizar el rendimiento adecuado de la aplicación, considerando el tamaño y la complejidad del sistema, así como la cantidad de usuarios concurrentes.

Arquitectura Del Software

Figura 3.

Arquitectura del Software



Fuente: Elaboración propia (2024).

La arquitectura de *software* descrita sigue un enfoque de capas bien definidas, típico de las aplicaciones basadas en el patrón Modelo-Vista-Controlador (MVC) de ASP.NET Core. En la parte transversal del diagrama se encuentra la Capa de Abstracciones, donde se definen los modelos de datos y las interfaces que especifican los contratos para las operaciones del sistema. Esta capa desempeña un papel crucial en la inversión de dependencias, donde los módulos de bajo nivel dependen de las abstracciones y no de los detalles concretos. Este enfoque permite una mayor flexibilidad y extensibilidad en el sistema, lo que facilita la sustitución de componentes sin afectar a otras partes del sistema.

En la parte superior del diagrama se encuentra la capa de interfaz de usuario, donde se gestionan todos los aspectos relacionados con la presentación y la interacción con el usuario. Aquí se incluyen componentes como vistas y controladores que permiten al usuario interactuar con el sistema de manera intuitiva y eficiente.

La capa de reglas de negocio está estrechamente relacionada con la capa de lógica de negocio y se encarga de implementar las validaciones y reglas específicas del negocio, asegurando la integridad y consistencia de los datos. En esta capa, se definen las normativas y restricciones que guían el comportamiento del sistema, tales como la validación de formatos de datos, restricciones de integridad referencial, reglas de acceso y otras reglas de negocio complejas.

Además, la capa de reglas de negocio se encarga de garantizar que los datos ingresados cumplan con los estándares establecidos por la organización, con lo que se evita la introducción de datos incorrectos o inconsistentes.

Por otro lado, la capa de lógica de negocio es el corazón del sistema, donde reside la mayor parte de la funcionalidad. Aquí se encuentran los servicios que encapsulan la lógica de la aplicación. Esta es la capa que se encarga de coordinar la interacción entre los distintos componentes del sistema y llevar a cabo las operaciones empresariales necesarias para cumplir con los objetivos del negocio. También, se encarga de tareas como la gestión de transacciones, la seguridad de los datos y la generación de informes. Su diseño está orientado a la reutilización, la cohesión y la modularidad, y permite que los servicios sean fácilmente escalables y mantenibles a medida que el sistema evoluciona.

Justo debajo de la capa de lógica de negocio se encuentra la capa de datos, encargada de gestionar el acceso a los datos almacenados en la base de datos utilizando Entity Framework. Aquí se implementan los repositorios, que proporcionan una abstracción sobre la capa de persistencia de datos, lo que facilita su manipulación y mantenimiento.

Por último, los servicios de terceros se representan en una sección aparte del diagrama, mostrando las conexiones entre la capa de lógica de negocio o la capa de interfaz de usuario y los servicios externos, como *GitHub*, motor de base de datos, servicios de correo electrónico y APIs del Ministerio de Hacienda. Estos servicios permiten la integración con sistemas externos y amplían la funcionalidad del sistema.

Diseño De Entradas

En el marco del desarrollo del prototipo funcional para la gestión de ventas de la empresa Zapatitos Lindos S.A., ubicada en San José, se hace hincapié en la importancia del diseño de entradas, tanto gráficas como no gráficas, como elemento fundamental para garantizar una experiencia de usuario óptima. Este apartado del proyecto se centra en la presentación de las distintas pantallas que conforman prototipo funcional. Cabe resaltar que estas pantallas están estandarizadas para mantener una coherencia visual a lo largo de todo el proyecto.

Es fundamental destacar que estas pantallas están diseñadas siguiendo un enfoque *responsive*, lo que implica que se adaptan de manera eficiente a diversos dispositivos, con lo cual se asegura una accesibilidad óptima desde cualquier navegador y cualquier dispositivo. Esta

adaptabilidad es esencial para garantizar que los usuarios puedan interactuar con el prototipo de manera fluida y sin limitaciones tecnológicas.

Además, el diseño de las entradas no se limita únicamente a su aspecto visual, sino que también se ha cuidado su usabilidad y funcionalidad para asegurar una experiencia de usuario intuitiva y satisfactoria. Esto se logra mediante la aplicación de principios de diseño centrado en el usuario y la consideración de las necesidades y expectativas del cliente.

Figura 4.

Diseño de entradas - Ingreso al sistema



La imagen muestra una interfaz de usuario para el sistema ZAPATITOS. En la parte superior, se encuentra el logo 'ZAPATITOS' con un icono de un zapato azul y una línea de colores (verde, amarillo, rojo, azul) debajo. El título principal es 'Ingresar datos para iniciar sesión'. Hay dos campos de entrada: 'Usuario' con un ícono de persona a la derecha, y 'Contraseña' con un ícono de candado a la derecha. Debajo de los campos hay un botón azul con el texto 'Ingresar'. En la parte inferior, se muestra el texto 'Copyright © 2024'.

Fuente: Elaboración propia (2024).

Pantalla de ingreso al sistema, el usuario debe digitar los datos solicitadas y presionar el botón ingresar para poder hacer uso de las diferentes funcionalidades del sistema.

Figura 5.

Diseño entradas – Selección de módulo y sucursal



The image shows a login form for 'ZAPATIT'S'. At the top is the logo 'ZAPATIT'S' with a colorful underline. Below the logo, the text 'Seleccionar datos para ingresar al sistema' is centered. There are two dropdown menus: 'Módulo' with 'CIF' selected, and 'Sucursal' with 'San José' selected. A blue 'Continuar' button is at the bottom. A copyright notice 'Copyright © 2024' is at the very bottom.

Fuente: Elaboración propia (2024).

El prototipo funcional desarrollado es un sistema multisucursal con diversas funcionalidades organizadas en módulos.

Esta pantalla únicamente se mostrará cuando el usuario esté asociado a más de un módulo o sucursal, y en las listas de módulos y Sucursales únicamente se visualizará la información que se le asignó en el mantenimiento de usuarios; para continuar el usuario deberá seleccionar el módulo y la sucursal del sistema a la que desea conectarse.

Figura 6.

Diseño de entradas – Cambio de contraseña al iniciar sesión



The image shows a web form for changing a password. At the top, the logo 'ZAPATIT'S' is displayed with a colorful underline. Below the logo, a message reads: 'Debe realizar el cambio de contraseña para poder continuar!!!'. The form contains two input fields: 'Nueva Contraseña:' and 'Confirmar contraseña:', each with a lock icon on the right. Below these fields is a checkbox labeled 'Mostrar Contraseña'. At the bottom, there are two buttons: a blue button with a house icon and the text 'Cambiar la contraseña', and a red button with a right-pointing arrow and the text 'Salir'.

Fuente: Elaboración propia (2024).

Esta pantalla se le mostrará al usuario la primera vez que ingrese posterior a una creación o modificación de datos del usuario llevada a cabo por un usuario administrador de sistema, con el objetivo de asegurar que las contraseñas solo sean del conocimiento del usuario final.

Figura 7.

Diseño de entradas – Mantenimiento de usuarios

The screenshot displays the 'Agregar Usuarios' (Add Users) form within the 'Mantenimiento Usuarios' (User Maintenance) section. The interface includes a dark sidebar with navigation options like 'Inicio', 'Facturación', 'Inventarios', 'Compras', 'Mantenimientos', 'Seguridad', 'Usuarios', 'Cambiar Contraseña', and 'Cerrar sesión'. The main content area has tabs for 'Datos de usuario', 'Asignación de módulos', and 'Asignación de sucursales'. The 'Datos de usuario' tab is active, showing a form with the following fields:

- Usuario:** Input field with a search icon.
- Correo:** Input field with an email icon.
- Estado:** Dropdown menu with 'Activo' selected.
- Nombre:** Input field labeled 'Nombre del usuario'.
- Contraseña:** Input field with a lock icon.
- Rol:** Dropdown menu with 'Administrador' selected.
- Apellido:** Input field labeled 'Apellidos'.
- Confirmar contraseña:** Input field with a lock icon.

At the bottom of the form are 'Guardar' (Save) and 'Cancelar' (Cancel) buttons. The footer contains 'Copyright © 2024 Easy CIF. All rights reserved.' and 'Version 1.1.0'.

Fuente: Elaboración propia (2024).

Figura 8.

Diseño de entradas – Mantenimiento de usuarios – Asignación de módulos

The screenshot displays the 'Asignación de módulos' (Module Assignment) form within the 'Mantenimiento Usuarios' (User Maintenance) section. The interface is consistent with Figure 7, but the 'Asignación de módulos' tab is active. It shows two side-by-side lists for module assignment:

- Módulos:** A list containing 'CIF', 'CXC', 'CXP', and 'CONTABILIDAD'. It has a 'Filtrar' (Filter) input and a '>>' button to move items to the right list.
- Lista vacía:** An empty list with a 'Filtrar' (Filter) input and a '<<' button to move items back to the left list.

'Guardar' (Save) and 'Cancelar' (Cancel) buttons are at the bottom. The footer contains 'Copyright © 2024 Easy CIF. All rights reserved.' and 'Version 1.1.0'.

Fuente: Elaboración propia (2024).

Figura 9.

Diseño de entradas – Mantenimiento de usuarios – Asignación de sucursales

The screenshot shows the 'Agregar Usuarios' (Add Users) form in the 'Mantenimiento Usuarios' (User Maintenance) section. The 'Asignación de sucursales' (Branch Assignment) tab is active. It features two side-by-side dropdown menus for selecting branches. The left dropdown is labeled 'Ezcazú' and the right one is labeled 'San José'. Below the dropdowns are 'Guardar' (Save) and 'Cancelar' (Cancel) buttons. The interface includes a sidebar with navigation options like 'Inicio', 'Facturación', 'Inventarios', 'Compras', 'Mantenimientos', 'Seguridad', 'Usuarios', 'Cambiar Contraseña', and 'Cerrar sesión'. The top right shows the user 'dquesada' and the page title 'Seguridad / Usuarios'. Copyright information '© 2024 Easy CIF. All rights reserved.' and version 'Version 1.1.0' are visible at the bottom.

Fuente: Elaboración propia (2024).

Figura 10.

Diseño de entradas – Mantenimiento de usuarios – Buscar usuarios

The screenshot shows the 'Buscar usuarios' (Search Users) form in the 'Mantenimiento Usuarios' (User Maintenance) section. It includes search filters for 'Buscar por:' (Search by) with options for 'Usuario' (selected), 'Nombre completo', and 'Llave de búsqueda' (Search key). There is also an 'Estado' (Status) dropdown set to 'Activa'. Below these are more search criteria: 'Que inicie', 'Que contenga', and 'Que finalice'. A 'Buscar' (Search) button and a 'Limpiar' (Clear) button are present. The 'Resultados' (Results) section shows a table with 2 records. The table has columns for 'Usuario', 'Nombre Completo', 'Correo', 'Rol', and 'Estado'. Each record has a 'Bloquear' (Block) button. The first record is for 'dquesada' (Daniel David Quesada Fallas, Administrator) and the second is for 'dquesada2' (Daniel II Quesada Fallas, Standard User). The interface also shows 'Mostrar 10 registros' (Show 10 records) and pagination controls for 'Anterior', '1', and 'Siguiente'. The sidebar and top navigation are consistent with Figure 9.

	Usuario	Nombre Completo	Correo	Rol	Estado	
	dquesada	Daniel David Quesada Fallas	daniel13pz@gmail.com	Administrador	Activo	
	dquesada2	Daniel II Quesada Fallas	daniel13pz@gmail.com	Usuario Estándar	Activo	

Fuente: Elaboración propia (2024).

Figura 11.

Diseño de entradas – Mantenimiento de usuarios – Modificar usuario

The screenshot displays the 'ZAPATIT'S' web application interface. On the left is a dark sidebar with navigation options: Inicio, Facturación, Inventarios, Compras, Mantenimientos, Seguridad, Cambiar Contraseña, and Cerrar sesión. The main content area is titled 'Mantenimiento Usuarios' and contains a 'Modificar Usuarios' form. The form has three tabs: 'Datos de usuario' (selected), 'Asignación de módulos', and 'Asignación de sucursales'. The form fields are arranged in a grid:

* Usuario:	Correo:	* Estado
<input type="text" value="dquesada"/>	<input type="text" value="pruebas@gmail.com"/>	<input type="text" value="Activo"/>
* Nombre	* Contraseña:	* Rol
<input type="text" value="Daniel David"/>	<input type="password" value="*****"/>	<input type="text" value="Administrador"/>
Apellido	* Confirmar contraseña:	
<input type="text" value="Quesada Fallas"/>	<input type="password" value="*****"/>	

At the bottom of the form are 'Guardar' and 'Cancelar' buttons. The footer contains 'Copyright © 2024 Easy CIF. All rights reserved.' and 'Version 1.1.0'.

Fuente: Elaboración propia (2024).

El diseño de entradas correspondientes al mantenimiento de usuarios muestra las diferentes acciones que se puede realizar desde el mantenimiento tales como: crear usuario, modificar usuario, buscar usuarios, asignar módulo y asignar sucursales. Únicamente el usuario con el perfil administrador podrá ingresar a esta pantalla a gestionar los usuarios.

Figura 12.

Diseño de entradas – Mantenimiento de compañías (emisor factura electrónica)

The screenshot displays the 'Agregar Compañía' (Add Company) form within the ZAPATIT S system. The interface features a dark sidebar on the left with navigation icons for 'Inicio', 'Facturación', 'Inventarios', 'Compras', 'Mantenimientos', 'Sistema', 'Compañía', 'Sucursal', 'Inventarios', 'Seguridad', 'Cambiar Contraseña', and 'Cerrar sesión'. The main content area is titled 'Mantenimiento Compañías' and includes a search bar. The form is divided into three columns:

- Column 1 (Left):**
 - * Tipo identificación:** Cédula Física
 - * Identificación:** Identificación de la compañía
 - * Nombre:** Nombre
 - Nombre Comercial:** Nombre comercial
 - * Correo:** Correo
 - Código de área teléfono:** (506) COSTA RICA
 - Teléfono:** Teléfono
- Column 2 (Middle):**
 - Código de area fax:** (506) COSTA RICA
 - Fax:** Fax
 - * Provincia:** San José
 - * Cantón:** San José
 - * Distrito:** Carmen
 - * Barrio:** San José (parte)
 - Dirección (Otras señas):** Otras Señas
- Column 3 (Right):**
 - * Usuario Hacienda:** Usuario Hacienda
 - * Contraseña Hacienda:** Contraseña Hacienda
 - * Clave del certificado:** Clave del certificado
 - Certificado Firma Digital:** Examinar, Browse
 - Fecha Vencimiento:** Fecha Vencimiento
 - * Estado:** Activa

At the bottom of the form are 'Guardar' and 'Cancelar' buttons. A footer contains 'Copyright © 2024 Easy CIF. All rights reserved.' and 'Version 1.1.0'.

Fuente: Elaboración propia (2024).

Esta pantalla permite agregar, modificar, inactivar y activar la compañía. Es una de las pantallas más importantes del sistema ya que tiene información requerida para la configuración y para correcto funcionamiento de la facturación electrónica.

Figura 13.

Diseño de entradas – Pantalla de error (404 recurso no encontrado)

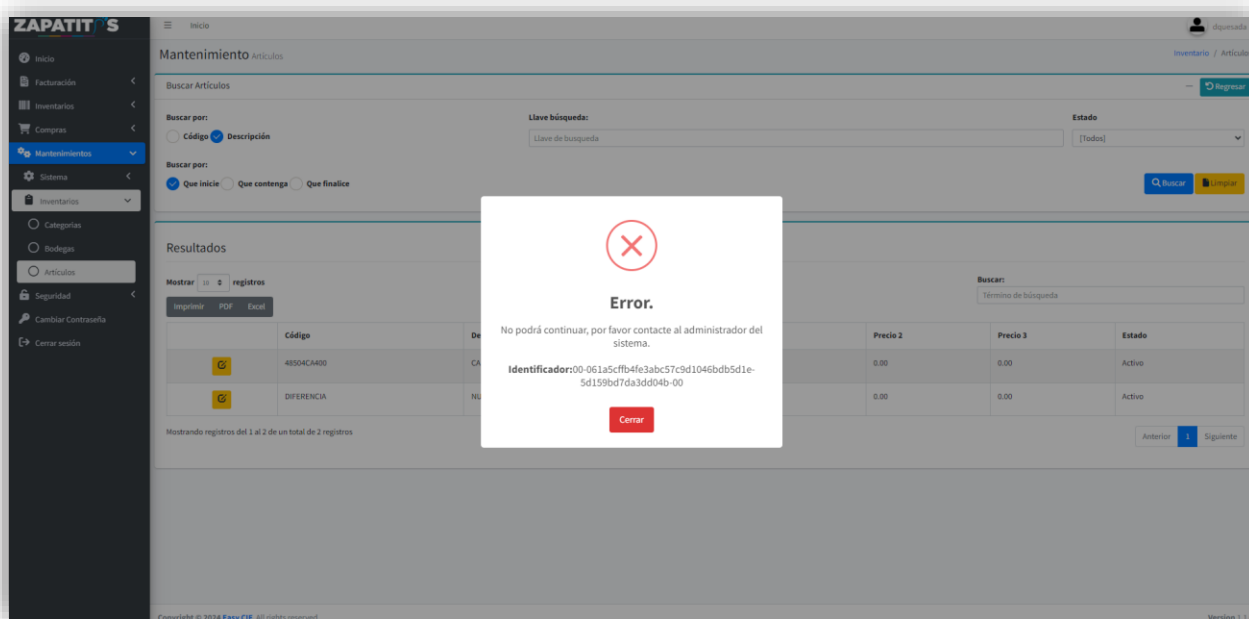


Fuente: Elaboración propia (2024).

Pantalla de error 404 cuando un recurso del sistema no es encontrado.

Figura 14.

Diseño de entradas – Pantalla de error (error controlado)



Fuente: Elaboración propia (2024).

El control de errores es uno de los puntos más importantes en todo sistema importante, por esa razón si se presentara algún error al hacer una petición al servidor del lado del cliente se mostrará la siguiente pantalla de error.

Figura 15.

Diseño de entradas – Pantalla de error (500 internal sever)



Fuente: Elaboración propia

En caso de presentarse un error del lado del servidor el sistema muestra la siguiente pantalla de error .

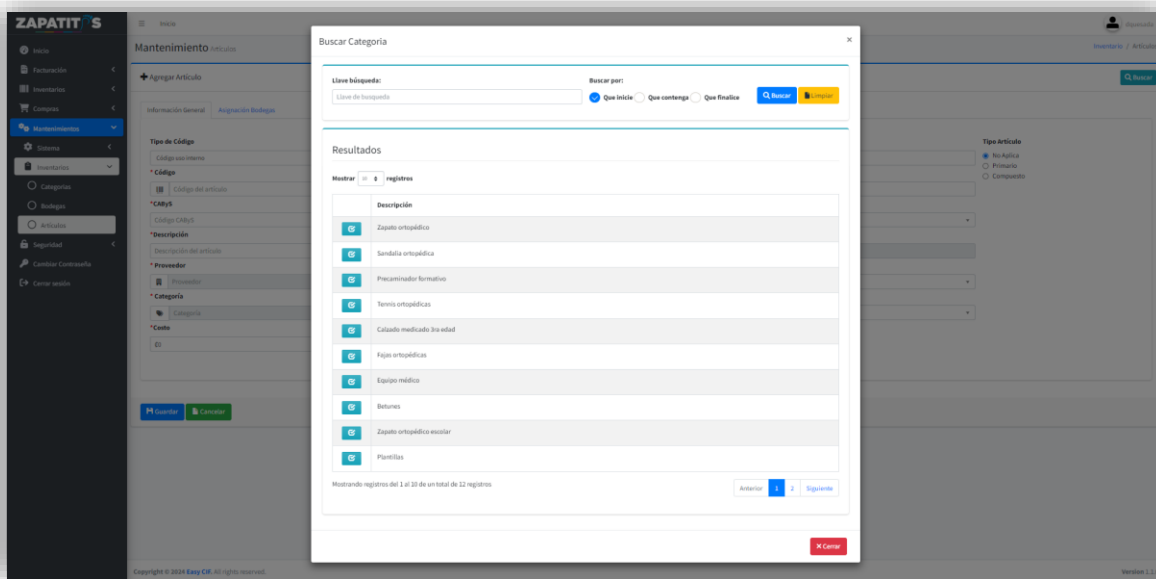
Figura 16.

Diseño de entradas – Mantenimiento de artículos

Fuente: Elaboración propia(2024).

Figura 17.

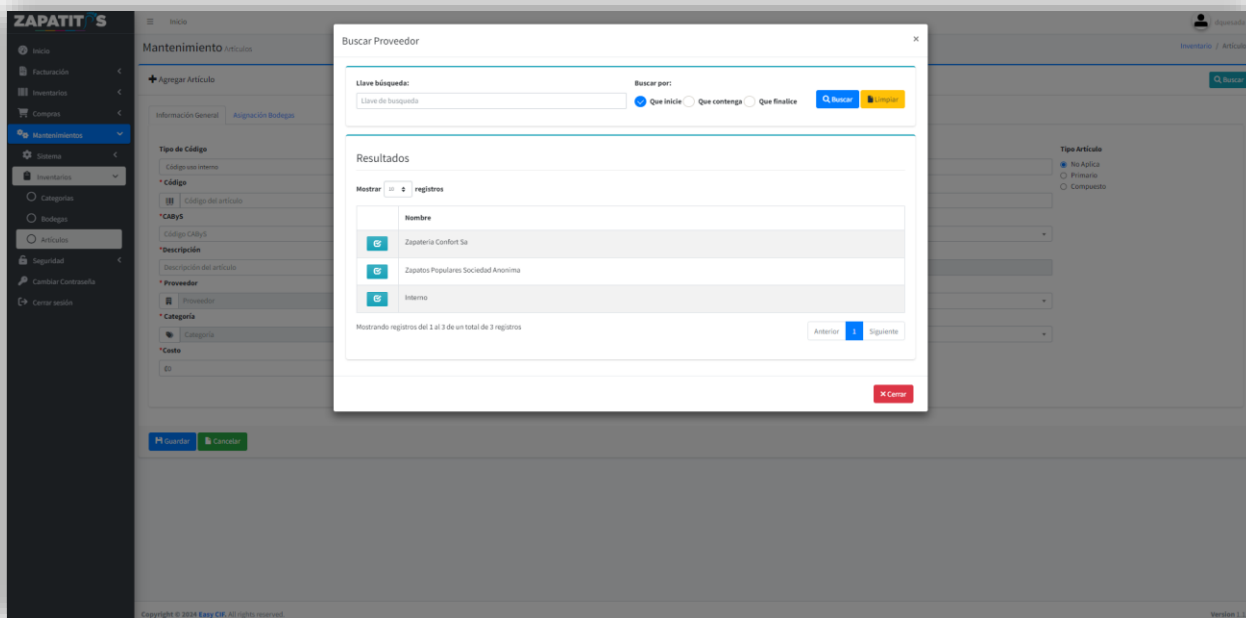
Diseño de entradas – Mantenimiento de artículos – Buscar categoría



Fuente: Elaboración propia (2024).

Figura 18.

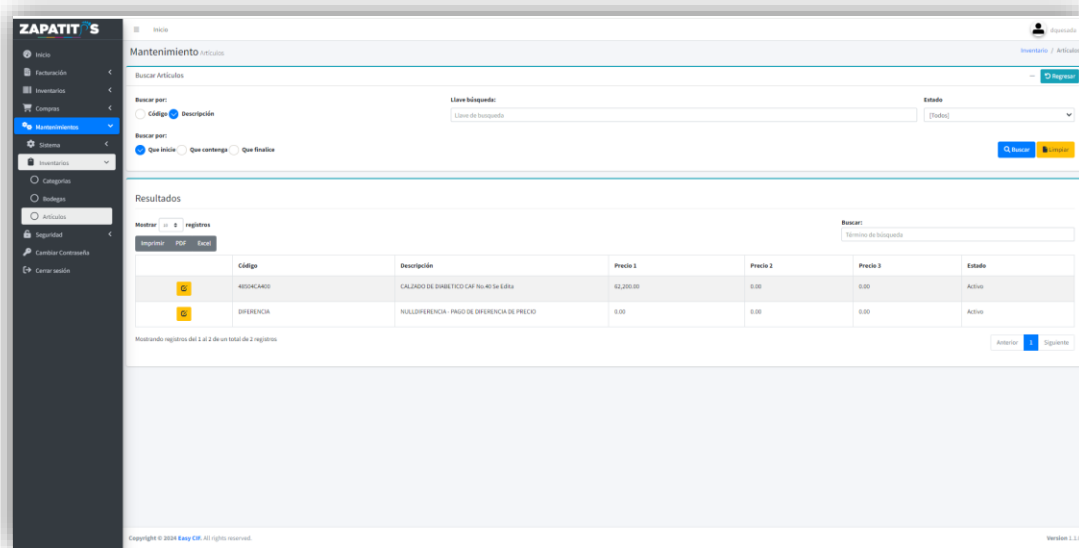
Diseño de entradas – Mantenimiento de artículos – Buscar proveedor



Fuente: Elaboración propia (2024).

Figura 19.

Diseño de entradas – Mantenimiento de artículos – Buscar artículos

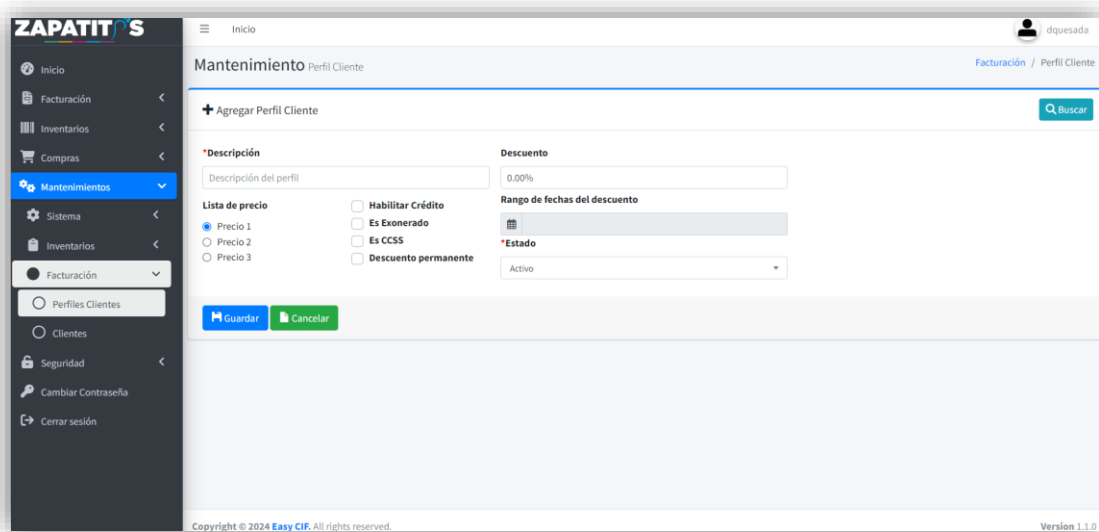


Fuente: Elaboración propia (2024).

Estas pantallas permiten agregar, buscar y modificar la información relacionada con los artículos del sistema, dentro de esta pantalla se encuentran dos búsquedas la de proveedores y categorías.

Figura 20.

Diseño de entradas – Mantenimiento de perfil clientes



Fuente: Elaboración propia (2024).

Figura 21.

Diseño de entradas – Mantenimiento de perfil clientes – Buscar

The screenshot displays the 'Mantenimiento Perfil Cliente' page in the ZAPATITOS system. The search filter section includes a search key, a state dropdown set to 'Activa', and search criteria options: 'Que inicie' (selected), 'Que contenga', and 'Que finalice'. The results table shows the following data:

	Descripción	Tiene Crédito	Es Exonerado	Es CCSS	Lista de Precio	Estado
	Cliente Gravado	No	No	No	Precio 1	Activo
	Cliente Exonerado	No	Sí	No	Precio 1	Activo
	Distribuidores	Sí	No	Sí	Precio 1	Activo
	Búsqueda CSS	Sí	No	Sí	Precio 2	Activo

Fuente: Elaboración propia (2024)

Figura 22.

Diseño de entradas – Mantenimiento de perfil clientes – Modificar

The screenshot displays the 'Modificar Perfil Cliente' page in the ZAPATITOS system. The form includes the following fields and options:

- *Descripción:** Cliente Exonerado
- Descuento:** 0.00%
- Lista de precio:**
 - Precio 1
 - Precio 2
 - Precio 3
- Habilitar Crédito:**
- Es Exonerado:**
- Es CCSS:**
- Descuento permanente:**
- Rango de fechas del descuento:** (Calendar icon)
- *Estado:** Activo

Buttons: Guardar, Cancelar

Copyright © 2024 Easy CIF. All rights reserved. Version 1.1.0

Fuente: Elaboración propia (2024)

Estas pantallas permiten agregar, buscar y modificar la información relacionada con los perfiles de cliente del sistema.

Figura 23.

Diseño de entradas – Mantenimiento de clientes

The screenshot displays the 'Agregar Cliente' form in the ZAPATIT S system. The form is organized into a grid layout with the following fields and sections:

- Perfil Cliente:** Cliente Gravado (dropdown)
- Tipo identificación:** Cédula Física (dropdown)
- Identificación:** Identificación (text input)
- Nombre:** Nombre (text input)
- Correo:** Correo (text input)
- Correo de facturación:** Correo para la facturación electrónica (text input)
- Código de área teléfono:** (0) NO DEFINIDO (dropdown)
- Teléfono:** Teléfono (text input)
- Código de área fax:** (0) NO DEFINIDO (dropdown)
- Fax:** Fax (text input)
- Provincia:** San José (dropdown)
- Cantón:** San José (dropdown)
- Distrito:** Carmen (dropdown)
- Barrio:** San José (parte) (dropdown)
- Dirección (Otras señas):** Otras Señas (text input)
- Límite de crédito:** 0 (text input)
- Plazo:** 0 (text input)
- Unidad ejecutora:** Unidad ejecutora (text input)
- Representante Legal:** Nombre del Representante Legal (text input)
- Contacto:** Información del contacto (text input)
- Contacto Cobro:** Contacto de cobro (text input)
- Estado:** Activo (dropdown)

Buttons for 'Guardar' (Save) and 'Cancelar' (Cancel) are located at the bottom left of the form area.

Fuente: Elaboración propia (2024).

Figura 24.

Diseño de entradas – Mantenimiento de clientes - Modificar

The screenshot displays the 'Modificar Cliente' form in the ZAPATIT S system. The form is pre-filled with the following data:

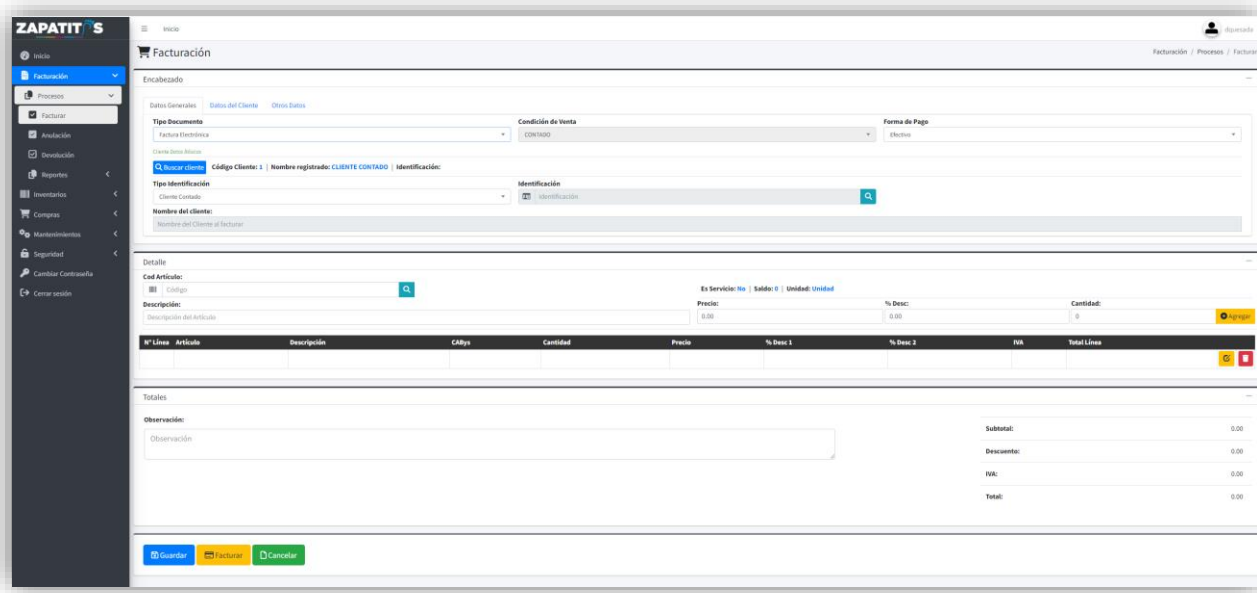
- Perfil Cliente:** Cliente CCSS (dropdown)
- Tipo identificación:** Cédula Jurídica (dropdown)
- Identificación:** 4-000-042147 (text input)
- Nombre:** HOSPITAL BLANCO CERVANTES_2202 (text input)
- Correo:** Correo (text input)
- Correo de facturación:** hospitalbl@outlook.com (text input)
- Código de área teléfono:** (506) COSTA RICA (dropdown)
- Teléfono:** Teléfono (text input)
- Código de área fax:** (506) COSTA RICA (dropdown)
- Fax:** Fax (text input)
- Provincia:** San José (dropdown)
- Cantón:** San José (dropdown)
- Distrito:** Carmen (dropdown)
- Barrio:** San José (parte) (dropdown)
- Dirección (Otras señas):** Dirección editada (text input)
- Límite de crédito:** ₡10,000,000 (text input)
- Plazo:** 30 (text input)
- Unidad ejecutora:** 2202 (text input)
- Representante Legal:** Nombre del Representante Legal (text input)
- Contacto:** Información del contacto (text input)
- Contacto Cobro:** Contacto de cobro (text input)
- Estado:** Activo (dropdown)

Buttons for 'Guardar' (Save) and 'Cancelar' (Cancel) are located at the bottom left of the form area.

Fuente: Elaboración propia (2024)

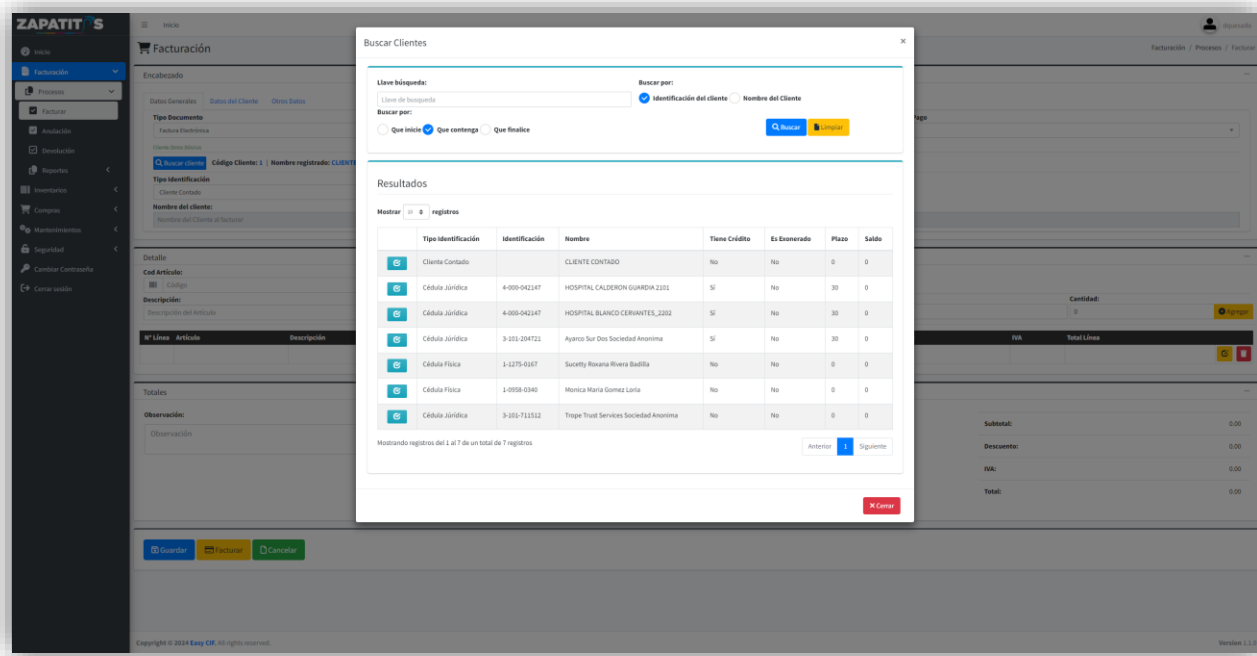
Estas pantallas permiten agregar, buscar y modificar la información relacionada con los clientes del sistema.

Figura 25.
Diseño de entradas – Pantalla de facturación



Fuente: Elaboración propia (2024).

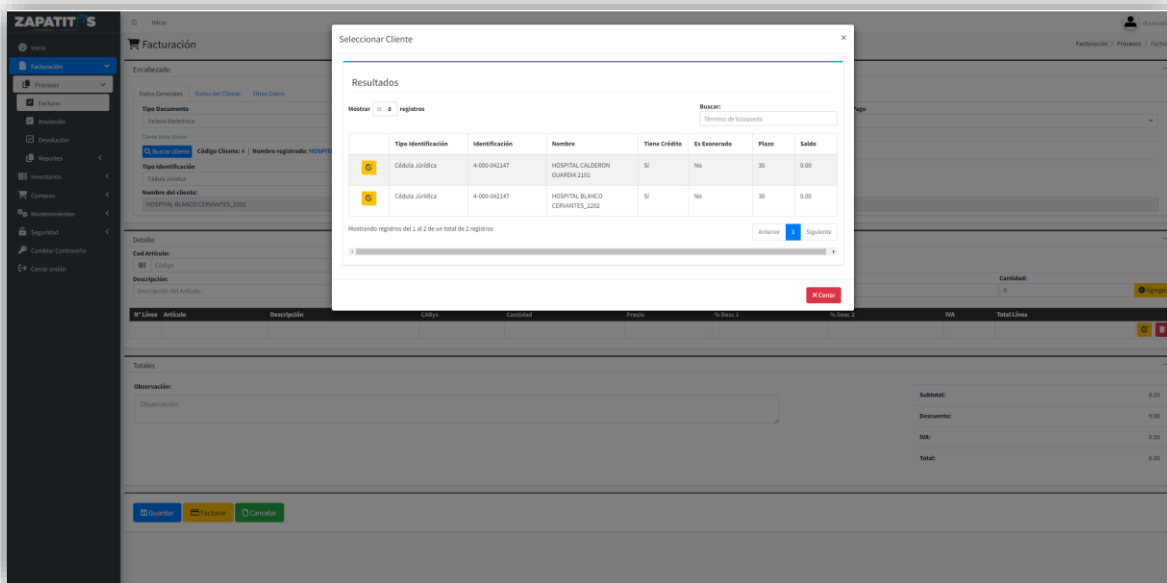
Figura 26.
Diseño de entradas – Pantalla de facturación – Buscar clientes



Fuente: Elaboración propia (2024).

Figura 27.

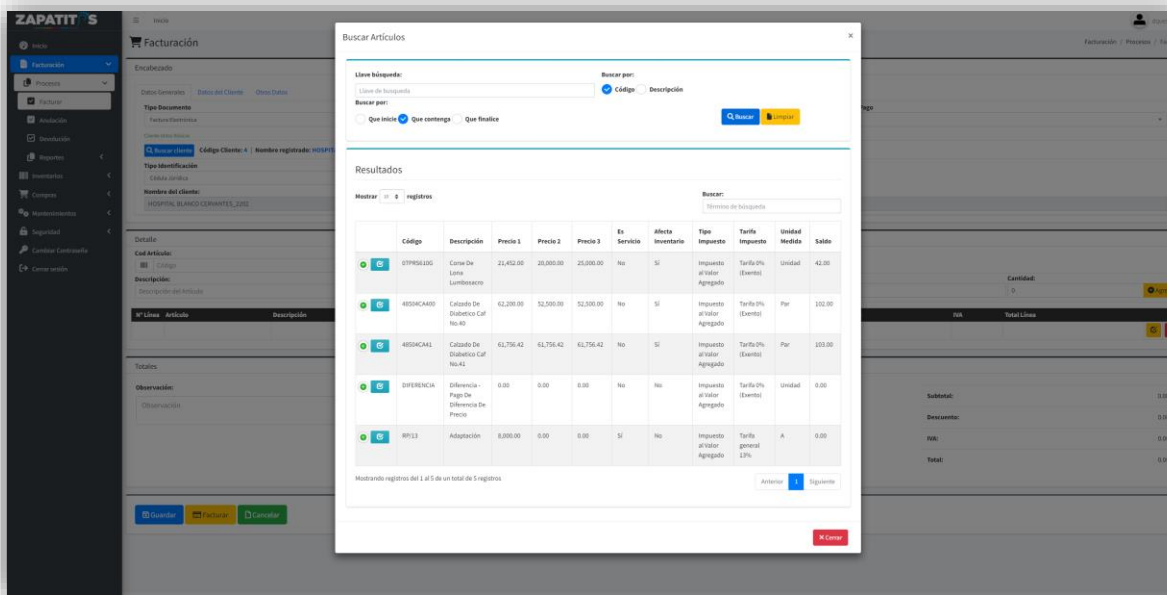
Diseño de entradas – Pantalla de facturación – Seleccionar cliente



Fuente: Elaboración propia (2024).

Figura 28.

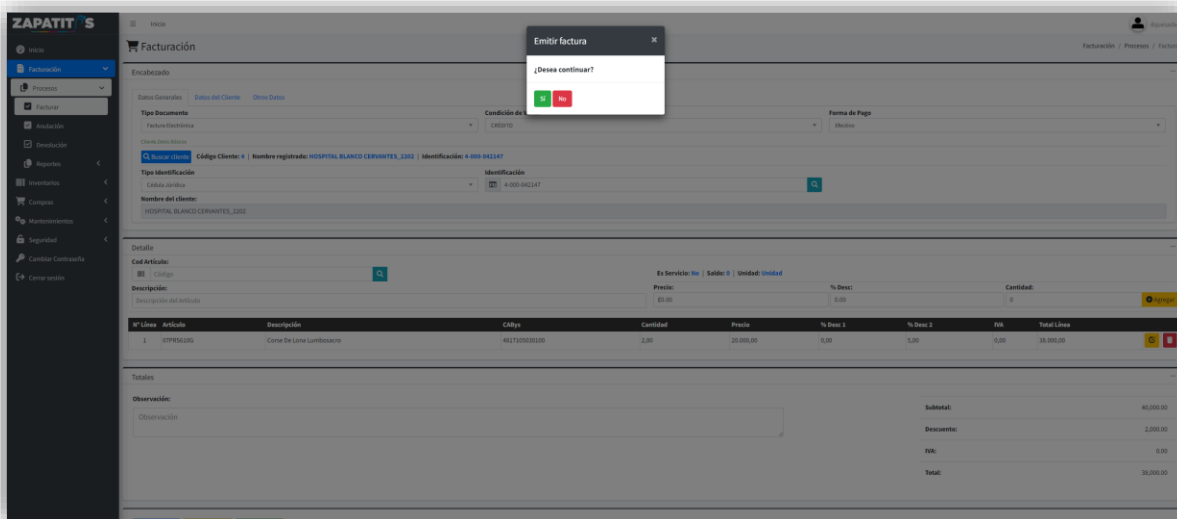
Diseño de entradas – Pantalla de facturación – Buscar articulos



Fuente: Elaboración propia (2024).

Figura 29.

