

UNIVERSIDAD INTERNACIONAL DE LAS AMÉRICAS

ESCUELA DE INGENIERÍA INFORMÁTICA

TRABAJO FINAL DE GRADUACIÓN PARA OPTAR POR EL GRADO DE

BACHILLERATO EN INGENIERÍA EN INFORMÁTICA

**Prototipo Funcional para la Gestión de Recursos Humanos para la Farmacia D'Angelo
Ubicada en Cartago**

Deyvin Francisco Polanco Espinoza

MBD. Olman Núñez Peralta

TUTOR

Sede Central

JULIO, 2024

DEDICATORIA

Con gran orgullo por culminar este proceso, quiero dedicar este logro primeramente a Dios, ya que, con su amor infinito, con su guía y fortaleza, ha sido fundamental en cada paso de este camino, me permitió seguir adelante siempre haciendo las cosas de la mejor manera posible. A mis padres, ya que gracias a su amor incondicional, su apoyo y constante sacrificio me han ayudado a llegar a la meta de este proceso; gracias por ser mi inspiración y por enseñarme el verdadero significado de la perseverancia y la dedicación.

AGRADECIMIENTOS

En primer lugar, agradezco a Dios por su infinita sabiduría y fortaleza, que me han guiado y sostenido a lo largo de este crucial proceso en mi vida, marcando el comienzo de todo lo que está por llegar. Sin su paciencia y presencia en mi vida, este logro no habría sido posible.

A mis padres, por su amor incondicional, su apoyo constante y sacrificios invaluable. Su confianza en mí y sus palabras de aliento me han motivado a superar cada obstáculo; gracias a ellos puedo dar un paso más en mi vida, siendo siempre un ejemplo y fuente de inspiración en mi vida. Agradezco a mi hermano por formar parte de este proceso, querer ser un ejemplo para él me motiva a hacer las cosas de la mejor manera posible.

A mis mentores en este proceso, máster Olda Bustillos Ortega, directora de la Escuela de Ingeniería en Informática, por siempre estar presente, así como a mi tutor MBD. Olman Núñez Peralta por brindarme la guía en esta última etapa, además de ser un excelente mentor en el transcurso de la carrera y brindarme guía y consejos.

A mis mentores en secundaria, Bismarck Méndez Suárez, Olinda de Fátima Cerda, Cándida Vidaurre, entre otros, gracias a ellos por la guía y la confianza en mí para seguir adelante. Siempre estaré agradecido y no solo en este proceso y logro, sino en todos los que están por llegar de ahora en adelante. Ellos han sido fundamentales en todos mis logros y en la formación de la persona que soy hoy.

A mis amigos, los cuales formaron parte de este proceso, gracias por el apoyo y por alentarme a seguir en el proceso, no fue fácil, pero sí muy enriquecedor en mi formación profesional y como persona. Estoy sumamente agradecido con todas las personas que formaron parte de este proceso y lo estaré siempre.

CONTENIDO

CAPÍTULO I: INTRODUCCIÓN	17
Planteamiento del Problema.....	17
Objetivos	18
Objetivo General.....	18
Objetivos Específicos	18
Justificación.....	19
Viabilidad Técnica.....	19
Viabilidad Operativa.....	21
Viabilidad Económica	22
Viabilidad Legal	23
Proyecciones.....	23
Alcance Funcional	23
Módulo de calcular planilla.	24
Módulo de calcular aguinaldo.....	24
Módulo de cálculo de vacaciones.	24
Módulo de calcular horas extras.	24
Módulo de control de permisos del colaborador.	24
Módulo de calcular liquidación.	25
Módulo de calcular incapacidades	25
Módulo mantenimientos.	25
Módulo consultas.....	25
Módulo reportes.....	25
Módulo seguridad.....	26
Módulo de control de asistencia.....	26

Alcance Metodológico.....	26
Fases del modelo cascada por implementar.....	27
Alcance Tecnológico	28
CAPÍTULO II: MARCO REFERENCIAL.....	29
CAPÍTULO III: MARCO METODOLÓGICO	43
Enfoques de la Investigación.....	43
Enfoque Cualitativo	43
Enfoque de Investigación Seleccionado	44
Tipos de Investigación	44
Investigación Descriptiva	45
Tipo de Investigación Seleccionado	45
Fuentes de información	45
Fuentes Primaria	46
Fuentes Secundaria	47
Fuentes Terciaria	47
Variables.....	48
Variable Conceptual	48
Variable Operacional	48
Variables Instrumentales	49
Cuadro de Variables	49
Instrumentos de recolección de datos.....	52
Entrevista	52
La observación.....	52
CAPÍTULO IV: ANÁLISIS DE RESULTADOS	53
Proceso de recolección y análisis de datos.....	53

Preguntas, Objetivos y Análisis por Módulo.....	54
Módulo de cálculo de planilla.....	54
Módulo para calcular aguinaldo.	56
Módulo para el cálculo de vacaciones.	58
Módulo para el cálculo de horas extras.	60
Módulo de control de permisos del colaborador.	63
Módulo de liquidación.	65
Módulo de registro de incapacidades.....	67
Observación.....	69
CAPÍTULO V: CONCLUSIONES Y RECOMENDACIONES	71
Conclusiones	71
Recomendaciones.....	72
CAPÍTULO VI: PROPUESTA.....	73
Requerimientos Funcionales	73
Indicador.....	83
Módulo	83
Requerimientos.....	83
Análisis.....	84
Análisis del Software.....	84
Módulo de calcular planilla.	84
Módulo de calcular aguinaldo.....	84
Módulo de cálculo de vacaciones.	84
Módulo de calcular horas extras.	85
Módulo de control de permisos del colaborador.	85
Módulo de calcular liquidación.	85

Módulo de mantenimiento.....	86
Módulo de consultas.....	86
Módulo de reportes.....	86
Módulo de seguridad.....	86
Módulo de control de asistencia.....	87
Análisis del Hardware.....	87
Análisis Detallado de los Elementos de Telecomunicaciones.....	88
Descripción Detallada de las Herramientas Técnicas Utilizadas.....	88
Descripción Detallada del Conocimiento Básico que Debe Tener el Recurso Humano que Operará el Sistema.....	90
Casos de uso.....	92
Diseño.....	109
Arquitectura del Sistema.....	110
Arquitectura de Software.....	111
Diseño de Entradas.....	112
Diseño Físico de la Base de Datos.....	115
Diccionario de Datos.....	116
Diseño de Procesos.....	125
Diseño de Salidas.....	133
Diagrama de Clases.....	135
Programación.....	136
Entradas.....	136
Salidas.....	143
Procesos.....	150
Pruebas Funcionales.....	154

REFERENCIAS	174
APÉNDICES	178
Apéndice 1	178
Apéndice 2	181

TABLAS

Tabla 1: Costo de desarrollo de la aplicación.	22
Tabla 2: Costos de los elementos para el desarrollo del prototipo funcional.....	22
Tabla 3: Cuadro de variables.....	49
Tabla 4: Requerimiento 1.....	73
Tabla 5: Requerimiento 2.....	74
Tabla 6: Requerimiento 3.....	75
Tabla 7: Requerimiento 4.....	76
Tabla 8: Requerimiento 5.....	77
Tabla 9: Requerimiento 6.....	78
Tabla 10: Requerimiento 7.....	79
Tabla 11: Requerimiento 8.....	80
Tabla 12: Requerimiento 9.....	80
Tabla 13: Requerimiento 10.....	81
Tabla 14: Requerimiento 11.....	81
Tabla 15: Requerimiento 12.....	82
Tabla 16: Matriz de requerimientos.	83
Tabla 17: Hardware requerido	87
Tabla 18: Análisis de telecomunicaciones	88
Tabla 19: Software requerido para la programación.....	89
Tabla 20: Caso de uso de planilla	92
Tabla 21: Caso de uso de cálculo de planilla	93
Tabla 22: Caso de uso de cálculo de aguinaldo	94
Tabla 23: Caso de uso de cálculo de aguinaldo	95
Tabla 24 : Caso de uso de vacaciones.....	96

Tabla 25: Caso de uso de vacaciones	97
Tabla 26: Caso de uso de horas extras	98
Tabla 27: Caso de uso de horas extras	99
Tabla 28: Caso de uso de control de permisos del colaborador	100
Tabla 29: Caso de uso de control de permisos del colaborador	101
Tabla 30: Caso de uso de cálculo de liquidación	102
Tabla 31: Caso de uso de cálculo de liquidación	103
Tabla 32: Caso de uso cálculo de liquidación.....	104
Tabla 33: Caso de uso cálculo de incapacidad.....	105
Tabla 34: Caso de uso de mantenimiento	106
Tabla 35: Caso de uso de consultas.....	107
Tabla 36: Caso de uso de reportes.....	108
Tabla 37: Caso de uso de seguridad.....	109
Tabla 38: Diccionario de datos aguinaldo.....	116
Tabla 39: Diccionario de datos cantón.....	116
Tabla 40: Diccionario de datos catálogo aprobaciones.....	116
Tabla 41: Diccionario de datos catálogo día	117
Tabla 42: Diccionario de datos catálogo género	117
Tabla 43: Diccionario de datos catálogo tipo horas extras.	117
Tabla 44: Diccionario de datos Catálogo tipo incapacidad.....	117
Tabla 45: Diccionario de datos Catálogo tipo liquidación.....	117
Tabla 46: Diccionario de datos catálogo tipo permiso.....	118
Tabla 47: Diccionario de datos catálogo tipo usuario.....	118
Tabla 48: Diccionario de datos tipo cédula.....	118
Tabla 49: Diccionario de datos control de marcas de asistencia.....	118

Tabla 50: Diccionario de datos correo electrónico	119
Tabla 51: Diccionario de bases de datos deducciones.	119
Tabla 52: Diccionario de datos dirección.....	119
Tabla 53: Diccionario de datos distrito	119
Tabla 54: Diccionario de datos empleado	120
Tabla 55: Diccionario de datos horario	120
Tabla 56: Diccionario de datos horas extras	120
Tabla 57: Diccionario de datos incapacidades	121
Tabla 58: Diccionario de datos liquidación	121
Tabla 59: Diccionario de datos movimiento	121
Tabla 60: Diccionario de datos permisos del colaborador	122
Tabla 61: Diccionario de datos persona	122
Tabla 62: diccionario de datos de planilla.....	122
Tabla 63: Diccionario de datos de provincia.....	123
Tabla 64: Diccionario de datos solicitud de extras	123
Tabla 65: Diccionario de datos teléfono	123
Tabla 66: Diccionario de datos tipo horario.....	123
Tabla 67: Diccionario de datos usuario.....	124
Tabla 68: Diccionario de datos vacaciones	124
Tabla 69: Prueba funcional inicio de sesión.	154
Tabla 70: Prueba funcional	155
Tabla 71: Prueba funcional	156
Tabla 72: Prueba funcional.	157
Tabla 73: Prueba funcional.	158
Tabla 74: Pruebas funcional.....	159

Tabla 75: Pruebas funcionales	160
Tabla 76: Pruebas funcionales	160
Tabla 77: Pruebas funcionales	161
Tabla 78: Pruebas funcionales	162
Tabla 79: Pruebas funcionales	163
Tabla 80: Pruebas funcionales	165
Tabla 81: Pruebas funcionales	166
Tabla 82: Pruebas funcionales	167
Tabla 83: Pruebas funcionales	168
Tabla 84: Pruebas funcionales	169
Tabla 85: Pruebas funcionales	170
Tabla 86: Pruebas funcionales	171
Tabla 87: Pruebas funcionales	172

FIGURAS

Figura 1: Proceso de desarrollo.....	27
Figura 2: MVC (modelo, vista y controlador).	33
Figura 3: Resultados del módulo de planilla.....	54
Figura 4: Resultados del módulo de planilla.....	55
Figura 5: Resultados del módulo de aguinaldo.....	56
Figura 6: Resultados del módulo de aguinaldo.....	57
Figura 7: Resultados del módulo de Vacaciones.	58
Figura 8: Resultados del módulo de vacaciones.	59
Figura 9: Resultados del módulo de horas extras.	60
Figura 10: Resultados del módulo de horas extras.	61
Figura 11: Resultados del módulo de horas extras.	62
Figura 12: Resultados del módulo de control de permisos.	63
Figura 13: Resultados del módulo de control de permisos.	64
Figura 14: Resultados del módulo de liquidación.....	65
Figura 15: Resultados del módulo de liquidación.....	66
Figura 16: Resultados del módulo de incapacidades.	67
Figura 17: Resultados del módulo de incapacidades.	68
Figura 18: Resultados del módulo de liquidación.....	69
Figura 19: Diagrama de casos de uso	91
Figura 20: Arquitectura de sistema	110
Figura 21: Arquitectura del software	111
Figura 22: Diseño de entradas solicitud de extras	112
Figura 23: Diseño de entradas solicitud de permisos.....	113
Figura 24: Diseño de entradas solicitud de vacaciones	113

Figura 25: Diseño de entradas crear incapacidad	114
Figura 26: Diseño de entradas crear horario	114
Figura 27: Diseño físico de base de datos.....	115
Figura 28: Diagramas de flujos inicio de sesión.....	125
Figura 29: Diagrama de flujo de vacaciones.....	126
Figura 30: Diagrama de flujos de horas extras.	127
Figura 31: Diagrama de flujos de incapacidad	128
Figura 32: Diagrama de flujo de permisos del colaborador.....	129
Figura 33: Diagrama de flujo de liquidación.	130
Figura 34: Diagrama de flujo de aguinaldo.	131
Figura 35: Diagrama de flujo de planilla	132
Figura 36: Diseño de salidas para horarios.	133
Figura 37: Diseño de salidas de planilla	133
Figura 38: Diseño de salidas de incapacidades.....	134
Figura 39: Diseño de salida para vacaciones.	134
Figura 40: Diseño de salida de solicitudes.....	134
Figura 41: Diagrama de clases.....	135
Figura 42: Entrada del módulo de planilla.....	136
Figura 43: Entrada del módulo de aguinaldo	137
Figura 44: Entrada del módulo de vacaciones	138
Figura 45: Entrada del módulo de horas extras	139
Figura 46: Entrada del módulo de permisos del colaborador.	140
Figura 47: Entrada del módulo de liquidación.....	141
Figura 48: Entrada del módulo de incapacidades	142
Figura 49: Salida del módulo de planilla	143

Figura 50: Salida del módulo de aguinaldo	144
Figura 51: Salida del módulo de vacaciones.....	145
Figura 52: Salida del módulo de horas extras	146
Figura 53: Salida del módulo de permisos del colaborador.....	147
Figura 54: Salida del módulo de liquidación	148
Figura 55: Salida del módulo de incapacidades.....	149
Figura 56: Proceso del cálculo de planilla.	150
Figura 57: Proceso del cálculo de aguinaldo.	151
Figura 58: Proceso del cálculo de vacaciones.....	151
Figura 59: Proceso para solicitar permisos.	152
Figura 60: Proceso para calcular la liquidación.....	152
Figura 61: Proceso para calcular la incapacidad.....	153

RESUMEN EJECUTIVO

En el transcurso de esta investigación, se abordan diversos aspectos relacionados con el desarrollo de un sistema de recursos humanos, con el objetivo de comprender a fondo la estructura y el contenido de este trabajo. El primer capítulo de la investigación está dedicado a la introducción. En esta sección, se presenta el problema por investigar, se definen los objetivos que se pretende alcanzar y se justifica la importancia del estudio, destacando las metas que se buscan con este trabajo.