Diseño de entradas – Pantalla de facturación – Emitir factura

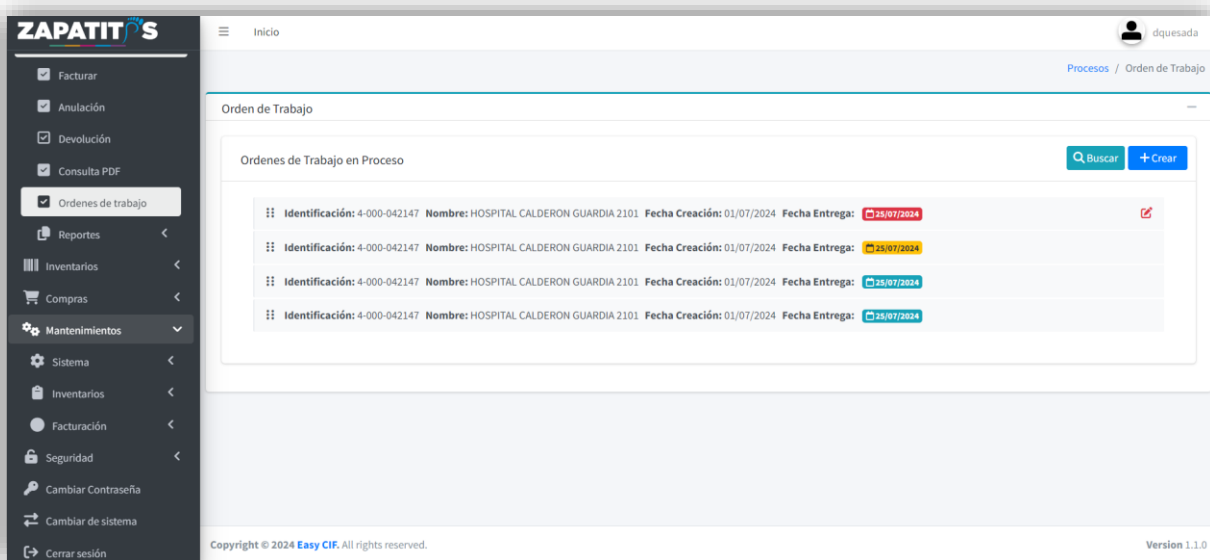


Fuente: Elaboración propia (2024)

Estas pantallas permiten la emisión de documento electrónicos: Tiquete y Factura, se selecciona un cliente, se seleccionan los artículos y se procede a facturar el pedido.

Figura 30.

Diseño de entradas – Pantalla de orden trabajo



Fuente: Elaboración propia (2024).

Figura 31.

Diseño de entradas – Pantalla de orden trabajo – Crear

Fuente: Elaboración propia (2024).

Figura 32.

Diseño de entradas – Pantalla de orden trabajo – Insumos

Fuente: Elaboración propia (2024).

Figura 33.

Diseño de entradas – Pantalla de orden trabajo – Datos plantilla

The screenshot displays the 'Orden de Trabajo' form in the ZAPATIT OS system. The 'Datos Plantilla' tab is active, showing the following fields:

- Arco Longitudinal:** Arco Longitudinal
- Espolón:** Espolón
- Arco Transverso:** Arco Transverso
- Soporte Ant:** Soporte Anterior
- Virón:** Virón
- Otros:** Otros

At the bottom of the form, there are 'Guardar' and 'Cancelar' buttons. The left sidebar shows the navigation menu with 'Ordenes de trabajo' selected.

Fuente: Elaboración propia (2024).

Figura 34.

Diseño de entradas – Pantalla de orden trabajo – Datos suela

The screenshot displays the 'Orden de Trabajo' form in the ZAPATIT OS system. The 'Datos Suela' tab is active, showing the following fields:

- Thomas:** Thomas
- Virón:** Virón
- Tacón corriente:** Tacón Corriente
- Otros:** Otros

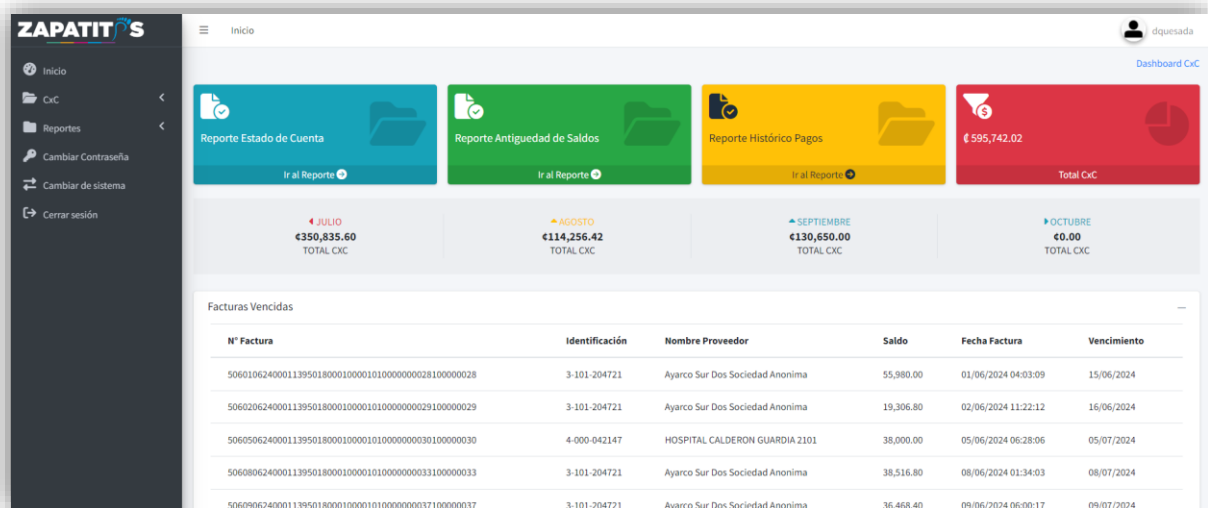
At the bottom of the form, there are 'Guardar' and 'Cancelar' buttons. The left sidebar shows the navigation menu with 'Ordenes de trabajo' selected.

Fuente: Elaboración propia (2024).

Estas pantallas permiten la emisión la creación y control del proceso de órdenes de trabajo, la primera pantalla a visualizar es una lista de las ordenes en procesos la cual se puede priorizar arrastrando el ítem a la posición deseada.

Figura 35.

Diseño de entradas – Dashboard CxC



Fuente: Elaboración propia (2024).

Esta pantalla es un Dashboard informativo de la situación de la cuentas cobrar de la empresa, con un acceso directo a los reportes.

Figura 36.

Diseño de entradas – Registro de notas de crédito electrónicas y financieras CxC

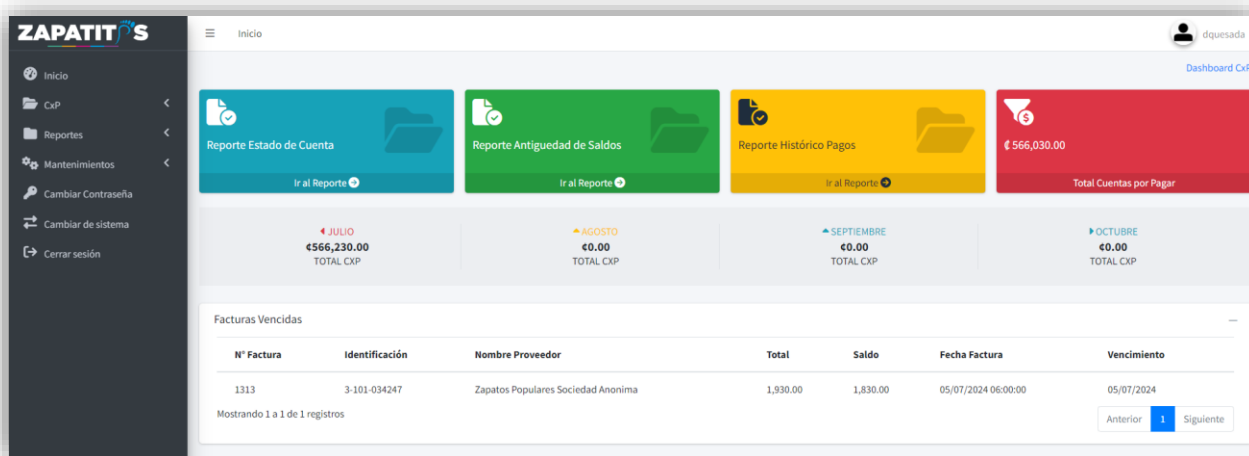
The screenshot shows the 'Registro Documentos' form in the ZAPATIT S system. The form is titled 'Registro de Documentos NC (+) /ND (-)'. It includes fields for 'Cliente', 'Identificación', 'Es Exonerado', 'Tiene Crédito', 'Doc Electrónico', and 'Plazo'. There are also sections for 'Tipo de Documento', 'Código Referencia', 'Nº Factura', 'Saldo Factura', 'Observación', 'Correo', and 'Tarifa impuesto'. At the bottom, there are summary fields for 'Monto Exento', 'Monto Gravado', 'Subtotal', 'Descuento', 'Impuesto', and 'Total', along with 'Guardar' and 'Cancelar' buttons.

Fuente: Elaboración propia (2024).

En esta pantalla se pueden crear la notas de crédito o débito que afectan los documentos electrónicos y que no tienen relación con un artículo específico de la factura, por ejemplo, crear una nota de crédito financiera para liquidar un saldo de retención de renta del dos por ciento.

Figura 37.

Diseño de entradas – Dashboard CxP



Fuente: Elaboración propia (2024).

Esta pantalla es un Dashboard informativo de la situación de las cuentas de la empresa, con un acceso directo a los reportes.

Figura 38.

Diseño de entradas – Registro de facturas cuentas por pagar

The screenshot shows the 'Registro de Facturas' form in the ZAPATIT'S system. The form includes the following fields and sections:

- Proveedor:** Search field for the provider name.
- Fecha de Factura:** Date field set to 09/07/2024.
- Tipo Compra:** Dropdown menu set to 'Factura'.
- N° Factura:** Input field for the invoice number.
- Cuenta Contable:** Dropdown menu set to 'Inventario en Tránsito (Colones)'. Below it is a 'Tarifa impuesto' dropdown set to 'Tarifa 0% (Exento)'.
- Plazo:** Input field for the payment term.
- Observación:** Text area for notes.
- Summary Section:**
 - Monto Exento:** Input field.
 - Monto Gravado:** Input field.
 - Subtotal:** Input field showing 0.
 - Descuento:** Input field showing 'Monto del Descuento'.
 - Impuesto:** Input field showing 0.
 - Total:** Input field showing 0.

At the bottom of the form are 'Guardar' and 'Cancelar' buttons.

Fuente: Elaboración propia (2024).

Esta pantalla permite el registro de facturas que no están directamente relacionadas con la compra de productos de inventario, se utiliza por ejemplo para ingresar una factura de gasto como puede ser el mantenimiento del sistema informático.

Figura 39.

Diseño de entradas – Registro de pagos Facturas CxP

Fuente: Elaboración propia (2024).

Esta pantalla permite el registro de abonos o pagos a facturas de crédito registradas desde el módulo de compras o cuentas por pagar.

Figura 40.

Diseño de entradas – Registro de asiento contables

Fuente: Elaboración propia (2024).

Esta pantalla permite el registro de asientos contables manuales al sistema.

Diccionario De Datos.

Tabla 39.

Diccionario de datos, tabla Rol

Nombre de la Tabla		Rol			
Descripción		Tabla para almacenar los roles de los sistema.			
Columna	Tipo Dato	Null	Pk	Fk	Descripción
IdRol	smallint		X		Identificador único de la tabla Rol
Nombre	varchar(5)				Nombre del Rol
Descripcion	varchar(50)				Descripción del Rol

Fuente: Elaboración propia (2024).

Tabla 40.

Diccionario de datos, tabla Usuario

Nombre de la Tabla		Usuario			
Descripción		Tabla para almacenar los usuarios			
Columna	Tipo Dato	Null	Pk	Fk	Descripción
IdUsuario	int		X		Identificador único de la tabla usuario
IdCompania	int			X	Identificador de la compañía - Fk Tabla Compania
IdRol	smallint			X	Identificador del rol - Fk Tabla Rol
NombreUsuario	varchar(50)				Nombre de usuario para ingresar al sistema
Nombre	nvarchar(50)				Nombre de la persona a la cual pertenece el usuario
Apellido	varchar(160)	X			Apellidos de la persona a la cual pertenece el usuario
Correo	varchar(100)	X			Correo electrónico del usuario
Contrasena	nvarchar(255)				Contraseña del usuario
FecRegistro	datetime				Fecha en la que se creó el usuario
FecUltActualizacion	datetime				Última fecha de actualización del usuario
Estado	smallint				Estado: 1- Inactivo, 2-Activo, 3-Bloqueado
IntentosFallidos	smallint				Cantidad de Intentos Fallidos
FecUltIntentoFallido	datetime	X			Fecha del último intento fallido de ingreso
DebeCambiarContrasena	bit				Indica si el usuario debe cambiar la contraseña

Fuente: Elaboración propia (2024).

Tabla 41.

Diccionario de datos, tabla Modulo

Nombre de la Tabla		Modulo			
Descripción		Tabla para almacenar los módulos del sistema			
Columna	Tipo Dato	Null	Pk	Fk	Descripción
CodModulo	int		X		Código del módulo
Descripcion	varchar(20)				Descripción o nombre del módulo
Estado	smallint				Estado: 1-Inactivo, 2-Activo

Fuente: Elaboración propia (2024).

Tabla 42.*Diccionario de datos, tabla Usuario_Modulo*

Nombre de la Tabla		Usuario_Modulo			
Descripción		Tabla para almacenar la relación usuarios-módulos			
Columna	Tipo Dato	Null	Pk	Fk	Descripción
IdUsuario	int		X	X	Identificador del usuario - Fk tabla usuario
CodModulo	int		X	X	Código del módulo - Fk tabla modulo

Fuente: Elaboración propia (2024).**Tabla 43.***Diccionario de datos, tabla TipoIdentificacion*

Nombre de la Tabla		TipoIdentificacion			
Descripción		Tabla para almacenar los tipos de identificación.			
Columna	Tipo Dato	Null	Pk	Fk	Descripción
CodTipoIdentificacion	varchar(2)		X		Código del Tipo de Identificación
Descripcion	varchar(50)				Descripción del tipo de identificación
Estado	smallint				Estado: 1-Inactivo, 2-Activo
TamanoMinimo	smallint				Cantidad mínima de caracteres que debe tener la identificación según el tipo
TamanoMaximo	smallint				Cantidad máxima de caracteres que debe tener la identificación según el tipo
Formato	nvarchar(50)	X			Formato de la identificación
ExpresionRegular	nvarchar(50)	X			Expresión regular para validar el formato de la identificaciones
Nota	varchar(150)	X			Nota de referencia a la documentación de factura electrónica DGT

Fuente: Elaboración propia (2024).**Tabla 44.***Diccionario de datos, tabla Provincia*

Nombre de la Tabla		Provincia			
Descripción		Tabla para almacenar los códigos de Provincia			
Columna	Tipo Dato	Null	Pk	Fk	Descripción
CodProvincia	smallint		X		Código de provincia
Descripcion	varchar(20)				Descripción o nombre de la provincia

Fuente: Elaboración propia (2024).**Tabla 45.***Diccionario de datos, tabla Canton*

Nombre de la Tabla		Canton			
Descripción		Tabla para almacenar los códigos de cantón			
Columna	Tipo Dato	Null	Pk	Fk	Descripción
CodProvincia	smallint		X	X	Código de provincia - Fk tabla Provincia
CodCanton	smallint		X		Código de cantón
Descripcion	varchar(50)				Descripción o nombre del cantón

Fuente: Elaboración propia (2024).

Tabla 46.

Diccionario de datos, tabla Distrito

Nombre de la Tabla		Distrito				
Descripción		Tabla para almacenar los códigos de Distrito				
Columna	Tipo Dato	Null	Pk	Fk	Descripción	
CodProvincia	smallint		X	X	Código de provincia - Fk tabla Provincia	
CodCanton	smallint		X	X	Código de cantón - Fk tabla Canton	
CodDistrito	smallint		X		Código de distrito - Fk tabla Distrito	
Descripcion	varchar(100)				Descripción o nombre del distrito	

Fuente: Elaboración propia (2024).

Tabla 47.

Diccionario de datos, tabla Barrio

Nombre de la Tabla		Barrio				
Descripción		Tabla para almacenar los barrios				
Columna	Tipo Dato	Null	Pk	Fk	Descripción	
CodProvincia	smallint		X	X	Código de provincia - Fk tabla Provincia	
CodCanton	smallint		X	X	Código de cantón - Fk tabla Canton	
CodDistrito	smallint		X	X	Código de distrito - Fk tabla Distrito	
CodBarrio	smallint		X		Código de barrio	
Descripcion	varchar(100)				Descripción o nombre del barrio	

Fuente: Elaboración propia (2024).

Tabla 48.

Diccionario de datos, tabla CodigoArea

Nombre de la Tabla		CodigoArea				
Descripción		Tabla para almacenar los códigos de áreas de los países				
Columna	Tipo Dato	Null	Pk	Fk	Descripción	
CodArea	smallint		X		Código de área del país	
Descripcion	nvarchar(50)				Descripción o nombre del código de área o país	

Fuente: Elaboración propia (2024).

Tabla 49.

Diccionario de datos, tabla Compania

Nombre de la Tabla		Compania				
Descripción		Tabla para almacenar las compañías (Emisores - Facturación Electrónica)				
Columna	Tipo Dato	Null	Pk	Fk	Descripción	
IdCompania	int		X		Identificador único de la compañía	
CodTipoIdentificacion	varchar(2)			X	Código tipo de identificación - Fk tabla TipoIdentificacion	
Identificacion	varchar(12)				Identificación de la compañía	
Nombre	nvarchar(100)				Nombre de la compañía	
NombreComercial	nvarchar(80)				Nombre comercial o fantasía de la compañía	
CodAreaTelefono	smallint			X	Código de área (País) del teléfono - Fk tabla CodigoArea	

Nombre de la Tabla		Compania			
Descripción		Tabla para almacenar las compañías (Emisores - Facturación Electrónica)			
Columna	Tipo Dato	Null	Pk	Fk	Descripción
Telefono	nvarchar(20)				Número de teléfono del emisor
CodAreaFax	smallint			X	Código de área (País) del Fax - Fk tabla CodigoArea
Fax	nvarchar(20)				Número de fax
Correo	nvarchar(160)				Dirección de correo electrónico del Emisor de los documentos electrónicos.
CodProvincia	smallint			X	Código de provincia - Fk tabla Provincia
CodCanton	smallint			X	Código de cantón - Fk tabla Canton
CodDistrito	smallint			X	Código de distrito - Fk tabla Distrito
CodBarrio	smallint				Código de barrio - Fk tabla barrio
Direccion	nvarchar(250)	X			Dirección de la compañía
RutaCertificado	nvarchar(Max)				Ruta del certificado para firmar electrónicamente
NombreCertificado	nvarchar(255)				Nombre del certificado de firma electrónica
ContrasenaCertificado	nvarchar(255)				Contraseña del certificado
FecVencimientoCertificado	datetime	X			Fecha de vencimiento del certificado
UsuarioHacienda	nvarchar(255)				Usuario del api de Hacienda
ContrasenaHacienda	nvarchar(Max)				Contraseña del api Hacienda
Estado	smallint				Estado: 1-Inactiva, 2- Activa

Fuente: Elaboración propia (2024).

Tabla 50.

Diccionario de datos, tabla Sucursal

Nombre de la Tabla		Sucursal			
Descripción		Tabla para almacenar las sucursales			
Columna	Tipo Dato	Null	Pk	Fk	Descripción
CodSucursal	int		X		Código de sucursal por compañía
IdCompania	int			X	Identificador de la compañía - Fk tabla Compania
Nombre	nvarchar(150)				Descripción o nombre de la sucursal
CodAreaTelefono1	smallint			X	Código de área (país) Teléfono 1 – Fk tabla CodigoArea
Telefono1	nvarchar(20)	X			Número de Teléfono 1
CodAreaTelefono2	smallint			X	Código de área (país) Teléfono 2 – Fk tabla CodigoArea
Telefono2	nvarchar(20)	X			Número de Teléfono 2
Direccion	nvarchar(250)	X			Dirección de la Sucursal
Correo	nvarchar(160)	X			Correo electrónico de la Sucursal
EncaFactura	nvarchar(255)	X			Leyenda Encabezado de factura
PieFactura	nvarchar(255)	X			Leyenda pie de factura
Estado	smallint				Estado: 1-Inactivo, 2-Activo

Fuente: Elaboración propia (2024).

Tabla 51.*Diccionario de datos, tabla Terminal*

Nombre de la Tabla		Terminal			
Descripción		Tabla para almacenar la terminales de las diferentes sucursales del sistema, requisito para la emisión de comprobantes electrónicos.			
Columna	Tipo Dato	Null	Pk	Fk	Descripción
IdCompania	int		X	X	Identificador de la compañía - Fk tabla compania
CodSucursal	int		X	X	Código de la sucursal - Fk tabla Sucursal
CodTerminal	int		X		Código de la terminal
Descripcion	varchar(50)				Descripción de la terminal

Fuente: Elaboración propia (2024).**Tabla 52.***Diccionario de datos, tabla TipoDocumento*

Nombre de la Tabla		TipoDocumento			
Descripción		Tabla para almacenar los tipos de documentos del sistema			
Columna	Tipo Dato	Null	Pk	Fk	Descripción
CodTipoDocumentoElectronico	nvarchar(2)		X		Código tipo de documento
Descripcion	varchar(255)				Descripción del tipo de documento
Estado	smallint				Estado: 1-Inactiva, 2- Activa
Etiqueta	varchar(255)				Etiqueta: Nombre del primer elemento del XML correspondiente tipo de documento cuando es electrónico
xmlns	varchar(255)				Atributo xmlns del primer elemento del XML correspondiente tipo de documento cuando es electrónico
xsi_schemaLocation	varchar(255)				Atributo xsi_schemaLocation del primer elemento del XML correspondiente tipo de documento cuando es electrónico
PolicyURL	varchar(255)				Url del documento pdf de la política que se requiere al realizar firma XAdES-EPES cuando es electrónico
VerEnFactura	bit				Identificador para conocer cuales tipos de documentos se deben cargar en la pantalla de facturación
EsElectronico	bit				True: Indica si el documento es electrónico

Fuente: Elaboración propia (2024).**Tabla 53.***Diccionario de datos, tabla ConsecutivoTipoDocumento*

Nombre de la Tabla		ConsecutivoTipoDocumento			
Descripción		Tabla para almacenar los consecutivos de los diferentes tipos de documentos electrónicos.			
Columna	Tipo Dato	Null	Pk	Fk	Descripción
IdCompania	int		X	X	Identificador de la compañía - Fk tabla Compania

CodSucursal	int		X	X	Código de la sucursal - Fk tabla Sucursal
CodTerminal	int		X	X	Código de la terminal - Fk tabla Terminal
CodTipoDocumento	varchar(2)		X	X	Código tipo de documento - Fk TipoDocumento
Consecutivo	float				Consecutivo por tipo de documento

Fuente: Elaboración propia (2024)

Tabla 54.

Diccionario de datos, tabla ConsecutivoInterno

Nombre de la Tabla		ConsecutivoInterno				
Descripción		Tabla para almacenar los consecutivos internos del sistema				
Columna	Tipo Dato	Null	Pk	Fk	Descripción	
IdCompania	int		X	X	Identificador de la compañía - Fk tabla Compania	
CodSucursal	int		X	X	Código de la sucursal - Fk tabla Sucursal	
CodConsecutivoInterno	varchar(15)		X		Código del Consecutivo interno	
Consecutivo	float				Consecutivo	

Fuente: Elaboración propia (2024).

Tabla 55.

Diccionario de datos, tabla ActividadComercial

Nombre de la Tabla		ActividadComercial				
Descripción		Tabla para almacenar los códigos de las actividades económicas del emisor relacionadas con los comprobantes electrónicos				
Columna	Tipo Dato	Null	Pk	Fk	Descripción	
IdCompania	int		X	X	Identificador de compañía – Fk Compania	
CodActividad	nvarchar(6)		X		Código de la actividad económica	
Descripcion	nvarchar(255)				Descripción o nombre de la actividad económica	
EsActividadPrincipal	bit				Si la compañía realiza varias actividades debe identificar cual es la actividad principal	
Estado	smallint				Estado: 1-Inactiva 2-Activa	

Fuente: Elaboración propia (2024).

Tabla 56.

Diccionario de datos, tabla Moneda

Nombre de la Tabla		Moneda				
Descripción		Tabla para almacenar las monedas				
Columna	Tipo Dato	Null	Pk	Fk	Descripción	
CodMoneda	varchar(2)		X		Código de la moneda, identificador de la moneda en el XML para la factura electrónica	
Descripcion	varchar(50)				Nombre por el cual se conoce a la moneda	
Estado	smallint				Estado: 1-Inactivo, 2- Activo	
NombrePais	varchar(50)				Nombre del país de origen de la moneda	
CodigoISO	varchar(4)				Código ISO de la Moneda	

Fuente: Elaboración propia (2024).

Tabla 57.*Diccionario de datos, tabla Categoria*

Nombre de la Tabla		Categoria			
Descripción		Tabla para almacenar las Categorías con las que se van a clasificar los artículos del inventario.			
Columna	Tipo Dato	Null	Pk	Fk	Descripción
IdCategoria	int		X		Identificador único de la categoría
IdCompania	int			X	Identificador de la compañía - Fk tabla Compania
Descripcion	varchar(150)				Descripción de la categoría
Estado	smallint				Estado: 1-Inactivo, 2- Activo

Fuente: Elaboración propia (2024).**Tabla 58.***Diccionario de datos, tabla Bodega*

Nombre de la Tabla		Bodega			
Descripción		Tabla para almacenar las bodegas			
Columna	Tipo Dato	Null	Pk	Fk	Descripción
IdBodega	int		X		Identificador único de la bodega
IdCompania	int			X	Identificador de la compañía - Fk tabla Compania
CodSucursal	int			X	Código de la sucursal - Fk tabla Sucursal
Descripcion	varchar(100)				Descripción de la bodega
Factura	bit				True: Permite facturar, False: solamente aprovisiona inventario
BajoSaldo	bit				True: Permite facturar bajo saldo False: No lo permite
BajoCosto	bit				True: Permite facturar bajo costo, False: No lo permite
EsPrincipal	bit				True: Indica si es la bodega principal la cual se cargará por defecto en la pantalla de facturación
Estado	smallint				Estado: 1-Inactivo, 2- Activo

Fuente: Elaboración propia (2024).**Tabla 59.***Diccionario de datos, tabla UnidadMedida*

Nombre de la Tabla		UnidadMedida			
Descripción		Tabla para almacenar las unidades de medida con la que se clasifican los artículos			
Columna	Tipo Dato	Null	Pk	Fk	Descripción
IdUnidadMedida	int		X		Identificador de la unidad de medida
Simbolo	nvarchar(15)				Símbolo con el que se identifica la unidad de medida en los documentos electrónicos
Descripcion	nvarchar(50)				Descripción de la unidad de medida
Orden	smallint				Orden en el que se visualizará
SeDigitaDescripcion	bit				True: Se obliga a que se digite el nombre de la Unidad de medida comercial
Estado	smallint				Estado: 1-Inactivo, 2- Activo

Fuente: Elaboración propia (2024).

Tabla 60.*Diccionario de datos, tabla TipoCodigoArticulo*

Nombre de la Tabla		TipoCodigoArticulo			
Descripción		Tabla para almacenar el tipo de código del artículo o servicio			
Columna	Tipo Dato	Null	Pk	Fk	Descripción
CodTipoCodigoArticulo	varchar(2)		X		Código del tipo de artículo o servicio
Descripcion	varchar(50)				Descripción del tipo de código
Estado	smallint				Estado: 1-Inactivo, 2- Activo

Fuente: Elaboración propia (2024).

Tabla 61.*Diccionario de datos, tabla TipoArticulo*

Nombre de la Tabla		TipoArticulo			
Descripción		Tabla para almacenar los tipos de artículos (primarios, compuestos)			
Columna	Tipo Dato	Null	Pk	Fk	Descripción
CodTipoArticulo	smallint		X		Código del tipo de artículo
Descripcion	varchar(50)				Descripción el tipo de artículo
Estado	smallint				Estado: 1-Inactivo, 2- Activo

Fuente: Elaboración propia (2024)

Tabla 62.*Diccionario de datos, tabla CondicionImpuesto*

Nombre de la Tabla		CondicionImpuesto			
Descripción		Tabla para almacenar las condiciones de impuesto			
Columna	Tipo Dato	Null	Pk	Fk	Descripción
CodCondicionImpuesto	varchar(2)		X		Código de la condición del impuesto
Descripcion	varchar(50)				Descripción del código de la condición del impuesto
Estado	smallint				Estado: 1-Inactivo, 2- Activo

Fuente: Elaboración propia (2024).

Tabla 63.*Diccionario de datos, tabla TipoImpuesto*

Nombre de la Tabla		TipoImpuesto			
Descripción		Tabla para almacenar los tipos de impuesto (impuesto valor agregado, impuesto selectivo de consumo, etc.)			
Columna	Tipo Dato	Null	Pk	Fk	Descripción
CodTipoImpuesto	varchar(2)		X		Código tipo de impuesto
Descripcion	varchar(50)				Descripción del tipo de impuesto
UsaFactorIVA	bit				Utiliza factor Iva
Estado	smallint				Estado: 1-Inactivo, 2- Activo

Fuente: Elaboración propia (2024).

Tabla 64.*Diccionario de datos, tabla TarifaImpuesto*

Nombre de la Tabla		TarifaImpuesto				
Descripción		Tabla para almacenar las tarifas de impuestos de valor agregado (Iva)				
Columna	Tipo Dato	Null	Pk	Fk	Descripción	
CodTarifaImpuesto	varchar(2)		X		Código para la tarifa de impuesta Iva	
Descripcion	varchar(50)				Descripción del código para la tarifa de impuesto Iva	
Porcentaje	float				Porcentaje de la Tarifa de Impuesto	
Estado	smallint				Estado: 1-Inactivo, 2- Activo	

Fuente: Elaboración propia (2024).**Tabla 65.***Diccionario de datos, tabla TipoDocumentoExoneracion*

Nombre de la Tabla		TipoDocumentoExoneracion				
Descripción		Tabla para almacenar los tipos de documentos de exoneración				
Columna	Tipo Dato	Null	Pk	Fk	Descripción	
CodTipoDocumentoExoneracion	varchar(2)		X		Código del tipo de exoneración	
Descripcion	varchar(50)				Descripción del código del tipo de exoneración	
Estado	smallint				Estado: 1-Inactivo, 2- Activo	

Fuente: Elaboración propia (2024).**Tabla 66.***Diccionario de datos, tabla TipoReferencia*

Nombre de la Tabla		TipoReferencia				
Descripción		Tabla para almacenar los códigos de referencia				
Columna	Tipo Dato	Null	Pk	Fk	Descripción	
CodTipoReferencia	varchar(2)		X		Código tipo de referencia	
Descripcion	varchar(50)				Descripción del código de tipo de referencia	
Estado	smallint				Estado: 1-Inactivo, 2- Activo	

Fuente: Elaboración propia (2024).**Tabla 67.***Diccionario de datos, tabla TipoDocumetoReferencia*

Nombre de la Tabla		TipoDocumentoReferencia				
Descripción		Tabla para almacenar los códigos de los diferentes tipos de documento de referencia de los documentos electrónicos				
Columna	Tipo Dato	Null	Pk	Fk	Descripción	
CodTipoDocumentoReferencia	varchar(2)		X		Código del tipo de documento de referencia	
Descripcion	varchar(150)				Descripción del código de tipo de referencia	
Estado	smallint				Estado: 1-Inactivo, 2- Activo	

Fuente: Elaboración propia (2024).

Tabla 68.*Diccionario de datos, tabla Articulo*

Nombre de la Tabla		Articulo			
Descripción		Tabla para almacenar los artículos (bienes/servicios)			
Columna	Tipo Dato	Null	Pk	Fk	Descripción
IdCompania	int		X	X	Identificador de la compañía Fk- Compania
CodArticulo	nvarchar(20)		X		Código del bien o servicio
CodTipoCodigoArticulo	varchar(2)			X	Tipo de código del artículo, clasificación para los documentos electrónicos - Fk tabla TipoCodigoArticulo
CodTipoArticulo	smallint			X	Código tipo de artículo (indica si es un artículo es normal, primario o compuesto) - Fk tabla TipoArticulo
Cabys	varchar(13)				Código CABys del artículo
Estado	smallint				Estado: 1-Inactivo, 2- Activo, 3-Descontinuado
Descripcion	nvarchar(200)				Descripción del artículo, solo permite 200 caracteres por requerimiento de la facturación electrónica
IdProveedor	int			X	Identificador del proveedor - FK tabla Proveedor
IdCategoria	int			X	Identificador de la categoría - FK tabla Categoria
CantMinima	int				Cantidad mínima en stock
CantMaxima	int				Cantidad máxima en stock
Margen1	bit				Margen de utilidad precio 1
Margen2	float				Margen de utilidad precio 2
Margen3	float				Margen de utilidad precio 3
Precio1	decimal(18,5)				Precio 1
Precio2	decimal(18,5)				Precio 2
Precio3	decimal(18,5)				Precio 3
Costo	decimal(18,5)				Costo promedio ponderado del artículo
EsServicio	bit				Indicador si el artículo es un servicio
AfectaInventario	bit				Indicador para el control de existencias
PermiteCambiarPrecio	bit				Indicador para ver si el artículo permite cambio de precio
CodTipoImpuesto	varchar(2)			X	Código del tipo de impuesto valor agregado – Fk - tabla TipoImpuesto
CodTarifaImpuesto	varchar(2)			X	Código de la tarifa de impuesto de valor agregado Fk - tabla TarifaImpuesto
IdUnidadMedida	int			X	Identificador de la unidad de medida - Fk tabla UnidadMedida
SeVende	Bit				Indicador, si el artículo está disponible para la venta
UnidadMedidaComercial	varchar(20)				Si el identificador de la unidad de medida es 94 se habilita el campo para colocar la descripción de la unidad de medida comercial

Fuente: Elaboración propia (2024).

Tabla 69.*Diccionario de datos, tabla Proveedor*

Nombre de la Tabla		Proveedor			
Descripción		Tabla para almacenar los proveedores			
Columna	Tipo Dato	Null	Pk	Fk	Descripción
IdProveedor	int		X		Identificador único del proveedor
IdCompania	int			X	Identificador de la compañía - FK Tabla Compania
CodTipoIdentificacion	varchar(2)			X	Código del Tipo de identificación - FK Tabla TipoIdentificacion
Identificacion	varchar(20)				Identificación del proveedor
Nombre	varchar(180)				Nombre del proveedor
Direccion	nvarchar(255)				Dirección del proveedor
Contacto	nvarchar(255)				Contacto del proveedor
CodAreaTelefono	smallint			X	Código de área del teléfono - FK Tabla CodigoArea
Telefono	nvarchar(20)				Teléfono del proveedor
Credito	bit				True: Es un proveedor de Crédito False: Es un proveedor de Contacto
LimiteCredito	decimal(18,5)				Límite de crédito del proveedor
Saldo	decimal(18,5)				Saldo Global de crédito Pendiente de pagar
Plazo	smallint				Plazo si el proveedor es de crédito
Correo	nvarchar(160)				Correo del proveedor
IdCuentaContable	int				Identificador de la cuenta contable
Otros	Text				Campo para colocar cualquier otro dato importante
Estado	smallint				Estado: 1-Inactivo, 2- Activo

Fuente: Elaboración propia (2024).

Tabla 70.*Diccionario de datos, tabla CondicionVenta*

Nombre de la Tabla		CondicionVenta			
Descripción		Tabla para almacenar la condición de venta de los documentos			
Columna	Tipo Dato	Null	Pk	Fk	Descripción
CodCondicionVenta	varchar(2)		X		Código de la condición de venta del documento
Descripcion	varchar(50)				Descripción de la condición de venta
Estado	smallint				Estado: 1-Inactivo, 2- Activo

Fuente: Elaboración propia (2024).

Tabla 71.*Diccionario de datos, tabla EstadoHacienda*

Nombre de la Tabla		EstadoHacienda			
Descripción		Tabla para almacenar los código de mensajes (Estados) de la respuestas de Hacienda			
Columna	Tipo Dato	Null	Pk	Fk	Descripción
CodEstadoHacienda	smallint		X		Código de mensaje de respuesta de Hacienda
Descripcion	varchar(50)				Descripción del código de mensaje de respuesta de Hacienda
Estado	smallint				Estado: 1-Inactivo, 2- Activo

Fuente: Elaboración propia (2024).

Tabla 72.*Diccionario de datos, tabla MedioPago*

Nombre de la Tabla		MedioPago				
Descripción		Tabla para almacenar los Medios de pago				
Columna	Tipo Dato	Null	Pk	Fk	Descripción	
CodMedioPago	varchar(2)		X		Código del Medio de pago	
Descripcion	varchar(50)				Descripción del Medio de pago	
Estado	smallint				Estado: 1-Inactivo, 2- Activo	

Fuente: Elaboración propia (2024).**Tabla 73.***Diccionario de datos, tabla Token*

Nombre de la Tabla		Token				
Descripción		Tabla para almacenar los token para la conexión con el Ministerio de Hacienda				
Columna	Tipo Dato	Null	Pk	Fk	Descripción	
IdCompania	int		X		Identificador de la compañía	
ValorToken	Text				Token de autenticación	
FecUltimoToken	datetime				Fecha hora de último token para controlar la solicitud y renovación	

Fuente: Elaboración propia (2024).**Tabla 74.***Diccionario de datos, tabla TipoDocumentoOtrosCargos*

Nombre de la Tabla		TipoDocumentoOtrosCargos				
Descripción		Tabla para almacenar los tipos de documento del tag Otros Cargos en los documentos electrónicos. Ejemplos (contribución parafiscal, timbre de la cruz Roja)				
Columna	Tipo Dato	Null	Pk	Fk	Descripción	
CodTipoDocumentoOtrosCargos	varchar(2)		X		Código del tipo de documentos del tag otros cargos	
Descripcion	varchar(150)				Descripción del código tipo de documento	
Estado	smallint				Estado: 1-Inactivo, 2- Activo	

Fuente: Elaboración propia (2024).**Tabla 75.***Diccionario de datos, tabla Artículo_Bodega*

Nombre de la Tabla		Articulo_Bodega				
Descripción		Tabla para almacenar los artículos por bodega y llevar el control de existencias				
Columna	Tipo Dato	Null	Pk	Fk	Descripción	
IdCompania	int		X	X	Identificador de la compañía - Fk tabla Compania	
CodSucursal	int		X	X	Código de la sucursal - Fk tabla Sucursal	
IdBodega	int		X	X	Identificador de la bodega - Fk tabla Bodega	
CodArticulo	nvarchar(20)		X	X	Código del artículo - Fk tabla Articulo	
Saldo	float				Saldo del artículo por bodega	
Comprometido	float				Saldo comprometido del artículo	
FecHoraUltVenta	datetime	X			Fecha hora de última venta	

Nombre de la Tabla		Articulo_Bodega			
Descripción		Tabla para almacenar los artículos por bodega y llevar el control de existencias			
Columna	Tipo Dato	Null	Pk	Fk	Descripción
FecHoraUltCompra	datetime	X			Fecha hora de última compra
FecHoraUltDevolucion	datetime	X			Fecha hora de última devolución

Fuente: Elaboración propia (2024).

Tabla 76.

Diccionario de datos, tabla KardexArticulo

Nombre de la Tabla		KardexArticulo			
Descripción		Tabla para almacenar el Kardex Global de los artículos para el control de costos y evidenciar las diferentes transacciones del sistema			
Columna	Tipo Dato	Null	Pk	Fk	Descripción
IdKardexArticulo	bigint		X		Identificador del registro del Kardex
IdCompania	int			X	Identificador de la compañía - Fk Compañía
CodSucursal	int			X	Código de la sucursal - Fk tabla Sucursal
CodArticulo	nvarchar(20)			X	Código del artículo - Fk tabla Articulo
CantEntra	float				Cantidad que ingresa
CantSale	float				Cantidad que sale
Saldo	float				Saldo del artículo
MontoEntra	decimal(18,5)				Costo Total de entrada
MontoSale	decimal(18,5)				Costo Total de Salida
CostoUniNuevo	decimal(18,5)				Costo Unitario Nuevo
CostoUniAnterior	decimal(18,5)				Costo Unitario Anterior
FecHora	datetime				Fecha y hora de la transacción
TipoMovimiento	varchar(20)				Tipo de Movimiento E=Entrada S= Salida

Fuente: Elaboración propia (2024).

Tabla 77.

Diccionario de datos, tabla KardexArticuloBodega

Nombre de la Tabla		KardexArticuloBodega			
Descripción		Tabla para almacenar el Kardex por bodega de los artículos para el control de costos y evidenciar las diferentes transacciones del sistema			
Columna	Tipo Dato	Null	Pk	Fk	Descripción
IdKardexArticulo	bigint		X		Identificador del registro del Kardex
IdCompania	int			X	Identificador de la compañía - Fk tabla Compañía
CodSucursal	int			X	Código de la sucursal - Fk tabla Sucursal
IdBodega	int			X	Identificador de la bodega - Fk tabla bodega
CodArticulo	nvarchar(20)			X	Código del artículo - Fk tabla Articulo
CantEntra	float				Cantidad que ingresa
CantSale	float				Cantidad que sale
Saldo	float				Saldo del artículo
MontoEntra	decimal(18,5)				Costo total de entrada
MontoSale	decimal(18,5)				Costo total de Salida
CostoUniNuevo	decimal(18,5)				Costo unitario nuevo
CostoUniAnterior	decimal(18,5)				Costo unitario anterior
FecHora	datetime				Fecha y hora de la transacción
TipoMovimiento	varchar(20)				Tipo de Movimiento E=Entrada S=Salida

Fuente: Elaboración propia (2024).

Tabla 78.*Diccionario de datos, tabla PerfilCliente*

Nombre de la Tabla		PerfilCliente			
Descripción		Tabla para almacenar los perfiles del cliente			
Columna	Tipo Dato	Null	Pk	Fk	Descripción
IdPerfilCliente	int		X		Identificador del perfil
IdCompania	int			X	Identificador de la compañía -Fk tabla Compania
Descripcion	varchar(50)				Descripción del perfil
Credito	bit				True: Se le puede facturar a crédito False: solo se puede facturar a contado
Exonerado	bit				True: Indica que el cliente es exonerado y obliga a completar los datos de la exoneración en el mantenimiento de cliente
CCSS	bit				True: Indica que es del tipo de cliente Caja Costarricense del Seguro Social
PorcentajeDescuento	float				Porcentaje de descuento que se aplicara al facturar
DescuentoPermanente	bit				True: Indica que el descuento es permanente, False: El descuento tiene fecha de inicio y fecha de fin
FecInicioDescuento	datetime	X			Fecha de inicio del descuento
FecFinDescuento	datetime	X			Fecha de fin del descuento
ListaPrecioAsignada	smallint				1: Precio 1, 2: Precio 2, 3: Precio 3
Estado	smallint				Estado: 1-Inactivo, 2- Activo

Fuente: Elaboración propia (2024).**Tabla 79.***Diccionario de datos, tabla Cliente*

Nombre de la Tabla		Cliente			
Descripción		Tabla para almacenar la información de los clientes			
Columna	Tipo Dato	Null	Pk	Fk	Descripción
IdCliente	int		X		Identificador único del Cliente
IdCompania	int			X	Identificador de la compañía - Fk tabla Compania
IdPerfilCliente	int			X	Identificador del perfil del Cliente - Fk tabla PerfilCliente
CodTipoIdentificacion	varchar(2)			X	Código del tipo de identificación- Fk tabla TipoIdentificacion
Identificacion	varchar(20)				Identificación del cliente
Nombre	varchar(80)				Nombre del cliente
CodProvincia	smallint			X	Código de provincia - Fk tabla Provincia
CodCanton	smallint			X	Código de cantón - Fk tabla Canton
CodDistrito	smallint			X	Código del distrito - Fk tabla Distrito
CodBarrio	smallint			X	Código del Barrio - Fk tabla Barrio
Dirección	nvarchar(300)				Dirección del cliente
CodAreaTelefono	smallint			X	Código de área (país) al que pertenece el teléfono - Fk tabla CodigoArea
Telefono	varchar(20)				Teléfono del cliente
CodAreaFax	smallint			X	Código de área (país) al que pertenece el fax- Fk Tabla CodigoArea

Nombre de la Tabla		Cliente			
Descripción		Tabla para almacenar la información de los clientes			
Columna	Tipo Dato	Null	Pk	Fk	Descripción
Fax	varchar(20)				Número de fax
Correo	nvarchar(160)				Correo Electrónico del cliente
CorreoFacturacion	nvarchar(160)				Correo Electrónico al que se van a enviar los documentos electrónicos
UnidadEjecutoraCCSS	int	X			Código Unidad Ejecutora de la Caja Costarricense, se utiliza para diferenciar a los hospitales con una misma cédula Jurídica
RepresentanteLegal	varchar(80)	X			Representante legal del cliente en caso de que sea una identificación Jurídica
Contacto	varchar(160)	X			Contacto del cliente
ContactoCobro	varchar(160)	X			Contacto de cobro en caso de que el Cliente sea de crédito
LimiteCredito	float				Límite de crédito del Cliente
Saldo	float				Saldo Global (Monto) de Facturas por pagar cuando el cliente es de crédito
Plazo	int				Plazo (Días) de crédito que se le otorga al cliente
FecUltVenta	datetime	X			Fecha hora de última venta que se le realizo al cliente
FecIngreso	datetime				Fecha hora de ingreso del registro
FecUltModificacion	datetime	X			Fecha hora de última modificación que se le realizo al cliente
IdUsuarioCreo	int			X	Identificador del usuario que creo registro - Fk tabla Usuario
IdUsuarioModifico	int			X	Identificador del usuario que modificó el registro - Fk Tabla Usuario
IdCuentaContable	int	X			Identificador de cuenta contable en caso de que los asientos y las cuentas de clientes de crédito se lleve de manera individual
CodTipoDocumentoExoneracion	varchar(2)	X		X	Código del tipo de documento exoneración en caso de que el perfil del cliente sea exonerado - Fk TipoDocumentoExoneracion
NumDocumentoExoneracion	varchar(40)	X			Número de documento que respalda la exoneración
InstitucionExoneracion	varchar(160)	X			Nombre de la institución que emitió la exoneración
FecExoneracion	datetime	X			Fecha hora del documento de exoneración
PorcExoneracionAutorizado	decimal(4,2)	X			Porcentaje de exoneración autorizado
CreditoCerrado	bit				Indicador si el crédito del cliente está cerrado
Estado	smallint				Estado: 1-Inactivo, 2- Activo

Fuente: Elaboración propia (2024).

Tabla 80.*Diccionario de datos, tabla Paciente*

Nombre de la Tabla		Paciente			
Descripción		Tabla para almacenar la información de los pacientes			
Columna	Tipo Dato	Null	Pk	Fk	Descripción
IdPaciente	int		X		Identificador único del paciente
IdCompania	int			X	Identificador de la compañía - Fk tabla Compania
CodTipoIdentificacion	varchar(2)			X	Código del tipo de identificación- Fk tabla TipoIdentificacion
Identificacion	varchar(20)				Identificación del paciente
Nombre	varchar(80)				Nombre del paciente
Observacion	nvarchar(1000)				Observación del paciente, campos para incluir algún otro tipo de dato que se considere necesario
Telefono	varchar(20)	X			Teléfono del cliente
Correo	nvarchar(160)	X			Correo electrónico del cliente
FecUltVenta	datetime	X			Fecha hora de última venta que se le realizo al cliente
FecIngreso	datetime				Fecha hora de ingreso del registro
FecUltModificacion	datetime	X			Fecha hora de última modificación que se le realizo al cliente
IdUsuarioCreo	int			X	Identificador del usuario que creo registro - Fk tabla Usuario
IdUsuarioModifico	int			X	Identificador del usuario que modificó el registro - Fk tabla Usuario
SuelaThomas	nvarchar(255)				Información Thomas de la suela del zapato
SuelaTaconCorriente	nvarchar(255)				Información Tacón corriente de la suela del zapato
SuelaViron	nvarchar(255)				Información Virón de la suela del zapato
SuelaOtros	nvarchar(1000)				Otros datos importantes de la suela
PlantillaArcoLongitudinal	nvarchar(255)				Información Arco Longitudinal de la plantilla
PlantillaArcoTransverso	nvarchar(255)				Información Arco Transverso de la plantilla
PlantillaViron	nvarchar(255)				Información Virón de la plantilla
PlantillaEspolon	nvarchar(255)				Información Espolón de la plantilla
PlantillaSoporteAnt	nvarchar(255)				Información Soporte anterior de la plantilla
PlantillaOtros	nvarchar(1000)				Otros datos importantes de la plantilla
Estado	smallint				Estado: 1-Inactivo, 2- Activo

Fuente: Elaboración propia (2024).**Tabla 81.***Diccionario de datos, tabla PadronIdentificacion*

Nombre de la Tabla		PadronIdentificacion			
Descripción		Tabla para almacenar el padrón nacional de identificaciones.			
Columna	Tipo Dato	Null	Pk	Fk	Descripción
CodTipoIdentificacion	varchar(2)		X	X	Código del tipo de identificación
Identificacion	varchar(12)		X		Identificación de la persona Física o Jurídica
Nombre	varchar(80)				Nombre completo de la persona Física o jurídica

Fuente: Elaboración propia (2024).

Tabla 82.*Diccionario de datos, tabla EncabezadoCompra*

Nombre de la Tabla		EncabezadoCompra			
Descripción		Tabla para almacenar el Encabezado de las compras			
Columna	Tipo Dato	Null	Pk	Fk	Descripción
IdCompania	int		X	X	Identificador de la compania - Fk tabla Compania
CodSucursal	int		X	X	Código de la sucursal - Fk tabla Sucursal
Compra	bigint		X		Consecutivo de la compra
IdProveedor	int			X	Identificador del proveedor - Fk tabla Proveedor
FecHora	datetime				Fecha hora del ajuste
FecHoraFactura	datetime	X			Fecha y hora de la factura de compra
FecHoraAplicacion	datetime	X			Fecha hora de aplicación de la compra
IdUsuarioCrea	int			X	Identificador del usuario que crea la compra
Estado	int				Estado: 1:Pendiente, 2:Aplicada, 3:Eliminada
NumFacturaProveedor	nvarchar(50)				Número de factura del proveedor
Observacion	varchar(255)				Observación del ajuste
TotalServGravado	decimal(18,5)				Total, de servicios Gravados
TotalServExentos	decimal(18,5)				Total, de servicios Exentos
TotalMercGravado	decimal(18,5)				Total, de artículos Gravados
TotalMercExenta	decimal(18,5)				Total, de artículos Exentos
Subtotal	decimal(18,5)				Subtotal compra
TotalDescuento	decimal(18,5)				Total, de descuentos
TotalNeto	decimal(18,5)				Total, Neto antes de impuesto (Subtotal- Total Descuento)
TotalImpuestos	decimal(18,5)				Total, de impuestos
TotalCompra	decimal(18,5)				Total, de la compra= Total Neto+ Total Impuestos
EsCredito	bit				Indica si la compra es Crédito
Plazo	int				Plazo de crédito de la factura de compra en caso de que sea de crédito

Fuente: Elaboración propia (2024).**Tabla 83.***Diccionario de datos, tabla DetalleCompra*

Nombre de la Tabla		DetalleCompra			
Descripción		Tabla para almacenar el Detalle de las compras			
Columna	Tipo Dato	Null	Pk	Fk	Descripción
IdCompania	int		X	X	Identificador de la compania - Fk tabla Compania
CodSucursal	int		X	X	Código de la sucursal - Fk tabla Sucursal
Compra	bigint		X	X	Consecutivo de la compra - Fk Encabezado Compra
Linea	int		X		Número de línea
CodArticulo	nvarchar(20)			X	Código de Artículo - Fk tabla Articulo
Cantidad	float				Cantidad
Costo	decimal(18,5)				Costo del articulo
CodigoTipoImpuesto	varchar(2)			X	Código el tipo de impuesto - Fk Tabla TipoImpuesto
CodTarifaImpuesto	varchar(2)			X	Código de la tarifa de Impuesto - Fk Tabla TarifaImpuesto
PorcImpuesto	float				Porcentaje de impuesto
EsServicio	bit				True: Indica si la línea es un servicio
Impuesto	decimal(18,5)				Monto de impuesto unitario

Nombre de la Tabla		DetalleCompra			
Descripción		Tabla para almacenar el Detalle de las compras			
Columna	Tipo Dato	Null	Pk	Fk	Descripción
Descuento	decimal(18,5)				Monto de Descuento unitario
PorcDescuento	float				Porcentaje de descuento

Fuente: Elaboración propia (2024).

Tabla 84.

Diccionario de datos, tabla Receta

Nombre de la Tabla		Receta			
Descripción		Tabla para almacenar las Recetas (formulas de producción) que se van a utilizar al generar las ordenes de trabajo para rebajar y controlar los insumos en el proceso de adaptación de los zapatos			
Columna	Tipo Dato	Null	Pk	Fk	Descripción
IdCompania	int		X	X	Identificador de la compania - Fk Tabla Compania
CodArticulo	nvarchar(20)		X	X	Código de artículo - Fk Tabla Articulo
Componente	nvarchar(20)		X	X	Código de artículo (componente) - Fk tabla Articulo
Cantidad	float				Cantidad por rebajar cuando se utilice el artículo compuesto

Fuente: Elaboración propia (2024).

Tabla 85.

Diccionario de datos, tabla MaestroFormula

Nombre de la Tabla		MaestroFormula			
Descripción		Tabla para almacenar el Maestro de fórmulas que se va a utilizar para generar los asientos contables			
Columna	Tipo Dato	Null	Pk	Fk	Descripción
CodMaestroFormula	char(1)		X		Código de la formula
Descripcion	varchar(20)			X	Código de sucursal - Fk Tabla Sucursal

Fuente: Elaboración propia (2024).

Tabla 86.

Diccionario de datos, tabla DocumentoElectronico

Nombre de la Tabla		DocumentoElectronico			
Descripción		Tabla para almacenar los diferentes archivos de los documentos electrónicos			
Columna	Tipo Dato	Null	Pk	Fk	Descripción
IdCompania	int		X	X	Identificador de la compañía - Fk tabla Compañía
CodSucursal	int		X	X	Código de estado de hacienda - Fk tabla EstadoHacienda
CodTerminal	int		X	X	Código de la terminal - Fk tabla Terminal
CodTipoDocumento	nvarchar(2)		X	X	Código del Tipo documento – Fk tabla TipoDocumento
ConsecutivoInterno	float		X		Consecutivo interno de los documentos
ConsecutivoNumerico	varchar(20)				Consecutivo numérico
ClaveNumerica	varchar(50)				Clave numérica
Documento	Text				Documento XML Base64 firmado electrónicamente
CodEstadoHacienda	smallint			X	Código de estado de hacienda - Fk tabla EstadoHacienda

Nombre de la Tabla		DocumentoElectronico				
Descripción		Tabla para almacenar los diferentes archivos de los documentos electrónicos				
Columna	Tipo Dato	Null	Pk	Fk	Descripción	
CorreoEnviado	bit				True= Indica si el correo ya enviado	
FecHoraProceso	datetime				Fecha hora del documento	
FecHoraEnvioCorreo	datetime	X			Fecha hora de envió del correo electrónico con los XML y el PDF	
EnvioHacienda	bit				Indicador: False está pendiente de enviar el documento a Hacienda	
FecHoraEnvioHacienda	datetime	X			Fecha hora en la que se envió el documento a Hacienda	

Fuente: Elaboración propia (2024).

Tabla 87.

Diccionario de datos, tabla DocumentoElectronicoRespuesta

Nombre de la Tabla		DocumentoElectronicoRespuesta				
Descripción		Tabla para almacenar los archivos de respuesta de los diferentes documentos electrónicos que se enviar al ministerio de Hacienda				
Columna	Tipo Dato	Null	Pk	Fk	Descripción	
IdCompania	int		X	X	Identificador de la compañía - Fk tabla Compañía	
CodSucursal	int		X	X	Código de estado de hacienda - Fk tabla EstadoHacienda	
CodTerminal	int		X	X	Código de la terminal - Fk tabla Terminal	
CodTipoDocumento	varchar(2)		X	X	Código del Tipo documento - Fk TipoDocumento	
ConsecutivoInterno	float		X		Consecutivo interno de los documentos	
ConsecutivoNumerico	varchar(20)				Consecutivo numérica	
ClaveNumerica	varchar(50)				Clave Numérica	
Respuesta	Text				Archivo de respuesta del Ministerio de Hacienda en Base64	
CodEstadoHacienda	smallint			X	Código de estado de hacienda - Fk Tabla EstadoHacienda	
FecHoraProceso	datetime				Fecha hora del documento en la que se obtiene respuesta	
Mensaje	Text				Mensaje de Hacienda	

Fuente: Elaboración propia (2024).

Tabla 88.

Diccionario de datos, tabla ParametroGlobal

Nombre de la Tabla		ParametroGlobal				
Descripción		Tabla para almacenar la configuración de parámetros Globales del sistema				
Columna	Tipo Dato	Null	Pk	Fk	Descripción	
CodParametro	nvarchar(25)		X		Código del parámetro	
Descripcion	varchar(25)				Descripción del parámetro	
ValorNumero	int	X			Valor numérico por defecto del parámetro	
ValorTexto	nvarchar(100)	X			Valor Texto por defecto del parámetro	
ValorFecha	smalldatetime	X			Valor Fecha por defecto del parámetro	
Campo	varchar(3)				Indica el tipo de parámetro: VN: Valor numérico, VT: Valor texto, VF: Valor Fecha	

Fuente: Elaboración propia (2024).

Tabla 89.

Diccionario de datos, tabla DocumentoEncabezado

Nombre de la Tabla		DocumentoEncabezado			
Descripción		Tabla para almacenar datos del encabezado y resumen de los distintos tipos de documentos			
Columna	Tipo Dato	Null	Pk	Fk	Descripción
IdCompania	int		X	X	Identificador de la compañía Fk - Tabla Compania
CodSucursal	int		X	X	Código de la sucursal Fk - Tabla Sucursal
CodTerminal	int		X	X	Código de la Terminal Fk - Tabla Terminal
CodTipoDocumento	nvarchar(2)		X	X	Código Tipo de Documento Electrónico FK - tabla TipoDocumento
ConsecutivoInterno	float		X		Consecutivo Interno del documento
FecHoraEmision	datetime				Fecha y hora del documento
FecVencimiento	date	X			Fecha de vencimiento del documento en caso de ser de crédito
TipoCambio	decimal(18,5)				Tipo de cambio, número decimal compuesto por 13 enteros y 5 decimales
CodMoneda	varchar(2)			X	Código de moneda del documento
CodCondicionVenta	varchar(2)			X	Código de la condición de venta- Fk tabla CondicionVenta
CantDiasCredito	smallint	X			Plazo para los documentos de crédito
NumOrdenCompra	varchar(20)				Almacena el número de orden de compra en caso de que el cliente lo requiera en el documento electrónico
Observacion	nvarchar(1000)				Observación o detalle que se le coloca a los documentos electrónicos
TotalServGravados	decimal(18,5)				Este campo será de condición obligatoria, cuando el servicio este gravado con IVA. Es un número decimal compuesto por 13 enteros y 5 decimales. En el caso que exista una exoneración Este campo se obtiene de la multiplicación (1-porcentaje de exoneración) por el monto de la venta
TotalServExentos	decimal(18,5)				Este campo será de condición obligatoria, cuando el servicio este exento de IVA. Es un número decimal compuesto por 13 enteros y 5 decimales.
TotalServExonerado	decimal(18,5)				Este campo será de condición obligatoria, cuando el servicio esté gravado y se preste a un cliente que goce de exoneración, se debe de indicar el monto equivalente al porcentaje exonerado. Se obtiene de la multiplicación del porcentaje de la exoneración por el monto del servicio.

Nombre de la Tabla		DocumentoEncabezado			
Descripción		Tabla para almacenar datos del encabezado y resumen de los distintos tipos de documentos			
Columna	Tipo Dato	Null	Pk	Fk	Descripción
					Es un número decimal compuesto por 13 enteros y 5 decimales.
TotalMercanciasGravadas	decimal(18,5)				Este campo será de condición obligatoria, cuando la mercancía este gravada con IVA. Es un número decimal compuesto por 13 enteros y 5 decimales. En el caso que exista una exoneración Este campo se obtiene de la multiplicación (1-porcentaje de exoneración) por el monto de la venta.
TotalMercanciasExentas	decimal(18,5)				Este campo será de condición obligatoria, cuando la mercancía esté exenta de IVA. Es un número decimal compuesto por 13 enteros y 5 decimales.
TotalMercExonerada	decimal(18,5)				Este campo será de condición obligatoria, cuando la mercancía o producto se venda a un cliente que goce de exoneración para la compra de esta. Se obtiene de la multiplicación del porcentaje de la exoneración por el monto del producto. Es un número decimal compuesto por 13 enteros y 5 decimales.
TotalGravado	decimal(18,5)				Se obtiene de la suma de los campos “total servicios gravados con IVA” mas “total de mercancías gravadas con IVA”.
TotalExento	decimal(18,5)				Se obtiene de la suma de los campos “total servicios exentos de IVA” mas “total de mercancías exentas de IVA”.
TotalExonerado	decimal(18,5)				Se obtiene de la suma de los campos “total servicios exonerados de IVA” mas “total de mercancías exoneradas del IVA”.
TotalVenta	decimal(18,5)				Se obtiene de la sumatoria de los campos “total gravado”, “total exento” y “Total Exonerado”.
TotalDescuentos	decimal(18,5)				Se obtiene de la suma de todos los campos de “monto de descuentos concedidos”.
TotalVentaNeta	decimal(18,5)				Se obtiene de la resta de los campos “total venta” menos “total descuento”.
TotalImpuesto	decimal(18,5)				Este campo es de condición obligatoria, cuando existen producto/servicios gravados con algún impuesto en las líneas de detalle el mismo se obtiene de la suma de todos los campos denominados “Monto del impuesto” cuando no posea exoneración más todos

Nombre de la Tabla		DocumentoEncabezado			
Descripción		Tabla para almacenar datos del encabezado y resumen de los distintos tipos de documentos			
Columna	Tipo Dato	Null	Pk	Fk	Descripción
					los campos “Impuesto Neto” de las líneas que poseen exoneración
TotalIVADevuelto	decimal(18,5)				Este campo será de condición obligatoria cuando se facturen servicios de salud y cuyo método de pago sea “Tarjeta”.
TotalOtrosCargos	decimal(18,5)				Este campo será de condición obligatoria, cuando se posean otros cargos.
TotalComprobante	decimal(18,5)				Se obtiene de la suma de los campos “total venta neta”, “monto total del impuesto” y “total otros cargos” menos “total IVA devuelto”, en caso de contar con dichos campos.
IdUsuario	decimal(18,5)			X	Identificador del usuario - FK Tabla Usuario
NumPedidoVenta	float				Consecutivo del pedido
NumOrdenTrabajo	float				Número de orden de trabajo.
CodActividadComercial	nvarchar(6)			X	Código de la actividad comercial
EsElectronico	bit				Indicador si el documento es electrónico porque puede ser una nota de crédito o débito que tenga un efecto contable o de ajuste en el sistema y no se envíe al Ministerio de hacienda

Fuente: Elaboración propia (2024).

Tabla 90.

Diccionario de datos, tabla DocumentoElectronicoRespuesta

Nombre de la Tabla		DocumentoElectronicoRespuesta			
Descripción		Tabla para almacenar los archivos de respuesta de los diferentes documentos electrónicos que se enviar al ministerio de Hacienda			
Columna	Tipo Dato	Null	Pk	Fk	Descripción
IdCompania	int		X	X	Identificador de la compañía - Fk tabla Compañía
CodSucursal	int		X	X	Código de estado de hacienda - Fk Tabla EstadoHacienda
CodTerminal	int		X	X	Código de la terminal - FK Tabla Terminal
CodTipoDocumento	nvarchar(2)		X	X	Código del Tipo documento - FK TipoDocumento
ConsecutivoInterno	float		X		Consecutivo interno de los documentos
ConsecutivoNumerico	varchar(20)				Consecutivo numérica
ClaveNumerica	varchar(50)				Clave Numérica
Respuesta	Text				Archivo de respuesta del Ministerio de Hacienda en Base64
CodEstadoHacienda	smallint			X	Código de estado de hacienda - Fk Tabla EstadoHacienda
FecHoraProceso	datetime				Fecha hora del documento en la que se obtiene respuesta
Mensaje	Text				Mensaje de Hacienda

Fuente: Elaboración propia (2024).

Tabla 91.*Diccionario de datos, tabla DocumentoDetalle*

Nombre de la Tabla		DocumentoDetalle			
Descripción		Tabla para almacenar datos del detalle de los distintos documentos			
Columna	Tipo Dato	Null	Pk	Fk	Descripción
IdCompania	int		X	X	Identificador de la compañía Fk - Tabla Compania
CodSucursal	int		X	X	Código de la sucursal FK- Tabla Sucursal
CodTerminal	int		X	X	Código de la Terminal FK- Tabla Terminal
CodTipoDocumento	varchar(2)		X	X	Código Tipo de Documento Electrónico FK- Tabla TipoDocumento
ConsecutivoInterno	float		X		Consecutivo Interno del documento
Linea	int		X		Número de línea del Detalle
CodArticulo	nvarchar(20)		X		Código del bien o servicio
CodTipoCodigoArticulo	varchar(2)			X	Tipo de código del artículo, clasificación para los documentos electrónicos - FK Tabla TipoCodigoArticulo
Cabys	varchar(13)				Código CAByS del artículo
Cantidad	decimal(16,3)				Cantidad de artículos
Descripcion	nvarchar(200)				Descripción del artículo, solo permite 200 caracteres por requerimiento de la facturación electrónica
PrecioUniarario	decimal(18,5)				Precio Unitario
MontoTotal	decimal(18,5)				Monto Total. Se obtiene de la multiplicación del campo "cantidad" por el campo "precio unitario".
PorcDescuento	decimal(18,5)				Porcentaje del descuento
MontoDescuento	decimal(18,5)				Este campo será de condición obligatoria, cuando exista un descuento.
NaturalezaDescuento	nvarchar(80)				Detalle del descuento
Subtotal	decimal(18,5)				Subtotal. Se obtiene de la resta del campo "monto total" menos "monto de descuento concedido".
EsServicio	bit				Indicador si el artículo es un servicio
EsBonificacion	bit				Indicador si el artículo es una bonificación
PartidaArancelaria	nvarchar(15)	X			Partida arancelaria. Se convierte en carácter obligatorio cuando se elija el tipo de comprobante "Factura electrónica de exportación" y la misma corresponda a la venta de una mercancía.
BaseImponible	decimal(18,5)				Base imponible del artículo. Se convierte en obligatorio cuando se seleccione en el campo "Código del impuesto" 07.
IdUnidadMedida	int			X	Identificador de la unidad de medida - FK Tabla UnidadMedida
UnidadMedidaComercial	varchar(20)				Si el identificador de la unidad de medida es 94 se habilita el campo para colocar la descripción de la unidad de medida comercial
ImpuestoNeto	decimal(18,5)				Impuesto Neto de la línea. Este campo será de condición obligatoria, cuando se incluya información en el campo "Tipo de

Nombre de la Tabla		DocumentoDetalle				
Descripción		Tabla para almacenar datos del detalle de los distintos documentos				
Columna	Tipo Dato	Null	Pk	Fk	Descripción	
					exoneración”. Dicho monto se obtiene de la multiplicación del campo “porcentaje de la exoneración” por el campo “Monto del Impuesto”.	
MontoTotalLinea	decimal(18,5)				Monto Total Linea. Existe dos maneras de obtener el resultado del campo: Cuando no existe exoneración, se obtiene de la sumatoria de los campos “subtotal”, “monto del impuesto - Cuando posee una exoneración, se obtiene de la sumatoria de los campos “Subtotal”, “Impuesto Neto”.	

Fuente: Elaboración propia (2024).

Tabla 92.

Diccionario de datos, tabla DocumentoDetalleImpuesto

Nombre de la Tabla		DocumentoDetalleImpuesto				
Descripción		Tabla para almacenar datos del detalle impuesto de los distintos documentos				
Columna	Tipo Dato	Null	Pk	Fk	Descripción	
IdCompania	int		X	X	Identificador de la compania Fk - tabla Compania	
CodSucursal	int		X	X	Código de la sucursal Fk- Tabla Sucursal	
CodTerminal	int		X	X	Código de la Terminal Fk- Tabla Terminal	
CodTipoDocumento	nvarchar(2)		X	X	Código Tipo de Documento Electrónico Fk- Tabla TipoDocumento	
ConsecutivoInterno	float		X		Consecutivo Interno del documento	
Linea	int		X	X	Número de línea del Detalle	
CodArticulo	nvarchar(20)		X	X	Código del bien o servicio	
CodTipoImpuesto	varchar(2)		X	X	Código del tipo de impuesto valor agregado - Fk Tabla TipoImpuesto	
CodTarifaImpuesto	varchar(2)		X	X	Código de la tarifa de impuesto de valor agregado - Fk TarifaImpuesto	
PorcImpuesto	decimal(18,5)				Porcentaje de impuesto	
MontoImpuesto	decimal(18,5)				Monto del impuesto. a. Este campo será de condición obligatoria, cuando el producto/servicio este gravado con algún impuesto. Se obtiene de la multiplicación del campo “subtotal” por “tarifa del impuesto”. b. En caso de utilizar el factor establecido para el régimen de bienes usados, este campo se obtiene de la multiplicación del factor menos 1 por el subtotal. c. En caso de utilizar el campo “Base	

Nombre de la Tabla		DocumentoDetalleImpuesto			
Descripción		Tabla para almacenar datos del detalle impuesto de los distintos documentos			
Columna	Tipo Dato	Null	Pk	Fk	Descripción
					imponible”, este campo se obtiene de la multiplicación “Base imponible” por la tarifa
CodTipoDocumentoExoneracion	varchar(2)			X	Código del tipo de documento exoneración en caso de que el perfil del cliente sea exonerado - Fk TipoDocumentoExoneracion
NumDocumentoExoneracion	varchar(40)				Número de Documento que respalda la exoneración
InstitucionExoneracion	varchar(160)				Nombre de la institución que emitió la exoneración
FecExoneracion	datetime	X			Fecha Hora del documento de exoneración
PorcExoneracionAutorizado	decimal(4,2)				Porcentaje de exoneración autorizado
MontoExoneracion	decimal(18,5)				Monto de la exoneración
FactorIVA	decimal(18,5)				Factor IVA. Este campo es de condición obligatoria, cuando el producto/servicio posea un factor para su cálculo. Cuando en el código de impuesto se defina IVA Bienes Usados se deberá utilizar este campo con el factor establecido por el Ministerio de Hacienda

Fuente: Elaboración propia (2024).

Tabla 93.

Diccionario de datos, tabla DocumentoDetalleReferencia

Nombre de la Tabla		DocumentoDetalleReferencia			
Descripción		Tabla para almacenar datos del encabezado y resumen del documento electrónico			
Columna	Tipo Dato	Null	Pk	Fk	Descripción
IdCompania	int		X	X	Identificador de la compania Fk - Tabla Compania
CodSucursal	int		X	X	Código de la sucursal FK- Tabla Sucursal
CodTerminal	int		X	X	Código de la Terminal FK- Tabla Terminal
CodTipoDocumento	nvarchar(2)		X	X	Código Tipo de Documento Electrónico FK- Tabla TipoDocumento
ConsecutivoInterno	float		X	X	Consecutivo Interno del documento
Linea	int		X		Número de línea de la información de referencia (Máximo 10 líneas por Documento)
ConsecutivoInternoReferencia	float			X	Consecutivo interno del documento de referencia
ConsecutivoNumerico	varchar(20)				Consecutivo numérico de 20 dígitos (Es un campo fijo de veinte posiciones. Solo permite ingresar números)

Nombre de la Tabla		DocumentoDetalleReferencia			
Descripción		Tabla para almacenar datos del encabezado y resumen del documento electrónico			
Columna	Tipo Dato	Null	Pk	Fk	Descripción
ClaveNumerica	varchar(50)				Clave Numérica del documento. Es un campo fijo de cincuenta posiciones. Solo permite ingresar números.
CodTipoDocumentoReferencia	varchar(2)			X	Código del tipo de documento de referencia (Factura Electrónica, Nota Débito, Nota Crédito Etc.) FK- Tabla TipoDocumentoReferencia
CodTipoReferencia	varchar(2)			X	Código del tipo de referencia del documento electrónico - FK Tabla TipoReferencia
ClaveNumericaReferencia	varchar(50)				Clave Numérica del documento de referencia.
FecHoraEmision	datetime				Fecha hora de emisión del documento de referencia.
RazonReferencia	nvarchar(180)				Detalle u observación del motivo por el cual se hace referencia al documento

Fuente: Elaboración propia (2024).

Tabla 94.

Diccionario de datos, tabla DocumentoReceptor

Nombre de la Tabla		DocumentoReceptor			
Descripción		Tabla para almacenar la información del receptor del documento			
Columna	Tipo Dato	Null	Pk	Fk	Descripción
IdCompania	int		X	X	Identificador de la compania Fk - Tabla Compania
CodSucursal	int		X	X	Código de la sucursal FK- Tabla Sucursal
CodTerminal	int		X	X	Código de la Terminal FK- Tabla Terminal
CodTipoDocumento	nvarchar(2)		X	X	Código Tipo de Documento Electrónico FK- Tabla TipoDocumento
ConsecutivoInterno	float		X		Consecutivo Interno del documento
IdCliente	int			X	Identificador del cliente - FK Tabla cliente
CodTipoIdentificacion	varchar(2)			X	Código tipo de identificación del cliente- FK Tabla TipoIdentificacion
Identificacion	varchar(20)				Identificación del cliente
NombreCliente	varchar(80)				Nombre del Cliente
CodProvincia	smallint			X	Código de provincia - FK Tabla Provincia
CodCanton	smallint			X	Código de Cantón - FK Tabla Canton
CodDistrito	smallint			X	Código del Distrito - FK Tabla Distrito
CodBarrio	smallint			X	Código del Barrio - FK Tabla Barrio
Dirección	nvarchar(300)				Dirección del Cliente
CodAreaTelefono	smallint			X	Código de área (País) al que pertenece el teléfono - FK Tabla CodigoArea
Telefono	varchar(20)				Teléfono del Cliente
CodAreaFax	smallint			X	Código de área (País) al que pertenece el Fax- FK Tabla CodigoArea

Nombre de la Tabla		DocumentoReceptor			
Descripción		Tabla para almacenar la información del receptor del documento			
Columna	Tipo Dato	Null	Pk	Fk	Descripción
Fax	varchar(20)				Número de fax
Correo	nvarchar(160)				Correo Electrónico del Cliente
CorreoCopiaCortesia	varchar(1024)	X			Correo Electrónico al que se van a enviar los documentos electrónicos
IdPaciente	int	X			Identificador del paciente
CodTipoIdentificacionPaciente	varchar(2)	X			Código tipo de identificación del paciente - Relación Tabla TipoIdentificacion
IdentificacionPaciente	varchar(20)	X			Identificación del paciente
NombrePaciente	varchar(80)	X			Nombre del paciente
UnidadEjecutoraCCSS	int	X			Código de unidad ejecutora cuando el cliente es la CCSS

Fuente: Elaboración propia (2024).

Tabla 95.

Diccionario de datos, tabla DocumentoDetalleOtrosCargos

Nombre de la Tabla		DocumentoDetalleOtrosCargos			
Descripción		Tabla para almacenar datos de la sección Otros Cargos de los documentos			
Columna	Tipo Dato	Null	Pk	Fk	Descripción
IdCompania	int		X	X	Identificador de la compania Fk - Tabla Compania
CodSucursal	int		X	X	Código de la sucursal FK- Tabla Sucursal
CodTerminal	int		X	X	Código de la Terminal FK- Tabla Terminal
CodTipoDocumento	nvarchar(2)		X	X	Código Tipo de Documento Electrónico FK- Tabla TipoDocumento
ConsecutivoInterno	float		X		Consecutivo Interno del documento
Linea	int		X		Número de línea del Detalle Otros Cargos. Se puede incluir un máximo de 15 repeticiones, cuando se posean otros cargos que forman parte del costo total de la línea de detalle.
IdentificacionTercero	varchar(20)				Número identificación del tercero
NombreTercero	varchar(100)				Nombre del tercero
Detalle	varchar(160)				Detalle - Descripción del cargo
Porcentaje	decimal(6,5)				En el caso que el cargo posea un porcentaje o monto para su cálculo se debe indicar
MontoCargo	decimal(18,5)				Monto total del cargo

Fuente: Elaboración propia (2024).

Tabla 96.*Diccionario de datos, tabla DocumentoDetalleMedioPago*

Nombre de la Tabla		DocumentoDetalleMedioPago			
Descripción		Tabla para almacenar los Medio de pagos			
Columna	Tipo Dato	Null	Pk	Fk	Descripción
IdCompania	int		X	X	Identificador de la compania Fk - Tabla Compania
CodSucursal	int		X	X	Código de la sucursal FK- Tabla Sucursal
CodTerminal	int		X	X	Código de la Terminal FK- Tabla Terminal
CodTipoDocumento	nvarchar(2)		X	X	Código Tipo de Documento Electrónico FK- Tabla TipoDocumento
ConsecutivoInterno	float		X		Consecutivo Interno del documento
Linea	int		X		Número de línea del medio de pago. Se puede incluir un máximo de 4 medios de pago.
CodMedioPago	varchar(2)			X	Código del Medio de pago
Monto	decimal(18,5)				Monto del Medio de pago

Fuente: Elaboración propia (2024).**Tabla 97.***Diccionario de datos, tabla TipoAsiento*

Nombre de la Tabla		TipoAsiento			
Descripción		Tabla para almacenar los Medio de pagos			
Columna	Tipo Dato	Null	Pk	Fk	Descripción
IdCompania	int		X	X	Identificador de la compania Fk - Tabla Compania
CodTipoAsiento	nvarchar(3)				Código del Tipo de asiento
Descripción	nvarchar(50)				Descripción del Tipo de Asiento
Indicador	smallint				Indicador del tipo de asiento 0=No Aplica, =Tipo de Asiento cierre fiscal, 2=Tipo de Asiento para gastos CxP
Estado	smallint				Estado: 1-Inactivo, 2- Activo

Fuente: Elaboración propia (2024).**Tabla 98.***Diccionario de datos, tabla TipoCuenta*

Nombre de la Tabla		TipoAsiento			
Descripción		Tabla para almacenar los Medio de pagos			
Columna	Tipo Dato	Null	Pk	Fk	Descripción
IdCompania	int		X	X	Identificador de la compania Fk - Tabla Compania
CodTipoCuenta	smallint		X		Código del tipo de cuenta
Descripción	nvarchar(50)				Descripción del tipo de cuenta: Activo, Pasivo Etc.
Estado	smallint				Estado: 1-Inactivo, 2- Activo

Fuente: Elaboración propia (2024).**Tabla 99.***Diccionario de datos, tabla CuentaContable*

Nombre de la Tabla		CuentaContable			
Descripción		Tabla para almacenar el Catálogo de Cuentas			
Columna	Tipo Dato	Null	Pk	Fk	Descripción
IdCompania	int			X	Identificador de la compania Fk - Tabla Compania
CodCuentaContable	nvarchar(40)			X	Número de cuenta contable

Nombre de la Tabla		CuentaContable			
Descripción		Tabla para almacenar el Catálogo de Cuentas			
Columna	Tipo Dato	Null	Pk	Fk	Descripción
CodTipoCuenta	smallint			X	Código del tipo de cuenta FK - Tabla Tipo Cuenta
Descripcion	nvarchar(150)				Descripción del número de cuenta contable
EsCuentaMayor	bit				Indica si es una cuenta mayor las cuentas mayores no tienen asientos.
UsarEnGasto	bit				Indica si la cuenta se puede utilizar en como cuenta de gastos en CxP
ImpuestoRenta	bit				Indica cual es la cuenta del Impuesto de Renta por Pagar
ResumenPeriodo	bit				Indica la cuenta Utilidad o Perdida del periodo
UtilPerdidaAcumulada	bit				Indica la cuenta utilidades o pérdidas acumuladas

Fuente: Elaboración propia (2024).

Tabla 100.

Diccionario de datos, tabla ConsecutivoXTipoAsiento

Nombre de la Tabla		ConsecutivoXTipoAsiento			
Descripción		Tabla para almacenar los consecutivos de los asiento por periodo y tipo de asiento			
Columna	Tipo Dato	Null	Pk	Fk	Descripción
IdCompania	int		X	X	Identificador de la compania Fk - Tabla Compania
CodTipoAsiento	nvarchar(3)		X	X	Código del tipo de asiento Fk - Tabla TipoAsiento
Anno	Int		X		Año del periodo contable
Mes	Int		X		Mes del periodo contable
Consecutivo	BigInt				Consecutivo del asiento por tipo de asiento

Fuente: Elaboración propia (2024).

Tabla 101.

Diccionario de datos, tabla Encabezado_Asiento

Nombre de la Tabla		Encabezado_Asiento			
Descripción		Tabla para almacenar el encabezado de los asiento contables			
Columna	Tipo Dato	Null	Pk	Fk	Descripción
IdCompania	int		X	X	Identificador de la compania Fk - Tabla Compania
Asiento	BigInt		X		Consecutivo del Asiento
CodTipoAsiento	nvarchar(3)		X	X	Código del tipo de asiento Fk - Tabla TipoAsiento
Anno	int		X		Año del periodo contable
Mes	int		X		Mes del periodo contable
Detalle	nvarchar(255)				Detalle del encabezado Asiento
Fechahora	datetime				Fecha hora para categorizar el asiento según el periodo
FechaProceso	datetime				Fecha hora del proceso
Usuario	int				Usuario creo

Fuente: Elaboración propia (2024).

Tabla 102.*Diccionario de datos, tabla Detalle_Asiento*

Nombre de la Tabla		Detalle_Asiento			
Descripción		Tabla para almacenar el detalle de los asiento contables			
Columna	Tipo Dato	Null	Pk	Fk	Descripción
IdCompania	int		X	X	Identificador de la compania Fk - Tabla Compania
Asiento	BigInt		X		Consecutivo del Asiento
CodTipoAsiento	nvarchar(3)		X	X	Código del tipo de asiento Fk - Tabla TipoAsiento
Anno	int		X		Año del periodo contable
Mes	int		X		Mes del periodo contable
Secuencia	int		X		Secuencia del detalle del asiento
NumDocumento	nvarchar(50)				Número del documento
FechaDocumento	datetime				Fecha del documento
Cuenta	nvarchar(40)				Cuenta contable
Debe	float				Debe - Monto del asiento contable
Haber	float				Haber - Monto del asiento contable
Detalle	nvarchar(255)				Detalle de la línea del asiento contable
CodTipoDoc	nvarchar(2)	X			Código del tipo de documento

Fuente: Elaboración propia (2024).**Tabla 103.***Diccionario de datos, tabla Factura_Abono*

Nombre de la Tabla		Factura_Abono			
Descripción		Tabla para almacenar los abonos a las facturas de crédito			
Columna	Tipo Dato	Null	Pk	Fk	Descripción
IdCompania	int		X	X	Identificador de la compania Fk - Tabla Compania
CodSucursal	int		X	X	Código de la sucursal FK- Tabla Sucursal
CodTerminal	int		X	X	Código de la Terminal FK- Tabla Terminal
CodTipoDocumento	nvarchar(2)		X		Código Tipo de Documento Electrónico FK- Tabla TipoDocumento
ConsecutivoInternoFact	float		X		Consecutivo Interno del documento (Factura)
ConsecutivoAbono	float		X		Consecutivo del abono
TipoAbono	smallint				TipoAbono:1 Recibo, 2 Nota de Crédito, 3 Nota de Débito
MontoAbono	decimal(18, 5)				Monto del abono
Usuario	int			X	Identificador del usuario
FechaHoraProceso	datetime				Fecha del abono

Fuente: Elaboración propia (2024).**Tabla 104.***Diccionario de datos, tabla Recibos*

Nombre de la Tabla		Recibos			
Descripción		Tabla para almacenar los recibos de pagos CxC			
Columna	Tipo Dato	Null	Pk	Fk	Descripción
IdCompania	int		X	X	Identificador de la compania Fk - Tabla Compania
CodSucursal	int		X	X	Código de la sucursal FK- Tabla Sucursal

Nombre de la Tabla		Recibos			
Descripción		Tabla para almacenar los recibos de pagos CxC			
Columna	Tipo Dato	Null	Pk	Fk	Descripción
CodTerminal	int		X	X	Código de la Terminal FK- Tabla Terminal
IdCliente	int		X		Código Tipo de Documento Electrónico FK- Tabla TipoDocumento
NumeroRecibo	float		X		Consecutivo del recibo
Monto	float		X		Monto del abono
Saldo	float				Saldo del recibo
Observacion	nvarchar(255)				Observación
FechaHoraProceso	datetime			X	Fecha hora del recibo

Fuente: Elaboración propia (2024).

Tabla 105.

Diccionario de datos, tabla MaestroFormulasPaseCxC

Nombre de la Tabla		MaestroFormulasPaseCxC			
Descripción		Tabla para almacenar el maestro de fórmula de los documentos de cuentas por cobrar para realizar los asientos contables.			
Columna	Tipo Dato	Null	Pk	Fk	Descripción
IdCompania	int		X	X	Identificador de la compania Fk - Tabla Compania
CodSucursal	int		X	X	Código de la sucursal FK- Tabla Sucursal
Cuenta	nvarchar(50)		X		Cuenta contable FK- Tabla Cuenta contable
Formula	nvarchar(50)		X		Formula del Documento A=gravado, B=exento, C=descuento Etc.
Tipo	int		X		Consecutivo del recibo
TipoDocumento	int		X		1= crédito 2= débito
CodTipoAsiento	nvarchar(3)		X		Código del tipo asiento

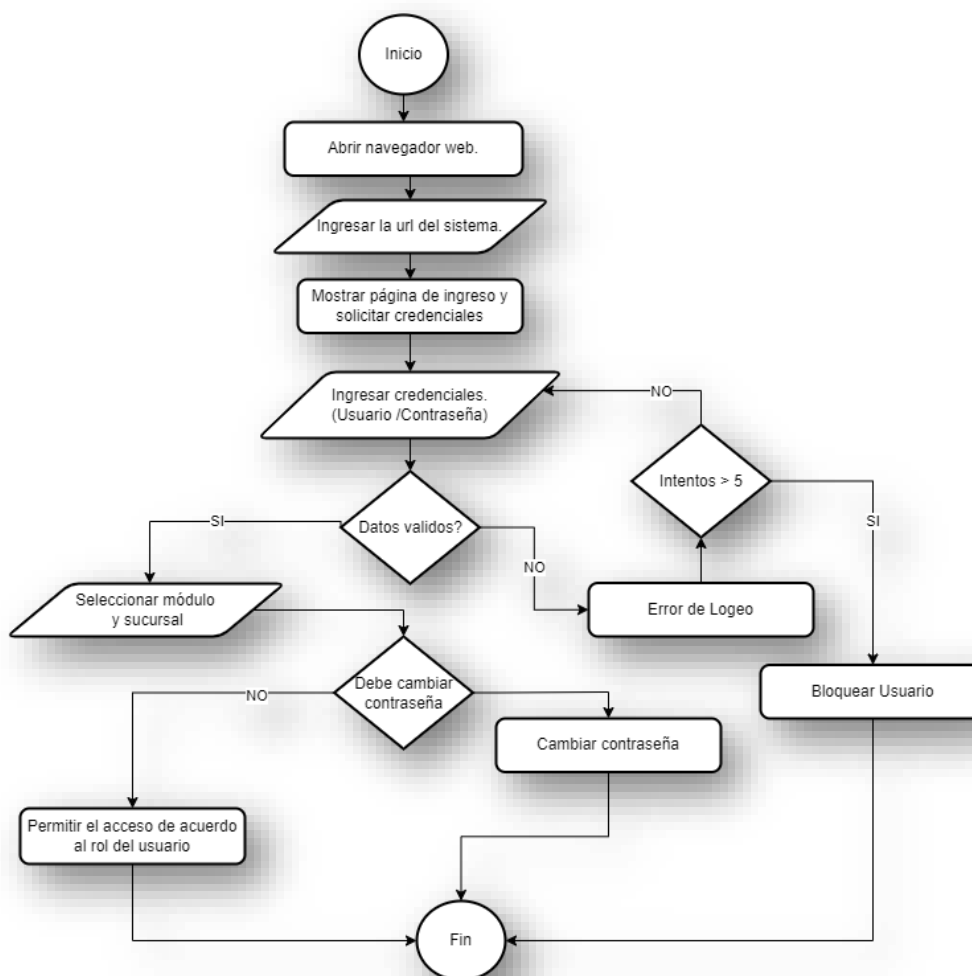
Fuente: Elaboración propia (2024).

Diseño De Procesos

Los diagramas de procesos representan los algoritmos que gobiernan las principales funcionalidades del prototipo, proporcionando una representación gráfica para comprender mejor el comportamiento del sistema. Los aspectos clave identificados en el prototipo para la gestión de ventas de la empresa Zapaticos Lindos S.A., ubicada en San José, incluyen el ingreso y modificación de registros de los mantenimientos del sistema, así como los procesos que conforman el sistema. A continuación, se presentan visualmente cada uno de estos procesos.