En el segundo capítulo, llamado Marco Referencial, se profundizan los temas claves que abarcan recursos humanos, tales como planilla, vacaciones, horas extras, permisos del colaborador, liquidación y aguinaldo. Adicional a esto, se describen las herramientas con las cuales se desarrolla el sistema con sus definiciones, tales como *Entity framework*, bases de datos, SQL, C#, HTML, CSS y Java Scripts; estos son esenciales para el desarrollo del prototipo funcional.

En el tercer capítulo llamado Marco Metodológico, se detallan los enfoques de la investigación, los tipos de investigación y la que se seleccionó para el desarrollo de este trabajo. Así mismo, se detallan las fuentes, variables y los instrumentos de recolección de datos, los cuales son fundamentales para obtener la información requerida en el desarrollo del prototipo.

En el capítulo cuatro, se presenta el análisis de resultados, donde se organiza y muestra de manera estructurada la información obtenida, así como los hallazgos y conclusiones derivados del análisis de los datos recopilados.

En el quinto capítulo se muestran conclusiones y recomendaciones del sistema, se detallan las conclusiones obtenidas con base en los objetivos, se ofrecen recomendaciones para la correcta aplicación del sistema y posibles mejoras. Finalmente, en el sexto capítulo se muestra la propuesta, en este se detalla el desarrollo, se muestra el análisis del *software*, casos de uso, diseño, programación y pruebas, los cuales concluyen con un proyecto exitoso.

CAPÍTULO I: INTRODUCCIÓN

La farmacia D'Angelo se dedica a la venta de medicamentos, productos farmacéuticos y artículos relacionados con la salud. Su principal función es proporcionar acceso a medicamentos recetados que tienen venta libre; de igual forma, ofrecer asesoramientos farmacéuticos a los clientes y orientar sobre el uso adecuado de los medicamentos, las interacciones medicamentosas, los efectos secundarios y otras consideraciones relacionadas con la salud.

Planteamiento del Problema

La farmacia enfrenta actualmente deficiencias en su sistema de gestión de recursos humanos, debido a que todos los procesos y cálculos se manejan de forma manual o mediante Microsoft Excel. Esta práctica llega a generar deficiencias en la efectividad de los cálculos internos como lo son:

- ***Falta de un sistema ágil para el cálculo de planilla de la empresa.*** La empresa en la actualidad no cuenta con un sistema ágil para realizar el cálculo de planilla, debido a que la mayoría de los registros se llevan en Excel y algunos a mano. Esto causa que los encargados de planilla del Departamento de Recursos Humanos (RR.HH.) tengan problemas en el pago por omitir datos importantes a la hora del cálculo causando en ocasiones errores de inconsistencias y registro, lo que lleva a un descontento por parte de los colaboradores.
- ***Ausencia de una herramienta que permita realizar un cálculo eficiente del aguinaldo.*** La empresa carece de un sistema formal para el pago del aguinaldo, ya que gestiona todo a través de Excel y algunos a mano, lo que genera dificultades en el proceso de pago de aguinaldos y provoca la insatisfacción de los colaboradores cada año.
- ***Falta de un registro de los días de vacaciones de los colaboradores.*** A causa de la falta de un adecuado control de los días de vacaciones de los colaboradores, en algunas situaciones se realizan pagos incorrectos relacionados con este concepto.
- ***No se cuenta con un sistema de registro que documente las horas extras realizadas por cada colaborador.*** No se tienen registros de las horas extras que realiza cada

colaborador, provocando cálculos incorrectos que desencadenan en inconformidades e incluso renunciaciones.

- ***Mala gestión de los permisos otorgados.*** La carencia de un sistema de gestión de permisos laborales en la empresa conlleva a que no se identifique el tipo de permiso ni se considere la reducción correspondiente en la remuneración de los colaboradores en la planilla.
- ***Carencia de sistema que permita el cálculo de liquidación de cada colaborador.*** La ausencia de un módulo de cálculo de liquidación en la empresa representa un desafío significativo. Sin esta herramienta, el proceso de liquidar las cuentas finales de los empleados al finalizar su relación laboral se vuelve manual y propenso a errores.
- ***Mal control para registrar los trámites de las incapacidades de los colaboradores.*** Debido a que la empresa lleva un mal control para registrar las incapacidades, no se cuenta de forma clara con los montos que se deben pagar por este concepto a los afectados.

Objetivos

Objetivo General

Desarrollar un prototipo funcional para la gestión de recursos humanos para la farmacia D'Angelo.

Objetivos Específicos

1. Analizar los requerimientos que serán implementados en el prototipo funcional satisfaciendo las necesidades de la entidad.
2. Diseñar la estructura del prototipo basado en los requerimientos, así como sus módulos.
3. Desarrollar la programación del prototipo con base en el diseño que se estableció.
4. Realizar pruebas para el funcionamiento óptimo del prototipo final.

Justificación

La empresa nunca ha contado con un sistema tecnológico que les permita facilitar el control del Departamento de Recursos Humanos, ya que en la actualidad los sistemas informáticos se han convertido en un pilar fundamental para un mejor desempeño de las empresas. De esta manera, la ejecución del prototipo funcional se convierte en un avance esencial, ya que esta percibe la tecnología como una fuente fundamental de innovación que transformará sus operaciones en el ámbito de recursos humanos.

La funcionalidad de este prototipo es brindar facilidades y apoyo tecnológico a la empresa con el fin de resolver los principales problemas relacionados, acelerando y aprovechando más el tiempo. Además, facilitará la creación de informe y el análisis de datos en el ámbito de recursos humanos, al brindar más facilidades a la empresa. El prototipo tiene la intención de superar las limitaciones de los métodos manuales y, de esta forma, eliminar los errores de cálculo; además, contará con una fuente confiable donde almacenar los datos, de tal modo que no podrán ser alterados por personas que no tengan autorización.

Este prototipo fomenta la correcta comunicación entre el área de recursos humanos y los colaboradores de la empresa, debido a que el colaborador realizará las peticiones por medio del programa y este notificará a Recursos Humanos acerca de alguna petición del colaborador. Además, se le responderá al mismo por medio del sistema evitando malentendidos y manteniendo una buena comunicación entre ambas partes.

Viabilidad Técnica

El prototipo requiere un equipo con una computadora que cuente con un sistema operativo compatible y recursos adecuados para el desarrollo, junto con un entorno de desarrollo integrado específico y herramientas de compilación. Además, se necesitan periféricos estándar como ratón, monitor y teclado ergonómicos. La conexión a Internet debe ser de alta velocidad y segura, con medidas como conexiones cifradas.

Es esencial mantener el sistema operativo y el *software* actualizados con parches y actualizaciones. Se aconseja establecer un sistema de respaldo regular para garantizar la seguridad de los datos. Además, la colaboración se facilita mediante el uso de herramientas en línea, como repositorios de código y plataformas de gestión de proyectos.

Un entorno de prueba riguroso, respaldado por herramientas de depuración, es crucial para evaluar y perfeccionar el prototipo antes de su implementación. Además, una documentación clara y detallada, que incluya instrucciones de instalación, configuración y uso, mejora la comprensión y la colaboración en el desarrollo del prototipo.

Para el desarrollo del prototipo funcional, se utiliza una aplicación web basada en la tecnología de Microsoft. Se hace uso de Visual Studio 2022 como el entorno de desarrollo principal y se usa el lenguaje de programación C# con el .NET Framework, específicamente en su versión 4.8.1, que incluye ASP.NET. Esta fusión de herramientas ofrece una plataforma completa y sólida para el desarrollo de aplicaciones web.

Para complementar el uso de este lenguaje, se utiliza Microsoft SQL Server 2022 con una licencia estándar; lo que representa una estrategia sólida para asegurar funcionalidades esenciales en el almacenamiento y gestión de datos, debido a que esta plataforma ofrece una base confiable que respalda los requisitos fundamentales de la empresa en el manejo de la información. Adicionalmente, cuenta con capacidades avanzadas para la generación de informes y análisis, consolidando una solución integral y robusta para la construcción del prototipo funcional.

Por su parte, para el diseño de las vistas del prototipo funcional, se implementa HTML con plantillas de Bootstrap, a fin de generar una interfaz amigable con el usuario, complementando con CSS para darle los estilos a todo el prototipo. También se hace uso de JavaScript con *DataTable* para generar vistas más amigables cuando se deba dar mantenimientos. Esta fusión de herramientas generará una interfaz de usuario atractiva y fácil de usar en la aplicación web.

La elección de HTML brinda una estructura sólida para las páginas web y al complementarlo con Bootstrap, se asegura de tener un buen diseño mejorando la experiencia de usuario.

En cuanto al uso de los estilos, la implementación de CSS desempeñará un papel crucial para la personalización de cada una de las vistas del prototipo. Cabe mencionar que todas llevarán el mismo estilo a fin de hacerlo atractivo para el usuario. Además, al aplicar estilos cohesivos, se logrará una apariencia profesional y alineada con la identidad visual de la empresa.

La inclusión de JavaScript, en particular con el uso de la librería *DataTable*, potenciará las capacidades interactivas del prototipo. *DataTable* ofrece funcionalidades avanzadas para la presentación de datos tabulares, como ordenamiento, filtrado y paginación, lo que facilita la

navegación y manipulación de grandes conjuntos de datos. Además, permite al usuario buscar de manera rápida algún registro, debido a que cuenta con funciones para buscar.

La fusión de estas herramientas no solo se centra en la estética, sino también en la usabilidad, ya que resultará en una interfaz intuitiva y fácil de navegar mejorando la experiencia del usuario y haciéndola amigable con este; lo cual permite que impacte de manera positiva en la empresa.

Viabilidad Operativa

El prototipo se desarrolla de forma que sea de fácil uso para los colaboradores que lo utilicen. Deberán contar con el conocimiento básico en equipo de cómputo; una vez esté implementado, permitirá a la persona encargada interactuar con el sistema mediante una interfaz amigable para gestionar de manera más eficaz y eficiente la planilla de la empresa. Adicionalmente, permitirá a los colaboradores realizar la petición de las vacaciones, permisos y horas extras de una mejor manera.

El desarrollo del prototipo permitirá que el proceso del cálculo de planilla se haga de manera eficiente reduciendo los errores que eran causados de forma manual. Así mismo, el prototipo ayudará a realizar el cálculo de horas extras, aguinaldo, liquidación, vacaciones; cálculo de incapacidades de manera automática y maneja de una mejor forma el control de permisos, ya que se notificará vía sistema. También asegura el cumplimiento de las normativas laborales reduciendo posibles riesgos legales, al proporcionar transparencia en los cálculos y permitir la generación automática de informes; finalmente, fortalece la confianza entre empleados y empleadores.

El desarrollo de este sistema constituye una herramienta diseñada para agilizar y optimizar de manera significativa la gestión de procesos; al mismo tiempo que simplifica las responsabilidades de la administración en la empresa. Este enfoque se logra a través de la implementación de un perfil administrador, que tiene la capacidad de acceder y supervisar todos los módulos del sistema, así como un perfil para el responsable de recursos humanos, para la correcta comunicación entre ellos. Adicionalmente, tendrán un usuario para los empleados donde podrán hacer las solicitudes permitiendo automatizar la empresa.

Viabilidad Económica

El desarrollo de este prototipo no tiene ningún costo, ya que, para crearlo se utilizan licencias gratuitas. Adicionalmente, se informó en la empresa que el prototipo se crea con fines universitarios para el proyecto de graduación.

Los costos de prototipo, basado en el Ministerio de Trabajo y Seguridad Social en el año 2024, consideran el salario mínimo de un programador no titulado, el cual es de ₡15,613.91 colones por día y este desempeñará muchas tareas en el desarrollo del proyecto. En la tabla 1, se muestra el costo aproximado en el desarrollo de la aplicación web.

Tabla 1:

Costo de desarrollo de la aplicación.

Etapa del proyecto	Costo por día	Cantidad de días aproximado	Costo
Análisis de requerimientos	₡15,613.91	5	₡78,069.55
Diseño	₡15,613.91	20	₡312,278.20
Desarrollo	₡15,613.91	40	₡624,556.40
Pruebas	₡15,613.91	5	₡78,069.55

Fuente: Elaboración propia, 2024.

En la tabla 2, se especifican los requerimientos de *hardware* y *software* para el desarrollo del prototipo.

Tabla 2:

Costos de los elementos para el desarrollo del prototipo funcional.

Requerimientos	Especificaciones Recomendadas	Costos
Computadora	-Procesador Intel i5 o superior. -Disco duro 500GB. -Memoria RAM 4GB o superior. -Sistema Operativo 64-bits.	₡0
Motor de base de datos.	SQL Server	₡0
Entorno de Desarrollo.	Microsoft Visual Studio	₡0
Herramienta de diseño de base de datos	MySQL Workbench	₡0

Fuente: Elaboración propia, 2024

Viabilidad Legal

Se habilitarán los módulos de manera confidencial y solamente el personal administrativo podrá hacer uso de la información del colaborador; con el fin de protegerla, no se podrá hacer uso de datos sensibles como nombre de empleados, números de teléfonos, números de cédula, montos salariales, información bancaria o similares, todo esto cumpliendo las siguientes leyes:

- Ley N.º 8148 sobre Delitos Informáticos (Costa Rica) y los siguientes artículos:
 - ✓ Artículo 196 bis, Sobre La Violación de Comunicaciones Electrónicas.
 - ✓ Artículo 217 bis, Sobre del Fraude Informático.
 - ✓ Artículo 229 bis, Sobre de Las Alteraciones de Datos y Sabotaje Informático.
- Ley N.º 4573 para reprimir y sancionar los delitos informáticos de la Asamblea Legislativa de la República de Costa Rica del año 2001.
- Ley de Derechos de Autor 6683 6683 por parte de la Asamblea Legislativa de la República de Costa Rica del año 1982.
- Ley N.º 8968 sobre la protección de la persona frente al tratamiento de sus datos personales.

Proyecciones

El objetivo del prototipo es mejorar las funciones de la empresa, ya que, al solventar los errores de cálculo humano, este le permite ser más eficiente. Además, el prototipo es fácil de usar y se rige bajo las leyes costarricenses, así como ayudará a la toma de decisiones, ya que tendrá una mayor productividad. Adicionalmente, esto reducirá la cantidad de horas laboradas en relación al cálculo de planilla, porque estará automatizado, permitirá un seguimiento minucioso de los colaboradores y tener una mayor satisfacción, ya que los cálculos se harán de manera correcta. Seguidamente, se encuentran los alcances respectivos para el prototipo funcional.

Alcance Funcional

El prototipo cuenta con diferentes módulos que ayudan al cálculo de planilla, cálculo de aguinaldo, cálculo de vacaciones, cálculo de horas extras, control de permisos del colaborador,

cálculo de liquidación y de incapacidades para la empresa D'Angelo. Los módulos se describen, a continuación:

Módulo de calcular planilla.