Figura 42.

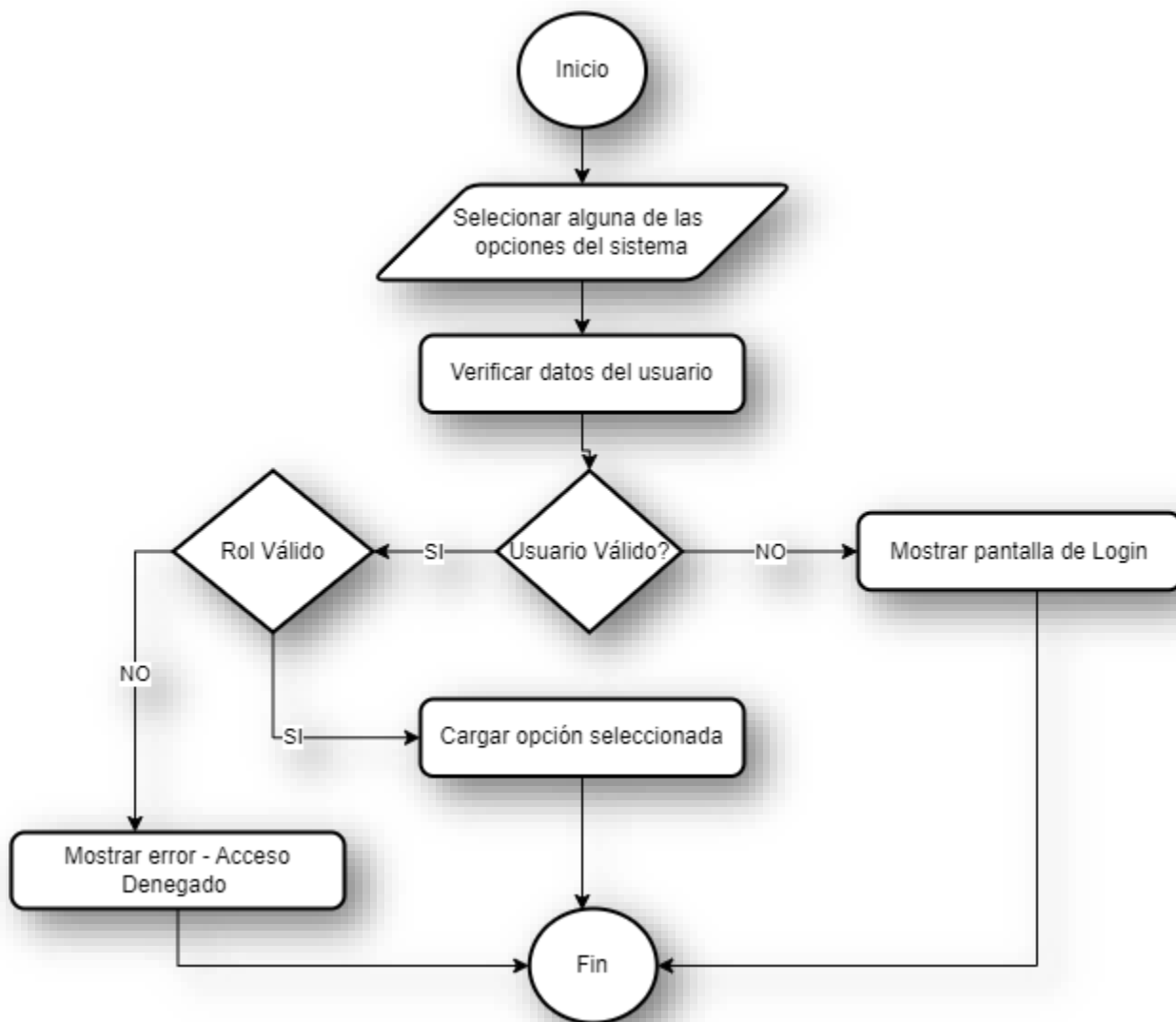
Diagrama de flujo - Ingreso al sistema



Fuente: Elaboración propia (2024).

Figura 43.

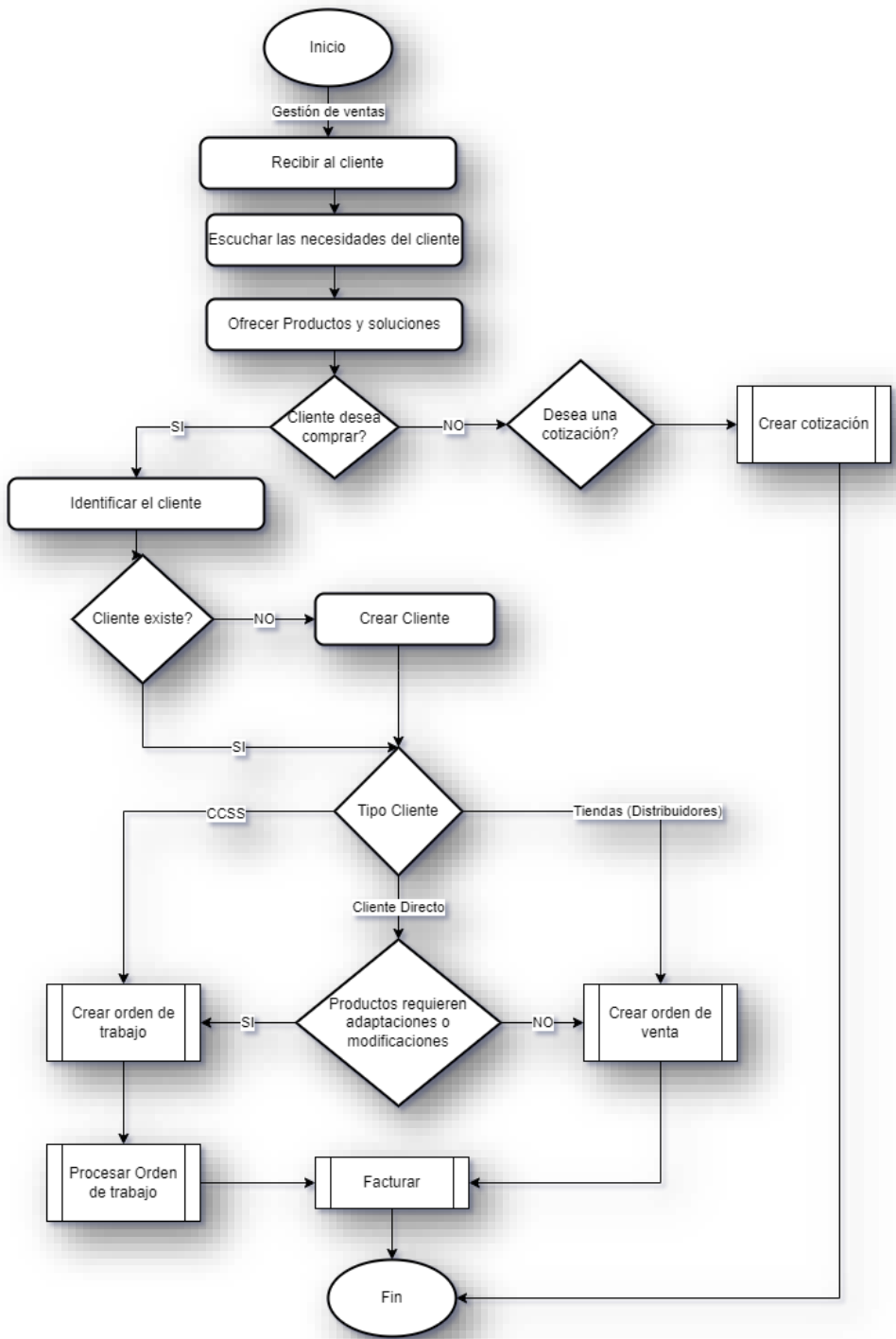
Diagrama de Flujo – Verificar acceso por rol de usuario



Fuente: Elaboración propia (2024).

Figura 44.

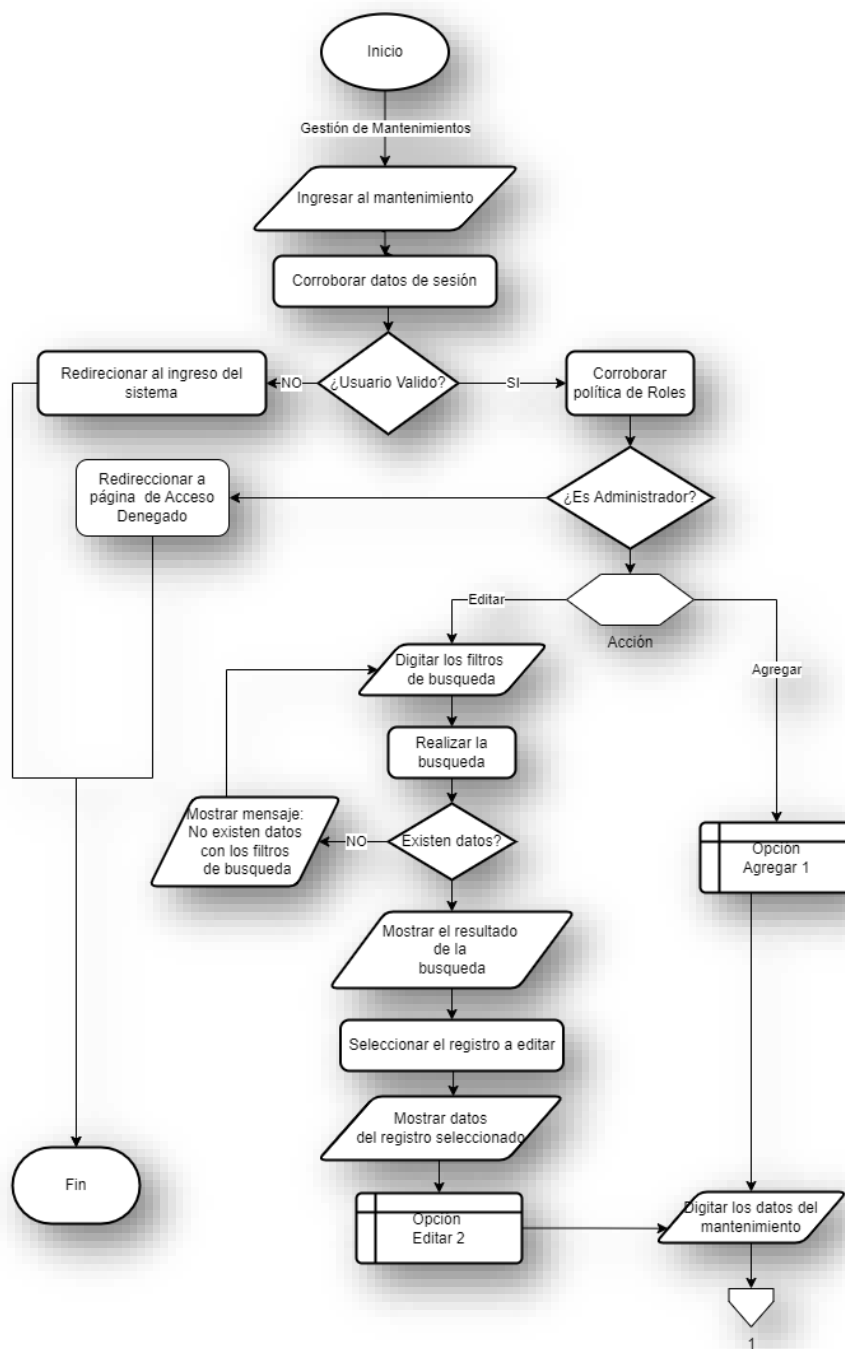
Diagrama de flujo – Gestión de ventas

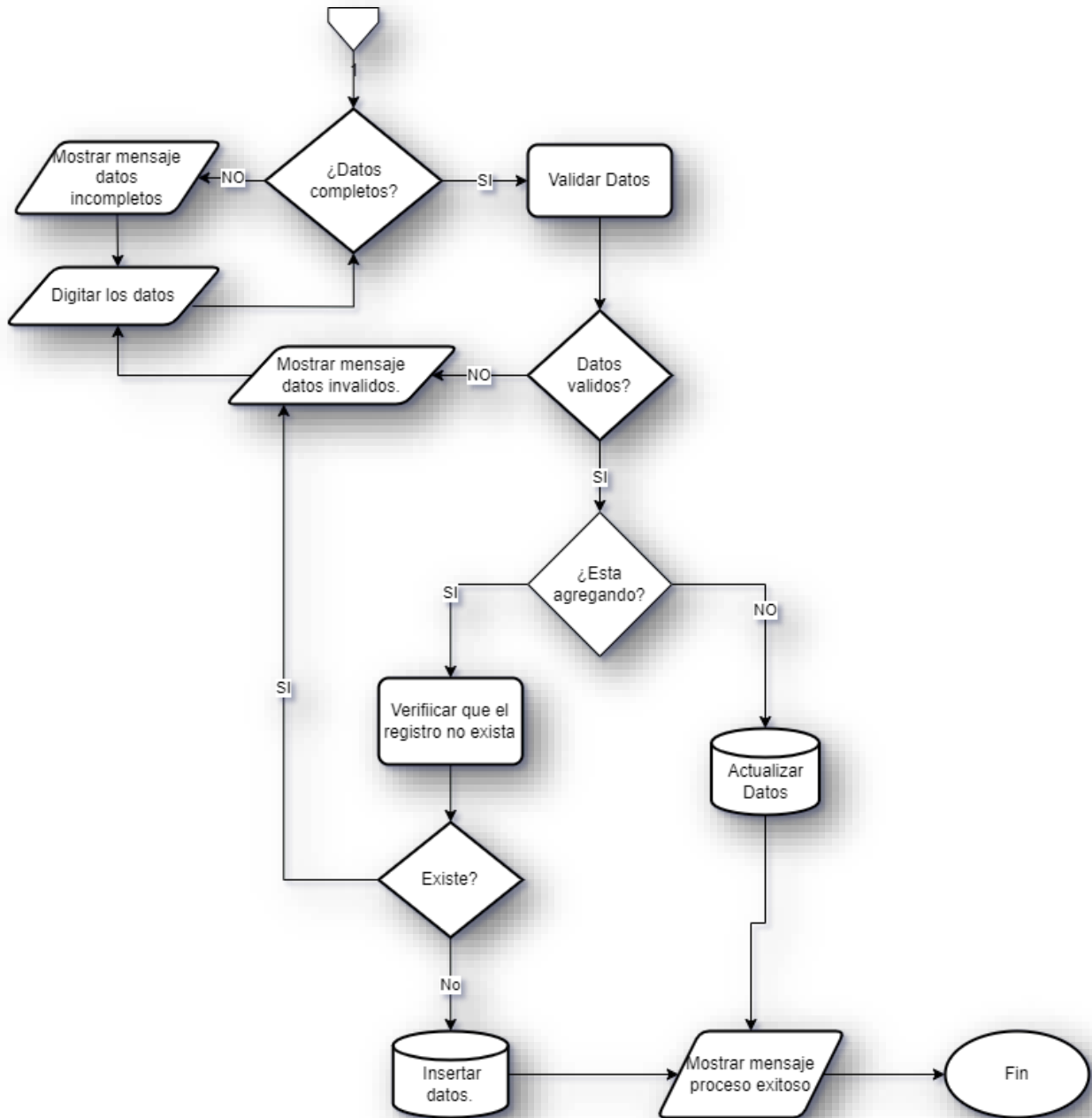


Fuente: Elaboración propia (2024).

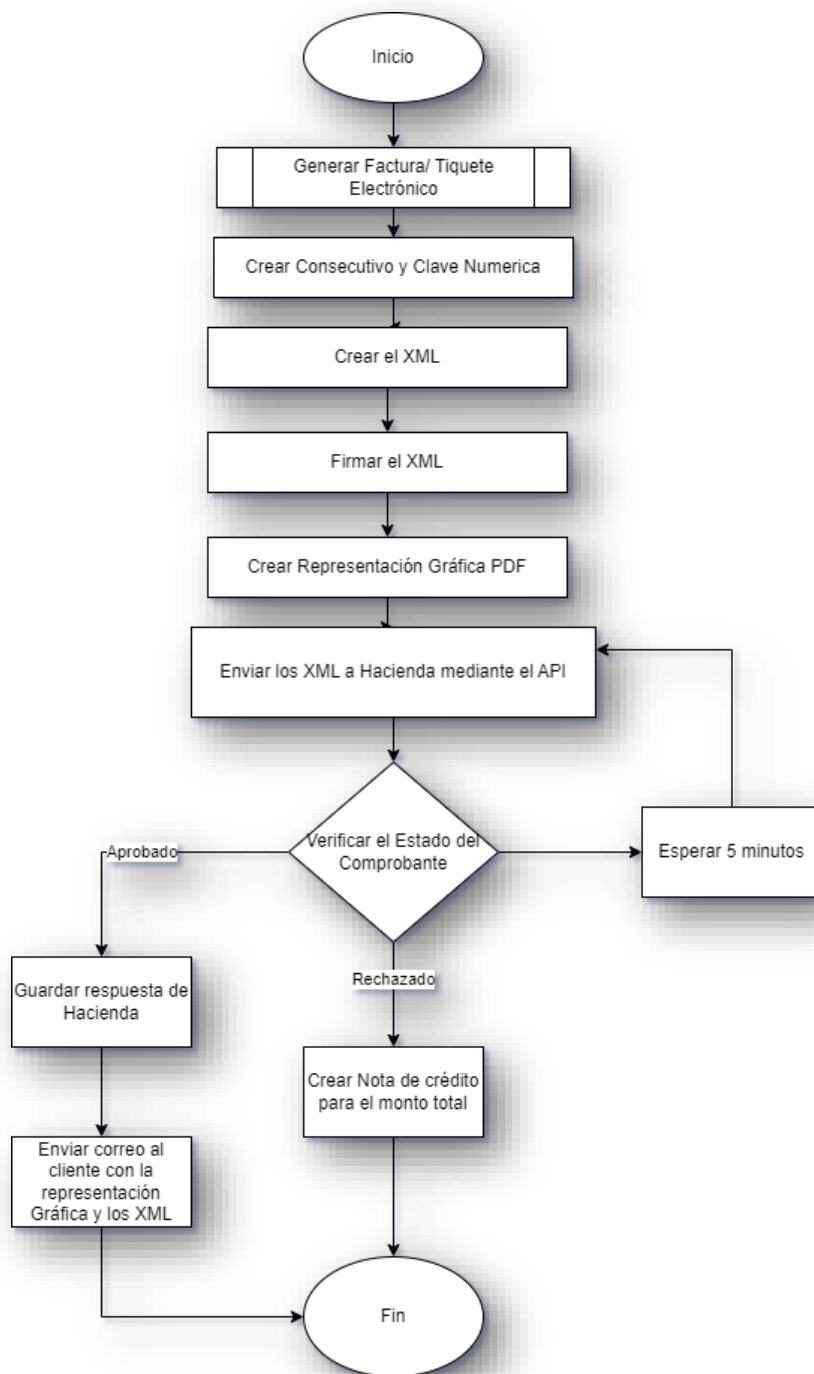
Figura 45.

Diagrama de flujo – Administrar mantenimientos





Fuente: Elaboración propia (2024).

Figura 46.*Diagrama de flujo – Generación comprobantes electrónicos*

Fuente: Elaboración propia (2024).

Diseño De Salidas

En esta sección se presentan los diseños de archivos, pantallas y reportes utilizados para la salida de la información. El objetivo es proporcionar una visión clara y estructurada de cómo se presenta la información generada por el sistema.

Los diseños de salidas son fundamentales para garantizar que los datos se entreguen de manera comprensible y útil para los usuarios finales. A continuación, se incluyen al menos cinco formatos representativos de salida para los datos:

Figura 47.


Representación gráfica factura de venta

Artículo	Descripción	CABys	Cant	Precio	IVA
07PRS610G	Corse De Lona Lumboacro	4817105030100	2.00	21,452.00	0.00
RP/13	Adaptación	8721000000000	2.00	8,000.00	1,872.00
DIFERENCIA	Diferencia - Pago De Diferencia De Precio	4817203990000	1.00	15,700.00	0.00

Observacion:	Subtotal:	€74,604.00
	Descuento:	€9,605.60
	IVA:	€1,872.00
	Total:	€66,870.40
	Sesenta y Seis Mil Ochocientos Setenta con Cuarenta	



Autorizado mediante la resolución DGT-R-033-2019 del veinte de junio de dos mil diecinueve de la Dirección General de Tributación.

Número de página: 1



Fuente: Elaboración propia (2024).

Figura 48.*Representación gráfica nota de crédito*

		Comprobante Electrónico V. 4.3 50619062400011395018000100001030000000024100000024			
Identificación: 1-1395-0180 Dirección: 450 Mtrs Sur de la escuela José Cubero Muñoz Teléfono: (506) - 85278762 Fax: (506) - 22562668 Correo: daniel13pz@gmail.com		Nota de Crédito N°: 00100001030000000024 Fecha y hora de emisión: 19/06/2024 11:29:56 Fecha de vencimiento: 19/06/2024 Actividad económica: 331103			
Cliente: Ayarco Sur Dos Sociedad Anonima Identificación: 3-101-204721 Dirección: Dirección Telefono: (506) - 24416038 Correo:		Condición de Venta: CRÉDITO Días de Crédito: 30 Medio de Pago: Efectivo Código de Moneda: COL			
Artículo	Descripción	CABys	Cant	Precio	IVA
07PRS610G	Corse De Lona Lumbosacro	4817105030100	2.00	21,452.00	0.00
RP/13	Adaptación	8721000000000	2.00	8,000.00	1,872.00
DIFERENCIA	Diferencia - Pago De Diferencia De Precio	4817203990000	1.00	15,700.00	0.00
Observacion: Se anula el documento completo		Subtotal:		€74,604.00	
		Descuento:		€9,605.60	
		IVA:		€1,872.00	
		Total:		€66,870.40	
		Sesenta y Seis Mil Ochocientos Setenta con Cuarenta			
Autorizado mediante la resolución DGT-R-033-2019 del veinte de junio de dos mil diecinueve de la Dirección General de Tributación.					
Número de página: 1					

Fuente: Elaboración propia (2024).

Figura 49.*Correo electrónico generado*

Zapatitos Lindos S.A Factura Electrónica-
00100001010000000070 Σ Recibidos x

Documentos Electrónicos - Zapatitos Lindos S.A <daniel13p... 23:27 (hace 2 minutos) ☆ 😊 ↶ ⋮
para mí ▼


DOCUMENTO ELECTRÓNICO 4.3 Zapatitos Lindos S.A

Tipo:	Factura Electrónica
Consecutivo número:	00100001010000000070
Clave numérica:	50619062400011395018000100001010000000070100000070
Estado:	Aceptado

DATOS EMISOR

Nombre	Zapatitos Lindos S.A
Identificación	1-1395-0180
Teléfono	(506) - 85278762
Email	daniel13pz@gmail.com

3 archivos adjuntos • Analizado por Gmail ⓘ ↓ 🗑️



PDF Factura Elect...

< />

Factura Electróni...

< />

MH-0010000101...

↶ Responder
↷ Reenviar
😊

Fuente: Elaboración propia (2024).

Figura 50.

XML del documento electrónico

```

<?xml version="1.0" encoding="utf-8"?>
<FacturaElectronica xmlns:ds="http://www.w3.org/2000/09/xmldsig#" xmlns:xsd=
"http://www.w3.org/2001/XMLSchema" xmlns:xsi=
"http://www.w3.org/2001/XMLSchema-instance" xmlns=
"https://cdn.comprobanteselectronicos.go.cr/xml-schemas/v4.3/facturaElectronica">
  <Clave>50619062400011395018000100001010000000070100000070</Clave>
  <CodigoActividad>331103</CodigoActividad>
  <NumeroConsecutivo>00100001010000000070</NumeroConsecutivo>
  <FechaEmision>2024-06-19T23:27:14.1086559-06:00</FechaEmision>
  <Emisor>
    <Nombre>Daniel David Quesada Fallas</Nombre>
    <Identificacion>
      <Tipo>01</Tipo>
      <Numero>113950180</Numero>
    </Identificacion>
    <NombreComercial>Zapatitos Lindos S.A</NombreComercial>
    <Ubicacion>
      <Provincia>1</Provincia>
      <Canton>08</Canton>
      <Distrito>04</Distrito>
      <Barrio>06</Barrio>
      <OtrasSenas> 450 Mtrs Sur de la escuela Jose Cubero Munoz </OtrasSenas>
    </Ubicacion>
    <Telefono>
      <CodigoPais>506</CodigoPais>
      <NumTelefono>85278762</NumTelefono>
    </Telefono>
    <Fax>
      <CodigoPais>506</CodigoPais>
      <NumTelefono>22562668</NumTelefono>
    </Fax>
    <CorreoElectronico>daniell13pz@gmail.com</CorreoElectronico>
  </Emisor>
  <Receptor>
    <Nombre>Ayaroo Sur Dos Sociedad Anonima</Nombre>
    <Identificacion>
      <Tipo>02</Tipo>
      <Numero>3101204721</Numero>
    </Identificacion>
    <Ubicacion>
      <Provincia>1</Provincia>
      <Canton>01</Canton>
      <Distrito>01</Distrito>
      <Barrio>01</Barrio>
      <OtrasSenas>Direccion</OtrasSenas>
    </Ubicacion>
    <Telefono>
      <CodigoPais>506</CodigoPais>
      <NumTelefono>24416038</NumTelefono>
    </Telefono>
    <CorreoElectronico/>
  </Receptor>
  <CondicionVenta>02</CondicionVenta>
  <PlazoCredito>30</PlazoCredito>
  <MedioPago>01</MedioPago>
  <DetalleServicio>
    <LineaDetalle>
      <NumeroLinea>1</NumeroLinea>
      <Codigo>4817105030100</Codigo>
      <CodigoComercial>
        <Tipo>04</Tipo>
        <Codigo>07PRS610G</Codigo>
      </CodigoComercial>
      <Cantidad>2</Cantidad>
      <UnidadMedida>Unid</UnidadMedida>
      <Detalle>Corse De Lona Lumbosacro</Detalle>
      <PrecioUnitario>21452</PrecioUnitario>
      <MontoTotal>42904</MontoTotal>
      <Descuento>
        <MontoDescuento>6435.6</MontoDescuento>
        <NaturalezaDescuento>Descuento de la linea 15% </NaturalezaDescuento>
      </Descuento>
    </LineaDetalle>
  </DetalleServicio>

```

Fuente: Elaboración propia (2024).

Figura 52.*Reporte*

Reporte Comprobación General

Contabilidad / Reporte Comprobación General

Filtros

Año: 2024 Mes: Nivel: 6 Nivel: 2 Generar

Reporte

Imprimir PDF Excel

Cuenta	Descripción	Debe	Haber
1	ACTIVO	145805.2	145805.2
1-1	ACTIVO CORRIENTE	145805.2	145805.2
2	PASIVO	3848	3848
2-1	PASIVO CORRIENTE	3848	3848
4	INGRESOS	146058.8	146058.800000000002
4-1	INGRESOS DEDUCIBLES	146058.8	146058.800000000002
6	COSTOS	21000	21000
6-1	COSTO MERCADERIA VENDIDA	21000	21000
	Total DEBE:	316712.00	Total HABER: 316712.00

Fuente: Elaboración propia (2024).

Los reportes tienen su salida en pantalla y se podrán Imprimir, Exportar a PDF o bien generar un documento en Excel.

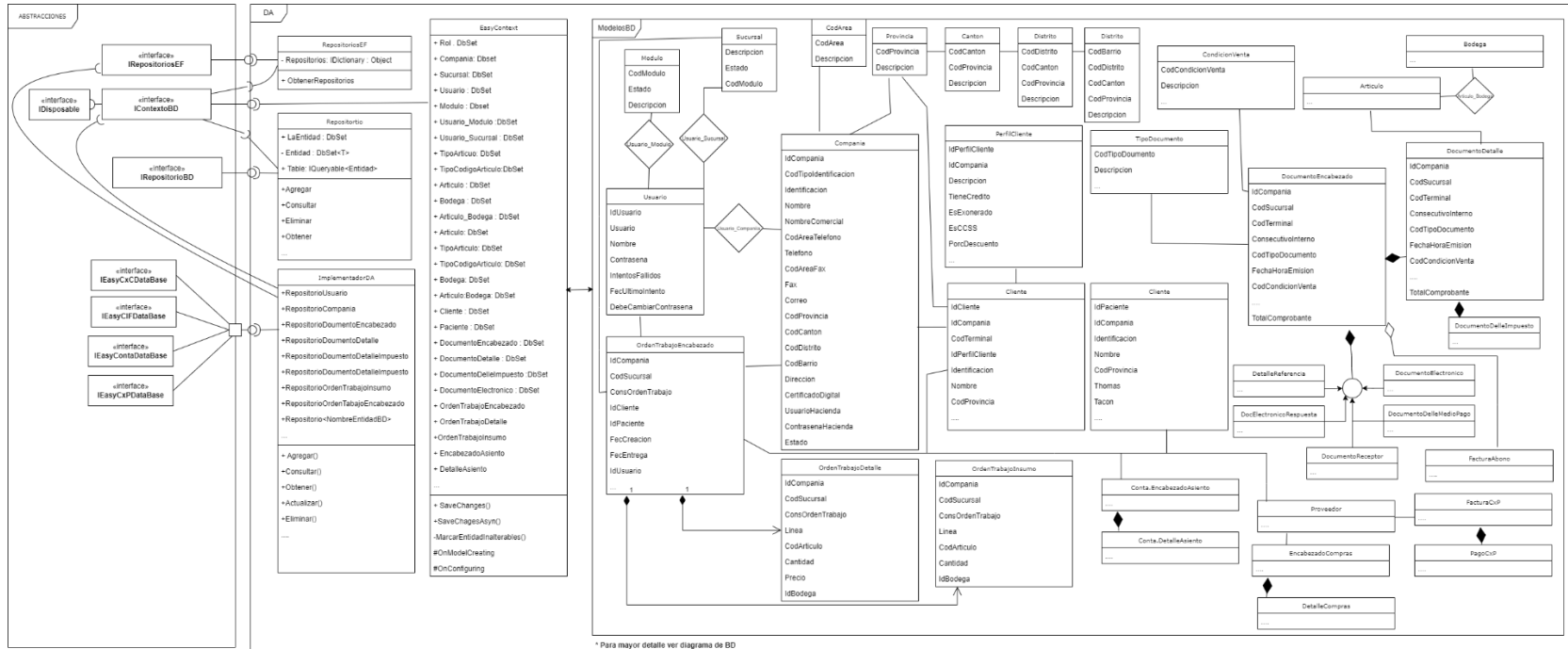
Diagramas UML

En el contexto de la programación orientada a objetos, la utilización de diagramas UML (Unified Modeling Language) es esencial para la planificación, visualización y documentación del diseño del sistema. Los diagramas UML proporcionan una representación gráfica que facilita la comprensión de la estructura y el comportamiento del sistema, así como la interacción entre sus componentes.

A continuación, se muestra el diagrama de clases el cual se fragmento para una mayor comprensión, el objetivo de este diagrama es mostrar a nivel general la estructura del prototipo funcional en cuanto a la capa abstracciones que contiene las interfaces con los contratos y los objetos.

Figura 53.

Diagrama de Clases DA

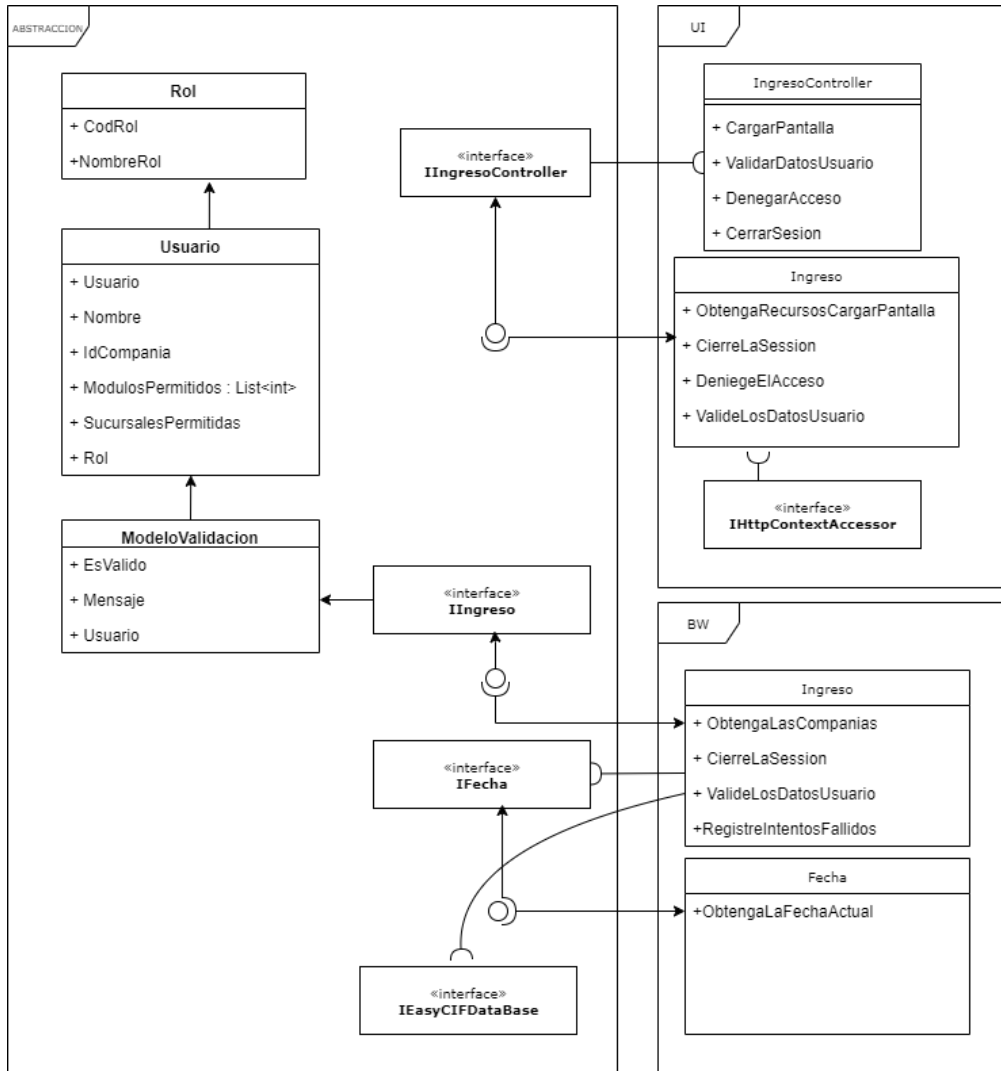


Fuente: Elaboración Propia (2024).

Este diagrama ejemplifica de manera general la estructura y el comportamiento de la capa de acceso a datos. Las interfaces con los contratos a implementar se encuentran en la capa de abstracciones, utilizando inyección de dependencias. También se puede observar la implementación del patrón repositorio para realizar operaciones CRUD a través del objeto Repositorio instanciado. A nivel de modelos, se incluyen los modelos de las entidades principales como ejemplo. Esto es necesario debido a la cantidad de tablas y maestros en la base de datos, asegurando que los modelos de entidad coincidan con el diagrama de la base de datos.

Figura 54.

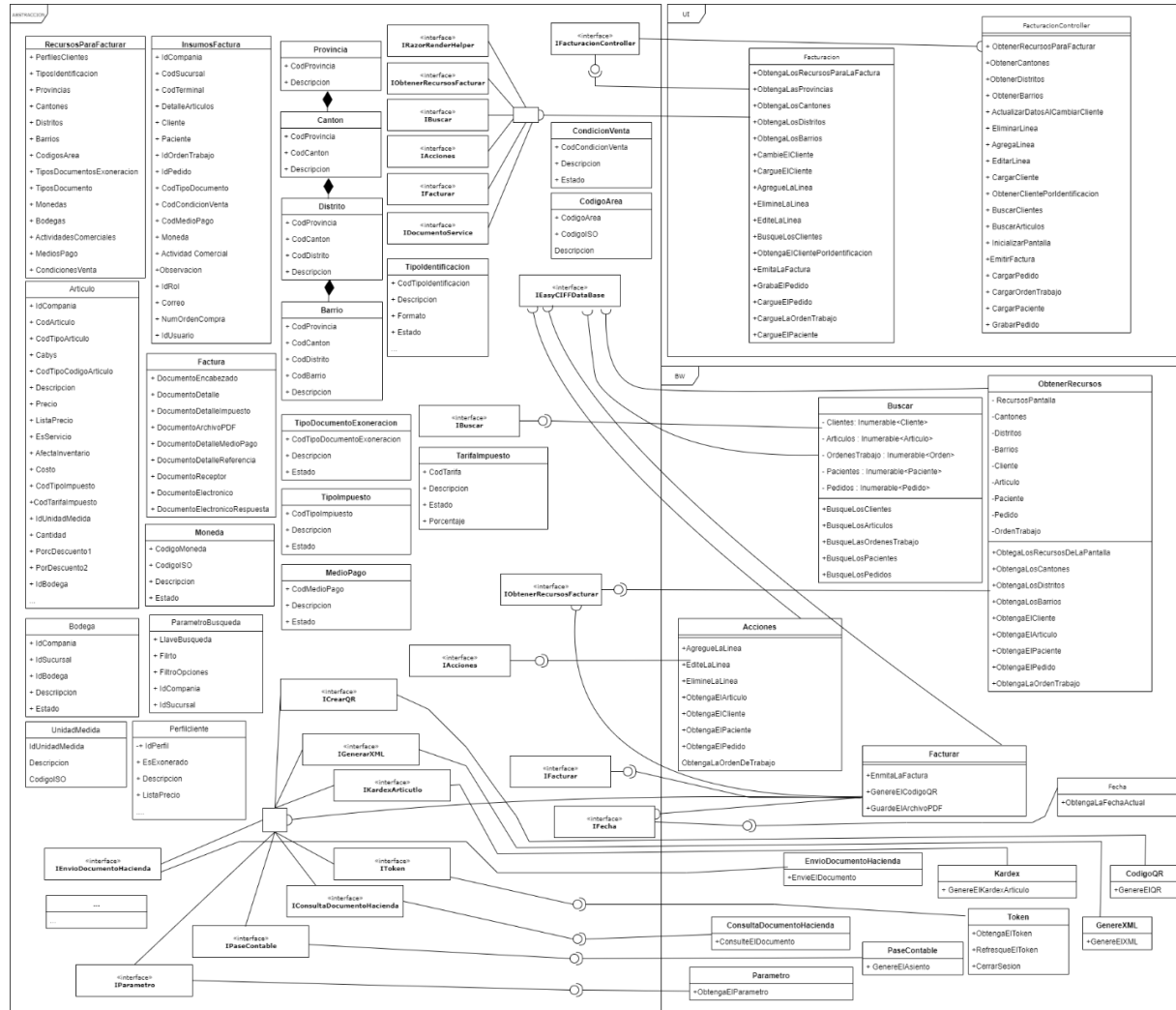
Diagrama de clases ingreso al sistema



Fuente: Elaboración propia (2024).

Figura 55.

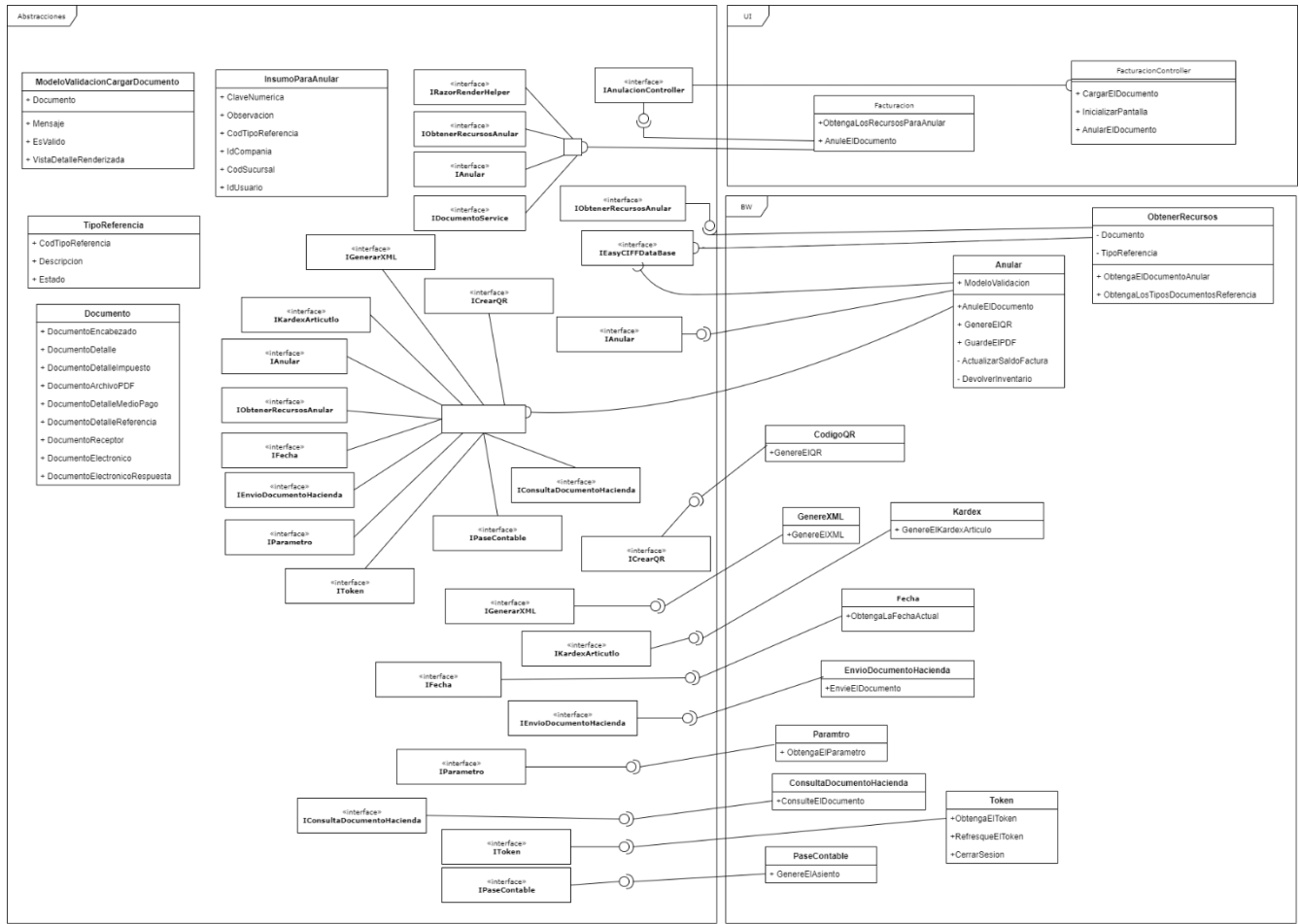
Diagrama de clases proceso facturación



Fuente: Elaboración propia (2024).

Figura 56.

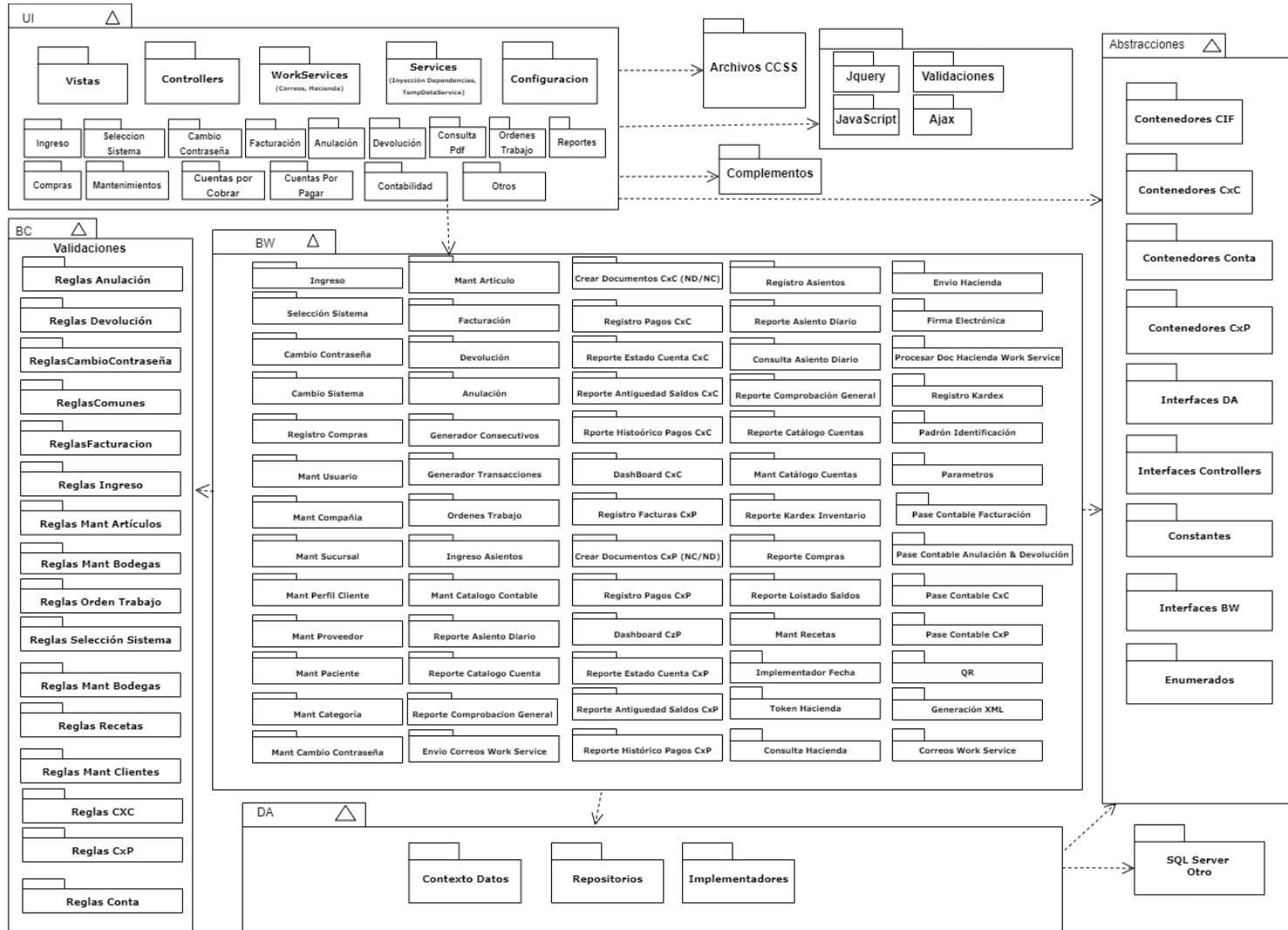
Diagrama de clases proceso Anulación



Fuente: Elaboración propia (2024).

Figura 57.

Diagrama de paquetes



Fuente: Elaboración propia (2024).

Programación

En esta sección se presentarán pequeños ejemplos del código utilizado en el desarrollo del prototipo funcional para la gestión de ventas de la empresa Zapatitos Lindos S.A. Estos ejemplos abarcarán diferentes aspectos críticos del sistema, incluyendo entradas, salidas, procesos y validaciones. Al proporcionar estos ejemplos, se busca ofrecer una visión clara y detallada de cómo se ha implementado cada componente del prototipo, asegurando así su correcto funcionamiento y efectividad en la gestión de ventas. De esta manera, se demostrará la funcionalidad y la robustez del sistema diseñado para Zapatitos Lindos S.A.

Entradas y salidas

A continuación, se presentan segmentos del código fuente utilizado en el desarrollo del prototipo de este proyecto. En estos ejemplos, se podrán visualizar las partes correspondientes a las entradas y salidas del sistema, demostrando cómo se gestionan y procesan los datos en el prototipo.

Para estructurar cada uno de los módulos propuestos en el alcance del prototipo, se crearon cuatro Layouts en el proyecto de Interfaz Gráfica en ASP.NET MVC. Estos Layouts actúan como contenedores para todas las vistas, proporcionando una estructura consistente y organizada para el diseño de la aplicación:

1. **_Layout:** Este contenedor permite estructurar y organizar las vistas de los siguientes módulos: generar Facturación, anular facturación, generar ordenes de trabajo, la seguridad y los mantenimientos relacionados con estos módulos.
2. **_Layout_CxC:** Permite estructurar y organizar las vistas del módulo gestión de las cuentas por cobrar.
3. **_Layout_CxP:** Permite estructurar y organizar las vistas del módulo gestión de las cuentas por pagar.
4. **_Layout_Conta:** Permite estructurar y organizar las vistas del módulo encargado de controlar los ingresos y gastos.

A continuación, se muestra las figuras que representan los diseños de entrada para cada uno de los Layout:

Figura 58.

Diseño entrada estructura común de los Layout

```

<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="utf-8" />
  <meta name="viewport" content="width=device-width, initial-scale=1.0" />
  <title>@ViewData["Title"] - EasyCxC</title>
  @await RenderSectionAsync("Estilos", required: false)
  <!-- Toastr -->
  <link rel="stylesheet" href="~/plugins/toastr/toastr.min.css">
  <!-- Google Font: Source Sans Pro -->
  <link rel="stylesheet" href="~/css/Fonts/fonts.css" as="font" crossorigin="anonymous">
  <link rel="stylesheet" href="~/css/Fonts/fonts@lyphicons.css">
  <!-- Font Awesome -->
  <link rel="stylesheet" href="~/plugins/fontawesome-free/css/all.min.css">
  <!-- Tempusdominus Bootstrap 4 -->
  <link rel="stylesheet" href="~/plugins/tempusdominus-bootstrap-4/css/tempusdominus-bootstrap-4.min.css">
  <!-- Theme style -->
  <link rel="stylesheet" href="~/dist/css/adminlte.css">
  <!-- overlayScrollbars -->
  <link rel="stylesheet" href="~/plugins/overlayScrollbars/css/OverlayScrollbars.min.css">
  <link rel="stylesheet" href="~/css/site.css">
  <!-- Ionicons -->
  <link rel="stylesheet" href="https://code.ionicframework.com/ionicons/2.0.1/css/ionicons.min.css">
</head>
<body class="hold-transition sidebar-mini layout-fixed text-sm sidebar-mini-md sidebar-mini-xs">
  <div class="wrapper">
    <!-- Preloader -->
    <div class="preloader flex-column justify-content-center align-items-center"></div>
    <div id="overlay" class="loadingOverlay"></div>
    <!-- Navbar -->
    <nav class="main-header navbar navbar-expand navbar-white navbar-light dropdown-legacy"></nav>
    <!-- /.navbar -->
    <!-- Main Sidebar Container -->
    <aside class="main-sidebar sidebar-dark-primary elevation-4"></aside>
    <!-- Content Wrapper. Contains page content -->
    <div class="content-wrapper">
      @RenderBody()
    </div>
    <!-- /.content-wrapper -->
    <footer class="main-footer text-sm">
      <strong>Copyright &copy; @DateTime.Now.Year <a href="#">Easy CxC</a></strong>
      All rights reserved.
      <div class="float-right d-none d-sm-inline-block">
        <b>Version /> 1.1.0
      </div>
    </footer>
  </div>
  <!-- jQuery -->
  <script src="~/plugins/jquery/jquery.min.js"></script>
  <!-- jQuery UI 1.11.4 -->
  <script src="~/plugins/jquery-ui/jquery-ui.min.js"></script>
  <!-- Resolve conflict in jQuery UI tooltip with Bootstrap tooltip -->
  <script></script>
  <!-- Bootstrap 4 -->
  <script src="~/plugins/bootstrap/js/bootstrap.bundle.min.js"></script>
  <!-- Tempusdominus Bootstrap 4 -->
  <script src="~/plugins/moment/moment.min.js"></script>
  <script src="~/plugins/tempusdominus-bootstrap-4/js/tempusdominus-bootstrap-4.min.js"></script>
  <!-- overlayScrollbars -->
  <script src="~/plugins/overlayScrollbars/js/jquery.overlayScrollbars.min.js"></script>
  <!-- AdminLTE app -->
  <script src="~/dist/js/adminlte.js"></script>
  <script src="~/js/site.js" asp-append-version="true"></script>
  <!-- Toastr -->
  <script src="~/plugins/toastr/toastr.min.js"></script>
  @await RenderSectionAsync("Scripts", required: false)
</body>
</html>

```

Fuente: Elaboración propia (2024).

En el encabezado de los Layouts se incluye todos los estilos CSS compartidos por las vistas, asegurando así un diseño uniforme en toda la aplicación. En la sección *body* de la página HTML, se encuentran los elementos *navbar* que contienen los menús, la sección *content-wrapper* o contenedor donde se realizará la renderización de las vistas del módulo, un *footer* o pie de página, y la referencia a los scripts base.

Los cuatro layouts del prototipo comparten la misma estructura; la única diferencia radica en la sección del menú, donde se cargan las opciones específicas de cada módulo gestionado.

Figura 59.

Diseño de entrada menú _Layout principal.

```

<ul class="nav nav-treeview">
  <li class="nav-item">
    <a href="#" class="nav-link">
      <i class="fas fa-copy nav-icon"></i>
      <p>
        Procesos
        <i class="right fas fa-angle-left"></i>
      </p>
    </a>
    <ul class="nav nav-treeview">
      <li class="nav-item">
        <a href="#" class="nav-link">
          <i class="fas fa-check-square nav-icon"></i>
          <p>Facturar</p>
        </a>
      </li>
      <li class="nav-item">
        <a href="#" class="nav-link">
          <i class="fas fa-check-square nav-icon"></i>
          <p>Anulación</p>
        </a>
      </li>
      <li class="nav-item">
        <a href="#" class="nav-link">
          <i class="fas fa-check-square nav-icon"></i>
          <p>Devolución</p>
        </a>
      </li>
      <li class="nav-item">
        <a href="#" class="nav-link">
          <i class="fas fa-check-square nav-icon"></i>
          <p>Consulta PDF</p>
        </a>
      </li>
      <li class="nav-item">
        <a href="#" class="nav-link">
          <i class="fas fa-check-square nav-icon"></i>
          <p>Órdenes de trabajo</p>
        </a>
      </li>
    </ul>
  </li>
</ul>
<li class="nav-item">
  <a href="#" class="nav-link">
    <i class="nav-icon fas fa-barcode"></i>
    <p>
      Inventarios
      <i class="right fas fa-angle-left"></i>
    </p>
  </a>
  <ul class="nav nav-treeview" style="display: none;">
    <li class="nav-item">
      <a href="#" class="nav-link">
        <i class="fas fa-copy nav-icon"></i>
        <p>
          Reportes
          <i class="right fas fa-angle-left"></i>
        </p>
      </a>
      <ul class="nav nav-treeview">
        <li class="nav-item">
          <a href="#" class="nav-link">
            <i class="fas fa-circle nav-icon"></i>
            <p>Kardex</p>
          </a>
        </li>
        <li class="nav-item">
          <a href="#" class="nav-link">
            <i class="fas fa-circle nav-icon"></i>
            <p>Listado de saldos</p>
          </a>
        </li>
      </ul>
    </li>
  </ul>
</li>
</ul>

```

Fuente: Elaboración propia (2024).

Los módulos de generar facturación, anular facturación y generar órdenes de trabajo están estructurados en el layout principal, el cual permite el acceso a las opciones de cada módulo. A nivel de interfaz de usuario, este módulo se denomina CIF, ya que su principal función es gestionar las compras, el inventario, la facturación, y los módulos mencionados forman parte de estas áreas.

Figura 60.

Diseño de entrada menú `_Layout_CxP`

```

<!-- Sidemenu -->
<div class="sidebar" style="margin-top:5px !important; ">
<!-- Sidebar Menu -->
<nav class="mt-2">
<ul class="nav nav-pills nav-sidebar flex-column nav-child-indent" data-widget="treeview" role="menu" data-accordion="false">
<li class="nav-item">
<a asp-controller="PrincipalCxP" asp-action="Index" class="nav-link">
<i class="nav-icon fas fa-tachometer-alt"></i>
<p>Inicio</p>
</a>
</li>
<li class="nav-item">
<a href="#" class="nav-link">
<i class="nav-icon fas fa-folder-open"></i>
<p>
CxP
<i class="right fas fa-angle-left"></i>
</p>
</a>
<ul class="nav nav-treeview">
<li class="nav-item">
<a asp-controller="RegistroFacturasCxP" asp-action="Index" class="nav-link">
<i class="far fa-circle nav-icon"></i>
<p>Registro Facturas</p>
</a>
</li>
<li class="nav-item">
<a asp-controller="RegistroDocumentosCxP" asp-action="Index" class="nav-link">
<i class="far fa-circle nav-icon"></i>
<p>Crear Documentos</p>
</a>
</li>
<li class="nav-item">
<a asp-controller="RegistroPagosCxP" asp-action="Index" class="nav-link">
<i class="far fa-circle nav-icon"></i>
<p>Registro de Pagos</p>
</a>
</li>
</ul>
</li>
<li class="nav-item">
<a href="#" class="nav-link">
<i class="nav-icon fas fa-folder"></i>
<p>
Reportes
<i class="right fas fa-angle-left"></i>
</p>
</a>
<ul class="nav nav-treeview" style="display: none; ">
<li class="nav-item">
<a asp-controller="RptEstadoCuentaCxP" asp-action="Index" class="nav-link">
<i class="far fa-circle nav-icon"></i>
<p>Estado de cuenta</p>
</a>
</li>
<li class="nav-item">
<a asp-controller="RptAntiguedadSalDOSCxP" asp-action="Index" class="nav-link">
<i class="far fa-circle nav-icon"></i>
<p>Antiguedad saldos</p>
</a>
</li>
<li class="nav-item">
<a asp-controller="RptHistoricoPagosCxP" asp-action="Index" class="nav-link">
<i class="far fa-circle nav-icon"></i>
<p>Histórico de pagos</p>
</a>
</li>
</ul>
</li>
<li class="nav-item">
<a href="#" class="nav-link">
<i class="nav-icon fas fa-gears"></i>
<p>
Mantenimientos
<i class="right fas fa-angle-left"></i>
</p>
</a>
<ul class="nav nav-treeview">
<li class="nav-item">
<a asp-controller="MantProveedor" asp-action="Index" class="nav-link">
<i class="fas fa-check-square nav-icon"></i>
<p>Proveedores</p>
</a>
</li>
</ul>
</li>
<li class="nav-item">
<a asp-controller="MantCambioDeContrasena" asp-action="CambioDeContrasenaCxP" class="nav-link">
<i class="nav-icon fas fa-key"></i>
<p>Cambiar Contraseña</p>
</a>
</li>

```

Fuente: Elaboración propia (2024).

Desde este Layout se puede ingresar a la pantalla para registrar facturas de compra o gastos que no tengan relación directa con el inventario, también se pueden crear notas de crédito y débito, el registro de pagos y los reportes del módulo.

Figura 61.

Diseño de entrada menú *_Layout_CxC*

```

<div class="sidebar" style="margin-top:5px !important;">
<!-- Sidebar Menu -->
<nav class="mt-2">
<ul class="nav nav-pills nav-sidebar flex-column nav-child-indent" data-widget="treeview" role="menu" data-accordion="false">
<li class="nav-item">
<a asp-controller="PrincipalCxC" asp-action="Index" class="nav-link">
<i class="nav-icon fas fa-tachometer-alt"/>
<p>Inicio</p>
</a>
</li>
<li class="nav-item">
<a href="#" class="nav-link">
<i class="nav-icon fas fa-folder-open"/>
<p>CxC
<i class="right fas fa-angle-left"/>
</a>
<ul class="nav nav-treeview">
<li class="nav-item">
<a asp-controller="RegistroDocumentosCxC" asp-action="Index" class="nav-link">
<i class="far fa-circle nav-icon"/>
<p>Crear Documentos</p>
</a>
</li>
<li class="nav-item">
<a asp-controller="RegistroPagosCxC" asp-action="Index" class="nav-link">
<i class="far fa-circle nav-icon"/>
<p>Registro de Pagos</p>
</a>
</li>
</ul>
</li>
<li class="nav-item">
<a href="#" class="nav-link">
<i class="nav-icon fas fa-folder"/>
<p>Reportes
<i class="right fas fa-angle-left"/>
</a>
<ul class="nav nav-treeview" style="display: none;">
<li class="nav-item">
<a asp-controller="RptEstadoCuenta" asp-action="Index" class="nav-link">
<i class="far fa-circle nav-icon"/>
<p>Estado de cuentas</p>
</a>
</li>
<li class="nav-item">
<a asp-controller="RptAntiguedadSalDOS" asp-action="Index" class="nav-link">
<i class="far fa-circle nav-icon"/>
<p>Antiguedad</p>
</a>
</li>
<li class="nav-item">
<a asp-controller="RptHistoricoPagos" asp-action="Index" class="nav-link">
<i class="far fa-circle nav-icon"/>
<p>Historial de pagos</p>
</a>
</li>
</ul>
</li>
<li class="nav-item">
<a asp-controller="MantCambioDeContrasena" asp-action="CambioDeContrasenaCxC" class="nav-link">
<i class="nav-icon fa fa-key"/>
<p>Cambiar Contraseña</p>
</a>
</li>
<li class="nav-item">
<a asp-controller="Sistema" asp-action="Index" class="nav-link">
<i class="nav-icon fas fa-right-left"/>
<p>Cambiar de sistema</p>
</a>
</li>
<li class="nav-item">
<a asp-controller="Ingreso" asp-action="Logout" class="nav-link">
<i class="nav-icon fas fa-sign-out"/>
<p>Cerrar sesión</p>
</a>
</li>

```

Fuente: Elaboración propia (2024).

Desde este Layout se pueden gestionar los procesos del módulo de cuentas por cobrar; tales como crear documento electrónicos o financieros que no tengan relación directa con un item del inventario y generar los reportes del módulo.

Figura 62.

Diseño de entrada menú `_Layout_Conta`

```

<!-- Sidebar Menu -->
<nav class="mt-2">
<ul class="nav nav-pills nav-sidebar flex-column nav-child-indent" data-widget="treeview" role="menu" data-accordion="false">
<li class="nav-item">
<a asp-controller="PrincipalConta" asp-action="Index" class="nav-link">
<i class="nav-icon fas fa-tachometer-alt"/>
<p>Inicio</p>
</li>
<li class="nav-item">
<a href="#" class="nav-link">
<i class="nav-icon fas fa-folder-open"/>
<p>Conta
<i class="right fas fa-angle-left"/>
</p>
</a>
<ul class="nav nav-treeview">
<li class="nav-item">
<a asp-controller="IngresoAsiento" asp-action="Index" class="nav-link">
<i class="far fa-circle nav-icon"/>
<p>Registro Asientos</p>
</li>
</ul>
</li>
<li class="nav-item">
<a href="#" class="nav-link">
<i class="nav-icon fas fa-folder"/>
<p>Reportes
<i class="right fas fa-angle-left"/>
</p>
</a>
<ul class="nav nav-treeview" style="display: none;">
<li class="nav-item">
<a asp-controller="RptAsientoDiario" asp-action="Index" class="nav-link">
<i class="far fa-circle nav-icon"/>
<p>Asientos de diario</p>
</li>
<li class="nav-item">
<a asp-controller="RptComprobacionGeneral" asp-action="Index" class="nav-link">
<i class="far fa-circle nav-icon"/>
<p>Comprobacion General</p>
</li>
<li class="nav-item">
<a asp-controller="RptCatalogoCuentas" asp-action="Index" class="nav-link">
<i class="far fa-circle nav-icon"/>
<p>Catálogo de cuentas</p>
</li>
</ul>
</li>
<li class="nav-item">
<a href="#" class="nav-link">
<i class="nav-icon fa fa-gears"/>
<p>Mantenimientos
<i class="right fas fa-angle-left"/>
</p>
</a>
<ul class="nav nav-treeview" style="display: none;">
<li class="nav-item">
<a asp-controller="MantCatalogoCuenta" asp-action="Agregar" class="nav-link">
<i class="far fa-circle nav-icon"/>
<p>Catalogo Cuentas</p>
</li>
</ul>
</li>
<li class="nav-item">
<a asp-controller="MantCambioDeContraseña" asp-action="CambioDeContraseñaCONTA" class="nav-link">
<i class="nav-icon fa fa-key"/>
<p>Cambiar Contraseña</p>
</li>
</ul>

```

Fuente: Elaboración propia (2024).

Desde este Layout se pueden gestionar los procesos del módulo de contabilidad, como los reportes base y el registro de asientos de diario manual.

Figura 63.

Diseño de entrada pantalla generar facturación

```

<div id="VistaPrincipal" class="view">
  <!-- Main content -->
  <section class="content">
    <div class="container-fluid">
      <!-- Form Facturar -->
      <!-- Default box -->
      <div class="card card-outline card-dark">
        <div class="card-header" style="padding-bottom: 5px !important; padding-top: 5px !important;">
          <h3 class="card-title" style="margin-top: 5px !important;">Encabezado /h3>
          <div class="card-tools">
            <button type="button" class="btn btn-tool" data-card-widget="collapse" title="Collapse">
              <i class="fas fa-minus"></i>
            </button>
          </div>
        </div>
        <div class="card-body pb-0">
          <partial name="_Encabezado" model=Model />
        </div>
      </div>
    </div>
    <!-- /card-body -->
  </div>
  <!-- /card -->
  <!-- Detalle -->
  <div class="card card-outline card-dark">
    <div class="card-header" style="padding-bottom: 5px !important; padding-top: 5px !important;">
      <h3 class="card-title" style="margin-top: 5px !important;">Detalle /h3>
      <div class="card-tools">
        <button type="button" class="btn btn-tool" data-card-widget="collapse" title="Collapse">
          <i class="fas fa-minus"></i>
        </button>
      </div>
    </div>
    <div class="card-body pt-1 pb-1">
      <div class="row pb-0">
        <div class="row p-0">
          <hr class="col-12 p-0 m-1" />
          <div class="col-md-12 p-0 dataTable_wrapper table-responsive">
            <table id="tblDetalleFactura" cellpadding="0" border="1" class="table table-bordered mt-2 tablaFacturar" style="width:100%;border-color: #333333 !important;">
              <thead>
                <tr style="color: #333333;background-color: #333333;font-weight:bold; height:15px;">
                  <th style="border: 1px solid #333333 !important; class="text-center">N° Línea</th>
                  <th style="border: 1px solid #333333 !important;">Artículo</th>
                  <th style="border: 1px solid #333333 !important;">Descripción</th>
                  <th style="border: 1px solid #333333 !important;">CABYS</th>
                  <th style="border: 1px solid #333333 !important;">Cantidad</th>
                  <th style="border: 1px solid #333333 !important;">Precio</th>
                  <th style="border: 1px solid #333333 !important;">% Desc 1</th>
                  <th style="border: 1px solid #333333 !important;">% Desc 2</th>
                  <th style="border: 1px solid #333333 !important;">IVA</th>
                  <th style="border: 1px solid #333333 !important;">Total Línea</th>
                </tr>
              </thead>
              <tbody id="tbody_DetalleArticulo">
                <partial name="_FileDetalleArticuloVacia" />
              </tbody>
            </table>
          </div>
        </div>
      </div>
    </div>
  </div>
  <!-- /card-body -->
  </div>
  <!-- /card -->
  <!-- TOTALES -->
  <div class="card card-outline card-dark">
    <div class="card-header"></div>
    <div class="card-body">
      <div class="row">
        </div>
    </div>
  </div>
  <!-- /card-body -->
  </div>
  <!-- /card -->
  <!-- BOTONES -->
  <div class="card card-outline card-secondary">
    <div class="card-body">
      <div>
        <button id="btnGuardarPedido" title="Guardar pedido" type="submit" class="btn btn-primary btn-lg" style="font-size:16px !important;" data-toggle="modal" data-target="#modal-grabar">
          Guardar
        </button>
        <button id="btnfacturar" title="Emitir la factura" type="submit" class="btn btn-warning btn-lg" style="font-size:16px !important;" data-toggle="modal" data-target="#modal-facturar">
          Facturar
        </button>
        <button id="btnCancelar" title="Cancelar" type="submit" class="btn btn-success btn-lg" style="font-size:16px !important;">
          Cancelar
        </button>
      </div>
    </div>
  </div>
  </div>

```

Fuente: Elaboración propia (2024).

La vista del proceso de facturación es una de las más extensas. Para estructurarla y organizarla de manera más efectiva, se decidió utilizar vistas parciales. Esto no solo ordena la vista, sino que también proporciona valor al permitir la actualización de secciones de la página de forma independiente, sin necesidad de refrescar toda la página.

Figura 64.

Diseño de entrada pantalla generación de orden de trabajo

```

<!-- Content Header (Page header) -->
<section class="content-header">
  <div class="container-fluid">
    <div class="row mb-2">
      <div class="col-sm-6">
        <h1></h1>
      </div>
      <div class="col-sm-6">
        <ol class="breadcrumb float-sm-right">
          <li class="breadcrumb-item"><a href="#">Procesos</a></li>
          <li class="breadcrumb-item active">Orden de Trabajo</li>
        </ol>
      </div>
    </div>
  </div><!-- /.container-fluid -->
</section>

<!-- Main content -->
<section class="content">
  <div id="lista" class="View">
    <partial name="~/Views/OrdenTrabajo/_OrdenesEnProceso.cshtml" />
  </div>
  <div id="VCrearOrden" class="View" style="display:none">
    <partial name="~/Views/OrdenTrabajo/_CrearOrden.cshtml" />
  </div>
  <div id="VBuscarOrden" class="View" style="display:none">
    <partial name="~/Views/OrdenTrabajo/_BuscarOrdenTrabajo.cshtml" />
  </div>
</div>
</section>
</form>

<!-- Modal -->
<div class="modal fade" id="modalCrearOrden">
  <div class="modal-dialog modal-sm">
    <div class="modal-content">
      <div class="modal-header bg-dark">
        <h5 class="modal-title text-light">Grabar Datos Orden De Trabajo</h5>
        <button type="button" class="close text-light" data-dismiss="modal" aria-label="Close">
          <span aria-hidden="true">&times;</span>
        </button>
      </div>
      <div class="modal-body">
        <strong class="text-md">¿Desea continuar?</strong>
      </div>
      <div class="modal-footer" style="display:block">
        <div style="float:left !important;">
          <button id="btnRegistraFacturaCompra" name="btnGrabarOrdenTrabajo" onClick='Ajax_GuardarOrdenTrabajo()' type="button" class="btn btn-success">Sí</button>
          <button type="button" class="btn btn-danger" data-dismiss="modal">No</button>
        </div>
      </div>
    </div>
  </div>
</div>
<!-- /.modal -->

<!-- Modal -->
<div class="modal fade" id="modalBuscarArticulo" tabIndex="-1" role="dialog" aria-labelledby="buscarArticulo" aria-hidden="true">
  <div class="modal-dialog modal-xl">
    <div class="modal-content">
      <div class="modal-header">
        <h5 class="modal-title" id="BuscarProveedor">Buscar Articulo</h5>
        <button type="button" class="close" data-dismiss="modal" aria-label="Close">
          <span aria-hidden="true">&times;</span>
        </button>
      </div>
      <div class="modal-body">
        <partial name="~/Views/OrdenTrabajo/_BuscarArticulos.cshtml" />
      </div>
      <div class="modal-footer">
        <div class="ml-auto mr-3">
          <button type="button" class="btn btn-danger" data-dismiss="modal"><i class="fa fa-close"></i> Cerrar</button>
        </div>
      </div>
    </div>
  </div>
</div>
</div>

```

Fuente: Elaboración propia (2024).

La vista correspondiente al módulo de gestión de órdenes de trabajo también se estructuró con vistas parciales. Durante el proceso de generación de órdenes de trabajo, se puede visualizar una lista de órdenes pendientes o en proceso de producción, las cuales pueden priorizarse mediante drag & drop. Además, es posible crear y modificar órdenes, y durante este proceso, se pueden buscar clientes, pacientes, artículos a modificar o adaptar, insumos utilizados para el proceso de adaptación, así como otra información relevante como los datos del paciente y de la receta médica.

Figura 65.

Diseño de entrada pantalla anular facturación

```

<!-- Main content -->
<section class="content">
  <div class="container-fluid">
    <form id="frmAnular">
      <!-- Default box -->
      <div class="card card-outline card-dark">
        <div class="card-header"></div>
        <div class="card-body p-0">
          <!-- Datos del Encabezado -->
          <div class="row pt-0 pb-0 mt-0 mb-2">
            <!-- Tipo Documento -->
            <div class="col-lg-5 col-md-5 col-sm-12">
              <div class="form-group-sm mb-1">
                <label class="pb-0 mb-0" for="ClaveNumerica">Clave Numérica:</label>
                <input id="ClaveNumerica" name="ClaveNumerica" placeholder="Digite la clave numérica del documento anular" type="text" class="form-control form-control-sm rounded-0">
                <span class="input-group-append">
                  <button id="btnCargarDocumento" onClick="Ajax_CargueElDocumento()" name="btnCargarDocumento" type="button" class="btn btn-warning btn-flat btn-sm"><i class="fa fa-plus-circle"></i> Cargar documento </button>
                </span>
              </div>
            </div>
            <div class="col-lg-3 col-md-3 col-sm-10"></div>
            <div class="col-lg-3 col-md-3 col-sm-2"></div>
          </div>
          <div class="row mt-1 mb-1">
            <div class="col-lg-12 col-md-12 col-sm-12">
              <text class="text-green">Datos <cite title="Source Title">Cliente:</cite></text>
              <hr class="pt-0 pb-1 mt-0 mb-0" />
            </div>
            <div class="col-lg-12 col-md-12 col-sm-12">
              <div class="row pt-0 pb-0 mt-0 mb-0">
                <div class="col-lg-4 col-md-4 col-sm-12">
                  <div class="form-group-sm mb-1">
                    <label class="pb-0 mb-0" for="TipoIdentificacion">Tipo Identificación:</label>
                    <input id="TipoIdentificacion" name="TipoIdentificacion" placeholder="Tipo Identificación" disabled type="text" class="form-control form-control-sm rounded-0">
                  </div>
                </div>
                <div class="col-lg-8 col-md-8 col-sm-12">
                  <div class="form-group-sm">
                    <label class="pb-0 mb-0" for="Identificacion">Identificación:</label>
                    <div class="input-group input-group-sm mb-1">
                      <div class="input-group-prepend">
                        <span class="input-group-text"><i class="fa fa-regular fa-id-card"></i></span>
                      </div>
                      <input id="Identificacion" name="Identificacion" placeholder="Identificación" disabled="" value="" type="text" class="form-control form-control-sm rounded-0">
                    </div>
                  </div>
                </div>
            </div>
            <div class="col-lg-8 col-md-8 col-sm-12">
              <div class="form-group-sm">
                <label class="pb-0 mb-0" for="NombreCliente">Nombre del cliente:</label>
                <div class="input-group input-group-sm mb-1">
                  <input id="NombreCliente" name="NombreCliente" disabled placeholder="Nombre del Cliente" disabled="" type="text" class="form-control form-control-sm rounded-0">
                </div>
              </div>
            </div>
          </div>
        </div>
      </div>
    </form>
  </div>
  <!-- /card-body -->
</div>
<!-- /card -->
<!-- Detalle -->
<div class="card card-outline card-dark">
  <div class="card-header"></div>
  <div class="card-body pt-1 pb-1">
    <div class="row p-0">
      <div class="col-md-12 p-0 dataTable_wrapper table-responsive">
        <table id="tblDetalleFactura" cellpadding="0" border="1" class="table table-bordered mt-2 tablaFacturar" style="width:100%;border-color: #333333 important;">
          <thead>
            <tr style="color: #333333;font-weight:bold; height:15px;">
              <th style="width:70px; border: 1px solid #333333 important; text-align:center;">N° Línea</th>
              <th style="border: 1px solid #333333 important;">Artículo</th>
              <th style="border: 1px solid #333333 important;">Descripción</th>
              <th style="border: 1px solid #333333 important;">CABys</th>
              <th style="border: 1px solid #333333 important;">Cantidad</th>
              <th style="border: 1px solid #333333 important;">Precio</th>
              <th style="border: 1px solid #333333 important;">% Desc 1</th>
              <th style="border: 1px solid #333333 important;">% Desc 2</th>
              <th style="border: 1px solid #333333 important;">IVA</th>
              <th style="border: 1px solid #333333 important;">Total Línea</th>
            </tr>
          </thead>
          <tbody id="tbody_DetalleArticulo">
            <partial name="FilaDetalleArticuloVacia" />
          </tbody>
        </table>
      </div>
    </div>
  </div>
</div>

```

Fuente: Elaboración propia (2024).

Desde esta pantalla se pueden cargar los documentos electrónicos y se puede generar su respectiva nota de crédito electrónica.

Figura 66.

Diseño de entrada pantalla ingreso compras

```

<div id="VistaPrincipal" class="View">
  <!-- main content -->
  <section class="content">
    <div id="frmRegistrarCompra">
      <!-- Default box -->
      <div class="card card-outline card-dark">
        <div class="card-header" style="padding-bottom:5px !important;padding-top:5px !important;";>
          <h3 class="card-title" style="margin-top:0px !important;";>Encabezado</h3>
        </div>
        <div class="card-tools">
          <button type="button" class="btn btn-tool" data-card-widget="collapse" title="Collapse">
            <i class="fas fa-minus"/>
          </button>
        </div>
        </div>
        <div class="card-body pt-1 pb-1">
          <div class="row pt-0 pb-0">
            <partial name="/Views/Compras/_Encabezado.cshtml" model=Model />
          </div>
        </div>
      </div>
      <!-- /card-body -->
    </div>
    <!-- /card -->
    <!-- Details -->
    <div class="card card-outline card-dark">
      <div class="card-header" style="padding-bottom:5px !important;padding-top:5px !important;";>
        <h3 class="card-title" style="margin-top:0px !important;";>Agregar Articulos</h3>
      </div>
      <div class="card-tools">
        <button type="button" class="btn btn-tool" data-card-widget="collapse" title="Collapse">
          <i class="fas fa-minus"/>
        </button>
      </div>
      </div>
      <div class="card-body pt-1 pb-1">
        <div class="row pt-0 pb-0">
          <partial name="/Views/Compras/_Detalle.cshtml" model=Model />
        </div>
      </div>
    </div>
    <!-- /card-body -->
  </div>
  <!-- /card -->
  <!-- Details -->
  <div class="card card-outline card-dark">
    <div class="card-header" style="padding-bottom:5px !important;padding-top:5px !important;";>
      <h3 class="card-title" style="margin-top:0px !important;";>Detalle</h3>
    </div>
    <div class="card-tools">
      <button type="button" class="btn btn-tool" data-card-widget="collapse" title="Collapse">
        <i class="fas fa-minus"/>
      </button>
    </div>
    </div>
    <div class="card-body pt-1 pb-1">
      <div class="col-md-12 p-0 dataTable_wrapper table-responsive">
        <table id="tblDetalleFactura" cellpadding="0" border="1" class="table table-bordered mt-2 tablaFacturar" style="width:100%;border-color:#333333 !important;";>
          <thead>
            <tr style="color:#333333;font-weight:bold; height:15px;";>
              <th style="width:70px; border: 1px solid #333333 !important;";>N° Línea</th>
              <th style="border: 1px solid #333333 !important;";>Artículo</th>
              <th style="border: 1px solid #333333 !important;";>Descripción</th>
              <th style="border: 1px solid #333333 !important;";>Cantidad</th>
              <th style="border: 1px solid #333333 !important;";>Costo</th>
              <th style="border: 1px solid #333333 !important;";>% Desc</th>
              <th style="border: 1px solid #333333 !important;";>Día</th>
              <th style="border: 1px solid #333333 !important;";>Total Línea</th>
              <th style="border: 1px solid #333333 !important;";>Bodega</th>
            </tr>
          </thead>
          <tbody id="tbody_DetalleArticulo">
            <partial name="/Views/Compras/_FilasDetalleArticuloVacia.cshtml" />
          </tbody>
        </table>
      </div>
    </div>
  </div>
  <!-- /card-body -->
</div>
<!-- /card -->

```

Fuente: Elaboración propia (2024).

La pantalla de ingreso de compras permite registrar las facturas que tienen relación con directa con el inventario.

Figura 68.

Diseño de salida del pdf documento electrónico

```

var isLastPage = i == pageCount - 1;
<section class="content">
  <div class="container-fluid">
    <div class="row">
      <div class="col-12">
        <!-- Main content -->
        <div class="invoice p-3 mb-3">
          <!-- title row -->
          <div class="row">
            <div class="col-12"></div>
          </div>
          <div class="mt-2"></div>
          <hr> <!-- Línea separadora -->
          <div class="mt-1"></div>
          <!-- Table row -->
          <div class="row mt-4" style="height:410px;">
            <div class="col-12 table-responsive">
              <table class="table table-striped">
                <thead>
                  <tr>
                    <th>Artículo</th>
                    <th>Descripción</th>
                    <th style="text-align: right;">Cant</th>
                    <th style="text-align: right;">Precio</th>
                    <th style="text-align: right;">IVA</th>
                  </tr>
                </thead>
                <tbody>
                  <!-- Table row -->
                  <!-- Línea separadora -->
                  <div style="overflow: hidden;">
                    <div style="float: left; width:60%;">
                      <div style="text-align: center; max-width:500px; font-size:12px">
                        Autorizado mediante la resolución DGT-R-033-2019 del veinte de junio de dos mil diecinueve de la Dirección General de
                        Tributación.
                      </div>
                    </div>
                    <!-- Sección derecha -->
                    <div style="float: right; margin-top:-10px">
                      
                    </div>
                    <div style="float: left; clear: both; margin-top:-15px">
                      <span> Número de página: @(i + 1)</span>
                    </div>
                  </div>
                </tbody>
              </table>
            </div>
          </div>
          <hr> <!-- Línea separadora -->
          <div class="mt-1"></div>
          <div style="overflow: hidden;">
            <div style="float: left; width:60%;">
              <div style="text-align: center; max-width:500px; font-size:12px">
                Autorizado mediante la resolución DGT-R-033-2019 del veinte de junio de dos mil diecinueve de la Dirección General de
                Tributación.
              </div>
            </div>
            <!-- Sección derecha -->
            <div style="float: right; margin-top:-10px">
              
            </div>
            <div style="float: left; clear: both; margin-top:-15px">
              <span> Número de página: @(i + 1)</span>
            </div>
          </div>
        </div>
      </div>
    </div>
  </div>
</section>

```

Fuente: Elaboración propia (2024).

La vista anterior exhibe la estructura utilizada para la generación del PDF de los documentos electrónicos. Es importante destacar que, para la generación de cualquier PDF en el sistema, se renderiza la vista y se convierte al formato correspondiente.

Figura 69.

Diseño de salida estructura documentos electrónicos

```

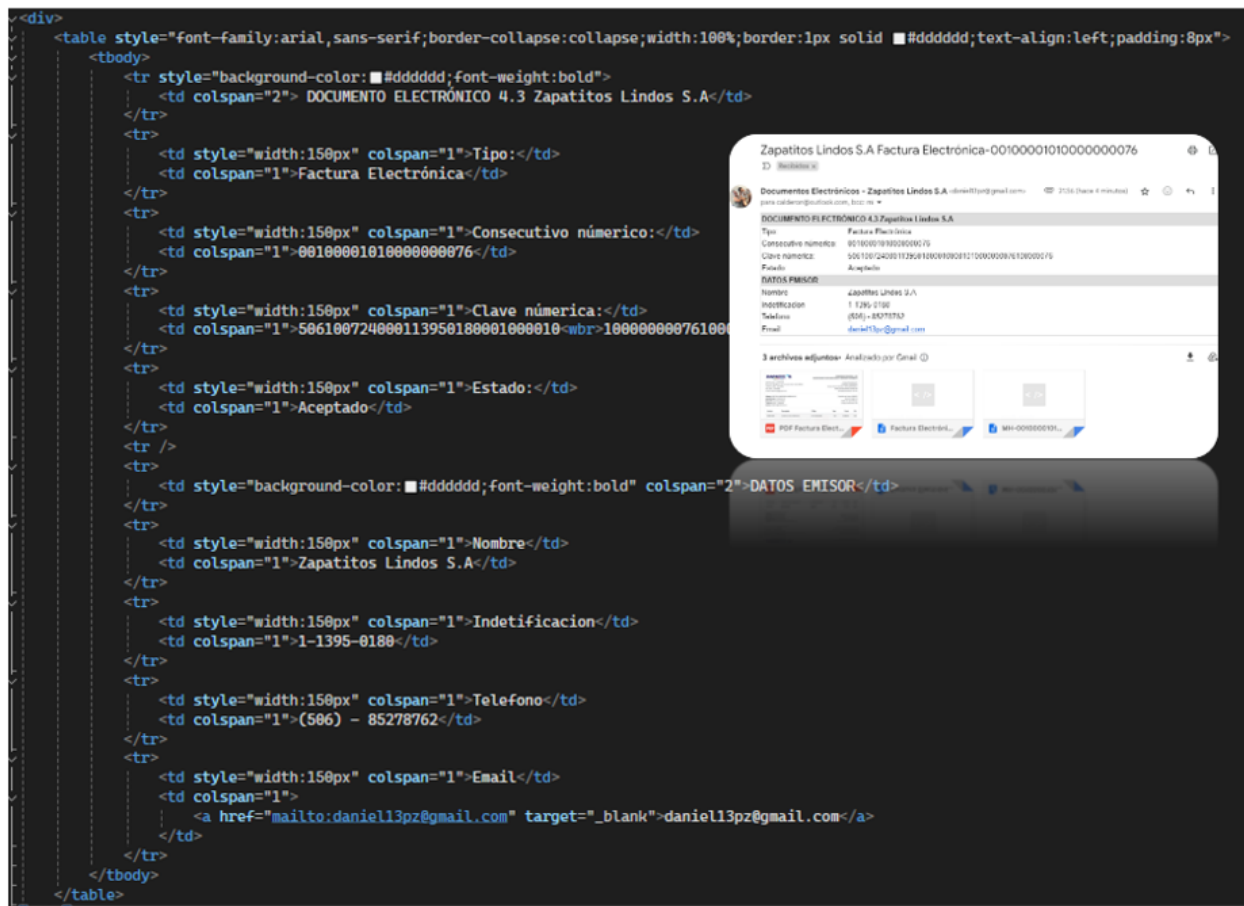
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns="https://cdn.comprobanteselectronicos.gob.cr/xml-schemas/v4.3/facturaElectronica"
xmlns:xs="http://www.w3.org/2001/XMLSchema" xmlns:vc="http://www.w3.org/2007/XMLSchema-versioning"
xmlns:ds="http://www.w3.org/2000/09/xmldsig#"
targetNamespace="https://cdn.comprobanteselectronicos.gob.cr/xml-schemas/v4.3/facturaElectronica"
elementFormDefault="qualified" attributeFormDefault="unqualified" version="4.3" vc:minVersion="1.1">
  <xs:element name="FacturaElectronica">
    <xs:annotation>
      <xs:documentation>Elemento Raiz de la Facturación Electrónica.</xs:documentation>
    </xs:annotation>
    <xs:complexType>
      <xs:sequence>
        <xs:element name="Clave" type="ClaveType">
          <xs:annotation>
            <xs:documentation>Corresponde a la clave del comprobante. Es un campo de 50 posiciones y se tiene que utilizar para la consulta del código QR. Ver nota 1 y 4.1.</xs:documentation>
          </xs:annotation>
        </xs:element>
        <xs:element name="CodigoActividad">
          <xs:annotation>
            <xs:documentation>Se debe indicar el código de la actividad económica a la cual corresponde el comprobante electrónico.</xs:documentation>
          </xs:annotation>
          <xs:simpleType>
            <xs:restriction base="xs:string">
              <xs:maxLength value="6"/>
              <xs:minLength value="6"/>
            </xs:restriction>
          </xs:simpleType>
        </xs:element>
        <xs:element name="NumeroConsecutivo" type="NumeroConsecutivoType">
          <xs:annotation>
            <xs:documentation>Numeración consecutiva del comprobante.</xs:documentation>
          </xs:annotation>
        </xs:element>
        <xs:element name="FechaEmision" type="xs:dateTime"/>
        <xs:element name="Emisor" type="EmisorType">
          <xs:annotation>
            <xs:documentation>Emisor del documento.</xs:documentation>
          </xs:annotation>
        </xs:element>
        <xs:element name="Receptor" type="ReceptorType">
          <xs:annotation>
            <xs:documentation>Receptor del documento.</xs:documentation>
          </xs:annotation>
        </xs:element>
        <xs:element name="CondicionesVenta">
          <xs:annotation>
            <xs:documentation>Condiciones de la venta: 01 Contado, 02 Crédito, 03 Consignación, 04 Apartado, 05 Arrendamiento con opción de compra, 06 Arrendamiento en función financiera, 07 Cobro a favor de
          </xs:annotation>
          <xs:simpleType>
            <xs:restriction base="xs:string">
              <xs:maxLength value="10"/>
            </xs:restriction>
          </xs:simpleType>
        </xs:element>
        <xs:element name="PlazoCredito" minOccurs="0">
          <xs:annotation>
            <xs:documentation>Plazo del crédito, es obligatorio cuando la venta del producto o prestación del servicio sea a crédito.</xs:documentation>
          </xs:annotation>
          <xs:simpleType>
            <xs:restriction base="xs:string">
              <xs:maxLength value="10"/>
            </xs:restriction>
          </xs:simpleType>
        </xs:element>
        <xs:element name="Redondeo" maxOccurs="1" minOccurs="0" type="xs:float"/>
        <xs:element name="DetalleServicio" minOccurs="0">
          <xs:annotation>
            <xs:documentation>Detalle de la mercancía o servicio prestado.</xs:documentation>
          </xs:annotation>
        </xs:element>
        <xs:complexType>
          <xs:sequence>
            <xs:element name="LineaDetalle" maxOccurs="1000">
              <xs:annotation>
                <xs:documentation>Cada línea del detalle de la mercancía o servicio prestado.</xs:documentation>
              </xs:annotation>
              <xs:complexType>
                <xs:sequence>
                  <xs:element name="NumeroLinea" type="xs:positiveInteger"/>
                  <xs:element name="Codigo">
                    <xs:annotation>
                      <xs:documentation>Código de la mercancía o servicio.</xs:documentation>
                    </xs:annotation>
                    <xs:simpleType>
                      <xs:restriction base="xs:string">
                        <xs:maxLength value="5"/>
                      </xs:restriction>
                    </xs:simpleType>
                  <xs:element name="CodigoComercial" type="CodigoType" minOccurs="0" maxOccurs="5"/>
                  <xs:element name="Cantidad">
                    <xs:annotation>
                      <xs:documentation>Cantidad de la mercancía o servicio.</xs:documentation>
                    </xs:annotation>
                    <xs:simpleType>
                      <xs:restriction base="xs:float">
                        <xs:totalDigits value="10"/>
                      </xs:restriction>
                    </xs:simpleType>
                  <xs:element name="UnidadMedida" type="UnidadMedidaType"/>
                  <xs:element name="UnidadMedidaCom" minOccurs="0" type="UnidadMedidaType"/>
                  <xs:element name="Detalle">
                    <xs:annotation>
                      <xs:documentation>Detalle de la mercancía transferida o servicio prestado.</xs:documentation>
                    </xs:annotation>
                    <xs:simpleType>
                      <xs:restriction base="xs:string">
                        <xs:maxLength value="1000"/>
                      </xs:restriction>
                    </xs:simpleType>
                  <xs:element name="Valor" type="xs:float"/>
                </xs:sequence>
              </xs:complexType>
            </xs:element>
          </xs:sequence>
        </xs:complexType>
      </xs:sequence>
    </xs:complexType>
  </xs:element>

```

Fuente: <https://atv.hacienda.gob.cr/> (2024).