Este módulo se encarga de calcular la remuneración de los colaboradores, considerando las deducciones aplicables en Costa Rica, así como el pago correspondiente por las horas extras trabajadas, emitida por la ley costarricense.

Módulo de calcular aguinaldo.

Este módulo es administrado con el propósito de efectuar el pago anual del aguinaldo a los colaboradores. Este cálculo se realiza considerando la acumulación de los 12 salarios mensuales y su distribución en los 12 meses del año trabajados, según lo emitido por la ley costarricense.

Módulo de cálculo de vacaciones.

Este módulo se encarga de gestionar las solicitudes que corresponden a las vacaciones, una vez aprobadas, los pagos se procesan de manera eficaz. El empleado realizará la solicitud vía sistema a la jefatura inmediata, cuando esta apruebe o repruebe, se le notificará vía sistema a RR.HH. y este le notificará al empleado también vía sistema.

Módulo de calcular horas extras.

Este módulo se administrado para realizar el cálculo de remuneración por horas extras para cada colaborador. El empleado realizará la solicitud vía sistema a la jefatura inmediata, cuando esta apruebe o repruebe, se le notificará vía sistema a RR.HH. y este le notificará al empleado también vía sistema.

Módulo de control de permisos del colaborador.

Este módulo se encarga de las solicitudes correspondientes a los permisos solicitados por los empleados, ya sean con goce o sin goce salarial. El empleado realizará la solicitud vía sistema

a la jefatura inmediata, cuando esta apruebe o repruebe, se le notificará vía sistema a RR.HH. y este le notificará al empleado también vía sistema.

Módulo de calcular liquidación.

La incorporación de un módulo de cálculo de liquidación optimiza la precisión y eficiencia en el proceso de finalización laboral, asegurando pagos exactos y evitando posibles demoras, según lo emitido por la ley costarricense.

Módulo de calcular incapacidades

Este módulo se gestiona para el registro correcto de las incapacidades, tomando en cuenta los tipos de incapacidades que existen en Costa Rica, además de recopilar información relevante, la fecha de inicio de la incapacidad, la fecha de finalización y el tipo de incapacidad.

Módulo mantenimientos.

Este módulo se encarga de realizar el borrado, inserción, modificación y actualización de datos.

Módulo consultas.

Este módulo se encarga de generar información detallada procedente de diversas tablas de datos. Su funcionalidad abarca la extracción, procesamiento y presentación de los datos de manera clara y concisa.

Módulo reportes.

Este módulo se encarga de generar información proporcionada de las diferentes tablas y procesos, pero con un formato específico, según lo solicite el usuario, podrá ser impreso o por pantalla.

Módulo seguridad.

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

Módulo de control de asistencia.

Este módulo se encarga de llevar el control de la entrada y salida del colaborador permitiendo automatizar el sistema y evitando errores de cálculo.

Alcance Metodológico

Para la implementación de este prototipo, la empresa D'Angelo se basa en el modelo de cascada. Dentro del alcance de los conceptos, Crehana (2023) indica lo siguiente:

El modelo en cascada o waterfall model, es la propuesta de un enfoque metodológico que consiste en ordenar de forma lineal las distintas etapas que debes de seguir al momento de desarrollar tu software.

Aunque existen muchas metodologías que te ayudarán a definir este proceso para el desarrollo de un software, una de las más comunes y conocidas es el modelo en cascada.

Este enfoque de cascada originalmente fue propuesto en 1970 por Winston W. Royce, es también conocido como modelo lineal o modelo de ciclo de vida de un programa, y millones de personas lo han incorporado a sus planes en las últimas cinco décadas. (párr. 4-6)

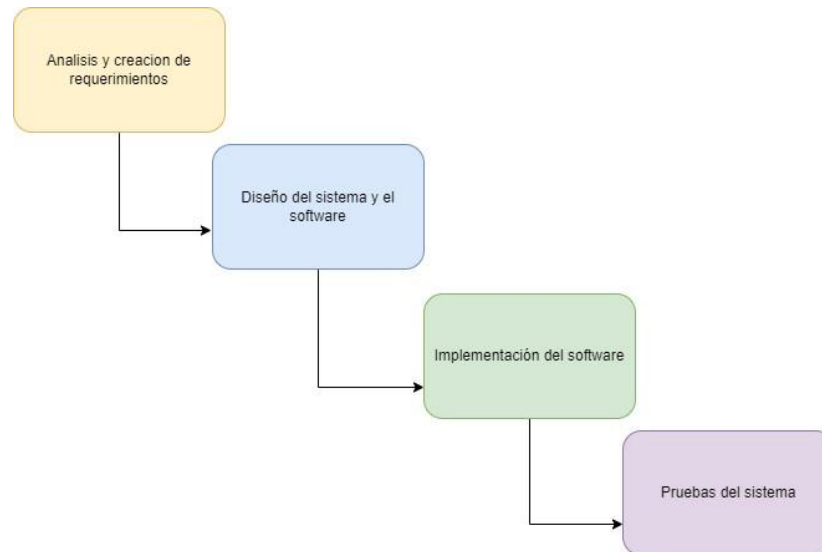
El modelo en cascada proporciona una estructura ordenada y comúnmente utilizada para guiar las diferentes etapas del desarrollo de *software*. Su relevancia radica en su simplicidad y claridad, convirtiéndolo en una opción popular para planificar y ejecutar proyectos de desarrollo de *software*.

El concepto anterior sugiere que el modelo en cascada es especialmente adecuado para proyectos en los que los requisitos son estables y pueden ser definidos con claridad desde el inicio, en el enfoque de desarrollo en cascada. El proceso se estructura en fases secuenciales, cada una de las cuales se inicia después de la finalización exitosa de la fase anterior.

- Las fases del modelo de cascada se listan en la figura 1.

- Análisis y creación de requerimientos.
- Diseño del sistema y el *software*.
- Implementación del *software*.
- Pruebas al sistema.

Figura 1:
Proceso de desarrollo.



Fuente: Elaboración propia.

A continuación, se detalla cada una de las fases a seguir del modelo cascada que se implementa en el prototipo.

Fases del modelo cascada por implementar

Análisis y creación de requerimientos. El primer paso de análisis es la etapa de preparación del proyecto, donde se detallan los requerimientos con el encargado de la empresa para solventar las necesidades y la problemática que tiene la compañía. Al crear los requerimientos, permitirá establecer una base sólida y garantizar que el proyecto esté alineado con los objetivos estratégicos de la empresa.

Diseño del sistema y el software. Dentro de esta etapa, se llevan a cabo los requerimientos, puesto que el programa se crea según las necesidades de la empresa que estarán detalladas en los requerimientos; de igual forma, es necesario definir la organización de la estructura y la de todos los elementos necesarios para el desarrollo del *software*.

Implementación del software. Una vez llegado a este punto, se realiza el diseño del prototipo, cada uno de los módulos o unidades de programa validando que cumpla con todas las especificaciones y que funcione de manera correcta.

Integración pruebas del sistema. Al llegar a este punto, en el modelo cascada se hace la integración de todos los módulos del prototipo, enlazándolos para que, de esta manera, funcione de la forma correcta, ya que el módulo de planilla depende de todos los demás para su correcto funcionamiento.

Alcance Tecnológico

Este prototipo se desarrolla en las siguientes herramientas:

Se utiliza una aplicación web basada en la tecnología de Microsoft. Además, se hace uso de Visual Studio 2022 como el entorno de desarrollo principal y se usa el lenguaje de programación C# con el .NET Framework, específicamente en su versión 4.8.1, que incluye ASP.NET. Esta fusión de herramientas ofrece una plataforma completa y sólida para el desarrollo de aplicaciones web, se complementa con MVC modelo, vista controlador y en los controladores se hacen mediante el lenguaje C#. Las vistas están complementadas con HTML, CSS y J JavaScript como modelo, se usa una conexión a base de datos que se describe, a continuación:

Como la base de datos Microsoft SQL Server 2022 tiene una licencia estándar, se aseguran funcionalidades esenciales para el almacenamiento y gestión de datos, así como capacidades avanzadas para la generación de informes y análisis, consolidando una solución integral y robusta para la construcción del prototipo funcional.

CAPÍTULO II: MARCO REFERENCIAL

La utilización de prototipos informáticos emerge como un elemento crucial en la fase de diseño y mejora de soluciones tecnológicas. Los prototipos informáticos son esenciales para agilizar el ciclo de desarrollo de *software* y reducir riesgos, al mismo tiempo que permiten evaluar la calidad del producto final de manera eficaz, según Jiménez (2024):

Un prototipo es una herramienta esencial en el desarrollo de productos. Sirve para validar conceptos, probar funcionalidades y mejorar la calidad final. Los tipos de prototipos varían en complejidad y propósito, y el proceso de creación implica diseño, construcción, pruebas y comunicación efectiva. Siguiendo estos pasos y consejos, podrás maximizar el valor de tus prototipos y acercarte al éxito en tus proyectos de desarrollo de productos. (párr. 2)

Los prototipos de *software* en el contexto de un sistema informático son representaciones funcionales tempranas que reproducen partes críticas del sistema interactivo propuesto. Estos prototipos están diseñados para probar aspectos específicos del sistema final, proporcionando una visión de cómo funcionará el sistema. Con base en esto, Euroinnova Business School (2023) indica lo siguiente:

Un sistema que nos permite almacenar y procesar información mediante una serie de partes interrelacionadas, cómo el hardware, el software y el personal. De hecho, estos son sus tres componentes fundamentales. En otras palabras, podemos decir que los sistemas informáticos son el conjunto de técnicas que nos permiten guardar y garantizar la seguridad de información mediante sistemas informatizados. (párr. 1)

Para lograr una implementación exitosa de un sistema informático, es fundamental contar con el *hardware* adecuado que pueda ejecutar eficientemente las aplicaciones y procesos requeridos. Esto implica asegurarse de que el *hardware* cumpla con los requisitos técnicos especificados, que pueden variar según las necesidades del sistema y las tecnologías utilizadas. De acuerdo con Apen (2022):

El hardware son aquellos elementos físicos o materiales que constituyen una computadora o un sistema informático. Es decir, son aquellas partes físicas de un sistema operativo tales cómo sus componentes eléctricos, electrónicos,

electromecánicos, mecánicos y cualquier elemento físico que esté involucrado. Todos ellos forman parte del hardware. Es la parte que puedes ver del ordenador, todos los componentes de su estructura física cómo la pantalla, el teclado, la torre, el ratón, etc. El hardware incluye los componentes físicos internos (disco duro, placa madre...) y periféricos, cómo la impresora. (párr. 1)

El funcionamiento efectivo de un sistema informático se basa en la estrecha colaboración entre el *software* y el *hardware*, es esencial para asegurar un rendimiento óptimo, debido a que ambos requieren de una asociación para garantizar la ejecución eficiente de tareas y procesamiento de datos. La combinación de estos es la esencia de los sistemas informáticos modernos, dentro de los conceptos de *software*. Al respecto, el Equipo editorial Etecé (2023) indica lo siguiente:

El término software es un vocablo inglés que fue tomado por otros idiomas y designa a todo componente intangible (y no físico) que forma parte de dispositivos cómo computadoras, teléfonos móviles o tabletas y que permite su funcionamiento.

El software está compuesto por un conjunto de aplicaciones y programas diseñados para cumplir diversas funciones dentro de un sistema. Además, está formado por la información del usuario y los datos procesados. (párr. 1-2)

Dentro de un sistema informático, la programación es el lenguaje que les da vida, permitiendo la creación de *software* funcional que interactúa con el *hardware* para realizar una amplia variedad de tareas. En el contexto de los sistemas informáticos descritos anteriormente, la programación es el elemento clave que une el *hardware* y el *software*, proporcionando las instrucciones y algoritmos necesarios para que los sistemas operen de manera eficiente y cumplan con las necesidades de los usuarios. De acuerdo con NeTEC Global Knowledge (2022), indica lo siguiente:

La programación informática es el arte del proceso por el cual se limpia, codifica, traza y protege el código fuente de programas computacionales, en otras palabras, es indicarle a la computadora lo que tiene que hacer. La programación informática es una de las habilidades esenciales que aprendes cuando estudias informática.

Detrás de todos los programas informáticos que conocemos y usamos de manera cotidiana para facilitarnos diversas actividades de nuestro día con día, existe

todo un proceso para poderlos crear. Este proceso es conocido como programación, conozcamos un poco más sobre lo que conlleva este proceso, por medio de la programación se establecen los pasos a seguir para la creación del código fuente de los diversos programas informáticos. (párr. 1-2)

Teniendo en cuenta lo anterior, para programar se utilizan diferentes lenguajes, ya que la diversidad de lenguajes de programación se debe a la variedad de necesidades y contextos en los que se utilizan los sistemas informáticos. Cada lenguaje de programación tiene sus propias características, sintaxis y paradigmas que lo hacen más adecuado para ciertos tipos de aplicaciones o situaciones. Dentro del concepto de lenguaje de programación, Mendoza (2023) indica lo siguiente:

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.

Los lenguajes de programación permiten a las computadoras procesar de forma rápida y eficientemente grandes y complejas cantidades de información. Por ejemplo, si a una persona se le da una lista de números aleatorios que van de uno a diez mil y se le pide que los coloque en orden ascendente, es probable que tome una cantidad considerable de tiempo e incluya algunos errores, mientras que si le das la misma instrucción a una computadora utilizando un lenguaje de programación, podrás obtener la respuesta en unos cuantos segundos y sin errores. (párr. 5-6)

Dentro de los lenguajes de programación, están los que se utilizan para el desarrollo de páginas web, juegan un papel fundamental en la creación de contenido dinámico y atractivo en Internet. Estos lenguajes permiten a los desarrolladores construir sitios web interactivos y funcionales que van más allá de simples páginas estáticas. Con relación al desarrollo web, Coppola (2023) indica lo siguiente:

Se conoce como desarrollo web al proceso de crear y mantener un sitio web que sea funcional en internet, a través de diferentes lenguajes de programación, según el modelo y la parte de la página que corresponda. Cada sitio tiene una URL única que lo distingue de los demás en la red informática mundial.

Un sitio web puede clasificarse de diferentes formas. Para cuestiones de desarrollo web principalmente se divide en dos partes.

Frontend. Es la parte que interactúa con el usuario, tanto en imagen cómo en función. Por ello está íntimamente relacionada con la experiencia del usuario (UX) y la interfaz de usuario (IU).