El diseño de salida de los documentos electrónicos en formato XML se basa en los esquemas XSD publicados en el sitio oficial del Ministerio de Hacienda. Cada uno de estos esquemas incorpora la estructura y las validaciones necesarias para garantizar el cumplimiento de la normativa vigente.

Figura 70.

Diseño de salida correos electrónicos


```

<div>
<table style="font-family:arial,sans-serif;border-collapse:collapse;width:100%;border:1px solid #dddddd;text-align:left;padding:8px">
<tbody>
<tr style="background-color:#dddddd;font-weight:bold">
<td colspan="2">DOCUMENTO ELECTRONICO 4.3 Zapatos Lindos S.A</td>
</tr>
<tr>
<td style="width:150px" colspan="1">Tipo:</td>
<td colspan="1">Factura Electrónica</td>
</tr>
<tr>
<td style="width:150px" colspan="1">Consecutivo número:</td>
<td colspan="1">00100001010000000076</td>
</tr>
<tr>
<td style="width:150px" colspan="1">Clave numérica:</td>
<td colspan="1">506100724000113950180001000010-wbr>1000000076100</td>
</tr>
<tr>
<td style="width:150px" colspan="1">Estado:</td>
<td colspan="1">Aceptado</td>
</tr>
<tr />
<tr>
<td style="background-color:#dddddd;font-weight:bold" colspan="2">DATOS EMISOR</td>
</tr>
<tr>
<td style="width:150px" colspan="1">Nombre</td>
<td colspan="1">Zapatos Lindos S.A</td>
</tr>
<tr>
<td style="width:150px" colspan="1">Indetificacion</td>
<td colspan="1">1-1395-0180</td>
</tr>
<tr>
<td style="width:150px" colspan="1">Telefono</td>
<td colspan="1">(506) - 85278762</td>
</tr>
<tr>
<td style="width:150px" colspan="1">Email</td>
<td colspan="1">
<a href="mailto:daniel13pz@gmail.com" target="_blank">daniel13pz@gmail.com</a>
</td>
</tr>
</tbody>
</table>

```

The image shows the HTML source code of an email on the left and a preview of the email on the right. The preview shows an email from 'Zapatos Lindos S.A' with the subject 'DOCUMENTO ELECTRONICO 4.3 Zapatos Lindos S.A'. The email content includes fields for 'Tipo' (Factura Electrónica), 'Consecutivo numerico' (00100001010000000076), 'Clave numerica' (506100724000113950180001000010-wbr>1000000076100), 'Estado' (Aceptado), and 'DATOS EMISOR' (Nombre: Zapatos Lindos S.A, Indetificacion: 1-1395-0180, Telefono: (506) - 85278762, Email: daniel13pz@gmail.com). There are also three attachments: PDF Factura Elect, Factura Electrónica, and WH-0000000101.

Fuente: Elaboración propia (2024).

El diseño de los correos electrónicos enviados desde el sistema de facturación electrónica sigue un formato estructurado tipo tabla, el cual presenta de manera organizada la información del documento electrónico y del emisor.

Figura 71.

Diseño de salida reporte comprobación general del módulo de conta

```

<!-- Main content -->
<section class="content">
  <div class="container-fluid">
    <!-- Default box -->
    <div class="card card-outline card-info">
      <div class="card-header" style="padding-bottom: 5px !important; padding-top: 5px !important;">
        <h3 class="card-title" style="margin-top: 0px !important;">Filtros</h3>
        <div class="card-tools">
          <button type="button" class="btn btn-tool" data-card-widget="collapse" title="Collapse">
            <i class="fas fa-minus"></i>
          </button>
        </div>
      </div>
      <div class="card-body">
        <div class="form">
          <div class="row">
            <div class="col-sm-12 col-md-3 col-lg-3">
              <div class="form-group">
                <label for="FiltroAnno">Año</label>
                <div class="mb-1">
                  <input id="FiltroAnno" name="FiltroAnno" placeholder="Año" type="text" class="form-control form-control-sm rounded-0 numero">
                </div>
              </div>
            </div>
            <div class="col-sm-12 col-md-3 col-lg-3">
              <div class="form-group">
                <label for="FiltroMes">Mes</label>
                <div class="mb-1">
                  <input id="FiltroMes" name="FiltroMes" placeholder="Mes" type="text" class="form-control form-control-sm rounded-0 numero">
                </div>
              </div>
            </div>
            <div class="col-sm-12 col-md-3 col-lg-3">
              <div class="form-group">
                <label class="mb-1">
                  <div class="form-control form-control-sm select2" style="width: 100% !important;">
                    <select id="FiltroNivel" name="FiltroNivel" class="form-control form-control-sm select2" style="width: 100% !important;">
                      <option value="1" selected="selected">Nivel 1</option>
                      <option value="2">Nivel 2</option>
                      <option value="3">Nivel 3</option>
                      <option value="4">Nivel 4</option>
                      <option value="5">Nivel 5</option>
                    </select>
                  </div>
                </label>
              </div>
            </div>
            <div class="col-sm-12 col-md-3 col-lg-3">
              <div class="form-group">
                <div style="float: left !important; margin: 0; padding: 0; text-align: center;">
                  <button id="btnGenerarReporte" title="Generar Reporte" name="btnGenerarReporte" type="button"
                    class="btn btn-primary">
                    <i class="fa fa-folder-open"></i> Generar
                  </button>
                </div>
              </div>
            </div>
          </div>
        </div>
      </div>
    </div>
  </div>
  <!-- / card-body -->
</div>
<!-- /card -->
<div class="card card-outline card-info">
  <div class="card-body">
    <div class="row">
      <div class="col-md-12">
        <div class="form-group">
          <h3 class="resultados">small-Reporte</small></h3>
        </div>
      </div>
    </div>
    <div class="row">
      <div class="col-md-12 table-responsive p-0">
        <table id="tblComprobacion" class="table table-hover text-nowrap mt-2" style="width: 100%;">
          <thead>
            <tr>
              <th>Cuenta</th>
              <th>Descripcion</th>
              <th>Debe</th>
              <th>Haber</th>
            </tr>
          </thead>
          <tbody>
            <tr>
              <td></td>
              <td></td>
              <td></td>
              <td></td>
            </tr>
            <tr>
              <td colspan="2"></td>
              <td><strong>Total Debe: 0.00</strong></td>
              <td><strong>Total Haber: 0.00</strong></td>
            </tr>
          </tbody>
          <tfoot>
            <tr>
              <td colspan="2"></td>
              <td></td>
              <td></td>
            </tr>
          </tfoot>
        </table>
      </div>
    </div>
  </div>
</div>

```

Fuente: Elaboración propia (2024).

A manera de ejemplo, se coloca el reporte de contabilidad para la comprobación general. En general, la mayoría de los reportes comparten una estructura similar a nivel de vista, variando únicamente los procesos para obtener, listar, exportar e imprimir.

Procesos

El proyecto del Prototipo Funcional para la gestión de ventas de Zapatitos Lindos S.A. se ha estructurado en base a interfaces y clases, diseñadas para proporcionar una arquitectura modular y escalable. Estas interfaces definen los contratos y métodos necesarios, mientras que las clases correspondientes implementan la lógica específica para gestionar eficientemente cada proceso del negocio. Este enfoque no solo asegura la coherencia y la flexibilidad del sistema, sino que también facilita su integración y mantenimiento continuo conforme a las necesidades cambiantes de la empresa.

A continuación, se visualizarán algunos ejemplos de cómo se estructura el código, es de resaltar que todo el sistema se comporta de la misma manera y cuenta con la misma estructura.

Figura 72.

Proceso de facturación, controller

```
[Authorize]
[Authorize(Roles = Role.User + "," + Role.Admin)]
1 reference | pc, 39 days ago | 1 author, 13 changes
public class FacturacionController : Controller
{
    private readonly IFacturacionController _IFacturacionController;

    0 references | 0 changes | 0 authors, 0 changes
    public FacturacionController(IFacturacionController facturacionController)
    {
        _IFacturacionController = facturacionController;
    }

    [Route("Facturacion/Procesos/Index")]
    [Route("Facturacion/Procesos/Facturas")]
    0 references | pc, 48 days ago | 1 author, 2 changes
    public async Task<ActionResult> Index()
    {
        return await _IFacturacionController.ObtengaLosRecursosParaFacturar((TempDataDictionary)TempData);
    }

    [HttpPost]
    0 references | pc, 48 days ago | 1 author, 2 changes
    public JsonResult Ajax_ObtenerCantonesPorProvincia(int provincia)
    {
        return _IFacturacionController.ObtengaCantonesPorProvincia((TempDataDictionary)TempData, provincia);
    }

    [HttpPost]
    0 references | pc, 48 days ago | 1 author, 2 changes
    public JsonResult Ajax_ObtenerDistritoPorCanton([FromBody] ParametroDistritoUI distrito)
    {
        return _IFacturacionController.ObtengaDistritoPorCanton((TempDataDictionary)TempData, distrito);
    }

    [HttpPost]
    0 references | pc, 48 days ago | 1 author, 2 changes
    public JsonResult Ajax_ObtenerBarrioPorDistrito([FromBody] ParametroBarrioUI barrio)
    {
        return _IFacturacionController.ObtengaBarrioPorDistrito((TempDataDictionary)TempData, barrio);
    }

    [HttpPost]
    0 references | pc, 48 days ago | 1 author, 2 changes
    public JsonResult Ajax_ActualiceElDetalleArticulosAlCambiarCliente()
    {
        return _IFacturacionController.ActualiceElDetalleArticulosAlCambiarCliente((TempDataDictionary)TempData);
    }

    [HttpPost]
    0 references | pc, 48 days ago | 1 author, 1 change
    public JsonResult Ajax_ElimineElArticulosDelDetalle(int linea)
    {
        return _IFacturacionController.ElimineElArticulosDelDetalle((TempDataDictionary)TempData, linea);
    }

    [HttpPost]
    0 references | pc, 48 days ago | 1 author, 1 change
    public async Task<JsonResult> Ajax_ObtengaElArticuloEditar(int linea)
```

Fuente: Elaboración Propia (2024).

Se procura que los controladores no contengan más de dos líneas de código para ejecutar o hacer llamados a las interfaces que contienen los contratos implementados en la capa de lógica de negocio.

Figura 73.

Proceso de facturación, dominio

```

2 references | pc, 3 days ago | 1 author, no changes
public class Facturar : Controller, IFacturacionController
{
    private readonly IRazorRenderHelper _IRazorRenderHelper;
    private readonly IObtengaLosRecursosParaFacturar _IObtengaLosRecursosParaFacturar;
    private readonly IBuscar _IBuscar;
    private readonly IProcesos _IProcesos;
    private readonly IFacturar _IFacturar;
    private readonly IDocumentService _documentService;

    0 references | 0 changes | 0 authors, 0 changes
    public Facturar(IRazorRenderHelper razorRenderHelper,
        IObtengaLosRecursosParaFacturar obtengaLosRecursosParaFacturar,
        IBuscar buscar,
        IProcesos procesos,
        IDocumentService documentService,
        IFacturar iFacturar)
    {
        _IRazorRenderHelper = razorRenderHelper;
        _IObtengaLosRecursosParaFacturar = obtengaLosRecursosParaFacturar;
        _IBuscar = buscar;
        _IProcesos = procesos;
        _documentService = documentService;
        _IFacturar = iFacturar;
    }

    2 references | pc, 36 days ago | 1 author, 2 changes
    public async Task<JsonResult> AgregueElArticuloAlDetalle(TempDataDictionary tempData, InsumoAgregarDetalleArticulo insumo)
    {
        insumo.IdCompania = TempDataService.ObtengaLaCompaniaDelContexto();
        insumo.CodSucursal = TempDataService.ObtengaLaSucursalDelContexto();
        Cliente cliente = TempDataService.Facturacion_ObtengaElCliente(tempData, true) ?? new Cliente();
        insumo.IdCliente = cliente.IdCliente;
        insumo.Rol = TempDataService.ObtengaElRolUsuarioDelContexto();
        List<Articulo> losArticulosAgregados = TempDataService.Facturacion_ObtengaElDetalleArticulos(tempData, true) ?? new();
        var elResultado = await _IProcesos.AgregueYRetorneLosArticuloDelDetalle(losArticulosAgregados, insumo);
        if (elResultado.EsValido)
        {
            if (elResultado.Articulos.Any())
            {
                elResultado.Articulos = elResultado.Articulos.OrderBy(X => X.Linea).ToList();
                TempDataService.Facturacion_GuardeElDetalleArticulos(tempData, elResultado.Articulos);
                string Vista = _IRazorRenderHelper.RenderPartialToString("~/Views/Facturacion/_FilaDetalleArticulo.cshtml", elResultado.Articulos);
                elResultado.Vista = Vista;
            }
        }

        return Json(elResultado!);
    }

    2 references | pc, 36 days ago | 1 author, 2 changes
    public JsonResult ActualiceElDetalleArticulosAlCambiarCliente(TempDataDictionary tempData) {...}
    2 references | pc, 36 days ago | 1 author, 2 changes
    public JsonResult ElimineElArticulosDelDetalle(TempDataDictionary tempData, int linea) {...}
    2 references | pc, 48 days ago | 1 author, 1 change
    public async Task<JsonResult> BusqueElClienteRegistrado(TempDataDictionary tempData, ParametroBusquedaClienteRegistrado parametros) {...}
    2 references | pc, 48 days ago | 1 author, 1 change
    public async Task<JsonResult> BusqueLosArticulos(TempDataDictionary tempData, ParametroBusquedaArticulo parametros) {...}
    2 references | pc, 36 days ago | 1 author, 2 changes
    public async Task<JsonResult> LimpieLaPantallaDeFacturacion(TempDataDictionary tempData) {...}
    2 references | pc, 52 days ago | 1 author, 1 change
    public IActionResult MuestreLaModalBuscarArticulos() {...}
    2 references | pc, 53 days ago | 1 author, 1 change
    public IActionResult MuestreLaModalBuscarClienteRegistrado() {...}
    2 references | pc, 36 days ago | 1 author, 2 changes
    public JsonResult ObtengaBarrioPorDistrito(TempDataDictionary tempData, ParametroBarrioUI barrio) {...}
    2 references | pc, 36 days ago | 1 author, 3 changes
    public JsonResult ObtengaCantonesPorProvincia(TempDataDictionary tempData, int provincia) {...}

```

Fuente: Elaboración propia (2024).

En el código, la clase Facturar se encarga de orquestar y controlar las diferentes interfaces que definen los contratos de los procesos relacionados con la facturación. Implementa dependencias como IRazorRenderHelper, IObtengaLosRecursosParaFacturar, IBuscar, IProcesos, y IDocumentService, las cuales se inyectan a través de su constructor. De esta manera, se centralizan las diversas funcionalidades que pueden ejecutarse desde la pantalla de facturación, permitiendo una gestión coherente y unificada de los recursos y procesos necesarios para esta operación.

Figura 74.

Proceso de facturación, lógica negocio, clase facturar

```

2 references | pc, 8 days ago | 1 author, 17 changes
public class Facturar : IFacturar
{
    private readonly IObtengaLosRecursosParaFacturar _IObtengaLosRecursosParaFacturar;
    private readonly IEasyCIFDataBase _IEasyCIFDataBase;
    private readonly IEnvioDocumentosHacienda _IEnvioDocumentosHacienda;
    private readonly ICrearQR _ICrearQR;
    private readonly IFecha _IFecha;
    private readonly IKardexArticulos _IKardexArticulos;
    private readonly IGeneraELXML _IGeneraELXML;
    private readonly IParametro _IParametro;
    private readonly IPaseContableCxC _IPaseContableCxC;

    #region Constructor
    0 references | pc, 31 days ago | 1 author, 1 change
    public Facturar(IObtengaLosRecursosParaFacturar iObtengaLosRecursosParaFacturar, IEasyCIFDataBase easyCIFDataBase,
        IConsultaDocumentosHacienda iConsultaDocumentosHacienda, IEnvioDocumentosHacienda envioDocumentosHacienda,
        ICrearQR crearQR, IFecha iFecha, IKardexArticulos kardexArticulos, IGeneraELXML generaELXML, IParametro parametro, IPaseContableCxC paseContableCxC)
    {
        _IObtengaLosRecursosParaFacturar = iObtengaLosRecursosParaFacturar;
        _IEasyCIFDataBase = easyCIFDataBase;
        _IEnvioDocumentosHacienda = envioDocumentosHacienda;
        _ICrearQR = crearQR;
        _IFecha = iFecha;
        _IKardexArticulos = kardexArticulos;
        _IGeneraELXML = generaELXML;
        _IParametro = parametro;
        _IPaseContableCxC = paseContableCxC;
    }
    #endregion

    private InsumoParaProcesarFactura ElInsumoParaFacturar = new();
    private RecursosParaFacturar LosRecursosParaFacturar = new();
    private ModeloValidacionEmiteLaFactura ModeloValidacion = new();
    private ParametrosLlaveDocEncabezado ParametrosLlaveDocEncabezado = new();

    private DateTime FechaHoraEmision;
    private Abstracciones.Contenedores.MantCompania.Compania Emisor = new();
    private DocumentoEncabezado ElDocumentoEncabezado = new();
    private List<DocumentoDetalle> ElDocumentoDetalle = new();
    private List<DocumentoDetalleImpuesto> ElDocumentoDetalleImpuesto = new();
    private DocumentoDetalleMedioPago ElDocumentoDetalleMedioPago = new();
    private DocumentoElectronico ElDocumentoElectronico = new();
    private DocumentoReceptor ElDocumentoReceptor = new();

    4 references | pc, 39 days ago | 1 author, 1 change
    private decimal TotalServGravados { get; set; }
    4 references | pc, 39 days ago | 1 author, 1 change
    private decimal TotalServExentos { get; set; }
    4 references | pc, 39 days ago | 1 author, 1 change
    private decimal TotalServExonerado { get; set; }
    4 references | pc, 39 days ago | 1 author, 1 change
    private decimal TotalMercanciasGravadas { get; set; }
    4 references | pc, 39 days ago | 1 author, 1 change
    private decimal TotalMercanciasExentas { get; set; }
    4 references | pc, 39 days ago | 1 author, 1 change
    private decimal TotalMercExonerada { get; set; }

```

Fuente: Elaboración propia (2024).

A nivel de desarrollo, la programación del prototipo sigue un mismo concepto. Se crean clases que implementan interfaces específicas y contienen los contratos para ejecutar acciones determinadas. Se utiliza la inyección de dependencias para lograr mayor modularidad y reducir el acoplamiento entre los componentes, asegurando que las abstracciones no dependan de implementaciones concretas. En lugar de crear instancias de las dependencias dentro de la clase Facturar, estas se inyectan desde el exterior, usualmente a través del constructor de la clase.

Figura 75.

Proceso de facturación, clase facturar, generar documento electrónico

```

private async Task GenerarDocumentoElectronico()
{
    await ObtengaLaCompania();
    FechaHoraEmision = _IFecha.ObtengaLaFechaActual();

    var TipoDocumentoElectronico = LosRecursosParaFacturar.TiposDocumento.Find(x => x.CodTipoDocumento == ELInsumoParaFacturar.CodTipoDocumento);
    if (TipoDocumentoElectronico?.esElectronico == false) EsDocElectronico = false;

    try
    {
        using (var scope = GeneradorTransacciones.ObtengaLaTransaccion())
        {
            await ObtengaLosConsecutivos();
            await ObtengaYCalculeELDetalleYDetalleImpuesto();

            if (ModeloValidacion.EsValido == false)
                throw new InvalidOperationException($"La validación ha fallado. {ModeloValidacion.Mensaje}");

            ObtengaYCompleteELEncabezado();
            ObtengaYCompleteELReceptorDelDocumento();
            ObtengaYCompleteELDocumentoDetalleDePago();

            if (EsDocElectronico)
            {
                await GenereELArchivoXMLFirmado();
                ObtengaELDocumentoElectronico();
            }

            await _IEasyCIFDataBase.Documento_AgregueDocumentoEncabezadoAsync(ELDocumentoEncabezado);
            await _IEasyCIFDataBase.Documento_AgregueDocumentoDetalleAsync(ELDocumentoDetalle);
            await _IEasyCIFDataBase.Documento_AgregueDocumentoDetalleImpuestoAsync(ELDocumentoDetalleImpuesto);
            await _IEasyCIFDataBase.Documento_AgregueDocumentoReceptorAsync(ELDocumentoReceptor);
            await _IEasyCIFDataBase.Documento_AgregueDocumentoMedioPagoAsync(new List<DocumentoDetalleMedioPago> { ELDocumentoDetalleMedioPago });
            await _IEasyCIFDataBase.Documento_AgregueDocumentoElectronicoAsync(ELDocumentoElectronico);

            var laCondicionVenta = LosRecursosParaFacturar.CondicionesVenta.Find(x => x.CodCondicionVenta == ELDocumentoEncabezado.CodCondicionVenta);
            var elTipoDocumento = LosRecursosParaFacturar.TiposDocumento.Find(x => x.CodTipoDocumento == ELDocumentoEncabezado.CodTipoDocumento);
            var elTipoIdentificacion = LosRecursosParaFacturar.TiposIdentificacion.Find(x => x.CodTipoIdentificacion == ELDocumentoReceptor.CodTipoIdentificacion);

            ELInsumoPaseCxC = new()
            {
                CodTipoAsiento = "101",
                CodTipoDoc = ELDocumentoEncabezado.CodTipoDocumento,
                Compania = ELDocumentoEncabezado.IdCompania,
                ConsecutivoInterno = (Long)ELDocumentoEncabezado.ConsecutivoInterno,
                Costo_E = (double)CostoTotal,
                Descuento_C = (double)TotalDescuentos,
                DesExento_F = (double)TotalDescuentosExento,
                DesGravado_G = (double)TotalDescuentosGravado,
                DetalleEncabezado = $"Asiento de diario FACTURACION {FechaHoraEmision!.ToString("dd/MM/yyyy")}",
                DetalleLinea = $"{elTipoDocumento!.Descripcion} de {laCondicionVenta!.Descripcion}, Cliente: {elTipoIdentificacion!.Descripcion}",
                Exento_B = (double)TotalExento,
                FechaDocumento = FechaHoraEmision,
                Gravado_A = (double)TotalGravado,
                Impuesto_D = (double)TotalImpuesto,
                NumDocumento = Clave,
                Sucursal = ELDocumentoEncabezado.CodSucursal,
                TipoDocumento = ELDocumentoEncabezado.CodCondicionVenta == "01" ? (int)Abstracciones.Enumerados.Conta.TipoDocumentoPaseCxC.Facti : 0,
                Total_H = (double)TotalComprobante,
                Usuario = ELInsumoParaFacturar.IdUsuario
            };
            await _IPaseContableCxC.RealiceELasientoContableCxC(ELInsumoPaseCxC);
            scope.Complete();
        }
    }
    catch
    {
        ModeloValidacion = new ModeloValidacion($"No podrá continuar, se ha presentado un problema a la hora de generar el documento.");
    }

    if (ModeloValidacion.EsValido)
        await EnviarDocumentoXML();
}

```

Fuente: Elaboración propia (2024).

Este procedimiento se encarga de llenar los modelos que contienen los datos que conforman la estructura de los documentos electrónicos. Se utiliza una interfaz incluso para obtener la fecha, con el objetivo de poder realizar pruebas unitarias sobre todo el código. Se sigue haciendo llamados a interfaces que a su vez son implementadas por otras clases para mantener el concepto de responsabilidad única. Por ejemplo, la función principal de esta clase es generar el

documento electrónico, pero se deben realizar otras acciones como registrar la cuenta por cobrar si la factura es de crédito, y realizar el asiento contable automático, para lo cual se llama a la interfaz `_IPaseContableCxC`. La generación y firma de los documentos electrónicos también se realiza mediante el llamado a una interfaz, lo que permite reutilizar la implementación de esta interfaz desde otros módulos como la anulación completa y la devolución parcial. También incluye la generación de notas de crédito electrónicas desde la pantalla de creación de documentos del módulo de cuentas por cobrar. A nivel de interfaces, esta clase también se encarga de registrar el Kardex de inventario, que corresponde a un registro de todos los movimientos de un artículo en el sistema y afecta las existencias de las bodegas.

Figura 76.

Proceso de envío de documentos a hacienda

```

2 referencias | pc: 24 days ago | 1 autor | 8 changes
public class EnvioDocumentosHacienda : IEnvioDocumentosHacienda
{
    private readonly ITokenHacienda _ITokenHacienda;
    private readonly IFecha _Ifecha;
    private readonly IEasyCIFDataBase _IEasyCIFDataBase;
    private readonly IConsultaDocumentosHacienda _IConsultaDocumentosHacienda;

    0 referencias | pc: 45 days ago | 1 autor | 1 change
    public EnvioDocumentosHacienda(ITokenHacienda tokenHacienda, IFecha ifecha, IEasyCIFDataBase easyCIFDataBase, IConsultaDocumentosHacienda iConsultaDocumentosHacienda)
    {
        _ITokenHacienda = tokenHacienda;
        _Ifecha = ifecha;
        _IEasyCIFDataBase = easyCIFDataBase;
        _IConsultaDocumentosHacienda = iConsultaDocumentosHacienda;
    }

    4 referencias | pc: 24 days ago | 1 autor | 7 changes
    public async Task<bool> EnvioDeDocumentos(ParametrosEnvioDocumentosHacienda parametros)
    {
        ControlDeToken controlDeToken = await _IEasyCIFDataBase.Hacienda.ObtenerTokenRegistrado(parametros.IDCompania);
        if (controlDeToken.Token == "" || EsInTokenValido(controlDeToken.FecultimoToken) == false)
            controlDeToken = await ObtengaToken(parametros.ParametrosToken, parametros.IDCompania);

        if (controlDeToken.Token == "") controlDeToken = await ObtengaToken(parametros.ParametrosToken, parametros.IDCompania);

        if (controlDeToken.Token != "")
        {
            await _IEasyCIFDataBase.Hacienda.GuardarElToken(controlDeToken);
            var respuesta = await Recepcion(parametros, controlDeToken.Token);
            if (respuesta == null)
            {
                var valoresActualizar = new Dictionary<string, object>
                {
                    { "Reenviar", true },
                    { "EnvioHacienda", false }
                };
            };
            InsumoParaConsultarDocumento InsumoConsultar = new()
            {
                CodSucursal = parametros.CodSucursal,
                CodTerminal = parametros.CodTerminal,
                CodIpoDocumento = parametros.CodIpoDocumento,
                ConsetuvoInterno = parametros.ConsetuvoInterno,
                IDCompania = parametros.IDCompania
            };
            await _IEasyCIFDataBase.Hacienda.ActualiceElEnvioDeDocumentoAlHacienda(InsumoConsultar, valoresActualizar);
        }
        else
        {
            await ConsultarElDocumento(parametros);
        }
    }
    return true;
}

1 referencias | pc: 25 days ago | 1 autor | 2 changes
private async Task ConsultarElDocumento(ParametrosEnvioDocumentosHacienda parametros, ...)

1 referencias | pc: 49 days ago | 1 autor | 0 changes
private static Emissor? ObtengaElEmisorDeDocumento(XmlDocument xmlDoc)

1 referencias | pc: 39 days ago | 1 autor | 0 changes
private static Receptor? ObtengaElReceptorDeDocumento(XmlDocument xmlDoc)

2 referencias | pc: 45 days ago | 1 autor | 2 changes
private static string ObtengaLaClaveNumerica(XmlDocument xmlDoc)

1 referencias | pc: 45 days ago | 1 autor | 1 change
private static string ObtengaElConsecutivoNumerico(XmlDocument xmlDoc)

1 referencias | pc: 49 days ago | 1 autor | 1 change
private static string ObtengaLaFechaDelDocumento(XmlDocument xmlDoc)

2 referencias | pc: 49 days ago | 1 autor | 1 change
private static XmlDocument ObtengaELXML(string documentoBase64)

1 referencias | pc: 45 days ago | 1 autor | 2 changes
private static string ObtengaElBsonAEviar(ParametrosEnvioDocumentosHacienda parametros)

3 referencias | pc: 46 days ago | 1 autor | 1 change
private async Task<ControlDeToken> ObtengaElToken(ParametrosToken parametre, int compania)

```

Fuente: Elaboración Propia (2024).

La clase anterior permite enviar los XML firmados de los documentos electrónicos al Ministerio de hacienda.

Figura 77.

Proceso obtención del Token del Ministerio de Hacienda

```

namespace EasyCIF.BW.Hacienda
{
    1 reference | pc, 53 days ago | 2 authors, 3 changes
    public class ObtengaELToken : ITokenHacienda
    {
        private string? accessToken = null;

        5 references | AdminPC, 297 days ago | 1 author, 1 change
        public async Task<string?> GetAccessToken(ParametrosToken parametros)
        {
            using var client = new HttpClient();
            string url = parametros.Url;
            client.DefaultRequestHeaders.Accept.Clear();
            client.DefaultRequestHeaders.Accept.Add(new MediaTypeWithQualityHeaderValue("application/json"));
            List<KeyValuePair<string, string>> list = new()
            {
                new KeyValuePair<string, string>("grant_type", "password"),
                new KeyValuePair<string, string>("client_id", parametros.ClienteID),
                new KeyValuePair<string, string>("username", parametros.Usuario),
                new KeyValuePair<string, string>("password", parametros.Contrasena)
            };
            List<KeyValuePair<string, string>> postData = list;
            FormUrlEncodedContent content = new(postData);
            try
            {
                HttpResponseMessage response = client.PostAsync(url, content).Result;
                string jsonString = await response.Content.ReadAsStringAsync();
                object responseData = JsonConvert.DeserializeObject(jsonString);
                accessToken = ((dynamic)responseData).access_token;
            }
            catch
            {
                return null;
            }
            return accessToken;
        }
    }
}

```

Fuente: Elaboración propia (2024).

La clase anterior muestra la implementación para obtener el Token, que es una clave de acceso necesaria para utilizar el servicio de facturación electrónica del Ministerio de Hacienda de Costa Rica. Este Token se obtiene registrándose y proporcionando las credenciales requeridas. Una vez obtenido, se incluye en cada solicitud al servicio de facturación para indicar que el usuario está autorizado. Esto garantiza que solo las personas con permiso puedan acceder y utilizar el servicio de manera segura.

Figura 78.

Proceso genere el XML de los documentos electrónicos

```

2 references | pc, 30 days ago | 1 author, 5 changes
public class GenererXML : IGenererXML
{
    public readonly IEasyCIFDataBase _IEasyCIFDataBase;
    0 references | pc, 30 days ago | 1 author, 1 change
    public GenererXML(IEasyCIFDataBase easyCIFDataBase)
    {
        _IEasyCIFDataBase = easyCIFDataBase;
    }
}

4 references | pc, 30 days ago | 1 author, 4 changes
public async Task<string> GenererFirmeELXML(InsumoParaCrearXML insumo)
{
    string sXml = "";
    using (Utf8StringWriter stringWriter = new()) {...}

    byte[] xmlBytes = Encoding.UTF8.GetBytes(sXml);
    string rutaArchivo = insumo.Emisor.RutaCertificado;
    byte[] bytesArchivo = System.IO.File.ReadAllBytes(rutaArchivo);
    string base64Cert = Convert.ToBase64String(bytesArchivo);
    byte[] certBytes = Convert.FromBase64String(base64Cert);

    X509Certificate2 cert = new(certBytes, insumo.Emisor.ContrasenaCertificado, X509KeyStorageFlags.MachineKeySet | X509KeyStorageFlags.PersistKeySet);

    FirmaXadesNet.Crypto.Signer signer = new(cert);

    XadesService xadesService = new();

    SignatureParameters parametros = new()
    {
        SignaturePackaging = SignaturePackaging.ENVELOPED,
        SignaturePolicyInfo = new()
        {
            {
                PolicyIdentifier = "https://tribunet.hacienda.go.cr/docs/esquemas/2016/v4.1/Resolucion_Comprobantes_Electronicos_DGT-R-48-2016.pdf",
                PolicyHash = @"0hixL6upD6av8N7pEVdABhEL6hME="
            },
        },
        DataFormat = new DataFormat { MimeType = "text/xml" },
        SignerRole = new FirmaXadesNet.Signature.Parameters.SignerRole()
    };
    parametros.SignerRole.ClaimedRoles.Add("ObligadoTributario");

    using (parametros.Signer = new FirmaXadesNet.Crypto.Signer(cert))
    {
        string resultString = "";
        using (MemoryStream stream = new(xmlBytes))
        {
            var docFirmado = xadesService.Sign(stream, parametros);
            xmlBytes = docFirmado.GetDocumentBytes();
            resultString = Encoding.UTF8.GetString(xmlBytes);
        }
    }

    string base64String = Convert.ToBase64String(xmlBytes);
    return base64String;
}

```

Fuente: Elaboración propia (2024).

Esta clase contiene la implementación necesaria para crear los archivos XML de los diferentes tipos de documentos electrónicos generados en el prototipo funcional, además de firmarlos.

Figura 79.

Proceso pase contable CxP

```

2 references | pc 5 days ago | 1 author | 1 change
public class PaseContableCxP : IPaseContableCxP
{
    public readonly IEasyCIFDataBase _IEasyCIFDataBase;
    public readonly IFecha _IFecha;

    private Dictionary<char, double> LosMontos = new();

    private List<MaestroFormulasPaseCxP> EL_MaestroFormulas = new();
    private List<Abstracciones.ContenedoresConta.Detalle_Asiento> ElDetalle_Asientos = new();
    private Abstracciones.ContenedoresConta.Encabezado_Asiento ElEncabezado_Asiento = new();
    private InsumoPaseCxP ElInsumo = new();

    private DateTime LaFechaHoraProceso = new();

0 references | pc 5 days ago | 1 author | 1 change
public PaseContableCxP(IEasyCIFDataBase easyCIFDataBase, IFecha fecha)
{
    _IEasyCIFDataBase = easyCIFDataBase;
    _IFecha = fecha;
}

5 references | pc 5 days ago | 1 author | 1 change
public async Task<bool> RealiceElAsientoContableCxP(InsumoPaseCxP insumo)
{
    ElInsumo = insumo;
    LaFechaHoraProceso = _IFecha.ObtengaLaFechaActual();
    AsigneLosMontos();
    EL_MaestroFormulas = await _IEasyCIFDataBase.MaestroFormulasCxP.ObtengaLaFormulaPaseCxP((InsumoConsultaFormulaPaseCxP)insumo);
    if (EL_MaestroFormulas != null && EL_MaestroFormulas.Any())
    {
        foreach (var item in EL_MaestroFormulas)
        {
            if (item.Cuenta == "0" || item.Cuenta == "")
            {
                item.Cuenta = ElInsumo.CuentaContable;
            }
        }
        await GenereElAsiento();
    }
    return true;
}

1 reference | pc 5 days ago | 1 author | 1 change
private void AsigneLosMontos()
{
    LosMontos = new Dictionary<char, double>
    {
        {'A', ElInsumo.Gravado_A}, // MontoGravado
        {'B', ElInsumo.Exento_B}, // MontoExento
        {'C', ElInsumo.Descuento_C}, // Descuento
        {'D', ElInsumo.Impuesto_D}, // Impuesto
        {'E', ElInsumo.Costo_E}, // Costo
        {'F', ElInsumo.DesExento_F}, // Des Exento
        {'G', ElInsumo.DesGravado_G}, // Des Gravado
        {'H', ElInsumo.Total_H} // Total
    };
}

```

Fuente: Elaboración propia (2024).

La clase anterior contiene la implementación de la lógica necesaria para realizar los asientos de las facturas de compra, facturas de gastos, y las notas de crédito y débito del módulo de cuentas por pagar. Se utiliza una estructura similar para el pase contable de las cuentas por cobrar, que incluye los asientos de facturación, anulación y devolución.

Figura 80.

Proceso registro de asientos manuales

```

2 references | pc, 18 days ago | 1 author, 2 changes
public class AccionesIngresoAsiento : IAccionesIngresarAsiento
{
    private readonly IEasyCIFDataBase _IEasyCIFDataBase;
    private readonly IObtengaLosRecursos _IObtengaLosRecursos;
    private readonly IFecha _IFecha;

    0 references | pc, 18 days ago | 1 author, 1 change
    public AccionesIngresoAsiento(IEasyCIFDataBase easyCIFDataBase, IObtengaLosRecursos iObtengaLosRecursos, IFecha fecha)
    {
        _IEasyCIFDataBase = easyCIFDataBase;
        _IObtengaLosRecursos = iObtengaLosRecursos;
        _IFecha = fecha;
    }

    2 references | pc, 18 days ago | 1 author, 1 change
    public async Task<ModeloValidacionRegistroAsientos> AgregueLaLinea(InsumoAgregaLineaDetalle insumo)
    {
        ModeloValidacionRegistroAsientos resultado = ReglasIngresoAsientos.ValidateElInsumoAgregaLinea(insumo);
        Asiento elAsiento = insumo.Asiento ?? new();
        Encabezado_Asiento encabezado_Asiento = elAsiento.EncabezadoAsiento ?? new();
        List<DetalleAsientoListar> detalle_Asiento = elAsiento.DetalleAsiento ?? new();

        string desCuenta = "";
        string destipoAsiento = "";
        int anno = 0;
        int mes = 0;
        long asiento = 0;
        int secuencia = 0;

        if (resultado.EsValido)
        {
            List<TipoAsiento> losTiposAsientos = await _IObtengaLosRecursos.ObtengaElCatalogoTipoDeAsiento(insumo.IdCompania);
            TipoAsiento? tipoAsiento = losTiposAsientos.Find(x => x.CodTipoAsiento == insumo.CodTipoAsiento);
            if (tipoAsiento == null) resultado = new("No podrá continuar el Tipo Asiento seleccionado es inválido.");

            if (resultado.EsValido)
                return resultado;
        }
    }

    2 references | pc, 18 days ago | 1 author, 1 change
    public ModeloValidacionRegistroAsientos ElimineLaLinea(Asiento asiento, int linea)
    {
    }

    2 references | pc, 18 days ago | 1 author, 2 changes
    public async Task<ModeloValidacion> GrabarAsiento(Asiento asiento, InsumoGrabarAsiento insumo)
    {
    }
}

```

Fuente: Elaboración propia (2024).

La clase anterior implementa el registro de asiento manuales en el módulo de contabilidad.

Figura 81.

Proceso generar transacción

```

28 references | AdminPC, 93 days ago | 1 author, 3 changes
public static class GeneradorTransacciones
{
    28 references | AdminPC, 93 days ago | 1 author, 3 changes
    public static TransactionScope ObtengaLaTransaccion()
    {
        return new TransactionScope(TransactionScopeOption.Required, new TransactionOptions
        {
            IsolationLevel = IsolationLevel.ReadCommitted,
            Timeout = TransactionManager.MaximumTimeout
        },
        TransactionScopeAsyncFlowOption.Enabled);
    }
}

```

Fuente: Elaboración propia (2024).

En el prototipo también se encuentran algunas clases que, debido a su naturaleza, no necesitan implementar una interfaz. Por ejemplo, la clase estática `GeneradorTransacciones` proporciona el nivel de aislamiento `ReadCommitted` para las operaciones en la base de datos.

Figura 82.

Proceso gestión de base de datos , abstracciones

```

public interface IRepositoryosEF
{
    61 references | AdminPC, 297 days ago | 1 author, 1 change
    IRepositoryosEF<T> ObtenerRepositorio<T>() where T : class;
}

64 references | pc, 32 days ago | 2 authors, 8 changes
public interface IRepositoryosEF<T> where T : class
{
    1 reference | AdminPC, 297 days ago | 1 author, 1 change
    IQueryable<T> Table { get; }
    1 reference | AdminPC, 297 days ago | 1 author, 1 change
    T Add(T entidad);
    25 references | AdminPC, 297 days ago | 1 author, 1 change
    Task<T> AddAsync(T entidad);
    1 reference | AdminPC, 297 days ago | 1 author, 1 change
    IList<T> AddRange(IList<T> entidades);
    12 references | AdminPC, 297 days ago | 1 author, 1 change
    Task<IList<T>> AddRangeAsync(IList<T> entidades);
    1 reference | AdminPC, 297 days ago | 1 author, 1 change
    int Count();
    2 references | AdminPC, 114 days ago | 1 author, 1 change
    int Count(Expression<Func<T, bool>> criteria);
    1 reference | AdminPC, 297 days ago | 1 author, 1 change
    Task<int> CountAsync();

    1 reference | AdminPC, 297 days ago | 1 author, 1 change
    int Delete<P>(P id);
    1 reference | AdminPC, 297 days ago | 1 author, 1 change
    Task<int> DeleteAsync<P>(P id);
    2 references | AdminPC, 297 days ago | 1 author, 1 change
    int DeleteEntity(T entidad);
    2 references | AdminPC, 297 days ago | 1 author, 1 change
    Task<int> DeleteEntityAsync(T entidad);
    1 reference | AdminPC, 297 days ago | 1 author, 1 change
    int DeleteRange(IEnumerable<T> entidades);
    3 references | AdminPC, 297 days ago | 1 author, 1 change
    Task<int> DeleteRangeAsync(IEnumerable<T> entidades);
    1 reference | AdminPC, 297 days ago | 1 author, 1 change
    int DeleteExpression(Expression<Func<T, bool>> criteria);
    4 references | AdminPC, 297 days ago | 1 author, 1 change
    Task<int> DeleteExpressionAsync(Expression<Func<T, bool>> criteria);

    1 reference | AdminPC, 297 days ago | 1 author, 1 change
    T? Find(Expression<Func<T, bool>> criteria);
    3 references | AdminPC, 297 days ago | 1 author, 1 change
    IEnumerable<T?> FindAll(Expression<Func<T, bool>> criteria);
    21 references | AdminPC, 297 days ago | 1 author, 1 change
    Task<IEnumerable<T?>> FindAllAsync(Expression<Func<T, bool>> criteria);
    11 references | AdminPC, 297 days ago | 1 author, 1 change
    Task<T?> FindAsync(Expression<Func<T, bool>> criteria);
    9 references | AdminPC, 297 days ago | 1 author, 1 change
    IQueryable<T?> FindBy(Expression<Func<T, bool>> criteria);
}

```

Fuente: Elaboración propia (2024).