Backend. Se refiere a la parte que está en contacto directo con el servidor; es donde se aplica el código de programación para crear la estructura. Permanece en un segundo plano a cargo de la accesibilidad, actualización, bases de datos y cambios del sitio. (párr. 4-8)

En la actualidad, la tecnología de Microsoft ASP.NET Framework es una herramienta fundamental en el mercado, debido a que tiene un desarrollo rápido y eficiente en el desarrollo web, proporciona una serie de herramientas y bibliotecas que ayudan a los desarrolladores a crear las aplicaciones web de forma más eficiente. Dentro de los conceptos, ASP.NET Framework, AltexSoft (2021) indica:

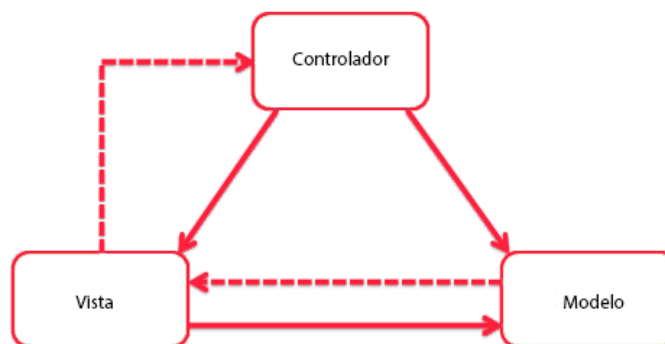
.NET es un marco y ecosistema de desarrollo de software diseñado y respaldado por Microsoft para permitir una fácil ingeniería de aplicaciones web y de escritorio. Es una plataforma gratuita popular que se utiliza actualmente para muchos tipos diferentes de aplicaciones, ya que proporciona el entorno de programación para la mayoría de las fases de desarrollo de software. .NET se adapta mejor a las empresas que buscan una amplia gama de funciones, cómo servicios basados en web, software de escritorio y soporte de infraestructura en la nube. (párr. 1)

La combinación de la tecnología de Microsoft ASP.NET Framework con el patrón MVC (Modelo-Vista-Controlador) ofrece una potente sinergia para el desarrollo de aplicaciones web. Al integrar ASP.NET Framework con el patrón MVC, los desarrolladores pueden aprovechar al máximo las ventajas de ambas tecnologías para crear aplicaciones web robustas y escalables, debido a que divide la programación en capas. En relación con los conceptos de modelo, vista y controlador, Caronte Studio. (2021) afirma lo siguiente:

Modelo-vista-controlador, abreviado MVC, es un tipo de arquitectura de software en el que se estipula que los datos (modelo), la interfaz de usuario (vista) y la lógica de control (control) son tres componentes distintos. Cada uno de estos componentes desempeña funciones específicas. (párr. 3)

El patrón MVC (Modelo-Vista-Controlador) establece una relación entre los tres componentes: la vista interactúa con el controlador y el modelo; el controlador interactúa con el modelo y la vista, y el modelo se relaciona con la vista y el controlador. En la figura 2, se aprecian las relaciones del patrón MVC.

Figura 2:
MVC (modelo, vista y controlador).



Fuente: Caronte Studio (2021).

La vista en el patrón de diseño MVC (Modelo-Vista-Controlador) juega un papel crucial en la experiencia del usuario en una aplicación web. Es el componente encargado de presentar la información de manera clara, intuitiva y atractiva para el usuario final. La vista es la capa de la aplicación que se comunica directamente con el usuario, mostrándole la interfaz gráfica con la que interactúa. De acuerdo con Rick-Anderson (2024), la definición de las vistas en el patrón MVC es:

Las vistas son los componentes que muestra la interfaz de usuario (IU) de la aplicación. Normalmente, esta interfaz de usuario se crea a partir de los datos del modelo. Un ejemplo sería una vista de edición de una tabla Productos que muestra cuadros de texto, listas desplegables y casillas basándose en el estado actual de un objeto Productos

En el desarrollo web, HTML es la columna vertebral para implementar y hacer visibles las vistas de una aplicación. Este lenguaje de marcado proporciona la estructura fundamental que define la disposición y el contenido de los elementos en una página web. En relación con los conceptos de HTML, Vadavo (2023) indica:

HTML es el lenguaje con el que se define el contenido de las páginas web. Corresponde a las siglas en inglés de Lenguaje de Marcado de Hipertexto,

básicamente son un conjunto de etiquetas que el navegador interpreta y se emplean para definir el texto y otros elementos que compondrán una página web, cómo imágenes, listas, tablas, vídeos, etc. (párr. 1)

El lenguaje de hipertexto HTML permite el diseño de las páginas web, pero, si se implementa solo, se tendría una página sin estilos. El lenguaje alcanza su máxima función al combinarlos con CSS y debido a que le darán estilos para que la página web sea más dinámica. En relación con los conceptos de CSS, Next U (2022) indica lo siguiente: “Cascading Style Sheets (CSS) que en español se traduce cómo “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).

Teniendo en cuenta lo anterior, CSS permite dar los estilos de la página, dentro de sus funciones, permite mejorar los estilos de las imágenes, letra, tabla y toda la información que complementa la página web. Al fusionar CSS y HTML, se obtiene un resultado agradable para el usuario. Sin embargo, si se complementa con JavaScript, se obtiene una página web más dinámica. En relación con los conceptos de JavaScript, Urrutia (2023) indica lo siguiente:

JavaScript es un lenguaje de programación diseñado en un principio para añadir interactividad a las páginas webs y crear aplicaciones web. A pesar de la similitud en el nombre, no está relacionado con Java. Se emplea en el desarrollo de páginas web para tareas cómo cambiar automáticamente la fecha de una página, hacer que una página aparezca en una ventana emergente al hacer clic en un enlace o que un texto o imagen cambien al pasar el ratón por encima. También suele emplearse para hacer encuestas y formularios. Se ejecuta en el ordenador del visitante a la web, por lo que no requiere descargas constantes desde el sitio web. (párr. 1)

El patrón MVC (Modelo-Vista-Controlador) alcanza su máxima funcionalidad cuando combina la vista con el modelo, ya que esta unión permite una separación clara y eficiente de las responsabilidades en una aplicación web. Al integrar la vista con el modelo, se establece una comunicación directa entre la presentación de la información al usuario. Según Rick-Anderson (2024):

Los objetos modelo son las partes de la aplicación que implementan la lógica del dominio de datos de la aplicación. A menudo, los objetos modelo recuperan y almacenan el estado del modelo en una base de datos. Por ejemplo, un objeto

Producto podría recuperar información de una base de datos, operar en ella y, después, escribir la información actualizada en una tabla Productos en SQL Server. (párr. 3)

Para almacenar información en el modelo, es necesario contar con una base que permite el acceso fácil y manipulación de la información. Así mismo, permite que los datos no se pierdan y que se tenga un control total, al almacenarlos en una base de datos facilita mantener un control de seguridad de los datos, ya que solo la persona encargada podrá tener acceso. De acuerdo con GoDaddy (2023), una base de datos es:

Una base de datos en informática es un sistema estructurado y organizado que permite almacenar, administrar y recuperar información de manera eficiente. Se compone de una colección de datos interrelacionados que se almacenan en una estructura definida con el objetivo de facilitar su acceso y gestión. Las bases de datos son ampliamente utilizadas en diversas aplicaciones informáticas, desde simples sistemas de almacenamiento hasta complejas soluciones empresariales con infinitud de objetivos diferentes. Proporcionan una forma estructurada de almacenar información, lo que permite la creación, modificación y eliminación de datos de manera controlada e incluso de forma remota si así se requiere. Las bases de datos se basan en un modelo de datos que define la estructura y las relaciones entre los datos. Los sistemas de gestión de bases de datos (SGBD) son el software encargado de administrar y controlar estas bases de datos. Además, proporcionan herramientas y lenguajes de consulta para interactuar con los datos, permitiendo realizar consultas, generar informes y mantener la integridad de la información almacenada. (párr. 1)

Para hacer uso de una base de datos en el desarrollo de aplicaciones web, SQL Server emerge como una solución robusta y escalable. SQL Server es un sistema de gestión de bases de datos relacional desarrollado por Microsoft, diseñado para manejar grandes volúmenes de datos y soportar múltiples usuarios simultáneos de manera eficiente. En relación con los conceptos de SQL Server, Hughes y Stedman (2021) comentan:

Microsoft SQL Server es un sistema de gestión de bases de datos relacionales (RDBMS) que admite una amplia variedad de aplicaciones de procesamiento de transacciones, inteligencia empresarial y análisis en entornos informáticos

corporativos. Microsoft SQL Server es una de las tres tecnologías de bases de datos líderes del mercado, junto con Oracle Database y DB2 de IBM. (párr. 1)

Una vez explicados los conceptos de vista y modelo en el patrón MVC (Modelo-Vista-Controlador), para el controlador se utiliza el lenguaje de programación C#, debido a que es uno de los lenguajes web con más impacto por su seguridad y el uso de librería permite a los programadores desarrollar de manera fácil y ordenada. En relación con los conceptos de C#, BillWagner (2023) indica lo siguiente:

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. Este paseo proporciona información general de los principales componentes del lenguaje en C# 8 y versiones anteriores. Si quiere explorar el lenguaje a través de ejemplos interactivos, pruebe los tutoriales de introducción a C#.

C# es un lenguaje de programación orientado a componentes, orientado a objetos. C# proporciona construcciones de lenguaje para admitir directamente estos conceptos, por lo que se trata de un lenguaje natural en el que crear y usar componentes de software. Desde su origen, C# ha agregado características para admitir nuevas cargas de trabajo y prácticas de diseño de software emergentes. En el fondo, C# es un lenguaje orientado a objetos. (párr. 1-2)

La elección de Visual Studio 2022 como el entorno de desarrollo principal para la creación de un prototipo funcional se fundamenta en varias razones sólidas. En primer lugar, Visual Studio 2022 es una herramienta altamente estable y confiable, respaldada por Microsoft, lo que garantiza un ambiente de desarrollo robusto y de alta calidad. Al respecto, Anandmeg (2023) detalla:

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.

Gracias a la variedad de características y lenguajes que admite Visual Studio, puede pasar de escribir su primer programa "Hola mundo" a desarrollar e implementar aplicaciones. Por ejemplo, compile, depure y pruebe aplicaciones de .NET y C++, edite páginas de ASP.NET en la vista del diseñador web, desarrolle aplicaciones móviles y de escritorio multiplataforma con .NET o cree interfaces de usuario web con capacidad de respuesta en C#. (párr. 1-2)

La creación de un prototipo funcional para la gestión de recursos humanos implica un enfoque centrado en las necesidades y procesos particulares de esta área dentro de la empresa. El Departamento de Recursos Humanos constituye un grupo de individuos que se integran en una organización con el propósito de llevar a cabo una diversidad de funciones y responsabilidades. Con base en recursos humanos, Gamarra (2023) explica:

Los recursos humanos son un departamento dentro de las empresas en el que se gestiona todo lo relacionado con las personas que trabajan en ella. Esto incluiría desde el reclutamiento, selección, contratación, onboarding o bienvenida, formación, promoción, nóminas y despidos. (párr. 1)

Dentro de las responsabilidades del Departamento de Recursos Humanos, el cálculo de la planilla es un módulo indispensable que garantiza la correcta remuneración de los colaboradores y el cumplimiento de las obligaciones legales y laborales por parte de la empresa. Con base en planilla, Proavance (2023) informa:

La planilla de pago es el documento en el que se especifican los detalles y aspectos vinculados con el sueldo de la persona. En ella aparece el sueldo bruto, las distintas bonificaciones, los aportes que realiza y las deducciones que experimenta según su tipo de contrato. (párr. 1)

El cálculo de planilla se implementa en el prototipo, ya que se busca gestionar el pago de forma correcta de los colaboradores. La implementación de este módulo deja atrás las hojas de cálculo o Excel, esto permite un ahorro de tiempo y mayor satisfacción en los colaboradores, debido que no existirán errores de cálculo en la planilla.

El cálculo del aguinaldo es un rubro de suma importancia en la elaboración de la planilla de pago de los empleados, ya que constituye un derecho laboral fundamental y una obligación para

los empleadores. Es una prestación económica adicional que se otorga a los trabajadores en ciertas épocas del año, generalmente a finales del año. De acuerdo con BG&A (2023):

El aguinaldo es un décimo tercer salario al que tienen derecho los trabajadores por disposición legal. El cálculo del aguinaldo debe incluir todas las remuneraciones que haya percibido el trabajador desde el 1° de diciembre del año anterior hasta el 30 de noviembre del presente año.

Todas las remuneraciones percibidas por el trabajador durante ese período deben ser tomadas en cuenta para ese cálculo. Con “remuneraciones” me refiero a los salarios, pago de horas extras, bonificaciones, comisiones, o bien, cualquier monto que se haya recibido de acuerdo con un determinado esquema de compensación. (párr. 3- 4)

El cálculo de planilla es un aspecto fundamental en la gestión de recursos humanos, ya que asegura la correcta remuneración de los colaboradores y el cumplimiento de las obligaciones legales y laborales por parte de la empresa. Este proceso no solo garantiza que los empleados reciban el pago correspondiente por su trabajo, sino que también abarca una serie de aspectos cruciales para la relación laboral

Además, el cálculo de planilla no solo tiene implicaciones financieras, sino que también impacta en la satisfacción y el bienestar de los empleados. Un cálculo preciso y transparente contribuye a generar confianza y motivación en el equipo de trabajo, al tiempo que evita posibles conflictos relacionados con la remuneración y los beneficios laborales. Adicional a esto, el prototipo tendrá modelos extras, los cuales son:

El cálculo de vacaciones es otro aspecto crucial en la gestión de la planilla de pago de los empleados, ya que garantiza el cumplimiento de otro derecho laboral fundamental. Las vacaciones representan un período de descanso y recuperación para los colaboradores, permitiéndoles recargar energías y mantener un equilibrio entre su vida laboral y personal. Estrada (2024) detalla lo siguiente:

Las vacaciones son un derecho laboral que responde a la necesidad biológica de descanso de toda persona trabajadora. En Costa Rica, toda persona trabajadora tiene derecho a disfrutar de dos semanas de vacaciones (equivalentes a 12 días + 2 días de descanso adicionales) como mínimo, por cada cincuenta semanas de labores continuas, al servicio de una misma persona empleadora, independientemente de la

jornada semanal que se labore, sea de ocho horas, seis horas, medio tiempo o una hora, de un día o varios días a la semana. (párr. 1)

La determinación de las horas extras constituye otro componente importante dentro del proceso de elaboración de la planilla de trabajadores, ya que representa una compensación adicional por el tiempo de trabajo realizado más allá de las horas regulares establecidas en el contrato laboral. Las horas extras son una herramienta comúnmente utilizada por las empresas para hacer frente a periodos de alta demanda de trabajo o para cubrir necesidades operativas específicas. *Ética Legal Costa Rica (2023)* comenta lo siguiente:

En Costa Rica, el cálculo y pago de las horas extra se rigen por el Código de Trabajo y los convenios colectivos que puedan aplicar. Las principales pautas son las siguientes:

Jornada laboral regular: La jornada diurna laboral regular en Costa Rica es de 8 horas diarias y 48 horas semanales, la jornada mixta de 7 horas diarias y 42 horas semanales, la nocturna de 6 horas diarias y 36 horas semanales, y próximamente será aprobada la jornada extraordinaria 4x3. Cualquier tiempo trabajado por encima de estas horas se considera horas extra.