La gestión de base de datos sigue el concepto utilizado en todo el desarrollo del prototipo funcional. La capa de acceso a datos utiliza la inyección de dependencias y contiene las implementaciones de las interfaces definidas en la capa de abstracciones. Además, la capa DA implementa el patrón repositorio, lo que permite gestionar los repositorios y las operaciones CRUD de manera ordenada y similar a la estructura de la base de datos SQL.

Figura 83.

Proceso gestión de base de datos, implementación de abstracciones

```

61 references | pc, 32 days ago | 2 authors, 11 changes
internal class Repositorio<T> : IRepositoryEF<T> where T : class
{
    private readonly IContextoBD _contextoBD;
    3 references | AdminPC, 297 days ago | 1 author, 1 change
    private DbSet<T> Entidad { get; set; }
    33 references | AdminPC, 101 days ago | 1 author, 2 changes
    public DbSet<T> LaEntidad {...}
    1 reference | AdminPC, 297 days ago | 1 author, 1 change
    public IQueryable<T> Table => Entidad;

    60 references | AdminPC, 297 days ago | 1 author, 1 change
    public Repositorio(IContextoBD contexto) => _contextoBD = contexto;

    1 reference | AdminPC, 297 days ago | 1 author, 1 change
    public T Add(T entidad) {...}
    25 references | pc, 46 days ago | 2 authors, 3 changes
    public async Task<T> AddAsync(T entidad)
    {
        await LaEntidad.AddAsync(entidad);
        await _contextoBD.SaveChangesAsync();
        return entidad;
    }

    1 reference | AdminPC, 297 days ago | 1 author, 1 change
    public IList<T> AddRange(IList<T> entidades) {...}
    12 references | pc, 46 days ago | 2 authors, 3 changes
    public async Task<IList<T>> AddRangeAsync(IList<T> entidades)
    {
        await LaEntidad.AddRangeAsync(entidades);
        await _contextoBD.SaveChangesAsync();
        return entidades;
    }

    1 reference | AdminPC, 297 days ago | 1 author, 1 change
    public int Count() {...}
    2 references | AdminPC, 114 days ago | 1 author, 1 change
    public int Count(Expression<Func<T, bool>> criteria) {...}
    1 reference | AdminPC, 297 days ago | 1 author, 1 change
    public async Task<int> CountAsync() {...}

    1 reference | AdminPC, 297 days ago | 1 author, 1 change
    public int Delete<P>(P id) {...}
    1 reference | pc, 46 days ago | 2 authors, 3 changes
    public async Task<int> DeleteAsync<P>(P id) {...}
    2 references | AdminPC, 297 days ago | 1 author, 1 change
    public int DeleteEntity(T entidad) {...}
    2 references | AdminPC, 111 days ago | 1 author, 2 changes
    public async Task<int> DeleteEntityAsync(T entidad) {...}
    1 reference | AdminPC, 297 days ago | 1 author, 1 change
    public int DeleteRange(IEnumerable<T> entidades) {...}
    3 references | pc, 46 days ago | 2 authors, 3 changes
    public async Task<int> DeleteRangeAsync(IEnumerable<T> entidades) {...}
    1 reference | AdminPC, 297 days ago | 1 author, 1 change
    public int DeleteExpression(Expression<Func<T, bool>> criteria) {...}
    4 references | AdminPC, 297 days ago | 1 author, 1 change
    public async Task<int> DeleteExpressionAsync(Expression<Func<T, bool>> criteria) {...}

```

Fuente: Elaboración propia (2024).

En la clase anterior se muestra la implementación de la interfaz IRepositoryEF, que contiene los métodos CRUD para acceder a la base de datos.

Figura 84.

Proceso gestión de base de datos, implementador del acceso a datos

```

77 references | pc 26 days ago | 1 author, 4 changes
public partial class ImplementadorDA : IEasyCIFDataBase
{
    4 references | pc 26 days ago | 1 author, 2 changes
    public async Task<bool> Hacienda_ActualiceElEnvioDelDocumentoAHacienda(InsumoParaConsultarDocumento insumo, Dictionary<string, object> valoresActualizar)
    {
        int registrosActualizados = await _repositorioDocumentoElectronico.UpdatePartialByExpressionAsync(x =>
            x.IdCompania == insumo.IdCompania &&
            x.CodSucursal == insumo.CodSucursal &&
            x.CodTerminal == insumo.CodTerminal &&
            x.CodTipoDocumento == insumo.CodTipoDocumento &&
            x.ConsecutivoInterno == insumo.ConsetuvoInterno, valoresActualizar);

        return true;
    }

    4 references | pc 38 days ago | 1 author, 1 change
    public async Task<bool> Documento_AgregueDocumentoElectronicoAsync(Abstracciones.Contenedores.Documentos.DocumentoElectronico documentoElectronico)
    {
        Modelos.DocumentoElectronico Doc = (Modelos.DocumentoElectronico)documentoElectronico;
        await _repositorioDocumentoElectronico.AddAsync(Doc);
        return true;
    }

    3 references | pc 32 days ago | 1 author, 1 change
    public async Task<Abstracciones.Contenedores.Documentos.DocumentoElectronico> Documento_ObtengaElDocumentoElectronicoPorClave(int compania, int sucursal, string claveNumerica)
    {
        var Documento = await _repositorioDocumentoElectronico.GetAsync(x =>
            x.IdCompania == (int)compania &&
            x.CodSucursal == (int)sucursal &&
            x.ClaveNumerica == claveNumerica);
        return Documento ?? new();
    }

    2 references | pc 26 days ago | 1 author, 2 changes
    public async Task<bool> Documento_ActualiceElEnvioCorreoDocumentoElectronico(Abstracciones.Contenedores.Hacienda.InsumoParaConsultarDocumento insumo, DateTime fechaHora)
    {
        var valoresActualizar = new Dictionary<string, object>
        {
            { "CorreoEnviado", true },
            { "FecHoraEnvioCorreo", fechaHora }
        };

        int registrosActualizados = await _repositorioDocumentoElectronico.UpdatePartialByExpressionAsync(x =>
            x.IdCompania == insumo.IdCompania &&
            x.CodSucursal == insumo.CodSucursal &&
            x.CodTerminal == insumo.CodTerminal &&
            x.CodTipoDocumento == insumo.CodTipoDocumento &&
            x.ConsecutivoInterno == insumo.ConsetuvoInterno, valoresActualizar);

        return true;
    }
}

```

Fuente: Elaboración propia (2024).

En la clase anterior se puede observar una sección del código de la implementación de la interfaz IEasyCIFDataBase, que contiene todos los contratos a los diferentes repositorios (tablas) existentes en la base de datos.

Validaciones

El sistema cuenta con dos tipos de validaciones fundamentales. En primer lugar, las validaciones a nivel de JavaScript se encargan de prevenir que se ingresen datos incorrectos y de notificar al usuario cuando un dato es requerido o no cumple con el formato específico. Por otro lado, las reglas de negocio, ubicadas en la capa de validaciones, permiten validar los datos antes y durante la ejecución de la lógica de negocio. Esta doble capa de validación proporciona seguridad adicional al sistema, asegurando que los datos sean válidos y consistentes incluso si las solicitudes al controlador son interceptadas o modificadas por software externo. Además, esta estructura garantiza la operación segura del sistema, aunque la ejecución de JavaScript esté deshabilitada en el cliente.

Figura 85.

Validaciones por script de datos requeridos.

```

(function () {
  $(camposFormantCompania.FormularioCompania).validate({
    rules: {
      TipoIdentificacion: {
        required: true,
      },
      Identificacion: {
        required: true,
      },
      Nombre: {
        required: true,
      },
      Correo: {
        required: true,
        email: true,
      },
      Provincia: {
        required: true,
      },
      Canton: {
        required: true,
      },
      Distrito: {
        required: true,
      },
      Barrio: {
        required: true,
      },
      UsuarioHacienda: {
        required: true,
      },
      Contraseña: {
        required: true,
      },
      ClaveCertificado: {
        required: true,
      },
      Estado: {
        required: true,
      },
      certificadoInput: {
        required: true,
      },
    },
    messages: {
      TipoIdentificacion: {
        required: "El campo \"Tipo identificación\" es requerido."
      },
      Identificacion: {
        required: "El campo \"Identificación\" es requerido."
      },
      Nombre: {
        required: "El campo \"Nombre\" es requerido."
      },
      Correo: {
        required: "El campo \"Correo\" es requerido.",
        email: "El campo \"Correo\" no cumple con los requisitos mínimos de una dirección de correo electrónico."
      },
      Provincia: {
        required: "El campo \"Provincia\" es requerido."
      },
      Canton: {
        required: "El campo \"Cantón\" es requerido."
      },
      Distrito: {
        required: "El campo \"Distrito\" es requerido."
      },
      Barrio: {
        required: "El campo \"Barrio\" es requerido."
      },
      UsuarioHacienda: {
        required: "El campo \"Usuario Hacienda\" es requerido."
      },
      Contraseña: {
        required: "El campo \"Contraseña Hacienda\" es requerido."
      },
      ClaveCertificado: {
        required: "El campo \"Clave del certificado\" es requerido."
      },
      Estado: {
        required: "El campo \"Estado\" es requerido."
      },
      certificadoInput: {
        required: "El campo \"Certificado Firma Digital\" es requerido."
      },
    },
    errorElement: 'span',
    errorPlacement: function (error, element) {
      error.addClass('invalid-feedback');
      element.closest('.form-group').append(error);
    },
    highlight: function (element, errorClass, validClass) {
      $(element).addClass('is-invalid');
    }
  });
}());

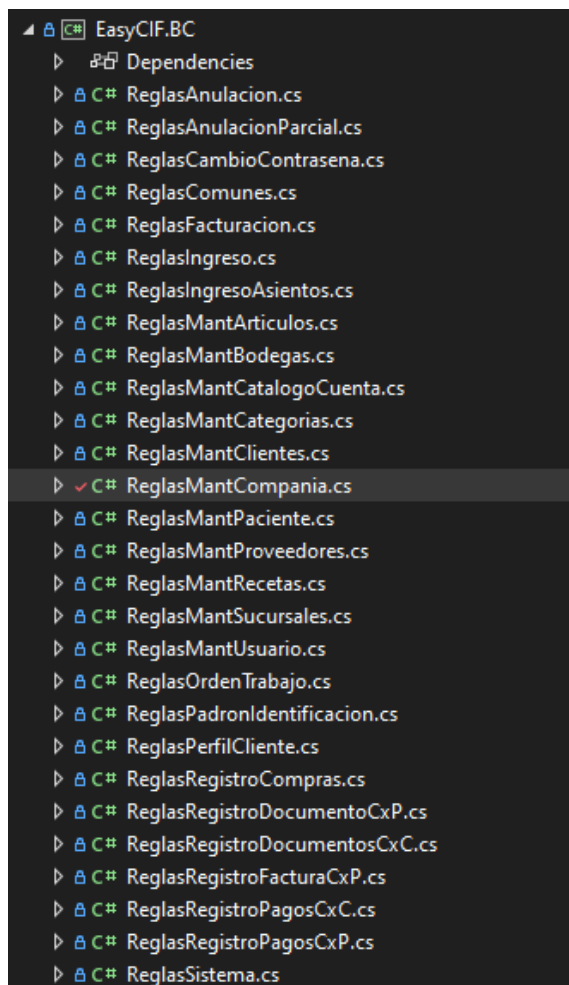
```

Fuente: Elaboración propia (2024).

Cada una de las vistas del UI, especialmente en los mantenimientos, incorpora validación de datos desde los archivos JavaScript. Esta práctica tiene como objetivo validar los datos necesarios desde el cliente para prevenir envíos de peticiones con información incompleta.

Figura 86.

Validaciones estructura de proyecto BC que contiene las reglas de negocio



Fuente: Elaboración propia (2024).

La capa BC es una de las más importantes del sistema, ya que contiene las reglas o validaciones aplicables a cada una de las pantallas del prototipo funcional. En la imagen se pueden visualizar algunas de las clases contenedoras de validaciones para los procesos de facturación, anulaciones, devoluciones, ingreso al sistema, validación de roles, registros de compras, gestión de órdenes de trabajo, e ingreso de asientos contables, entre otros

Figura 87.

Validaciones reglas mantenimiento de compañías

```

22 references | pc, 60 days ago | 2 authors, 10 changes
public static class ReglasMantCompania
{
    4 references | AdminPC, 101 days ago | 1 author, 1 change
    public static ModeloValidacion ValideLaExtensionDelArchivo(string? extension) {...}
    4 references | AdminPC, 101 days ago | 1 author, 1 change
    public static ModeloValidacion ValideSiSeAdjuntoArchivo(byte[]? archivo) {...}
    3 references | AdminPC, 297 days ago | 1 author, 1 change
    public static ModeloValidacion ValideLaFechaDeVencimiento(DateTime fechaCertificado, DateTime fechaActual)
    {
        ModeloValidacion modeloValidacion = new();

        if (fechaCertificado.Date <= fechaActual.Date)
        {
            modeloValidacion = new("No podrá continuar, el certificado digital se encuentra vencido.");
        }
        return modeloValidacion;
    }
    3 references | AdminPC, 297 days ago | 1 author, 1 change
    public static ModeloValidacion ValideSiLaContraseñaEsInvalida(string mensaje)
    {
        ModeloValidacion modeloValidacion = new();
        if (mensaje == "The specified network password is not correct.")
        {
            modeloValidacion = new("No podrá continuar, la clave del certificado digital es incorrecta.");
        }
        return modeloValidacion;
    }
    1 reference | AdminPC, 291 days ago | 1 author, 2 changes
    public static ModeloValidacion ValideLosDatosAlAgregar(InsumosCompaniaAgregarUI insumo, Catalogos catalogos) {...}
    1 reference | AdminPC, 290 days ago | 1 author, 2 changes
    public static ModeloValidacion ValideLosDatosAlModificar(InsumosCompaniaModificarUI insumo, CatalogosModificar catalogos) {...}
    2 references | AdminPC, 297 days ago | 1 author, 1 change
    public static ModeloValidacion ValideCertificadoDigitalRequerido(bool cargo) {...}
    2 references | AdminPC, 297 days ago | 1 author, 1 change
    public static ModeloValidacion ValideELToken(string? token) {...}
    1 reference | AdminPC, 294 days ago | 1 author, 1 change
    public static ModeloValidacion ValideSiExisteLaCompaniaAlAgregar(int id) {...}
    1 reference | pc, 83 days ago | 2 authors, 4 changes
    public static ModeloValidacion ValideSiExisteLaCompaniaAlEditar(int id) {...}
    1 reference | AdminPC, 297 days ago | 1 author, 1 change
    private static ModeloValidacion ValideElTipoIdentificacion(string? tipoIdentificacion, List<TipoIdentificacion> tipoIdentificaciones) {...}
    1 reference | AdminPC, 297 days ago | 1 author, 1 change
    private static ModeloValidacion ValideLaIdentificacion(string? identificacion, TipoIdentificacion tipoIdentificacion) {...}
    2 references | AdminPC, 290 days ago | 1 author, 3 changes
    private static ModeloValidacion ValideElNombre(string? nombre) {...}
    2 references | AdminPC, 291 days ago | 1 author, 2 changes
    private static ModeloValidacion ValideElNombreComercial(string nombreComercial) {...}
    2 references | AdminPC, 297 days ago | 1 author, 1 change
    private static ModeloValidacion ValideElCodAreaTelefono(short? codArea, List<CodigoAreaPais> codigosAreas) {...}
    2 references | AdminPC, 297 days ago | 1 author, 1 change
    private static ModeloValidacion ValideElCodAreaFax(short? codArea, List<CodigoAreaPais> codigosAreas) {...}
    2 references | AdminPC, 297 days ago | 1 author, 1 change
}

```

Fuente: Elaboración propia (2024).

La figura anterior muestra un ejemplo de las validaciones realizadas para el mantenimiento de compañías. Este mantenimiento es importante en el prototipo porque permite ingresar los datos necesarios para iniciar el proceso de facturación electrónica, incluyendo el certificado con el cual se firman los documentos electrónicos.

Figura 88.

Validaciones reglas para registrar asientos manuales

```

public static class ReglasIngresoAsientos
{
    1 reference | pc, 18 days ago | 1 author, 1 change
    public static ModeloValidacion ValideLosInsumosParaObtenerCuenta(int compania, string cuenta)
    {
        ModeloValidacion resultado = ValideLaCompania(compania);

        if (resultado.EsValido && string.IsNullOrEmpty(cuenta))
            resultado = new ModeloValidacion("El campo \"Cuenta\" es requerido.");

        return resultado;
    }

    1 reference | pc, 18 days ago | 1 author, 1 change
    public static ModeloValidacion ValideLosInsumosParaCargaAsientos(InsumoCargaAsiento insumo)...

    1 reference | pc, 18 days ago | 1 author, 1 change
    public static ModeloValidacion ValideElInsumoAGrabar(InsumoGrabarAsiento insumo)
    {
        ModeloValidacion resultado = ValideLaCompania(insumo.IdCompania);
        if (resultado.EsValido) resultado = ValideElUsuario(insumo.IdUsuario);
        if (resultado.EsValido) resultado = ValideLaFechaDelDocumento(insumo.Fecha, "Fecha de Asiento");
        if (resultado.EsValido) resultado = ValideLaObservacion(insumo.Observacion);
        return resultado;
    }

    1 reference | pc, 18 days ago | 1 author, 1 change
    public static ModeloValidacion ValideElInsumoAgregarLinea(InsumoAgregarLineaDetalle insumo)
    {
        ModeloValidacion resultado = ValideLaCompania(insumo.IdCompania);
        if (resultado.EsValido) resultado = ValideElUsuario(insumo.Usuario);
        if (resultado.EsValido) resultado = ValideLaCuenta(insumo.Cuenta);
        if (resultado.EsValido) resultado = ValideElDetalle(insumo.Detalle);
        if (resultado.EsValido) resultado = ValideElTipoAsiento(insumo.CodTipoAsiento);
        if (resultado.EsValido) resultado = ValideElNumeroDeDocumento(insumo.NumDocumento);
        if (resultado.EsValido) resultado = ValideLaFechaDelDocumento(insumo.FechaAsiento, "Fecha de Asiento");
        if (resultado.EsValido) resultado = ValideLaFechaDelDocumento(insumo.FechaDocumento, "Fecha del Documento");
        if (resultado.EsValido) resultado = ValideMontos(insumo.Debe, insumo.Haber);
        return resultado;
    }

    1 reference | pc, 18 days ago | 1 author, 1 change
    private static ModeloValidacion ValideMontos(double debe, double haber)...

    3 references | pc, 18 days ago | 1 author, 1 change
    private static ModeloValidacion ValideLaFechaDelDocumento(DateTime fecha, string campo)...

    1 reference | pc, 18 days ago | 1 author, 1 change
    private static ModeloValidacion ValideElNumeroDeDocumento(string numDocumento)...

    1 reference | pc, 18 days ago | 1 author, 1 change
    private static ModeloValidacion ValideElTipoAsiento(string tipoAsiento)...

    1 reference | pc, 18 days ago | 1 author, 1 change
    private static ModeloValidacion ValideLaCuenta(string cuenta)...

    1 reference | pc, 18 days ago | 1 author, 1 change
    private static ModeloValidacion ValideElDetalle(string detalle)...

    1 reference | pc, 18 days ago | 1 author, 1 change
    private static ModeloValidacion ValideLaObservacion(string observacion)...

    2 references | pc, 18 days ago | 1 author, 1 change
    private static ModeloValidacion ValideElUsuario(int usuario)...

    4 references | pc, 18 days ago | 1 author, 1 change
    private static ModeloValidacion ValideLaCompania(int compania)...

```

Fuente: Elaboración propia (2024).

La figura anterior muestra algunas de las validaciones que se invocan desde la pantalla de ingreso manual de asientos en el módulo de contabilidad.

Figura 89.

Validaciones reglas para facturar

```

1 reference | pc: 40 days ago | 1 author | 1 change
public static ModeloValidacion ValideElInsumoParaProcesarFactura(InsumoParaProcesarFactura insumo, RecursosParaFacturar recursos)
{
    ModeloValidacion resultado = ValideQueElClienteContengaDatos(insumo.Cliente.IdCliente);
    if (resultado.EsValido) resultado = ValideQueHayaAgregadoArticulo(insumo.Cliente.IdCliente);
    if (resultado.EsValido) resultado = ValideElTipoDocumentoYCondicionVenta(insumo, recursos);
    if (resultado.EsValido) resultado = ValideElCodMedioPago(insumo.CodMedioPago, recursos.MediosPago);
    if (resultado.EsValido) resultado = ValideElCorreoElectronico(insumo.Correo);
    if (resultado.EsValido) resultado = ValideLaMoneda(insumo.Moneda, recursos.Monedas);
    if (resultado.EsValido) resultado = ValideLaActividadComercial(insumo.CodActividadComercial, recursos.ActividadesComerciales);
    if (resultado.EsValido) resultado = ValideLaObservacion(insumo.Observacion);
    if (resultado.EsValido && insumo.Cliente.CodTipoIdentificacion == "00") resultado = ValideLaInformacionDelClienteDeContadoNoRegistrado(insumo, recursos);
    return resultado;
}
1 reference | pc: 40 days ago | 1 author | 3 changes
public static ModeloValidacion ValideLosParametrosAlConsultarClientePorIdentificacionYTipo(Abstracciones.Contenedores.Facturacion.ParametroIdentificacion parametro)
2 references | pc: 40 days ago | 1 author | 1 change
public static ModeloValidacion ValideElInsumoParaAgregarElArticuloAlDetalle(Abstracciones.Contenedores.Facturacion.InsumoAgregarDetalleArticulo insumo)
2 references | pc: 40 days ago | 1 author | 2 changes
public static ModeloValidacion ValideQueSePuedaAgregarElArticuloAlDetalle(Articulo articulo, Cliente cliente, InsumoAgregarDetalleArticulo insumo)
2 references | pc: 40 days ago | 1 author | 1 change
public static ModeloValidacion ValideElSaldoDeArticulo(double saldo, double cantidad)
2 references | pc: 40 days ago | 1 author | 1 change
public static ModeloValidacion ValideElPorcentajeDescuentoGlobal(double descuentoCliente, double descuentoLinea)
1 reference | pc: 39 days ago | 1 author | 2 changes
private static ModeloValidacion ValideLaInformacionDelClienteDeContadoNoRegistrado(InsumoParaProcesarFactura insumo, RecursosParaFacturar recursos)
1 reference | pc: 39 days ago | 1 author | 2 changes
private static ModeloValidacion ValideLaIdentificacionYTipo(string? tipoIdentificacion, string? identificacion, List<Abstracciones.Contenedores.CatalogoTipoIdentificacion.TipoIdentificacion> tipoIdentific
1 reference | pc: 40 days ago | 1 author | 1 change
private static ModeloValidacion ValideElNombre(string? nombre)
1 reference | pc: 40 days ago | 1 author | 1 change
private static ModeloValidacion ValideElCodAreaTelefono(short? codArea, List<CodigoAreaPais> codigosAreas)
1 reference | pc: 40 days ago | 1 author | 1 change
private static ModeloValidacion ValideElCodAreaFax(short? codArea, List<CodigoAreaPais> codigosAreas)
1 reference | pc: 40 days ago | 1 author | 1 change
private static ModeloValidacion ValideElTelefono(string? telefono)
1 reference | pc: 40 days ago | 1 author | 1 change
private static ModeloValidacion ValideElFax(string? telefono)
1 reference | pc: 40 days ago | 1 author | 1 change
private static ModeloValidacion ValideLaDistribucionAdministrativa(InsumoParaProcesarFactura insumo, RecursosParaFacturar recursos)
1 reference | pc: 40 days ago | 1 author | 1 change
private static ModeloValidacion ValideLaObservacion(string observacion)
{
    const int CANTIDAD_MAXIMA_CARACTERES_NOMBRE = 300;
    ModeloValidacion resultado = new();
    string laObservacion = observacion.Trim();
    if (laObservacion.Length > CANTIDAD_MAXIMA_CARACTERES_NOMBRE)
        resultado = new($"No podrá continuar, la \"{observacion}\" supera los {CANTIDAD_MAXIMA_CARACTERES_NOMBRE} caracteres permitidos.");
    if (resultado.EsValido)
        resultado = RegLasComunes.ValideSiElTextoTieneCaracteresEspeciales(laObservacion, $"{observacion}");
    return resultado;
}
1 reference | pc: 40 days ago | 1 author | 1 change
private static ModeloValidacion ValideLaActividadComercial(string codActividad, List<Abstracciones.Contenedores.CatalogoActividadComercial.ActividadComercial> actividades)
1 reference | pc: 40 days ago | 1 author | 1 change
private static ModeloValidacion ValideLaMoneda(string codMoneda, List<Abstracciones.Contenedores.CatalogoMoneda.Moneda> monedas)
1 reference | pc: 40 days ago | 1 author | 1 change
private static ModeloValidacion ValideElCorreoElectronico(string? correo)
{
    ModeloValidacion resultado = new();
    string elCorreo = (correo ?? "").Trim();
    if (elCorreo != string.Empty)
    {
        string patron = "\\s+\\w+([-+.,~\\w]*)+@\\w+([-+.]\\w+)+\\.\\w+([-+.]\\w+)+\\s+";
        bool esValido = Regex.IsMatch(elCorreo, patron);
        if (esValido == false) resultado = new($"No podrá continuar, el \"{correo}\" no cumple con los requisitos mínimos de una dirección de correo electrónico.");
    }
    return resultado;
}
1 reference | pc: 40 days ago | 1 author | 1 change
private static ModeloValidacion ValideElCodMedioPago(string codMedioPago, List<Abstracciones.Contenedores.CatalogoMedioPago.MedioPago> mediosPago)

```

Fuente: Elaboración propia (2024).

En la figura anterior, se pueden visualizar algunas de las validaciones que se realizan al momento de emitir o tratar de generar una factura desde el prototipo funcional.

Pruebas

Scripts De Pruebas Unitarias

A continuación, se presenta una serie de scripts de pruebas unitarias que ilustran la validación de algunas funcionalidades del sistema. Estas pruebas están diseñadas para asegurar que cada componente del software opere conforme a las especificaciones establecidas, garantizando así su correcto funcionamiento y robustez. El objetivo principal de estas pruebas unitarias es verificar que los métodos del sistema retornen los resultados esperados cuando se les proporcionan entradas específicas, asegurando la integridad y precisión del código.

Tabla 106.

Script de prueba unitaria para encriptar información

Script de prueba unitaria para encriptar información.	Código del Script:	SC-01
Descripción: Este script verifica que el método “Encripta” de la clase “Encriptación” retorne un texto que no sea fácilmente reconocible o descifrable. El objetivo es proteger información sensible, como contraseñas, códigos de sesión y transacciones.		
Código: <pre>string elResultadoEsperado = "a4Kf0eN55ut62VVehoAdOIQ5POUEV4BaVr1ojrK4r/4=";</pre> <pre>string elTextoAEncriptar = "Texto sin encriptar";</pre> <pre>string elResultadoObtenido = Encriptacion.Encripta(elTextoAEncriptar);</pre> <pre>Assert.IsTrue(elResultadoEsperado.EsIgualA(elResultadoObtenido));</pre>		
Resultado esperado: Valor Esperado: “a4Kf0eN55ut62VVehoAdOIQ5POUEV4BaVr1ojrK4r/4=";		
Resultado Obtenido: Valor Obtenido: “a4Kf0eN55ut62VVehoAdOIQ5POUEV4BaVr1ojrK4r/4=";		

Fuente: Elaboración propia (2024).

Figura 90.

Evidencia método de prueba encriptar resultado válido

```
[TestMethod]
| 0 references | 0 changes | 0 authors, 0 changes
public void Encriptar_InValido()
{
    string elResultadoEsperado = "a4Kf0eN55ut62VVehoAdOIQ5POUEV4BaVr1ojrK4";
    string elTextoAEncriptar = "Texto sin encriptar";
    string elResultadoObtenido = Encriptacion.Encripta(elTextoAEncriptar);
    Assert.IsFalse(elResultadoEsperado.EsIgualA(elResultadoObtenido));
}
```

Fuente: Elaboración propia (2024).

Tabla 107.*Script de prueba unitaria para desencriptar información*

Script de prueba unitaria para desencriptar información.	Código del Script:	SC-02
Descripción: Este script verifica que el método “DesEncripta” de la clase “Encriptación” retorne un texto sea fácilmente reconocible a partir de un encriptado. El objetivo es poder desencriptar los datos para poder manipularlos o procesarlos dentro del sistema.		
Código: <pre>string elTextoADesEncriptar = "a4Kf0eN55ut62VVehAdOIQ5POUEV4BaVr1ojrK4r/4="; string elResultadoEsperado = "Texto sin encriptar"; string elResultadoObtenido = Encriptacion.DesEncripta(elTextoADesEncriptar); Assert.IsTrue(elResultadoEsperado.EsIgualA(elResultadoObtenido));</pre>		
Resultado esperado: Valor Esperado: Texto sin encriptar		
Resultado Obtenido: Valor Obtenido: Texto sin encriptar		

Fuente: Elaboración propia (2024).**Figura 91.***Evidencia método de prueba desencriptar resultado válido*

```
[TestMethod]
| 0 references | 0 changes | 0 authors, 0 changes
public void DesEncriptar_ResultadoValido()
{
    string elTextoADesEncriptar = "a4Kf0eN55ut62VVehAdOIQ5POUEV4BaVr1ojrK4r/4=";
    string elResultadoEsperado = "Texto sin encriptar";

    string elResultadoObtenido = Encriptacion.DesEncripta(elTextoADesEncriptar);

    Assert.IsTrue(elResultadoEsperado.EsIgualA(elResultadoObtenido));
}
```

Fuente: Elaboración propia (2024).

Tabla 108.*Script de prueba unitaria convertir números a letras*

Script de prueba unitaria para descriptar información.	Código del Script:	SC-03
Descripción: Este script convierte un valor numérico a una cadena de texto que representa dicho valor en palabras. En la representación gráfica en formato pdf de los documentos electrónicos es requerido mostrar el valor en letras del total del documento.		
Código: <pre>string elResultadoEsperado = "Sesenta y Dos Mil Quinientos Treinta y Cinco"; string elResultadoObtenido = ConviertaNumeroALetra.ConvertirNumeroALetras(62535); Assert.IsTrue(elResultadoEsperado.EsIgualA(elResultadoObtenido));</pre>		
Resultado esperado: True Valor Esperado: Sesenta y Dos Mil Quinientos Treinta y Cinco		
Resultado Obtenido: True Valor Obtenido: Sesenta y Dos Mil Quinientos Treinta y Cinco		

Fuente: Elaboración propia (2024).**Figura 92.***Evidencia método de prueba convertir número a letras 62535 resultado válido*

```
[TestMethod]
● | 0 references | 0 changes | 0 authors, 0 changes
public void ConvertirNumeroALetras_62535_ResultadoValido()
{
    string elResultadoEsperado = "Sesenta y Dos Mil Quinientos Treinta y Cinco";
    string elResultadoObtenido = ConviertaNumeroALetra.ConvertirNumeroALetras(62535);
    Assert.IsTrue(elResultadoEsperado.EsIgualA(elResultadoObtenido));
}
```

Fuente: Elaboración propia (2024).

Tabla 109.

Script de prueba unitaria obtenga y valide los datos ingreso al sistema válidos

Script de prueba unitaria para obtener y validar los datos de usuario al realizar el ingreso al sistema.	Código del Script:	SC-04
<p>Descripción: Este script obtiene los datos del usuario por el nombre de usuario y compañía y procede a validar la contraseña para posteriormente devolver un modelo de validación que contiene el mensaje de error en caso de ser inválido y la identidad del usuario para establecer un contexto de datos que se usara durante la permanencia del usuario en el sistema.</p>		
<p>Código:</p> <pre>InsumosIngreso datos Ingreso = new() { Compania = 1, Contraseña = "123", Usuario = "User1", ValidaSeleccionCompania = false }; DatosValidacionIngreso elResultadoEsperado = new() { EsValido = true, Mensaje = "", Usuario = ObtengaElClaimsPrincipalUsuarioValido() }; UsuarioIngreso usuario = ObtengaElUsuarioValido(); _easyCIFDataBase.Ingreso_ObtengaLosDatosDelUsuarioPorNombreUsuarioYCompania(datosIngreso.Usuario, datosIngreso.Compania).Returns(usuario); DatosValidacionIngreso elResultadoObtenido = await Ingreso.ObtengaYValideLosDatosDeIngresoAlSistema(datosIngreso); var claimsIdentity = (ClaimsIdentity)elResultadoObtenido!.Usuario!.Identity!; //El Código de sesión es un GUID por lo tanto se elimina del Claim a comparar. var claimAEliminar = claimsIdentity?.Claims.FirstOrDefault(claim => claim.Type == "CodSesion"); if (claimAEliminar != null) claimsIdentity!.RemoveClaim(claimAEliminar); Assert.IsTrue(elResultadoEsperado.EsIgualA(elResultadoObtenido));</pre>		
<p>Resultado esperado: EsValido: True Mensaje: ""</p>		
<p>Resultado Obtenido: Esvalido: True Mensaje: ""</p>		

Fuente: Elaboración propia (2024).

Figura 93.

Evidencia método de prueba obtenga y valide los datos de ingreso al sistema resultado válido

```
[TestMethod]
| 0 references | pc, 72 days ago | 2 authors, 2 changes
public async Task ObtengaYValideLosDatosDeIngresoAlSistema_Valido()
{
    InsumosIngreso datosIngreso = new() { Compania = 1, Contraseña = "123", Usuario = "User1", ValidaSeleccionCompania = false };
    DatosValidacionIngreso elResultadoEsperado = new() { EsValido = true, Mensaje = "", Usuario = ObtengaElClaimsPrincipalUsuarioValido() };

    Abstracciones.Contenedores.Ingreso.UsuarioIngreso usuario = ObtengaElUsuarioValido();
    _easyCIFDataBase.Ingreso.ObtengaLosDatosDelUsuarioPorNombreUsuarioYCompania(datosIngreso.Usuario, datosIngreso.Compania).Returns(usuario);

    DatosValidacionIngreso elResultadoObtenido = await Ingreso.ObtengaYValideLosDatosDeIngresoAlSistema(datosIngreso);

    var claimsIdentity = (ClaimsIdentity)elResultadoObtenido!.Usuario!.Identity!;

    //El Código de sesión es un GUID por lo tanto se elimina del Claim a comparar.
    var claimAEliminar = claimsIdentity?.Claims.FirstOrDefault(claim => claim.Type == "CodSesion");
    if (claimAEliminar != null)
        claimsIdentity!.RemoveClaim(claimAEliminar);

    Assert.IsTrue(elResultadoEsperado.EsIgualA(elResultadoObtenido));
}
```

Fuente: Elaboración propia (2024).

Tabla 110.

Script de prueba unitaria obtenga y valide los datos ingreso al sistema campo contraseña requerido

Script de prueba unitaria para obtener y validar los datos de usuario al realizar el ingreso al sistema; campo contraseña requerido.	Código del Script:	SC-05
<p>Descripción:</p> <p>Este script obtiene los datos del usuario por el nombre de usuario y compañía y procede a validar la contraseña para posteriormente devolver un modelo de validación que contiene el mensaje de error en caso de ser inválido y la identidad del usuario para establecer un contexto de datos que se usara durante la permanencia del usuario en el sistema.</p> <p>En los datos de ingreso no se proporciona la contraseña del usuario por lo que se rechaza el ingreso al sistema y se le indica que el campo es requerido.</p>		
<p>Código:</p> <pre>InsumosIngreso datosIngreso = new() { Compania = 0, Contraseña = "123", Usuario = "User1" }; DatosValidacionIngreso elResultadoEsperado = new("El campo \"Compañía\" es requerido."); DatosValidacionIngreso elResultadoObtenido = await Ingreso.ObtengaYValideLosDatosDeIngresoAlSistema(datosIngreso); Assert.IsTrue(elResultadoEsperado.EsIgualA(elResultadoObtenido, true));</pre>		
<p>Resultado esperado:</p> <p>EsValido: False Mensaje: “El campo “Compañía” es requerido.”</p>		
<p>Resultado Obtenido:</p> <p>Esvalido: False Mensaje: “El campo “Compania” es requerido.”</p>		

Fuente: Elaboración propia (2024).

Figura 94.

Evidencia método de prueba obtenga y valide los datos ingreso al sistema campo contraseña requerido

```
[TestMethod]
0 references | AdminPC, 105 days ago | 1 author, 1 change
public async Task ObtengaYValideLosDatosDeIngresoALsistema_Compania_NoSeleccionada()
{
    InsumosIngreso datosIngreso = new() { Compania = 0, Contraseña = "123", Usuario = "User1" };
    DatosValidacionIngreso elResultadoEsperado = new("El campo \"Compañía\" es requerido.");
    DatosValidacionIngreso elResultadoObtenido = await Ingreso.ObtengaYValideLosDatosDeIngresoALsistema(datosIngreso);
    Assert.IsTrue(elResultadoEsperado.EsIgualA(elResultadoObtenido, true));
}
```

Fuente: Elaboración propia (2024).

Tabla 111.

Script de prueba unitaria buscar categorías sin resultados

Script de prueba unitaria para buscar categorías sin resultados	Código del Script:	SC-06
<p>Descripción: Este script prueba la búsqueda de categorías que se realiza desde el mantenimiento de categorías y desde el mantenimiento de artículos. De acuerdo con los parámetros ingresado la búsqueda no retorna datos.</p>		
<p>Código:</p> <pre>IEnumerable<CategoriaListar> elResultadoEsperado = new List<CategoriaListar>(); ParametrosBusquedaCategoria parametro = new() { Estado = 2, FiltroOpciones = "FiltroInicieCategoria", IdCompania = 1, LlaveBusqueda = "Abc" }; _easyCIFDataBase.ObtengaLasCategoriaPorParametrosDeBusquedaAsync(Arg.Is<ParametrosBusquedaCategoria>(x => x.EsIgualA(parametro, false))).Returns(new List<CategoriaListar>()); IEnumerable<CategoriaListar> elResultadoObtenido = await _IBuscarCategoria.BuscarLasCategorias(parametro); Assert.IsTrue(elResultadoEsperado.EsIgualA(elResultadoObtenido, true));</pre>		
<p>Resultado esperado: Sin datos. IEnumerable<CategoriaListar> elResultadoEsperado = new List<CategoriaListar>();</p>		
<p>Resultado Obtenido: Sin datos IEnumerable<CategoriaListar> elResultadoEsperado = new List<CategoriaListar>(); La prueba es exitosa.</p>		

Fuente: Elaboración propia (2024).

Figura 95.

Evidencia método de prueba buscar categorías sin resultados

```

[TestMethod]
public async Task BuscarLasCategorias_SinResultados()
{
    IEnumerable<Categorialistar> elResultadoEsperado = new List<Categorialistar>();

    ParametrosBusquedaCategoria parametro = new()
    {
        Estado = 2,
        FiltroOpciones = "FiltroInicieCategoria",
        IdCompania = 1,
        LlaveBusqueda = "Abc"
    };

    _easyCIFDataBase.Obtengalascategoriasporparametrosdebusquedaasync(Arg.Is<ParametrosBusquedaCategoria>(x => x.EsIgualA(parametro, false))).
    Returns(new List<Categorialistar>());

    IEnumerable<Categorialistar> elResultadoObtenido = await _IBuscarCategoria.BuscarLasCategorias(parametro);
    Assert.IsTrue(elResultadoEsperado.EsIgualA(elResultadoObtenido, true));
}

```

Fuente: Elaboración propia (2024).

Tabla 112.

Script de prueba unitaria valide que el asiento contable este balanceado resultado inválido

Script de prueba unitaria para validar que el asiento contable este balanceado resultado inválido,	Código del Script:	SC-07
Descripción:		
Este script prueba que no se puedan registrar asiento contables si no se encuentran balanceados el monto del debe y el monto del haber.		
Código:		
decimal elTotalDebe = 123560; decimal elTotalHaber = 123580; ModeloValidacion elResultadoEsperado = new(); ModeloValidacion elResultadoObtenido = ReglasIngresoAsientos.ValideQueElAsientoContableSeEncuentreBalanceado(elTotalDebe, elTotalHaber); Assert.IsTrue(elResultadoEsperado.EsIgualA(elResultadoObtenido));		
Resultado esperado:		
ModeloValidacion elResultadoEsperado = new() { EsValido=false, Mensaje= "No podrá continuar, el asiento no se encuentra balanceado." };		
Resultado Obtenido:		
ModeloValidacion elResultadoEsperado = new() { EsValido=false, Mensaje= "No podrá continuar, el asiento no se encuentra balanceado." };		

Fuente: Elaboración propia (2024).

Figura 96.

Evidencia método de prueba valide que el asiento contable se encuentra balanceado inválido

```
[TestMethod]
| 0 references | 0 changes | 0 authors, 0 changes
public void ValideQueElAsientoContableSeEncuentreBalanceado_Invalido()
{
    decimal elTotalDebe = 123560;
    decimal elTotalHaber = 123580;
    ModeloValidacion elResultadoEsperado = new() { EsValido = false, Mensaje = "No podrá continuar, el asiento no se encuentra balanceado." };
    ModeloValidacion elResultadoObtenido = ReglasIngresoAsientos.ValideQueElAsientoContableSeEncuentreBalanceado(elTotalDebe, elTotalHaber);
    Assert.IsTrue(elResultadoEsperado.EsIgualA(elResultadoObtenido));
}
```

Fuente: Elaboración propia (2024).

Tabla 113.

Script de prueba unitaria valide el saldo del artículo al facturar inválido

Script de prueba unitaria para validar el saldo del artículo al facturar.	Código del Script:	SC-08
Descripción: Este script prueba que no se puedan facturar artículos cuya existencia es cero en bodega.		
Código: double saldoDelArticulo = 0; double cantidadAFacturar = 10; ModeloValidacion elResultadoEsperado = new() { EsValido = false, Mensaje = "No podrá continuar; el artículo no cuenta con el saldo suficiente." }; ModeloValidacion elResultadoObtenido = ReglasFacturacion.ValideElSaldoDelArticulo(saldoDelArticulo, cantidadAFacturar); Assert.IsTrue(elResultadoEsperado.EsIgualA(elResultadoObtenido));		
Resultado esperado: ModeloValidacion elResultadoEsperado = new() { EsValido = false, Mensaje = "No podrá continuar; el artículo no cuenta con el saldo suficiente." };		
Resultado Obtenido: ModeloValidacion elResultadoEsperado = new() { EsValido = false, Mensaje = "No podrá continuar; el artículo no cuenta con el saldo suficiente." };		

Fuente: Elaboración propia (2024).

Figura 97.

Evidencia método de prueba valide que el asiento contable se encuentra balanceado inválido

```
[TestMethod]
| 0 references | 0 changes | 0 authors, 0 changes
public void ValideElSaldoDelArticulo_Invalido()
{
    double saldoDelArticulo = 0;
    double cantidadAFacturar = 10;
    ModeloValidacion elResultadoEsperado = new() { EsValido = false, Mensaje = "No podrá continuar; el articulo no cuenta con el saldo suficiente." };
    ModeloValidacion elResultadoObtenido = ReglasFacturacion.ValideElSaldoDelArticulo(saldoDelArticulo, cantidadAFacturar);
    Assert.IsTrue(elResultadoEsperado.EsIgualA(elResultadoObtenido));
}
```

Fuente: Elaboración propia (2024).

Casos De Prueba

A continuación, se presentan una serie de casos de prueba diseñados para validar las diversas funcionalidades del sistema. Estos casos de prueba incluyen datos de entrada, acciones a ejecutar y resultados esperados, proporcionando un método estructurado para evaluar cómo el *software* responde en diferentes situaciones. La implementación de estos casos de prueba asegura que el sistema cumpla con los requisitos definidos y opere de manera confiable y efectiva.

Tabla 114.