Pago de las horas extra: Las horas extra deben pagarse a una tasa mayor que la hora regular. Según el Código de Trabajo, las horas extra se pagan calculando la hora ordinaria multiplicada por 1.5, lo que es igual a pagar el 150% de la hora ordinaria, lo que comúnmente llamamos "a tiempo y medio". (párr. 3-5)

La gestión de permisos de colaborador es otra faceta esencial dentro del proceso de administración de la planilla. Los permisos representan una parte fundamental de las relaciones laborales, ya que permiten a los trabajadores ausentarse del trabajo por diversas razones, como enfermedad, asuntos personales o vacaciones. De acuerdo con el Poder Ejecutivo de Costa Rica (1954), se afirma lo siguiente según el Decreto Ejecutivo 21:

Los jefes podrán conceder licencia hasta por una semana con goce de sueldo en los casos de matrimonio del servidor, el fallecimiento de cualquiera de sus padres, hijos, hermanos o cónyuge. También podrán conceder este derecho a aquellos servidores padres de hijos nacidos dentro o fuera del matrimonio. En este último caso solo cuando sean hijos reconocidos y en su función paternal. (art.33)

Existen dos tipos de permiso: primero están los permisos con goce de salario; son aquellos en los cuales el empleado tiene derecho a ausentarse del trabajo por motivos válidos, como enfermedad, citas médicas, asuntos personales o familiares, entre otros, y continúa recibiendo su salario completo durante el período de ausencia. Según Ministerio de Educación Pública de Costa Rica (2024):

Es servicio consiste en otorgar licencia o permiso con goce de salario de acuerdo con el artículo 165 del Estatuto del Servicio Civil y 33 inciso a) del Reglamento del Estatuto de Servicio Civil. En el cual, para tramitarlos por enfermedad grave de algún familiar debidamente comprobada, fallecimiento de un familiar, matrimonio y fuerza mayor o caso fortuito. (párr. 1)

El otro tipo es el sin goce de salario, son aquellos en los cuales el empleado tiene autorización para ausentarse del trabajo por un período determinado, pero no recibe remuneración económica durante ese tiempo de ausencia. Este tipo de permisos pueden ser otorgados por diversas razones. De acuerdo con Ministerio de Educación Pública de Costa Rica (2024), los permisos sin goce de salario son los siguientes:

Este servicio consiste en otorgar una Licencia o Permiso sin goce de salario de acuerdo con el Artículo 36 del Reglamento de Servicio de Conserjería de las Instituciones Educativas Oficiales, a una persona funcionaria regular con base a su solicitud para ausentarse de su lugar de trabajo, la cual impide percibir salario por el periodo otorgado. (párr. 1)

La inclusión del cálculo de liquidación en el proceso de administración de la planilla garantiza que los empleados reciban los montos que les corresponden de manera justa y oportuna al finalizar su relación laboral con la empresa. Este proceso requiere una cuidadosa revisión de los registros y documentos relacionados con la compensación del empleado, así como una comprensión clara de las políticas y regulaciones laborales aplicables. Al respecto, ASOBANCARIA (2021) detalla:

Cuando se habla de liquidación, normalmente se refiere a la liquidación del contrato de trabajo (cuando termina la relación entre la empresa y el empleado) o a la liquidación de la nómina mensual o quincenal según el periodo de pago pactado en el contrato de trabajo (básicamente, el salario). Para este artículo nos centraremos en la liquidación del contrato de trabajo. (párr. 1)

Cabe mencionar que, a la hora de realizar la liquidación, deben tomarse en cuenta muchos rubros, tales como salario y horas extras pendientes, debido a que el pago se realiza a fin de mes, se tiene que remunerar los días de vacaciones no disfrutados, la proporción del aguinaldo incluyendo horas extras y cualquier otro pago o bonificación relacionada con el salario.

Además de tener en cuenta cuando aplicar estas incapacidades, se va a mostrar la información para generar la incapacidad al colaborador. Dentro de la información, León (2022) indica lo siguiente:

La incapacidad es un periodo de reposo que le permite a la persona recuperar su estado de salud. Se considera incapacitante a una dolencia o padecimiento que le impida a la persona laborar de manera ordinaria. Sin embargo, esta decisión está fuera del margen del trabajador, pues depende de un criterio médico tras la respectiva evaluación. (párr. 5)

Por lo cual, en este módulo el patrono tiene ocho días hábiles para enviar un aviso de accidente al INS o a la CCSS que le permitirá al trabajador asistir a un centro de estas entidades a la cual le corresponda asistir. Sobre esto Valverde (2020) indica:

Recordemos que la incapacidad por enfermedad cubre a partir del cuarto día de la incapacidad. Los primeros tres días son cubiertos por el patrono, pagando un 50% del salario como subsidio por enfermedad. A partir del cuarto día, el seguro de salud de la CCSS paga el 60% del salario como subsidio. (párr. 4)

El módulo de mantenimiento es una adición esencial al prototipo en desarrollo, ya que proporciona funcionalidades clave para la gestión eficiente de datos. Este módulo estará diseñado para ejecutar operaciones fundamentales como borrado, inserción, modificación y actualización de datos en el sistema. Es una herramienta poderosa que permite a los usuarios administradores mantener la integridad y la precisión de la información almacenada.

Es importante destacar que únicamente los usuarios con privilegios de administrador tendrán acceso y autorización para realizar modificaciones en los datos a través de este módulo. Esto garantiza un control riguroso sobre los cambios realizados en el sistema, mitigando riesgos de errores o manipulaciones indebidas.

El módulo de mantenimiento proporcionará una interfaz intuitiva y segura para llevar a cabo estas operaciones, lo que facilitará la administración y la actualización de datos de manera eficiente. Además, se implementarán medidas de seguridad adicionales, como registros de

auditoría y restricciones de acceso, para proteger la integridad y la confidencialidad de la información almacenada en el sistema.

El módulo de consultas representa una valiosa adición al prototipo, ya que ofrece a los usuarios una herramienta poderosa para realizar búsquedas de datos de manera eficiente y precisa. Este módulo estará diseñado para facilitar la recuperación de información específica dentro del sistema, lo que permitirá a los usuarios acceder rápidamente a los datos relevantes que necesiten en un momento dado.

Una de las principales ventajas del módulo de consultas es su capacidad para optimizar el proceso de búsqueda, lo que se traduce en ahorro de tiempo y esfuerzo por parte de los usuarios. Ofrecerá una interfaz intuitiva y fácil de usar, lo que garantizará que los usuarios, incluso aquellos con poca experiencia técnica, puedan aprovechar al máximo sus capacidades. Se implementarán funciones de búsqueda avanzada, como la búsqueda por palabras clave, filtros por categorías y opciones de ordenamiento, para brindar una experiencia de usuario fluida y eficiente.

Por su parte, el módulo de reportes es una función esencial que enriquecerá significativamente la capacidad del prototipo para analizar y presentar datos de manera clara y comprensible. Este módulo permitirá a los usuarios generar informes detallados y personalizados sobre diversos aspectos del sistema, proporcionando una visión completa y en tiempo real de la información relevante; esto se genera en PDF para que los colaboradores puedan tener una copia de dicho registro.

El módulo de seguridad es de vital importancia en cualquier sistema, y su función principal será garantizar la protección de la información mediante la autenticación de contraseñas y la definición de perfiles de usuario. En primer lugar, la autenticación de contraseñas asegura que solo los usuarios autorizados tengan acceso al sistema. Esto se logra mediante la verificación de las credenciales de inicio de sesión, como nombres de usuario y contraseñas, frente a una base de datos segura.

CAPÍTULO III: MARCO METODOLÓGICO

El marco metodológico está orientado a cómo será el desarrollo de la investigación y las metodologías que se implementan. Además, es clave debido que establece los principios que guiarán el estudio y se puede realizar un plan de actividades; proporciona una estructura que permite obtener mejores resultados al implementar los métodos, técnicas y herramientas que se utilizarán en la recolección y análisis de datos.

Enfoques de la Investigación

El enfoque de la investigación está relacionado con la orientación metodológica que se desarrollará. Este enfoque constituye la estrategia general para abordar, plantear, construir y solucionar el problema planteado, además, muestra el rumbo que seguirá la investigación, brindando una guía sobre cómo se llevará a cabo el estudio. También ayuda a definir los métodos, técnicas e instrumentos empleados.

Enfoque Cualitativo

El enfoque cualitativo es muy importante, ya que este se basa en la reelección de datos y permite generar hipótesis, esto ayuda a generar nuevas ideas que pueden ser exploradas. Además, tiene flexibilidad, debido a que permite investigar a medida que avanza, así como la exploración de nuevos temas. Al respecto, Hernández et al. (2017) indican:

En la investigación cualitativa, el diseño también implica el plan o estrategia para recolectar la información que se requiere a fin de responder al planteamiento del problema. Aunque resulta más flexible que el diseño cuantitativo en cualquier modalidad, porque la indagación cualitativa es en gran medida emergente y se encuentra aún más sujeta a las circunstancias de cada contexto o ambiente.

Asimismo, el diseño cualitativo puede subrayar un aspecto cultural, un proceso, la experiencia humana, una historia de vida o varias, la resolución de una problemática específica, el entendimiento de las relaciones entre variables o la comprensión de un caso, lo cual le otorga un sello diferente o clase de abordaje, y esto se deriva directamente del planteamiento del problema. (p.122)

Según lo mencionado anteriormente, esta investigación se enfoca en recolectar la información necesaria para responder al planteamiento del problema. Además, es flexible, ya que permite adaptarse a situaciones cambiantes durante la investigación. Está basado en un proceso de experiencia humana de una vida o de varias y permite la resolución de un problema en específico.

Enfoque de Investigación Seleccionado

Para el proyecto se seleccionó el enfoque de investigación cualitativo como principal vía de exploración. Esta elección permite profundizar en el análisis del tema seleccionado, ayuda a detectar de forma minuciosa cualquier inconveniente, así como las deficiencias de la empresa. Todo esto es expuesto mediante los datos recopilados. Este enfoque permite la solución de problemas y lo hace ideal para el proyecto que se está desarrollando, debido a que se basa en un proceso de experiencia humana de una vida o de varias; permite recopilar información del problema que se está tratando; su flexibilidad ayuda a investigar a medida que avanza la investigación y permite la exploración de nuevos temas.

Tipos de Investigación

Los proyectos pueden ser enfocados a diferentes tipos de investigación dependiendo de las necesidades específicas y el enfoque deseado. Cada tipo de investigación tiene su propia finalidad, lo que determina la forma en que se lleva a cabo la recopilación de datos. Al respecto, Guía Interactiva (s.f) afirma lo siguiente:

Los proyectos de investigación pueden ser clasificados con base en los siguientes criterios: por el propósito, la investigación puede ser básica o aplicada; según los medios usados para obtener los datos, puede ser documental, de campo o experimental; atendiendo al nivel de conocimientos que se adquieren, podrá ser exploratoria, descriptiva o explicativa; dependiendo del campo de conocimientos en que se realiza, es científica o filosófica; conforme al tipo de razonamiento empleado, es espontánea, racional o empírico-racional; acorde con el método utilizado, es analítica, sintética, deductiva, inductiva, histórica, comparativa, etc.; y conforme al número de investigadores que la realizan, es individual o colectiva. (párr. 1)

Investigación Descriptiva

La investigación descriptiva se basa en definir características de un fenómeno o situación de grupos de individuos, lo que la hace ideal para dar información más detallada, sin manipulación y enfocada en la realidad actual; ya que esta se centra en los hechos tal como se presentan en el momento del estudio. Al respecto, Cimec (2023) detalla:

La investigación descriptiva es aquella que busca, cómo su propio nombre indica, describir de manera detallada alguna variable relacionada con el estudio, cómo pueden ser las características de la población, un fenómeno concreto, etc. Es decir, se centra en brindar una representación precisa y detallada de los hechos observados, sin tratar de establecer relaciones de causa y efecto. (párr. 2)

La importancia de la investigación descriptiva se basa en la capacidad que tiene para la recolección y el análisis de datos. Permite tener un contexto exacto y mucho más claro de la información, debido a que esta se centra en los hechos tal como se presentan en el momento del estudio, haciendo que se vuelva indispensable para el desarrollo de la investigación

Tipo de Investigación Seleccionado

El tipo de investigación seleccionada es la descriptiva, debido a que esta brinda datos más exactos para la problemática de la empresa y cómo se llevan los registros. Este tipo de investigación permite recopilar información de los colaboradores acerca de las posibles fallas que puedan existir a la hora el pago de planilla. Esto permitirá que se obtengan los datos correctos y con información confiable, así mismo permite que el desarrollo del prototipo sea más eficiente, ya que este tipo de investigación es la más confiable para la recolección de datos.

Fuentes de información

Las fuentes de información son los recursos que proporcionan datos, hechos y opiniones sobre un tema en específico; ayudan tener una mejor comprensión de temas que se desconocen.

Sin embargo, se debe evaluar la credibilidad de las fuentes donde se recopila información con el fin de garantizar la calidad de la investigación. Al respecto, Barbara (2024) detalla:

Las fuentes de información son la base de cualquier trabajo académico. Representan la evidencia y el sustento necesario para respaldar las ideas y los argumentos presentados. Estas fuentes pueden ser tanto impresas como digitales, y abarcan una amplia gama de recursos, como libros, revistas científicas, sitios web especializados, bases de datos académicas y archivos históricos, por mencionar algunos.

La utilización de fuentes de información confiables y actualizadas es fundamental para garantizar la veracidad y la solidez de un trabajo académico. Al acceder a fuentes de calidad, los estudiantes demuestran su capacidad para investigar de manera crítica, evaluar la información y utilizarla de manera ética. Algo esencial si quieres aprobar el TFG, TFM o tesis doctoral. (párr. 4-5)

Las fuentes de información son de suma importancia, debido a que respaldan la investigación. Estas pueden variar ampliamente dependiendo del tipo de información que se proporciona y fiabilidad de estas. Las fuentes de información pueden ser primarias, secundarias y terciarias, por lo que es importante que se seleccione la fuente para el tema que se está investigando.

Fuentes Primaria

Las fuentes primarias son los materiales originales que brindan información directa sobre un tema. Estas son las pruebas de lo que ocurrió sin que nadie más lo haya interpretado antes. Las fuentes primarias pueden ser libros, diarios, fotografías entre otros. Al respecto, Suárez (2024) indica:

Las fuentes primarias son aquellas que proporcionan datos e información original y directa sobre un tema específico. Estas fuentes son producidas por los actores directamente involucrados en los hechos, por ejemplo: documentos oficiales, diarios, cartas, grabaciones de audio, entrevistas, encuestas, entre otros. (párr.7)

Las fuentes primarias aportan en el proceso de la investigación, ya que estas brindan autenticidad y credibilidad. Además, proporcionan una visión auténtica de los eventos y sin la distorsión que se puede llevar mediante el paso del tiempo o malas interpretaciones. Además, son la base sobre la cual se construyen interpretaciones históricas y teorías en otros campos.

Fuentes Secundaria

Las fuentes secundarias proporcionan una información o análisis de información existente, en otras palabras, de las fuentes primarias. Estas están escritas después de los eventos que se describen y están basadas en múltiples informaciones de las fuentes primarias. Al respecto, Suárez (2024) indica:

Las fuentes secundarias son aquellas que proporcionan datos e información indirecta, es decir, que se basan en información ya existente y tienden a expresar un punto de vista. Estas fuentes son producidas por investigadores, historiadores, académicos, periodistas, entre otros, que utilizan fuentes primarias para elaborar sus estudios. A este tipo de información también se le denomina vulgarmente “de segunda mano” y corre el riesgo de ser alterada o manipulada. Algunos ejemplos de fuentes secundarias son: libros de historia, artículos académicos, ensayos, reseñas, etc. (párr. 9)

Las fuentes secundarias son de importancia, ya que ayudan a contextualizar y analizar información de las fuentes primarias. Además, proporcionan diferentes perspectivas y enfoque de algún tema o evento. Estas suelen ser más accesibles y fáciles de entender para el público en general en comparación con las fuentes primarias, debido a que estas son interpretaciones.

Fuentes Terciaria

Las fuentes terciarias recopilan y organizan información de fuentes primarias y secundarias. Estas presentan la información resumida o clasificada para facilitar la comprensión o para hacer alguna consulta rápida. Algunos ejemplos de fuentes de información son índices, bibliografías y catálogos de biblioteca. Al respecto, Investigadores (2022) indica:

Una fuente terciaria consolida y organiza las fuentes primarias y secundarias juntas en una sola fuente para facilitar el acceso rápido a la información. Las fuentes terciarias son buenos puntos de partida para proyectos de investigación porque a menudo extraen el significado esencial o los aspectos más importantes de grandes cantidades de información en un formato conveniente. (párr. 16)

La importancia de las fuentes terciarias radica en su capacidad de proporcionar una visión general de un tema sin necesidad de consultar las fuentes primarias y secundarias. Esto hace que

sea útil para aquellos que desean obtener una comprensión básica o introductoria sobre un tema antes de profundizar en investigaciones más detalladas.

Variables

Las variables juegan un papel importante dentro de la investigación, debido a que proporcionan un marco claro para medir y observar algún fenómeno en específico; permiten a los investigadores cuantificar y registrar los datos de manera sistemática. Las variables en una investigación se trabajan con base en los objetivos y tienen su origen en cada objetivo específico.

Variable Conceptual

Una variable conceptual es un concepto o una característica que se requiere estudiar en una investigación, es decir, es aquella que dice qué se entenderá por su propio concepto. Al respecto, SalusPlay. (s. f.) afirma:

Sería la definición real. Se trata de definiciones de diccionarios o de libros especializados que describen la esencia o las características de una variable, objeto o fenómeno. Constituyen la adecuación de la definición conceptual a los requerimientos prácticos de la investigación. Ej. el término actitud se definiría como “una tendencia o predisposición a evaluar de cierta manera un objeto o un símbolo de este objeto”. (párr. 8)

Estas variables son importantes para la investigación, debido a que vinculan directamente con la realidad, al proporcionar la conexión entre los conceptos teóricos y la realidad. Al definir claramente las variables conceptuales, los investigadores pueden identificar los aspectos específicos de la realidad que se pretende estudiar.

Variable Operacional

Las variables operacionales son fundamentales para la investigación, ya que buscan organizar y dar claridad para la confección de los instrumentos de recolección de datos. Son la base para el diseño de los instrumentos, como cuestionarios o encuestas. También buscan garantizar la validez y confiabilidad de las medidas utilizadas en la investigación.

Variables Instrumentales

Las variables instrumentales, tal como sugiere su denominación, representan los medios o instrumentos específicos que se utilizan para recoger la información necesaria en el estudio. Estos instrumentos varían dependiendo del tipo de investigación y de los datos que se desean obtener en general; pueden incluir cuestionarios, entrevistas y observaciones, entre otros. La elección de los instrumentos es importante, debido a que estos garantizan que los datos recolectados sean válidos y confiables.

Cuadro de Variables

El cuadro de variables es de suma importancia para la investigación porque proporciona una estructura organizada y clara de las variables que se estudian en el proyecto. En este apartado, se muestran las variables conceptuales y operacionales, así como sus definiciones y relaciones, lo que permite a los investigadores tener una visión general y detallada de cómo se relacionan las diferentes variables en el estudio. La tabla 3 muestra el uso de las diversas variables de acuerdo con los objetivos específicos.

Tabla 3:
Cuadro de variables

Objetivo específico	Variable	Variable Conceptual	Variable Operacional	Variable Instrumental
Analizar los requerimientos que serán implementados en el prototipo funcional con el propósito de cumplir con las necesidades de la entidad.	Requerimientos funcionales y no funcionales	Northware (2022) indica: “En general, los requerimientos funcionales describen acciones específicas que el ingeniero de software debe ser capaz de realizar durante el desarrollo de software. Los requerimientos funcionales a menudo se dividen en reglas de negocio y casos de uso.	Entrevista	Guía de entrevista

Objetivo específico	Variable	Variable Conceptual	Variable Operacional	Variable Instrumental
		<p>Las reglas de negocio son declaraciones de alto nivel que definen lo que un sistema debe hacer, mientras que los casos de uso son descripciones más detalladas de cómo debe funcionar el sistema.</p> <p>los requerimientos no funcionales describen características específicas que el software debe poseer durante el desarrollo de la aplicación. Por lo general, se dividen en tres categorías: rendimiento, seguridad y calidad” (párr. 11-12).</p>		
<p>Diseñar la estructura del prototipo basado en los requerimientos, así como sus módulos.</p>	<p>Prototipo módulos.</p>	<p>Alegsa (2023) indica lo siguiente: “En programación, un módulo es un software que agrupa un conjunto de subprogramas y estructuras de datos. Los módulos son unidades que pueden ser compiladas por separado y los hace reusables y permite que múltiples programadores trabajen en diferentes módulos en forma simultánea, produciendo ahorro en los tiempos de desarrollo” (párr. 2).</p>	<p>Arquitectura del software Diseño base de datos.</p>	<p>Draw.io MySQL Workbench.</p>
<p>Desarrollar la programación</p>	<p>Desarrollo de software.</p>	<p>Bello (2023) indica:</p>	<p>Desarrollo del prototipo</p>	<p>Visual Studio 2022</p>

Objetivo específico	Variable	Variable Conceptual	Variable Operacional	Variable Instrumental
del prototipo, con base en el diseño que se estableció.		“Podríamos definir el desarrollo de software como los procesos involucrados a la hora de crear un programa de software, que incorpora todas las etapas a lo largo del ciclo de vida del desarrollo de sistemas. Esto lo hacen gracias a la utilización de uno o más lenguajes de programación específicos que brindan funcionalidad para abordar objetivos comerciales o personales” (párr. 1).	Construcción de base de datos.	Microsoft SQL Server 2022.
Realizar pruebas para el funcionamiento óptimo del prototipo final.	Pruebas del proyecto	Atlassian. (s. f.). afirma: “Las pruebas de software son un proceso organizativo dentro del desarrollo de software en el que se verifica la corrección, la calidad y el rendimiento del software crucial para el negocio Las pruebas de software sirven para garantizar que las funciones de producto y los sistemas empresariales previstos se comporten correctamente como se esperaba” (párr. 1).	Casos de prueba	Guía de casos de prueba

Fuente: Elaboración propia, 2024.

Instrumentos de recolección de datos

Los instrumentos de recolección de datos son las herramientas implementadas para recopilar información. La recolección de datos es importante en un proyecto, debido a que representa la base sobre la cual se realizan los análisis y ayudan a tomar decisiones mediante la información recolectada.

Entrevista

La entrevista es una técnica de recolección de datos que implica hacer preguntas a una persona o a un grupo determinado para obtener información detallada sobre un tema en específico. Es un proceso interactivo donde el entrevistador busca obtener respuestas que puedan facilitar la comprensión de algún problema mediante preguntas estratégicas enfocadas al problema.

Para la implementación de la entrevista a la farmacia D'Angelo, se plantean 16 preguntas a cada colaborador, con el fin de obtener respuestas sobre las carencias que presenta la empresa a la hora de realizar el cálculo de planilla. Es importante mencionar que la entrevista se aplica de manera individual, debido a que, si se hace en grupo, se puede influenciar en las respuestas de los demás colaboradores. Las respuestas se emplean únicamente con el fin de mejorar el prototipo, estas son confidenciales y no son compartidas con la jefatura de la empresa.

La observación

La observación es una técnica de recolección de datos por la cual se puede observar el comportamiento o procesos que se llevan a cabo dentro de la empresa; mediante esta se ven los procesos del cálculo de planilla, aguinaldo, vacaciones, permisos del colaborador, liquidación, incapacidades y horas extras; lo cual se lleva en una tabla detallada para ver si esta cumple con las funciones básicas del cálculo de planilla dentro de la empresa.

CAPÍTULO IV: ANÁLISIS DE RESULTADOS

El presente capítulo tiene como objetivo examinar los resultados obtenidos para el desarrollo del prototipo funcional para la gestión de recursos humanos en la farmacia D'Angelo. Se implementaron diversas actividades para la recolección de datos con el fin de evaluar el cálculo de planilla, cálculo de aguinaldo, cálculo de vacaciones, cálculo de horas extras, control de permisos de colaborador, cálculo de liquidación e incapacidades. Mediante este análisis, se busca la creación óptima de un prototipo que permita agilizar los cálculos, además, erradicar los errores de cálculos de los módulos antes mencionados.

Proceso de recolección y análisis de datos

Para la recolección y análisis de datos, se implementaron los métodos de recolección de datos, los cuales fueron la entrevista y la observación. Con el fin de realizar la entrevista, se pactó una fecha con el encargado de la farmacia para aplicar una serie de preguntas y, de esta forma, obtener información confiable con el fin de mejorar los cálculos en la empresa mediante la creación de un prototipo que permita agilizar los procesos. Al implementar la observación, se elaboró una guía para ver si cumplen con los procesos y observar las oportunidades de mejora. Cada uno de estos instrumentos de recolección de datos fue aplicado a 14 personas.

Con el fin de llevar a cabo la entrevista, se acordó entre ambas partes un día en específico que no afectara las labores de la farmacia D'Angelo. Se realizó una visita a la empresa para implementar la entrevista; a fin de facilitar la información, la administradora Paola Andrea Niño Gallego brindó un tiempo para aplicarla a los colaboradores y, de esta manera, implementar el desarrollo de forma óptima.

Para llevar a cabo la entrevista, se eligió el día que harían el cálculo de la planilla con el fin de observar las fórmulas y ver las deficiencias que tiene actualmente cada uno de los módulos. Esto se hace con el fin de recopilar información para la creación del prototipo y erradicar todos los errores en el proceso actual de los módulos.

Preguntas, Objetivos y Análisis por Módulo

Módulo de cálculo de planilla.

1. ¿Con qué frecuencia se presentan errores en el cálculo de la planilla?

Figura 3:

Resultados del módulo de planilla

14 respuestas



Fuente: Elaboración propia.

Objetivo. Evaluar la eficacia y precisión del proceso actual del cálculo de planilla en la empresa.

Análisis. Con respecto a la frecuencia de los errores acerca del cálculo de planilla, los colaboradores indican que se presentan errores de cálculo ocasionalmente, debido a que en ocasiones se omiten aspectos tales como horas extras. Se detalla que es un problema que no es frecuente, pero en ocasiones se omiten detalles, debido a que no siempre se lleva un buen manejo de los datos.

2. ¿Considera que la falta de un sistema ágil para el cálculo de planilla ha afectado la precisión en el pago de los empleados?

Figura 4:

Resultados del módulo de planilla.

14 respuestas



Fuente: Elaboración propia, 2024.

Objetivo. Evaluar la opinión de los empleados sobre un sistema ágil de cálculo de planilla y, de esta manera, reducir los errores.

Análisis. Debido a la falta de un sistema ágil para el cálculo de planilla, se ha afectado la precisión del pago de los colaboradores, muchas veces estos errores son mínimos y se presentan ocasionalmente, pero al implementar un sistema ágil se tendrá un mejor manejo de los datos y se podrá erradicar este tipo de inconvenientes.

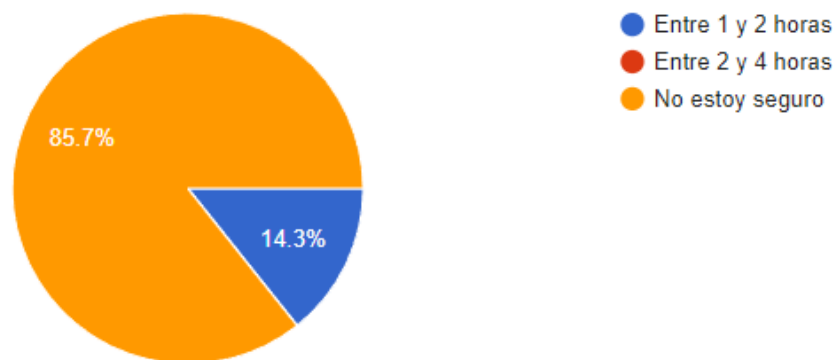
Módulo para calcular aguinaldo.

1. ¿Cuánto tiempo toma el proceso actual de pago del aguinaldo?

Figura 5:

Resultados del módulo de aguinaldo.

14 respuestas



Fuente: Elaboración propia, 2024.

Objetivo. Verificar el tiempo que toma el proceso del cálculo de aguinaldo para lograr una optimización.

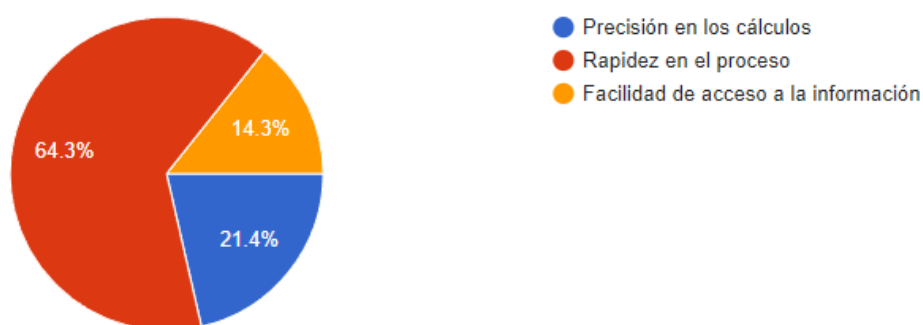
Análisis. Respecto al tiempo que toma el proceso actual del cálculo de aguinaldo, se habló con el encargado de recursos humanos y afirma que el proceso tarda entre dos y cuatro horas. Aunque se lleva un registro detallado de montos que se necesitan para el cálculo de aguinaldo, se afirma que se pierde mucho tiempo en ver los registros, ya que todo está en Microsoft Excel y muchas veces se tiene que validar de manera minuciosa los datos para poder brindar un cálculo óptimo del aguinaldo de cada colaborador de la farmacia. Las respuestas en naranja son de los colaboradores que desconocen acerca del tiempo del cálculo.

2. ¿Cuál de los siguientes aspectos del proceso de pago del aguinaldo considera que necesita mayor mejora?

Figura 6:

Resultados del módulo de aguinaldo.

14 respuestas



Fuente: Elaboración propia, 2024.

Objetivo. Validar los aspectos a mejorar en el cálculo de aguinaldo tomando en cuenta la opinión de los empleados para brindar un mejor resultado en el desarrollo del prototipo.

Análisis. Los aspectos para mejorar dentro del cálculo de aguinaldo son la precisión en los cálculos, en ocasiones no se realiza el cálculo de forma correcta, debido a errores humanos, ya que los cálculos se realizan mediante fórmulas de Excel, aunque todas las fórmulas están elaboradas según la ley costarricense, el error humano causa ineficiencia en el cálculo. Adicional a esto, otro aspecto a mejorar es la rapidez y facilitar el acceso de la información, debido a que muchas veces se pierde mucho tiempo intentando encontrar registros que son necesarios para realizar el cálculo de aguinaldo de los colaboradores.