Caso de prueba de ingreso al sistema y cierre de sesión, usuario estándar con acceso al módulo de facturación

Caso de prueba de ingreso al sistema y cierre de sesión, usuario estándar con acceso al módulo de facturación	Código CP:	CP-01
<p>Descripción: Este caso de prueba se centra en verificar el proceso de ingreso y cierre de sesión de un usuario con perfil estándar en el sistema. Durante la prueba de ingreso, se valida que al usuario no se le solicite la selección del sistema ya que únicamente cuenta con acceso al módulo llamado CIF que contiene los procesos: compras, inventario y facturación. Además, se prueba que el sistema permita al usuario acceder únicamente a las opciones autorizadas según su rol, denegando el acceso a cualquier opción para la cual no esté autorizado. Durante el cierre de sesión, se verifica que el usuario pueda finalizar la sesión de manera adecuada y que se le redirija a la página de inicio de sesión de manera segura.</p>		
<p>Prerrequisitos:</p> <ol style="list-style-type: none"> 1. Se debe contar con un usuario válido con el estado “Activo”. 2. El usuario debe estar asociado a un solo módulo (CIF) y una sola sucursal (San José). 3. El usuario debe conocer el nombre de usuario y contraseña. 4. El usuario debe contar con el rol “Usuario Estándar”. 5. El usuario debe contar con acceso a internet. 		
<p>Pasos:</p> <p>Acción ingresar al sistema.</p> <ol style="list-style-type: none"> 1. Ingresar al sistema web desde cualquier navegador. Url: https://easycif-001-site1.gtempurl.com/ 2. Ingresar las credenciales de usuario en los siguiente campos: Usuario, Contraseña. 3. Presionar el botón “Ingresar” <p>Acción ingresar a una opción con acceso restringido por rol de usuario.</p> <ol style="list-style-type: none"> 1. Inicia en el paso 3 de la acción ingresar al sistema. 2. Ingresar a la siguiente opción del menú: Facturación > Procesos > Anulación. <p>Acción Cerrar sesión</p> <ol style="list-style-type: none"> 1. Inicia en el paso 3 de la acción ingresar al sistema. 2. Seleccionar la opción Cerrar Sesión. 		

Resultados esperados:

1. Se espera que el sistema permita el ingreso; si el usuario y contraseña son correctos, y si el usuario está habilitado según su estado y rol.
2. Se espera que el sistema no muestre la pantalla de sistema en donde se solicita seleccionar el módulo y la sucursal a la cual se desea ingresar, debido a que el usuario solo está relacionado a un solo módulo y sistema.
3. Se espera que al intentar ingresar a una opción no permitida según el rol de usuario el sistema muestre la pantalla de acceso denegado.
4. Se espera que el sistema permita al usuario cerrar la sesión.

Resultados obtenidos:

1. El sistema permite el ingreso del usuario sistema y muestra el menú asignado al módulo CIF el cual contiene los accesos para: facturación, anulación, devoluciones, órdenes de trabajo, compras, mantenimiento de artículo, etc.
2. El sistema no muestra la pantalla de selección de sistema, se redirecciona al usuario directamente a la pantalla principal del módulo asignado.
3. El sistema mostró la pantalla de Acceso Denegado !!! cuando el usuario con el rol Usuario Estándar intentó ingresar a la opción Anulación la cual está habilitada únicamente para usuarios con el rol administrador.
4. Al cerrar sesión el sistema eliminó los datos de la sesión y mostró nuevamente la pantalla de ingreso al sistema.

Fuente: Elaboración propia (2024).

Tabla 115.

Caso de prueba de ingreso al sistema y cierre de sesión, usuario Administrador con acceso total todos los módulos y sucursales

Caso de prueba de ingreso al sistema y cierre de sesión, usuario Administrador con acceso total todos los módulos y sucursales	Código CP:	CP-02
<p>Descripción: Este caso de prueba se centra en verificar el proceso de ingreso y cierre de sesión de un usuario con el rol Administrador en el sistema. Durante la prueba de ingreso, se valida que al usuario se le solicite la selección del sistema ya cuenta con acceso global. Además se corrobora que el usuario pueda ingresar a la pantalla de compras que está habilitada únicamente para usuarios con rol Administrador. Durante el cierre de sesión, se verifica que el usuario pueda finalizar la sesión de manera adecuada y que se le redirija a la página de inicio de sesión de manera segura.</p>		
<p>Prerrequisitos:</p> <ol style="list-style-type: none"> 1. Se debe contar con un usuario válido con el estado “Activo”. 2. El usuario debe tener acceso a todos los módulos y sucursales registradas en el sistema. 3. El usuario debe conocer el nombre de usuario y contraseña. 4. El usuario debe contar con el rol “Usuario Administrador”. 5. El usuario debe contar con acceso a internet. 		

Pasos:**Flujo ingresar al sistema.**

1. Ingresar al sistema web desde cualquier navegador.
Url: <https://easycif-001-site1.gtempurl.com/>
2. Ingresar las credenciales de usuario en los siguiente campos: Usuario, Contraseña.
3. Presionar el botón “Ingresar”
4. Seleccionar el módulo o sistema al cual desea ingresar para el caso de prueba ingresar al módulo CIF.
5. Seleccionar la sucursal.
6. Presionar el botón continuar.

Flujo ingresar a una opción con acceso restringido por rol de usuario.

1. Inicia en el paso 6 de la acción ingresar al sistema.
2. Ingresar a la siguiente opción del menú: Compras > Compras.

Flujo Cerrar sesión

1. Inicia en el paso 6 de la acción ingresar al sistema.
2. Seleccionar la opción Cerrar Sesión.

Resultados esperados:

1. Se espera que el sistema permita el ingreso si el usuario y contraseña son correctos, y si el usuario está habilitado según su estado y roles.
2. Se espera que el sistema muestre la pantalla de sistema en donde se solicita seleccionar el módulo (CIF, CxC, CxP, CONTABILIDAD) y la sucursal (San José, Escazú).
3. Se espera que el ingresar a la opción de registro de compras el sistema permita el ingreso y muestra la pantalla.
4. Se espera que el sistema permita al usuario cerrar la sesión.

Resultados obtenidos:

1. El sistema permite el ingreso del usuario sistema. }
2. El sistema mostro la pantalla de sistema para la selección de módulo y sucursal, y le permitió al usuario continuar hasta la pantalla principal de módulo seleccionado.
3. El sistema permite el ingreso del usuario a la pantalla de registro de compras.
4. Al cerrar sesión el sistema elimino los datos de la sesión y mostró nuevamente la pantalla de ingreso al sistema.

Fuente: Elaboración propia (2024).

Tabla 116.

Caso de prueba solicitud de cambio de contraseña al primer ingreso

Caso de prueba solicitud de cambio de contraseña al primer ingreso	Código CP:	CP-03
<p>Descripción: Este caso de prueba verifica que sea obligatorio el cambiar la contraseña de usuario cuando se ingresa por primera vez al sistema, o si la contraseña es reseteada por un usuario administrador.</p>		
<p>Prerrequisitos:</p> <ol style="list-style-type: none"> 1. Ingresar por primera vez al sistema con usuario nuevo. 2. Solicitar a un usuario administrador que le asigne una contraseña nueva porque la olvido. 3. El usuario debe conocer el nombre de usuario y contraseña actual que fue asignada por el usuario administrador. 4. El usuario debe contar con acceso a internet. 		
<p>Pasos:</p> <p>Flujo base ingresar al sistema por primera vez</p> <ol style="list-style-type: none"> 1. El usuario administrador crea un nuevo registro de usuario y le proporciona las credenciales al usuario nuevo. 2. Ingresar al sistema web desde cualquier navegador. Url: https://easycif-001-site1.gtempurl.com/ 3. Ingresar las credenciales de usuario en los siguiente campos: Usuario, Contraseña. 4. Presionar el botón “Ingresar”. 5. Ingresar y confirmar la nueva contraseña la cual debe ser robusta. 6. Presionar el botón Cambiar la contraseña. <p>Flujo base ingresar al sistema posterior a un cambio de contraseña realizado por un usuario administrador.</p> <ol style="list-style-type: none"> 1. Realizar el ingreso al sistema con un usuario administrador. 2. Modificar la contraseña de un usuario existente. 3. Proporcionar las credencial modiflicas al usuario que solicito el cambio. 4. Cerrar sesión. 5. Ingresar al sistema con el usuario al cual se le modifico la contraseña. 6. Ingresar y confirmar la nueva contraseña la cual debe ser robusta. 7. Presionar el botón Cambiar la contraseña. 		
<p>Resultados esperados:</p> <ol style="list-style-type: none"> 1. Se espera que la primera vez que un usuario nuevo ingrese al sistema y digite las credenciales que le proporciono un usuario administrador el sistema le solicite obligatoriamente el cambio de contraseña. 2. Que se cumplan todas validaciones del caso uso “Cambiar contraseña al ingresar”. 3. Que el sistema permita el cambio de contraseña. 		

Resultados obtenidos:

1. El sistema permite el ingreso del usuario sistema.
2. El sistema solicito el cambio de contraseña al primer ingreso y al ingreso posterior al cambio de contraseña realizada por un usuario administrador,
3. El sistema permitió el ingreso a la opciones de menú de sistema una vez que se modificó la contraseña.
4. El sistema ejecuto las validaciones del caso de uso.

Fuente: Elaboración propia (2024).

Tabla 117.

Caso de prueba administrar compañías

Caso de prueba administrar compañías	Código CP:	CP-04
<p>Descripción: Este caso de prueba se busca realizar el proceso completo de ingreso o actualización de un registro en el mantenimiento de compañías, permitiendo agregar y modificar los datos de la compañía. El mantenimiento de compañías es uno de los más importantes en todo el sistema debido que es donde se ingresa los datos necesario para poder emitir documentos electrónicos.</p>		
<p>Prerrequisitos:</p> <ol style="list-style-type: none"> 1. Contar con un usuario válido con el rol “Administrador”. 2. Estar inscrito en sistema del Ministerio de Hacienda como emisor de factura electrónica. 3. Contar con el usuario, contraseña y certificado de firma electrónica de la plataforma del Ministerio de hacienda. https://atv.hacienda.go.cr/ 		
<p>Pasos: Caso de uso: Administrar compañías.</p> <p>Flujo agregar datos.</p> <ol style="list-style-type: none"> 1. Ingresar al sistema. 2. Ejecutar el flujo base del caso de uso. <p>Flujo modificar datos</p> <ol style="list-style-type: none"> 1. Ingresar al sistema. 2. Ejecutar el sub flujo Buscar. 3. Ejecutar el subflujo Editar. 4. Continuar en el paso 2 del flujo base. <p>Flujo validaciones</p> <ol style="list-style-type: none"> 1. Ejecutar el sub flujo validaciones. 		
<p>Resultados esperados:</p> <ol style="list-style-type: none"> 1. Se espera que el usuario permite registrar la información para una nueva compañía. 2. Se espera que el sistema permita al usuario consultar las compañías que se encuentran registradas en el sistema. 3. Se espera que el sistema permita al usuario seleccionar un registro del resultado de búsqueda. 		

4. Se espera que el sistema permita al usuario modificar la información.
5. Se espere que el sistema ejecute el flujo validaciones.

Resultados obtenidos:

1. El sistema permitió agregar datos.
2. El sistema permitió realizar la búsqueda de la compañías según el criterio de búsqueda ingresado por el usuario.
3. El sistema permitió al usuario modificar los datos en pantalla y actualizarlos en la base de datos.
4. Se ejecutaron la validaciones exitosamente:
 - a. Datos requeridos.
 - b. Formato identificación.
 - c. Estructura de los correos electrónicos ingresados.
 - d. La clave del certificado digital.
 - e. El usuario y contraseña de hacienda.
 - f. Que no existan registros repetidos.

Fuente: Elaboración propia (2024).

Tabla 118.

Caso de prueba administrar categorías

Caso de prueba administrar categorías	Código CP:	CP-05
<p>Descripción: Este caso de prueba se busca realizar el proceso completo de ingreso o actualización de un registro en el mantenimiento de categorías, permitiendo agregar y modificar los datos de la categoría.</p>		
<p>Prerrequisitos:</p> <ol style="list-style-type: none"> 1. Contar con un usuario válido con el rol “Administrador”. 2. Ingresar al sistema. 		
<p>Pasos: Caso de uso: Administrar categorías. Flujo agregar datos.</p> <ol style="list-style-type: none"> 1. Ingresar al sistema. 2. Ejecutar el flujo base del caso de uso. <p>Flujo modificar datos</p> <ol style="list-style-type: none"> 1. Ingresar al sistema. 2. Ejecutar el sub flujo Buscar. 3. Ejecutar el subflujo Editar. 4. Continuar en el paso 2 del flujo base. <p>Flujo validaciones</p> <ol style="list-style-type: none"> 1. Ejecutar el sub flujo validaciones. 		
<p>Resultados esperados:</p> <ol style="list-style-type: none"> 1. Se espera que el usuario permite registrar la información para una nueva categoría. 		

<ol style="list-style-type: none"> 2. Se espera que el sistema permita al usuario consultar las categorías que se encuentran registradas en el sistema. 3. Se espera que el sistema permita al usuario seleccionar un registro del resultado de búsqueda. 4. Se espera que el sistema permita al usuario modificar la información. 5. Se espere que el sistema ejecute el flujo validaciones.
<p>Resultados obtenidos:</p> <ol style="list-style-type: none"> 1. El sistema permitió agregar datos. 2. El sistema permitió realizar la búsqueda de categorías según el criterio de búsqueda ingresado por el usuario. 3. El sistema permitió al usuario modificar los datos en pantalla y actualizarlos en la base de datos. 4. Se ejecutaron la validaciones exitosamente: <ol style="list-style-type: none"> a. Datos requeridos. b. Que no existan registros repetidos.

Fuente: Elaboración propia (2024).

Tabla 119.

Caso de prueba administrar artículos y servicios

Caso de prueba administrar categorías	Código CP:	CP-06
<p>Descripción: Este caso de prueba se busca realizar el proceso completo de ingreso o actualización de un registro en el mantenimiento de artículos y servicios, permitiendo agregar y modificar los datos de la categoría.</p>		
<p>Prerrequisitos:</p> <ol style="list-style-type: none"> 1. Contar con un usuario válido con el rol “Administrador”. 2. Ingresar al sistema. 		
<p>Pasos: Caso de uso: Administrar artículos y servicios. Flujo agregar datos. <ol style="list-style-type: none"> 1. Ingresar al sistema. 2. Ejecutar el flujo base del caso de uso. Flujo modificar datos <ol style="list-style-type: none"> 5. Ingresar al sistema. 1. Ejecutar el sub flujo Buscar. 2. Ejecutar el subflujo Editar. 3. Continuar en el paso 2 del flujo base. Flujo validaciones <ol style="list-style-type: none"> 1. Ejecutar el sub flujo validaciones. </p>		
<p>Resultados esperados:</p> <ol style="list-style-type: none"> 1. Se espera que el usuario permite registrar la información para una nuevo artículo. 2. Se espera que el sistema permita al usuario consultar los artículos que se encuentran registradas en el sistema. 3. Se espera que el sistema permita al usuario seleccionar un registro del resultado de búsqueda. 4. Se espera que el sistema permita al usuario modificar la información. 		

5. Se espere que el sistema ejecute el flujo validaciones.
<p>Resultados obtenidos:</p> <ol style="list-style-type: none"> 1. El sistema permitió agregar datos. 2. El sistema permitió realizar la búsqueda de artículos y servicios según el criterio de búsqueda ingresado por el usuario. 3. El sistema permitió al usuario modificar los datos en pantalla y actualizarlos en la base de datos. 4. Se ejecutaron la validaciones exitosamente: <ol style="list-style-type: none"> a. Datos requeridos. b. Que no existan registros repetidos. c. Al editar que no se puedan desasociar el artículo de bodegas que contienen saldos. d. Se verifica que los campos no contengan información inválida o con caracteres especiales ya no son permitidos en la generación de los archivos Xml. Caracteres como < & no son permitidos. e. Qué los artículos compuestos no se puedan asociar a bodegas.

Fuente: Elaboración propia (2024).

Tabla 120.

Caso de prueba generar factura electrónica

Caso de prueba administrar categorías	Código CP:	CP-07
<p>Descripción:</p> <p>Este caso de prueba se busca realizar el proceso completo de facturación para el cual se cuenta con dos tipos de comprobantes electrónicos, la factura y el ticket, la diferencia entre ambos es que el ticket no se puede utilizar como crédito fiscal por lo tanto es opcional que contenga la información del cliente. También se busca corroborar que el proceso de facturación realice todas las operaciones básicas, como lo son el cálculo de impuestos, descuentos, totales, el rebajo del inventario, la inserción del Kardex, la actualización de las cuentas por cobrar y la generación del asiento contable.</p>		
<p>Prerrequisitos:</p> <ol style="list-style-type: none"> 1. Contar con un usuario válido con el rol “Administrador”. 2. Ingresar al sistema. 3. Completar la información de la compañía en el mantenimiento compañía. 4. Crear al menos una sucursal. 5. Deben existir los artículos que se van a facturar. 6. Si es una factura electrónica de crédito es obligatorio que el cliente esté registrado desde el mantenimiento de clientes. 		
<p>Pasos:</p> <p>Flujo generar factura</p> <ol style="list-style-type: none"> 1. Ingresar al sistema. 2. Ejecutar el flujo base del caso de uso generar facturación. 		
<p>Resultados esperados:</p> <ol style="list-style-type: none"> 1. El sistema permita generar la factura de con los campos requeridos de acuerdo con el tipo de documento seleccionado. 2. Que no sea obligatorio ingresar un cliente para los tickets electrónicos. 		

3. Que para la facturación de crédito sea obligatorio ingresar el cliente.
4. Que si un cliente cuenta con perfil que posee un descuento global este se aplique al total de la factura.
5. Que si la factura es crédito se cree el registro de cuentas por cobrar.
6. Que permita al usuario agregar productos o servicios al detalle de la factura.
7. Que permita al usuario modificar o eliminar líneas del detalle.
8. Que permita al usuario modificar la descripción del artículo en caso de una solicitud en particular o si no quiere que se consoliden líneas en el detalle.
9. Que permita al usuario digitar el precio para los servicios y artículos que no afectan el inventario.
10. Que si el usuario cuenta con rol Usuario Estándar no se le permita ingresar descuentos, facturar a crédito.
11. Que no se permita la facturación de artículos que afectan el inventario que no cuenten con saldo en bodega.
12. Si selecciono un tipo de identificación diferente a cliente de contado el sistema valide los datos del cliente como requeridos y valide el formato de la identificación.
13. Que el sistema realice la búsqueda del nombre del cliente por identificación aunque este no exista la base de datos del cliente, que la consulta la realice contra el padrón de identificaciones nacionales.
14. Que el sistema valide la estructura del campo correo electrónico.
15. Que si el cliente es un cliente exonerado se pueden visualizar los datos en la sección otros datos.
16. Que el sistema permita guardar el pedido en caso que no se requiera emitir la factura al instante.
17. Al cargar un pedido que el sistema no cargue aquellas líneas que no cuenten con saldo.
18. Que se realicen correctamente el cálculo de totales.
19. Que valide el ingreso de caracteres especiales en los campos de entrada.
20. Que al facturar registre la factura, rebaje el inventario, registre el Kardex de artículo, genere la cuenta por cobrar en caso de ser de crédito, que se cree el asiento contable, que se cree y se firme el comprobante electrónico, que se envíe, se valide y se consulte el documento contra la plataforma del ministerio de hacienda.
21. Que se imprima en pantalla la representación gráfica de la factura.
22. Que se envíe un correo al cliente y al emisor de la factura electrónica con los archivos xml y con la representación gráfica.

Resultados obtenidos:

Se genero por sistema una factura por cada tipo de documento y condición venta, se procedió a verificar los saldos del inventario, el registro de cuentas por cobrar, los asientos contables, los documentos electrónicos enviados al correo electrónico y se certificó que cada una de la validaciones y los resultados esperados se ejecutaran con éxito.

Fuente: Elaboración propia (2024).

REFERENCIAS

- Álvarez, P. G. (29 de Julio de 2021). *¿Qué es un módulo?* Obtenido de <https://ainotes298786558.wordpress.com/2021/07/29/que-es-un-modulo/>
- Amazon. (s.f.). *¿Qué es JavaScript?* Obtenido de [aws.amazon.com: https://aws.amazon.com/es/what-is/javascript/](https://aws.amazon.com/es/what-is/javascript/)
- Amazon. (s.f.). *¿Qué es un IDE (entorno de desarrollo integrado)?* Obtenido de [aws.amazon.com: https://aws.amazon.com/es/what-is/ide/](https://aws.amazon.com/es/what-is/ide/)
- Anderson, T. (2 de Marzo de 2021). *The Register*. Obtenido de https://www.theregister.com/2021/03/02/microsoft_debuts_lowcode_power_fx/
- Asamblea Legislativa de la República de Costa Rica. (1992, Abril 03). *Ley Reguladora de Exoneraciones Vigentes, Derogatorias y Excepciones*. San José Costa Rica. Obtenido de http://www.pgrweb.go.cr/scij/Busqueda/Normativa/Normas/nrm_texto_completo.aspx?nValor1=1&nValor2=32135
- Assembler Institute. (31 de Marzo de 2022). *¿Que es un Framework en programación y sus principales usos?* Obtenido de [assemblerinstitute.com: https://assemblerinstitute.com/blog/framework-programacion/](https://assemblerinstitute.com/blog/framework-programacion/)
- Beynon-Davies, P. (2018). *Sistemas de información: Introducción a la informática en las organizaciones*. España: Editorial Reverté.
- BillWagner. (15 de Febrero de 2023). *Paseo por el lenguaje C#*. Obtenido de [learn.microsoft.com: https://learn.microsoft.com/es-es/dotnet/csharp/tour-of-csharp/](https://learn.microsoft.com/es-es/dotnet/csharp/tour-of-csharp/)
- Blasco, J., & Pérez, J. (2008). *Metodologías de investigación en educación física y deporte*. En C. U. Gamma (Ed.). España.
- campusMVP . (11 de Febrero de 2020). *Acceso a datos con .NET: Diferencias entre Entity Framework 6 “tradicional” y Entity Framework Core*. Obtenido de [www.campusmvp.es: https://www.campusmvp.es/recursos/post/que-diferencias-hay-entre-entity-framework-6-tradicional-y-entity-framework-core.aspx](https://www.campusmvp.es/recursos/post/que-diferencias-hay-entre-entity-framework-6-tradicional-y-entity-framework-core.aspx)
- Coufano, G. (12 de Agosto de 2023). *Diagrama de contexto*. Obtenido de [FourWeekMBA.: https://fourweekmba.com/es/diagrama-contextual/](https://fourweekmba.com/es/diagrama-contextual/)
- da Silva, D. (04 de enero de 2023). *¿Qué es gestión de ventas en una empresa?: guía para iniciantes*. Obtenido de [Zendesk MX.: https://www.zendesk.com.mx/blog/gestion-de-ventas/](https://www.zendesk.com.mx/blog/gestion-de-ventas/)
- Dirección General de Tributación Directa. (2019, Julio 01). *Resolución general sobre las disposiciones técnicas de los comprobantes electrónicos para efectos tributarios N° DGT-R-033-2019*. Obtenido de

- http://www.pgrweb.go.cr/scij/Busqueda/Normativa/Normas/nrm_texto_completo.aspx?nValor1=1&nValor2=89094
- Dobaño, R. (2 de Agosto de 2023). *Libro diario: qué es y cómo hacerlo*. Obtenido de Quipu Blog: <https://getquipu.com/blog/libro-diario/>
- Euroinnova Business School. (24 de Julio de 2022). *¿Qué son los pasivos y activos en contabilidad?* Obtenido de euroinnova.edu.es: <https://gbm.com/academy/que-es-un-activo-definicion-tipos-y-como-se-calcula/>
- Galindo, E. M. (09 de Marzo de 2018). *Metodología de investigación, pautas para hacer Tesis*. Obtenido de tesis-investigacion-cientifica.blogspot.com: <https://tesis-investigacion-cientifica.blogspot.com/2018/03/definicion-instrumental-de-las-variables.html>
- García, J. O. (2017). *Contabilidad General* (Vol. 631). Deusto.
- GBM. (18 de Febrero de 2022). *¿Qué es un activo?: definición, tipos y cómo se calcula*. Obtenido de gbm.com: <https://gbm.com/academy/que-es-un-activo-definicion-tipos-y-como-se-calcula/>
- GS1. (s.f.). *Factura Electrónica*. Obtenido de www.gs1cr.org: <https://www.gs1cr.org/factura-electronica/>
- Guzmán, C. (24 de Enero de 2022). *¿Qué es el Balance General?* Obtenido de ceupe: <https://www.ceupe.mx/blog/que-es-el-balance-general.html>
- Herranz, J. I. (13 de Abril de 2023). *TDD como metodología de diseño de software*. Obtenido de paradigmadigital.com: <https://www.paradigmadigital.com/dev/tdd-como-metodologia-de-diseno-de-software/>
- Herrera, J. (02 de Septiembre de 2020). *Patrón de diseño MVC. ¿Qué es y cómo puedo utilizarlo?* Obtenido de Easy App CODE: <https://www.easyappcode.com/patron-de-diseno-mvc-que-es-y-como-puedo-utilizarlo>
- Huaman, W. C. (7 de Septiembre de 2018). *Los 10 patrones comunes de arquitectura de software*. Obtenido de Medium: <https://medium.com/@maniakhitoccori/los-10-patrones-comunes-de-arquitectura-de-software-d8b9047edf0b>
- IBM. (s.f.). *¿Qué es la gestión de inventario?* Obtenido de ibm.com: <https://www.ibm.com/es-es/topics/inventory-management>
- IBM. (08 de Agosto de 2021). *Claves primarias y foráneas*. Obtenido de www.ibm.com: <https://www.ibm.com/docs/es/ida/9.1.2?topic=entities-primary-foreign-keys>
- IONOS. (24 de Julio de 2020). *El diagrama de casos de uso en UML*. Obtenido de IONOS Digital Guide: <https://www.ionos.es/digitalguide/paginas-web/desarrollo-web/diagrama-de-casos-de-uso/>
- Kaner. (17 de Mayo de 2015). *Definición de testing CEM Kaner*. Obtenido de kaner.com: <http://www.kaner.com/pdfs/KanerSocialScienceSTEP.pdf>

- keepcoding. (26 de Septiembre de 2022). *¿Qué es un modelo entidad-relación?* Obtenido de keepcoding BootCamps: <https://keepcoding.io/blog/modelo-entidad-relacion/>
- Ken, A. (16 de 08 de 2023). *Requisitos no funcionales: ¿Por qué son importantes?* Obtenido de Gluo: <https://www.gluo.mx/blog/requisitos-no-funcionales-por-que-son-importantes>
- Lucidchart. (s.f.). *¿Qué es el lenguaje unificado de modelado (UML)?* Obtenido de www.lucidchart.com: <https://www.lucidchart.com/pages/es/que-es-el-lenguaje-unificado-de-modelado-uml>
- lucidchart. (s.f.). *¿Qué es un diagrama entidad-relación?* Obtenido de www.lucidchart.com: <https://www.lucidchart.com/pages/es/que-es-un-diagrama-entidad-relacion>
- María Silvestrini Ruiz, & Jacqueline Vargas Jorge. (Enero de 2008). *Fuentes de información Primarias, Secundarias y Terciarias*. Obtenido de ponce.inter.edu: <https://ponce.inter.edu/cai/manuales/FUENTES-PRIMARIA.pdf>
- Marquez, C. N. (2013). *Proyecto de inversión. Creación de páginas web para potenciar ventas de emprendimientos*. México: Universidad de Guayaquil.
- Martín, M. J. (22 de Noviembre de 2018). *SOLID: los 5 principios que te ayudarán a desarrollar software de calidad*. Obtenido de <https://profile.es/blog/principios-solid-desarrollo-software-calidad/>
- Martin, S. (10 de Agosto de 2020). *Las ventajas de crear prototipos para tu proyecto digital*. Blog de Hiberus Tecnología. Obtenido de Hiberus: <https://www.hiberus.com/crecemos-contigo/las-ventajas-de-crear-prototipos-para-tu-proyecto-digital/>
- Master SQL. (21 de Abril de 2023). *Diccionario de datos SQL*. Obtenido de www.programarsql.com: <https://www.programarsql.com/diccionario-de-datos-sql/>
- Medina Cruz, J., Pineda Ballesteros , E., & Téllez Acuña, F. R. (2019). *Requerimientos de software: prototipado, software heredado y análisis de documentos*. (Vol. 37).
- Medina Cruz, J., Pineda Ballesteros, E., & Téllez Acuña, F. R. (01 de Abril de 2019). *Requerimientos de software: prototipado, software heredado y análisis de documentos*. (F. U. Norte, Ed.) doi:<https://doi.org/10.14482/inde.37.2.1053>
- Melanie. (3 de Octubre de 2022). *¿Qué es el diseño del software?* Obtenido de <https://streambe.com/es/que-es-el-diseno-de-software/>
- Mendoza, M. L. (16 de Julio de 2020). *Qué es un lenguaje de programación*. Obtenido de www.openwebinars.net: <https://openwebinars.net/blog/que-es-un-lenguaje-de-programacion/>
- Microsoft. (08 de Mayo de 2023). *¿Qué es Visual Studio?* Obtenido de learn.microsoft.com: <https://learn.microsoft.com/es-es/visualstudio/get-started/visual-studio-ide?view=vs-2022>

- Microsoft. (30 de Noviembre de 2023). *Introducción a ASP.NET Core*. Obtenido de learn.microsoft.com: <https://learn.microsoft.com/es-es/aspnet/core/introduction-to-aspnet-core?view=aspnetcore-8.0>
- Ministerio de Hacienda Costa Rica. (22 de Marzo de 2019). *Catálogo de Bienes y Servicios (Cabys)*. Obtenido de www.hacienda.go.cr: www.hacienda.go.cr/docs/Catalogobienesyservicios.pdf
- Ministerio de Hacienda Costa Rica. (24 de Agosto de 2023). *Comprobantes Electrónicos*. Obtenido de www.hacienda.go.cr: <https://www.hacienda.go.cr/docs/AspectosGeneralesComprobantesElectronicos.pdf>
- Monteserín, P. (13 de Mayo de 2020). *Definición de Modelo entidad-relacion*. Obtenido de pablomonteserin.com: <https://pablomonteserin.com/curso/bases-de-datos-con-sql/definicion-de-modelo-entidad-relacion/>
- Nérida Gladys Rey Córdova, & Ángel Velázquez Fernández. (2019). *Las variables de investigación. Alternativa financiera*. Obtenido de Universidad de San Martín de Porres, Perú: <https://www.aulavirtualusmp.pe/ojs/index.php/AF/article/view/1686/2055>
- NextU LATAM. (24 de Mayo de 2022). *¿Qué es CSS y para qué sirve?* Obtenido de www.nextu.com: <https://www.nextu.com/blog/que-es-css-rc22/>
- NextU LATAM. (7 de Octubre de 2022). *¿Qué es jQuery y cuál es su uso?* Obtenido de Blog | NextU LATAM: <https://www.nextu.com/blog/que-es-jquery-y-como-implementarlo-rc22/>
- Northware. (26 de Mayo de 2022). *Northware*. Obtenido de Northware: <https://www.northware.mx/blog/requerimientos-en-el-desarrollo-de-software-y-aplicaciones/#:~:text=requerimientos%20de%20software%20son%20simplemente,las%20necesidades%20de%20los%20usuarios>.
- Novet, J. (8 de Mayo de 2019). *CNBC*. Recuperado el 2023, de CNBC: <https://www.cnbc.com/2019/05/08/what-is-microsofts-power-platform.html>
- Palomo, Sebastián Rubén Gómez; Eduardo Antonio Moraleda Gil;. (2020, Marzo 2). *Aproximación a la Ingeniería del Software*. Editorial Centro de Estudios Ramon Areces.
- Parra, A. (13 de Enero de 2020). *¿Cuáles son los tipos de variables en una investigación?* Obtenido de QuestionPro: <https://www.questionpro.com/blog/es/tipos-de-variables-en-una-investigacion/>
- Pereyra, L. E. (2022). Metodología de la investigación. En L. E. Pereyra (Ed.). 1: Klik.
- Pittet, S. (s.f.). *Los distintos tipos de pruebas de software*. Obtenido de <https://www.atlassian.com/es/continuous-delivery/software-testing/types-of-software-testing>

- Platzi. (2017). *Platzi*. Obtenido de Platzi: <https://platzi.com/blog/que-es-scrum-y-los-roles-en-scrum/>
- pmoinformatica.com. (06 de 05 de 2015). *Requerimientos no funcionales*. Obtenido de pmoinformatica.com: <http://www.pmoinformatica.com/2015/05/requerimientos-no-funcionales-ejemplos.html>
- pmoinformatica.com. (30 de Mayo de 2018). *¿Qué es un requerimiento funcional?* Obtenido de pmoinformatica.com: <http://www.pmoinformatica.com/2018/05/que-es-requerimiento-funcional.html>
- Poder Ejecutivo. (2019, Junio 6). *Reglamento de la Ley del Impuesto sobre el Valor Agregado N° 41779*. Obtenido de http://www.pgrweb.go.cr/SCIJ/Busqueda/Normativa/Normas/nrm_texto_completo.aspx?nValor1=1&nValor2=88953
- proyectosagiles.org. (04 de Agosto de 2018). *¿Qué es SCRUM?* Obtenido de proyectosagiles.org: <https://proyectosagiles.org/que-es-scrum/>
- Ramos, J. A., & Segura, J. B. (11 de Enero de 2022). *DIAGRAMAS DE SECUENCIA*. Obtenido de [portalacademico.cch.unam.mx](https://portalacademico.cch.unam.mx/cibernetica1/analisis-y-diseno-en-poo/diagramas-de-secuencia): <https://portalacademico.cch.unam.mx/cibernetica1/analisis-y-diseno-en-poo/diagramas-de-secuencia>
- Richardson, A., & Rymer, J. (9 de Junio de 2014). *Forrester*. Obtenido de Forrester: <https://www.forrester.com/report/New-Development-Platforms-Emerge-For-CustomerFacing-Applications/RES113411>
- Roman, S. (26 de Agosto de 2023). *Prototipo: ¿Qué es y para que sirve?* Obtenido de saulromanjimenez.com: <https://saulromanjimenez.com/que-prototipo-sirve-ejemplos/>
- Ruiz Medina, M. I., Borboa Quintero, M., & Rodríguez Valdez, J. C. (13 de Agosto de 2019). *Grupo Eumed*. Obtenido de Grupo Eumed: <https://www.eumed.net/rev/tlatemoani/13/estudios-fiscales.pdf>
- SafetyCulture. (3 de Abril de 2023). *SafetyCulture*. Obtenido de SafetyCulture: <https://safetyculture.com/es/temas/recoleccion-de-datos/tecnicas-de-recoleccion-de-datos/>
- Sampieri, R. H. (2014). *Metodología de la Investigación* (sexta ed.). México, México: McGRAW-HILL / INTERAMERICANA EDITORES, S.A. doi:978-1-4562-2396-0
- Sampieri, R. H., Fernández Collado, C., & Baptista Luicio, P. (2014). Metodología de la investigación. En McGraw-Hill (Ed.).
- Sánchez, J. L. (29 de Noviembre de 2022). *Clean Code: ¿Qué es y por dónde empezar?* Obtenido de viewnext.com: <https://www.viewnext.com/clean-code-que-es-y-por-donde-empezar/>
- Software Engineering Institute. (2020). *Arquitectura de software*. Obtenido de [www.sei.cmu.edu](https://www.sei.cmu.edu/our-work/software-architecture/): <https://www.sei.cmu.edu/our-work/software-architecture/>

- Solís, L. D. (7 de Mayo de 2019). *Investigalia*. Obtenido de <https://investigaliacr.com/investigacion/el-enfoque-de-investigacion-la-naturaleza-del-estudio/#:~:text=Cuando%20hablamos%20de%20enfoque%20de,el%20desarrollo%20de%20la%20perspectiva>
- Souto, V. R. (24 de Agosto de 2023). ¡Qué es GitHub y por qué es útil? *www.hackaboss.com*. Obtenido de <https://www.hackaboss.com/blog/github-utilidad-copilot>
- Stein, L. G. (23 de mayo de 2020). *Universidad de Costa Rica* . Obtenido de Escuela de Ciencias Sociales: <https://www.ts.ucr.ac.cr/binarios/docente/pd-000169.pdf>
- techtarget. (20 de Septiembre de 2023). *What is Web Application (Web Apps) and its Benefits*. Obtenido de techtarget: <https://www.techtarget.com/searchsoftwarequality/definition/Web-application-Web-app>
- Trabajo, O. I. (05 de 09 de 1996). Protección de la maternidad. En *Código de Trabajo* (pág. 1). Costa Rica, Costa Rica: La Gaceta. Obtenido de Organización Internacional del Trabajo : https://www.ilo.org/dyn/natlex/natlex4.detail?p_isn=45267&p_lang=es
- Useche, M., Artigas, W., Queipo, B., & Perozo, É. (2019). *Técnicas e instrumentos de recolección de datos cuali-cuantitativos*. . Colombia, Colombia : Universidad de La Guajira. doi:9789566037040
- UTEC Universidad de Ingeniería y Tecnología. (28 de Mayo de 2018). *UTEC Universidad de Ingeniería y Tecnología*. Obtenido de UTEC Universidad de Ingeniería y Tecnología.
- Visure. (10 de Julio de 2022). *¿Qué son los requisitos funcionales?* Obtenido de <https://visuresolutions.com/es/blog/requerimientos-funcionales/>
- write. (s.f.). *¿Qué son los requisitos técnicos en la gestión de proyectos?* Obtenido de [wrike.com: https://www.wrike.com/es/project-management-guide/faq/que-son-los-requisitos-tecnicos-en-la-gestion-de-proyectos/](https://www.wrike.com/es/project-management-guide/faq/que-son-los-requisitos-tecnicos-en-la-gestion-de-proyectos/)

ÁPENDICES

Apéndice A: Guía de entrevista semiestructurada aplicada

Entrevista Semiestructurada Aplicada

Nombre del entrevistador	Daniel Quesada Fallas
Nombre de las personas entrevistadas	Kathleen Wilson Mora (Gerente General)
	Ana Cristina Chaves (Administradora)
Fecha	Sábado 04 de mayo del 2024

Pregunta N° 1 ¿La empresa Zapatitos Lindos dispone actualmente de una infraestructura informática para respaldar la administración de sus operaciones de venta?

Pregunta N° 2 ¿Cuáles son los obstáculos más destacados o desafíos que la empresa está experimentando actualmente al utilizar el software actual?

Pregunta N° 3 ¿En su opinión, qué procesos identifica como los que tienen la mayor cantidad de inconsistencias y, por lo tanto, necesitan un control más riguroso?

Pregunta N° 4 ¿Qué módulos o funcionalidades forman parte del sistema en la actualidad?

Pregunta N° 5 ¿Puede describir cómo se desarrolla el proceso de ventas en Zapatitos Lindos S.A.?

Pregunta N° 6 ¿Puede explicar el proceso de gestión de órdenes de trabajo relacionado con los apartados en la empresa?

Pregunta N° 7 ¿Podría proporcionar una visión general de los elementos que se gestionan en el catálogo de productos de la empresa?

Pregunta N° 8 En relación con el elemento "Diferencia de precio" mencionado previamente, ¿podría explicar su función o propósito en el contexto de la empresa?

Pregunta N° 9 ¿Hay algún otro elemento o detalle que le gustaría incluir en el prototipo propuesto?

para la gestión de ventas?

Pregunta N° 10 Con relación a la gestión de pacientes, ¿cuáles son los datos esenciales que deben registrarse obligatoriamente en el sistema?

Pregunta N° 11 ¿Hay algún dato importante de las recetas de CCSS que no se esté registrando actualmente en las órdenes de trabajo o que le gustaría capturar en un campo específico?

Pregunta N° 11 En cuanto a la contabilidad, ¿cuáles son los datos o insumos que extraen del sistema actual y envían al departamento de contabilidad para su procesamiento?

Pregunta N° 13 ¿Con qué periodicidad se llevan a cabo las importaciones de zapatos desde México?

Apéndice B: Guía de Observación

Entrevista Semiestructurada Aplicada

Nombre del observador	Daniel Quesada Fallas
Fecha	Sábado 11 de mayo del 2024

Tabla 121.

Guía de observación

Apartado Hardware	
Aspecto por evaluar	Observación
Cantidad de Computadoras	Cuentan con 4 computadoras: 2 de escritorio y 2 laptops.
Estado y Antigüedad de las Computadoras	Las computadoras están en buen estado y cumplen con los requisitos para ejecutar el prototipo propuesto.
Impresoras y Escáneres	La empresa tiene dos impresoras y un escáner de códigos de barras y QR. Estos equipos pueden seguir siendo utilizados sin problemas y el escáner facilita la lectura de los códigos QR de las facturas electrónicas.
Conexión a Internet y Red Interna	Proveedor: Tigo. Velocidad: 10 Mbps. Estabilidad: Presenta intermitencias. Tipo de red: Cableada e inalámbrica. La velocidad del internet y la conexión de las computadoras es aceptable.

Apartado Software	
Aspecto por evaluar	Observación
Funcionalidades del Sistema Actual	1. Ingreso de compras. 2. Mantenimiento de proveedores. 3. Mantenimiento de artículos. 4. Mantenimiento de clientes. 5. Mantenimiento de pacientes. 6. Facturación contado. 7. Facturación de crédito. 8. Anulación de facturas. 9. Apartados (órdenes de trabajo). Esto permitió levantar una lista de funcionalidades necesarias para el prototipo funcional propuesto.
Cuenta con Factura Electrónica	La empresa emite facturación electrónica, pero no es una función habilitada en el sistema actual. Utilizan un tercero (GTI) para la emisión de facturas electrónicas.
Capacidad de integración del sistema actual con otros sistemas.	La opción de integración es casi nula debido al costo y a que, según la administración, cada vez que se toca algo se daña otra cosa.
Inconsistencias o problemas del sistema	1. Desconexiones frecuentes. 2. Errores en cálculos. 3. Inconsistencias en el inventario. 4. Lentitud. 5. Otros errores. Se registraron muchos problemas, principalmente debido a la intermitencia del internet y a la falta de manejo adecuado del sistema. Ejemplo: una factura simple tardó 15 minutos en generarse debido a fallos y lentitud del sistema. Los usuarios, con conocimiento básico, suelen reiniciar la computadora cuando el sistema se queda pegado.
Seguridad y copias de seguridad.	La empresa no cuenta con un plan de recuperación ante desastres ni con políticas de respaldos. Esto es desconocido para la administración. Se observó un archivo de texto en una de las computadoras que contenía la URL, usuario y contraseña del hosting sin encriptar, lo cual es una práctica muy deficiente.
Apartado Empleados	
Aspecto por evaluar	Observación
Cantidad de empleados	La empresa cuenta con 6 empleados.
Cantidad de empleados presente durante el proceso de observación	Durante la observación, estaban presentes dos empleados, la administradora y la gerente general. También estaba el doctor que toma las recetas de los pacientes, quien trabaja por servicios profesionales no más de dos veces al mes.
Nivel de conocimiento del sistema informático	El nivel de conocimiento del sistema informático es básico. Las dos vendedoras solo realizan funciones relacionadas con la atención al cliente, la gestión del proceso de ventas y la generación de facturas u órdenes de trabajo.

Apartado Procesos	
Aspecto por evaluar	Observación
Control de inventarios	La empresa no cuenta con un control de inventarios fiable. Utilizan un cuaderno de papel para llevar el control de lo que sale y entra. Esto se realiza de manera mensual. Anteriormente, usaban un inventario perpetuo con seguimiento en tiempo real y método de valuación de costo promedio ponderado. La propuesta del prototipo incluye este método de valuación.
Gestión de cuentas por cobrar	Se llevan de manera manual. Encontrar una factura es tedioso, ya que la numeración de los documentos en el sistema no coincide con la de la factura electrónica, que es realizada por un tercero.
Gestión de cuentas por pagar	Se realiza de forma manual.
Gestión Contable (Ingresos y Gastos)	No cuentan con un sistema contable propio. Utilizan un tercero para este servicio, quien les proporciona copias físicas de las transacciones.
Gestión de ventas	El servicio al cliente es bueno, pero el proceso de facturación es lento, lo que genera impaciencia en los clientes, quienes deben esperar hasta 15 minutos para recibir su factura. Este proceso se había detallado en la entrevista aplicada, sin embargo, la observación permitió determinar que la emisión de la factura es un proceso que demanda mucho tiempo y que se debe mejorar.
Políticas de Devoluciones y Garantías	El proceso de anulación o devoluciones presenta muchas inconsistencias y solo lo realiza la administradora.

Fuente: Elaboración propia (2024).