Módulo para el cálculo de vacaciones.

1. En promedio, ¿cuántos días de vacaciones se pierden por empleado al año?

Figura 7:

Resultados del módulo de Vacaciones.

14 respuestas



Fuente: Elaboración propia, 2024.

Objetivo. Ayudar a determinar la pérdida de los días de vacaciones debido a una mala gestión y mejorar la eficiencia del módulo.

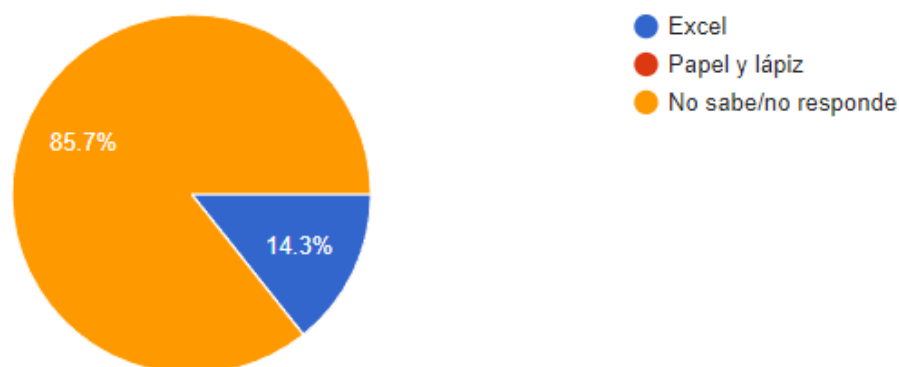
Análisis. Con base en los días de vacaciones perdidos, se estima que son entre uno y tres días al año, esto sucede por el mal manejo de los datos o descuido del encargado de recursos humanos. Pese a que se cumplen las leyes del Ministerio de Trabajo brindando un día de vacaciones por cada mes laborado, los errores consisten en los registros de los días. Por lo tanto, se enfatizó en la importancia de un sistema ágil que calcule los días de manera automática y que lleve un mejor control de los días disfrutados por parte de los colaboradores, de esta manera, se evitan pérdidas para los colaboradores y la farmacia.

2. ¿Qué método o herramienta utiliza actualmente la empresa para llevar el registro de los días de vacaciones de los colaboradores?

Figura 8:

Resultados del módulo de vacaciones.

14 respuestas



Fuente: Elaboración propia, 2024.

Objetivo. Conocer los métodos y herramientas mediante las cuales se lleva un registro de los días de vacaciones.

Análisis. Al respecto de las herramientas utilizadas para llevar un registro de los días de vacaciones de los colaboradores, estos se llevan en Microsoft Excel y papel y lápiz, para registrar las vacaciones primeramente se llevan en papel y lápiz, luego se pasan a Excel para aplicar las fórmulas. Esto causa que en ocasiones se omitan datos causando la inconformidad de los colaboradores de la farmacia, en este caso, se tomaron las respuestas de las personas encargadas de llevar el registro, ya que las respuestas naranja son de los colaboradores que desconocen del tema.

Módulo para el cálculo de horas extras.

1. ¿Con qué frecuencia realiza horas extras en su trabajo?

Figura 9:

Resultados del módulo de horas extras.

14 respuestas



Fuente: Elaboración propia, 2024.

Objetivo. Conocer la frecuencia con la que los empleados realizan las horas extras.

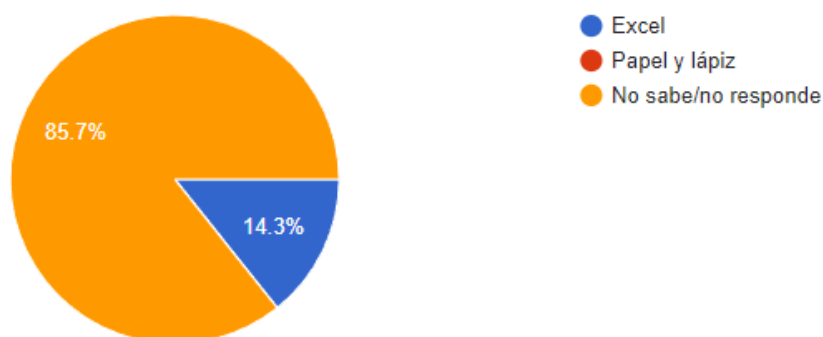
Análisis. La frecuencia con la que se realizan las horas extras en la farmacia es ocasionalmente, debido a que no es una organización en la cual se puedan realizar muchas horas extras, pero sí hay ocasiones en las cuales los colaboradores pueden hacerlo.

2. ¿Qué método o herramienta utiliza actualmente la empresa para llevar el registro de las horas extras?

Figura 10:

Resultados del módulo de horas extras.

14 respuestas



Fuente: Elaboración propia, 2024.

Objetivo. Validar herramientas que se utilizan en la actualidad en la empresa para el cálculo de horas extras

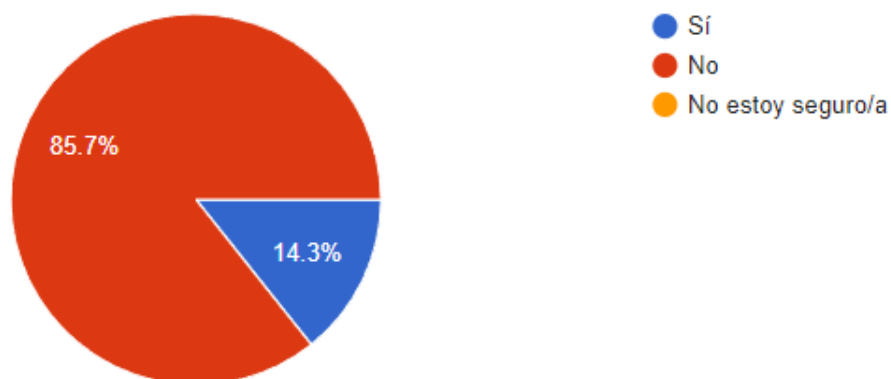
Análisis. La herramienta que se utiliza para el cálculo de horas extras es Microsoft Excel, el encargado de recursos humanos indicó que las horas extras están reguladas según el Ministerio de Trabajo, ya que se aplica la hora diurna y nocturna cada una con un costo diferente dependiendo de la hora en la cual se labore. En este caso, se tomaron las respuestas de las personas encargadas de llevar el registro, ya que las respuestas naranjas de los colaboradores desconocen del tema.

3. ¿Hay un protocolo para corregir errores en el cálculo de horas extras?

Figura 11:

Resultados del módulo de horas extras.

14 respuestas



Fuente: Elaboración propia, 2024.

Objetivo. Validar si la empresa tiene algún protocolo en caso de cometer errores en el registro de horas extras.

Análisis. Al respecto de si existe un protocolo para corregir errores en horas extras, no existe uno como tal. El encargado lo que hace es que agrega las horas extras en el siguiente pago del colaborador, ya que la planilla al ser aprobada no se puede modificar.

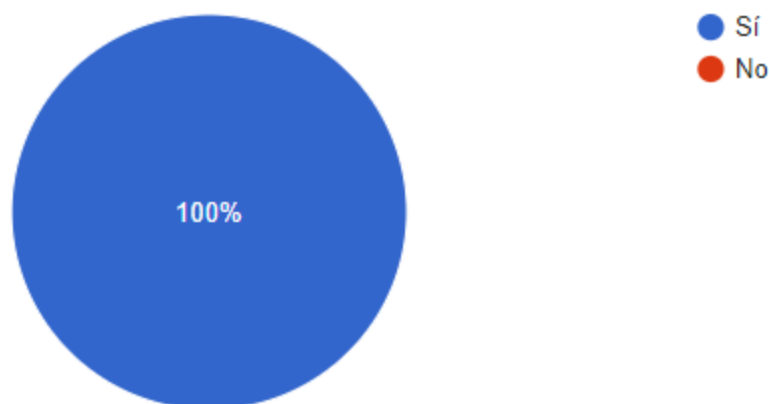
Módulo de control de permisos del colaborador.

1. ¿Ha experimentado problemas con el registro o la deducción correspondiente en la remuneración de sus permisos laborales en la empresa?

Figura 12:

Resultados del módulo de control de permisos.

14 respuestas



Fuente: Elaboración propia, 2024.

Objetivo. Conocer si la empresa tiene problemas en los permisos y solventar posibles errores en esta área.

Análisis. En cuanto a si ha experimentado problemas con el registro o la deducción correspondiente en la remuneración de sus permisos laborales en la empresa, los colaboradores indican que sí, generalmente por que en ocasiones los permisos con goce salarial no los aplican de forma correcta. Aunque no es algo que sea muy frecuente, es importante erradicar este problema para evitar inconvenientes con los colaboradores de la farmacia.

2. ¿Cuánto tiempo tarda actualmente el proceso de solicitud y aprobación de permisos laborales?

Figura 13:

Resultados del módulo de control de permisos.

14 respuestas



Fuente: Elaboración propia, 2024.

Objetivo. Validar el tiempo que tarda el proceso de solicitud y aprobación de los permisos laborales, además de la comunicación de la empresa.

Análisis. Respecto al tiempo que tarda actualmente el proceso de solicitud y aprobación de permisos laborales, se indica que es entre 1 y 3 días, a pesar de que no es un mal tiempo de respuesta a la hora de implementar el prototipo, se busca que mejore y que se lleve el registro de manera ordenada. Además, los colaboradores se ahorran hablar con los encargados de recursos humanos, ya que lo podrán hacer vía sistema.

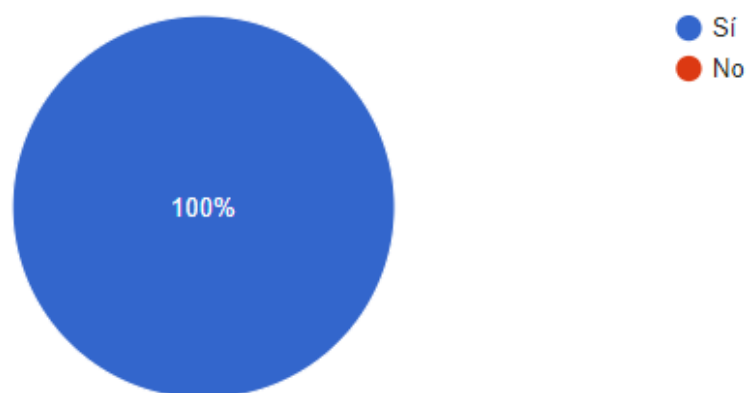
Módulo de liquidación.

1. ¿Ha experimentado problemas con los cálculos de liquidación de empleados que hayan resultado en errores en los pagos finales?

Figura 14:

Resultados del módulo de liquidación.

14 respuestas



Fuente: Elaboración propia, 2024.

Objetivo. Conocer si se han experimentado muchos problemas en el cálculo de planilla mediante las respuestas de los colaboradores.

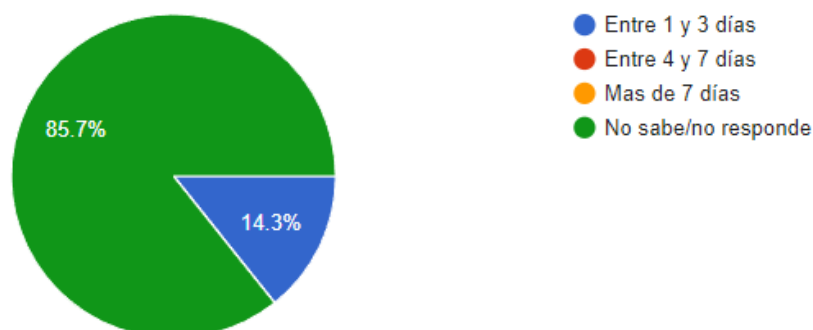
Análisis. Al respecto sobre si se ha experimentado problemas con los cálculos de liquidación de empleados que hayan resultado en errores en los pagos finales, se afirma que sí, por la dificultad que se tiene para acceder a la información; debido a que para este módulo se deben tomar en cuenta muchos aspectos como las vacaciones que llevan hasta el momento, aguinaldo entre otras, y los datos al no estar registrados en sistema dificultan el acceso causando errores de cálculo.

2. ¿Cuánto tiempo en promedio toma completar el proceso completo de liquidación de un empleado, desde la recolección de datos hasta la emisión del pago?

Figura 15:

Resultados del módulo de liquidación.

14 respuestas



Fuente: Elaboración propia, 2024.

Objetivo. Evaluar el tiempo que tarda el proceso de liquidación de los colaboradores y mejorar la forma en la que se está haciendo.

Análisis. En cuanto al tiempo en promedio que toma completar el proceso completo de liquidación de un empleado, desde la recolección de datos hasta la emisión del pago, es entre 4 y 7 días, debido a dificultad que se tiene de acceder a toda la información, ya que la herramienta que se utilizan para llevar el registro de datos en la actualidad en la empresa es Microsoft Excel. Esto dificulta el acceso a la información, ya que, como se menciona anteriormente, para realizar el cálculo de manera correcta de la liquidación, se deben tomar en cuenta muchos aspectos. En este caso se toman las respuestas en azul, debido a que son los encargados de realizar el proceso.

Módulo de registro de incapacidades.

1. ¿Con qué frecuencia ha experimentado problemas en el registro de incapacidades en la empresa?

Figura 16:

Resultados del módulo de incapacidades.

14 respuestas



Fuente: Elaboración propia, 2024.

Objetivo. Evaluar la frecuencia con la que los empleados experimentan problemas en el registro de incapacidades en la empresa debido a la complejidad de este módulo.

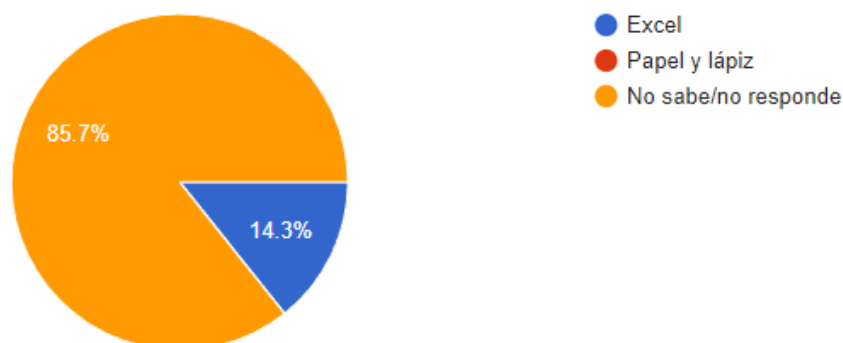
Análisis. Respecto a la frecuencia con que ha experimentado problemas en el registro de incapacidades en la empresa, es ocasionalmente. Esto debido a la complejidad del módulo, aunque no es algo frecuente, sí se han experimentado errores causando inconformidad en los colaboradores de la farmacia.

2. ¿Qué método o herramienta utiliza actualmente la empresa para llevar el registro de incapacidades?

Figura 17:

Resultados del módulo de incapacidades.

14 respuestas



Fuente: Elaboración propia, 2024.

Objetivo. Conocer el método o herramienta que se esté utilizando para el cálculo de las incapacidades para, de esta manera, mejorar en caso de que existan inconvenientes.

Análisis. El método que se utiliza para el registro de las incapacidades es mixto entre papel y lápiz y Microsoft Excel, ya que primero se registra en papel y luego se pasa a Excel. Esto causa que muchas veces los registros no se hagan de la manera correcta y omiten datos o días en los que el empleado estuvo incapacitado causando errores a la hora del pago correspondiente.

3. ¿Estaría de acuerdo con que se utilice un sistema automatizado para el cálculo de planilla que mejore la precisión y claridad en los montos a pagar?

Figura 18:

Resultados del módulo de liquidación.

14 respuestas



Fuente: Elaboración propia, 2024.

Objetivo. Conocer la opinión de los empleados sobre la implementación de un sistema automatizado debido a que este tendrá mejores especificaciones de los pagos realizados.

Análisis. En cuanto a si están de acuerdo con que se utilice un sistema automatizado para el cálculo de planilla que mejore la precisión y claridad en los montos a pagar, se obtuvo una total aceptación, ya que al implementar se eliminarán los errores de cálculos de cada uno de los módulos. Además, los colaboradores tendrán mucha más información e interacción con el sistema, ya que todo estará reflejado en el sistema, aspectos como incapacidades, vacaciones, permisos y planilla permitiendo mejorar la comunicación entre el empleado, la jefatura y el encargado de recursos humanos.

Observación

La observación fue enfocada en el proceso del cálculo de planilla. El mayor problema para realizar este cálculo se basa en la información, debido a que muchas veces se pierden las horas extras y causa inconformidad en los colaboradores. Cabe mencionar que todos los procesos se realizan con base en la ley costarricense y las fórmulas están de manera correcta, el problema en la

mayoría de los errores que se tiene es la pérdida de información y en ocasiones error de cálculo de los encargados de los procesos.

Todos estos errores se solucionarán con la creación del prototipo, debido a que los datos estarán seguros y tendrá registros confiables. Además, facilitará el cálculo de cada uno de los módulos y mejorará la comunicación a la hora de solicitar permisos o vacaciones, para el colaborador todo estará más detallado evitando inconvenientes y malentendidos.

CAPÍTULO V: CONCLUSIONES Y RECOMENDACIONES

Conclusiones

Tras un análisis exhaustivo de los procesos de recursos humanos en la farmacia D'Angelo, ubicada en Cartago, se identificó una serie de problemas relacionados a los cálculos manuales utilizados en la gestión del personal. Estos errores incluyen cálculos incorrectos de planilla, aguinaldo, incapacidades, liquidación, permisos, vacaciones y horas extras.

Los objetivos específicos establecidos para el desarrollo del prototipo funcional se han cumplido de manera exitosa, contribuyendo significativamente a la optimización y eficiencia de los cálculos de recursos humanos para la farmacia D'Angelo. A continuación, se detallan los objetivos específicos y cómo se han logrado:

- Analizar los requerimientos, se evaluaron las necesidades del prototipo mediante encuesta y observaciones para poder cumplir con las necesidades, donde se tomó en cuenta la automatización de recursos humanos y una gestión centralizada de datos mediante la creación de una base de datos centralizada y segura.
- Diseñar la estructura del prototipo: se diseñó un prototipo para la gestión de recursos humanos basado en los requerimientos. Se desarrolló el sistema de manera meticulosa para la eficiencia, precisión y usabilidad del sistema, los módulos fueron desarrollados con base en las necesidades de la farmacia.
- Desarrollar la programación del prototipo: se programó la implementación del prototipo funcional en la farmacia D'Angelo. Este proceso ha involucrado la implementación de módulos clave, la integración de funcionalidades esenciales y la optimización del rendimiento y la usabilidad del sistema.
- Realizar pruebas para el funcionamiento óptimo del prototipo final: se realizaron pruebas exhaustivas del sistema cumpliendo con todos los requerimientos y expectativas. Dentro de las pruebas se validó la precisión, estabilidad, usabilidad y seguridad del sistema asegurando que esté listo para su implementación en la farmacia.

Estos logros permitieron la automatización del sistema resolviendo los problemas que tenían en recursos humanos. Además, aportó valor a la empresa en términos de organización,

eficiencia, precisión y satisfacción del personal, ya que ahora ellos podrán visualizar información que anteriormente era desconocida.

Recomendaciones

El sistema diseñado para Farmacia D'Angelo ha sido desarrollado con gran atención a las necesidades de gestión de recursos humanos y procesos administrativos, asegurando que todas las funciones clave estén cubiertas eficientemente. Para garantizar una transición exitosa, se recomienda a Jefatura y Recursos Humanos que primero utilicen el sistema en un entorno de pruebas. Esta fase permitirá que los usuarios se familiaricen con el sistema y validen la precisión de los cálculos automatizados, reduciendo el riesgo de errores durante la implementación en un entorno real.

Además, este periodo de pruebas ofrecerá una oportunidad para identificar y corregir posibles problemas antes de la implementación definitiva, mientras se construye confianza en el personal. La formación y capacitación durante esta etapa garantizarán que los usuarios puedan operar el sistema de manera efectiva desde el primer día de su uso oficial, asegurando una transición fluida y sin contratiempos hacia las operaciones diarias en la Farmacia D'Angelo.

Se recomienda a la jefatura implementar, en un plazo menor a 15 días, el llenado de información. Este proceso debe incluir todos los datos desde el 1 de diciembre de 2023 hasta la fecha actual, tomando en cuenta la información que pueda variar los resultados. El propósito de esta acción es garantizar que el sistema cuente con toda la información necesaria para realizar los cálculos de manera precisa y eficiente, esto garantiza que a la hora de migrar al nuevo sistema todo esté correcto.

Se recomienda a la jefatura implementar en el módulo de registro de marcas datos biométricos, específicamente huellas dactilares, en un plazo de 15 días. Esta implementación tiene como objetivo automatizar y mejorar la precisión y seguridad del registro de asistencia en el módulo de gestión de recursos humanos, lo cual permite una mayor eficiencia operativa del sistema. Al adoptar esta tecnología, se espera alcanzar una mayor eficiencia operativa del sistema y mejorar la experiencia de usuario, además, esta tecnología permitirá que existan errores de marcas.

CAPÍTULO VI: PROPUESTA

Requerimientos Funcionales

En las tablas presentadas en este apartado se detallan los requerimientos funcionales del sistema desarrollado para la farmacia.

Tabla 4:

Requerimiento 1.

Identificador Del Requerimiento Funcional	RF-01
Nombre	Cálculo de planilla
Características	El usuario podrá realizar el cálculo de planilla de todos los colaboradores de la farmacia
Descripción	<p>Se requiere un programa que permita realizar el cálculo de planilla de todos los colaboradores tomando en cuenta.</p> <ul style="list-style-type: none"> • Monto de renta con base en los salarios, si es mayor a ₡929,000 y menor o igual a ₡1,363,000 se le resta el 10%, si es mayor a 1,363,000 pero menor o igual a ₡2,392,000 se le resta el 15%, si es mayor a ₡2,392,000 pero menor o igual a ₡4,783,000 se le resta el 20%, si es mayor a ₡4,783,000 en adelante se le resta el 25% • Deducciones se resta lo que es el seguro de CCSS que es el 10.67% • Incapacidades • Total de salario • Cédula del colaborador <p>Todo esto aplicado según la ley costarricense.</p>
Prioridad	Alta

Fuente: Elaboración propia, 2024.

Tabla 5:
Requerimiento 2.

Identificador Del Requerimiento Funcional	RF-02
Nombre	Cálculo de aguinaldo
Características	El usuario podrá realizar el cálculo de aguinaldo de los colaboradores.
Descripción	<p>El programa debe permitir ingresar los datos necesarios para calcular el aguinaldo de un colaborador, incluyendo su salario base, las horas extras realizadas y el período de cálculo, el programa deberá calcular el salario diario promedio considerando las horas extras y luego multiplicarlo por los días correspondientes al aguinaldo, el sistema deberá mostrar:</p> <ul style="list-style-type: none"> • Total, del salario • Monto del aguinaldo • Cédula del colaborador • Nombre del colaborador <p>Todos los cálculos con base en la ley costarricense.</p>
Prioridad	Alta

Fuente: Elaboración propia, 2024.

Tabla 6:
Requerimiento 3.

Identificador Del Requerimiento Funcional	RF-03
Nombre	Cálculo de vacaciones
Características	El usuario podrá realizar el cálculo de los días de vacaciones.
Descripción	<p>El usuario deberá poder ingresar para ver los días de vacaciones con los que cuenta y en caso de tener días disponibles, el empleado realizará la solicitud vía sistema a la jefatura inmediata, cuando ésta apruebe o repruebe y este le notificará al empleado también vía sistema. El sistema actuará mediante la ley costarricense que indica que al mes tiene derecho a 1 día de vacaciones. El sistema hará este proceso de manera automática tomando en cuenta la fecha de ingreso del colaborador, el sistema deberá mostrar:</p> <ul style="list-style-type: none"> • Fecha de inicio • Fecha de fin • Cantidad de días • Nombre del colaborador • Estado de las vacaciones que mostrará si se aprueban o rechazan
Prioridad	Alta

Fuente: Elaboración propia, 2024.

Tabla 7:
Requerimiento 4.

Identificador Del Requerimiento Funcional	RF-04
Nombre	Cálculo de horas extras
Características	El usuario podrá realizar el cálculo de horas extras.
Descripción	<p>El sistema permitirá realizar el cálculo de las horas extras de los colaboradores tomando en cuenta el tipo de hora extra, en el caso de ser hora diurna, se multiplicará 1.5 por lo que él gana por hora. En caso de ser nocturna, se realizará la multiplicación por 2.5 por el monto que él gana por hora, el sistema deberá mostrar lo siguiente:</p> <ul style="list-style-type: none"> • Cantidad de horas • Tipo de hora, hora diurna o hora nocturna. • Monto total de las horas laboradas • Colaborador • todo esto con base en la ley costarricense.
Prioridad	Alta

Fuente: Elaboración propia, 2024.

Tabla 8:
Requerimiento 5.

Identificador Del Requerimiento Funcional	RF-05
Nombre	Control de permisos del colaborador
Características	El usuario podrá llevar el control de los permisos y aceptar o rechazar permisos.
Descripción	<p>El sistema llevará un control de los permisos realizados por los colaboradores, además, se encargará de las solicitudes correspondientes a los permisos solicitados por los colaboradores, además los empleados podrán realizar la solicitud vía sistema y una vez tomada la decisión se le notificará vía sistema, deberá mostrar los siguientes campos:</p> <ul style="list-style-type: none"> • Fecha de solicitud • Descripción para validar el tipo de permiso • Estado que mostrará si se aprueba o rechaza el permiso • Colaborador
Prioridad	Alta

Fuente: Elaboración propia, 2024.

Tabla 9:
Requerimiento 6.

Identificador Del Requerimiento Funcional	RF-06
Nombre	Cálculo de liquidación
Características	El usuario podrá realizar el cálculo de liquidación de los colaboradores.
Descripción	<p>El sistema podrá realizar el cálculo de liquidación de los colaboradores, en caso de que no aplique, no tendrá remuneración económica, pero en caso de aplicar deberá mostrar lo siguiente:</p> <ul style="list-style-type: none"> • Monto cesantía este va de 3 a menos de 6 meses cuenta con 7 días de salario, 6 meses a menos de 1 año cuenta con 14 días de salario, 1 año cuenta con 19.5 días por año laborado, 2 años cuenta con 20 días por año laborado o fracción superior a seis meses, 3 años cuenta con 20.5 días por año laborado o fracción superior a seis meses, 4 años cuenta con 21 días por año laborado o fracción superior a seis meses, 5 años cuenta con 21.24 días por año laborado o fracción superior a seis meses, 6 años cuenta con 21.5 días por año laborado o fracción superior a seis meses, 8 años cuenta con 22 días por año laborado o fracción superior a seis meses. • Monto preaviso, si un colaborador tiene de 0 a 3 meses no contará con monto de preaviso, si tiene más de 3 meses y menos de 6 recibe una semana del monto preaviso, si tiene más de 6 meses y menos de 1 año recibe 15 días de salario, si tiene más de 1 año recibe 1 mes completo de monto preaviso, estos montos es según la antigüedad. • Monto vacaciones • Monto aguinaldo • Total, liquidación <p>Todo esto se aplicará según la ley costarricense.</p>
Prioridad	Alta

Fuente: Elaboración propia, 2024.

Tabla 10:
Requerimiento 7.

Identificador Del Requerimiento Funcional	RF-07
Nombre	Cálculo de incapacidades
Características	El usuario podrá agregar y consultar las incapacidades de los colaboradores.
Descripción	<p>El sistema podrá generar el cálculo de las incapacidades, el sistema en caso de agregar una incapacidad, se asignará el tipo de incapacidad, así como la fecha de inicio y la fecha de fin, dependiendo del tipo de incapacidad se mostrará reflejado en el pago del colaborador, el sistema deberá mostrar lo siguiente:</p> <ul style="list-style-type: none"> • Fecha de inicio • Fecha fin • Tipo incapacidad por la CCSS y INS • Monto de incapacidad • Colaborador <p>Se aplicarán las incapacidades con base en la ley costarricense.</p>
Prioridad	Alta

Fuente: Elaboración propia, 2024.

Tabla 11:*Requerimiento 8.*

Identificador Del Requerimiento Funcional	RF-08
Nombre	Control de asistencia
Características	El usuario podrá ingresar la hora de entrada y salida.
Descripción	<p>Se requiere que el sistema, a la hora que el colaborador ingresa, le permita ingresar la hora de entrada y hora de salida para poder llevar el control de las horas.</p> <ul style="list-style-type: none"> • El sistema deberá validar que la hora de entrada sea anterior a la hora de salida. • El sistema deberá validar que la hora de entrada y salida no sean anteriores a la fecha actual.
Prioridad	Alta

Fuente: Elaboración propia, 2024.

Tabla 12:*Requerimiento 9.*

Identificador Del Requerimiento Funcional	RF-09
Nombre	Mantenimiento
Características	El usuario podrá realizar ajustes a todos los módulos del sistema.
Descripción	Se requiere que el sistema cuenta con un CRUD, que permite crear, leer, modificar y eliminar elementos en todos los módulos, esto con el fin de poder insertar registros nuevos, visualizar los existentes y en el caso de una equivocación en algún registro poder modificarlos o eliminarlos.
Prioridad	Alta

Fuente: Elaboración propia, 2024.

Tabla 13:*Requerimiento 10.*

Identificador Del Requerimiento Funcional	RF-10
Nombre	Consultas
Características	El usuario podrá realizar ajustes a todos los módulos del sistema.
Descripción	Se requiere que el sistema cuente con consultas que permita facilitar la información de cada módulo permitiendo ahorrar tiempo a la hora de buscar algún registro.
Prioridad	Alta

Fuente: Elaboración propia, 2024.

Tabla 14:*Requerimiento 11.*

Identificador Del Requerimiento Funcional	RF-11
Nombre	Reportes
Características	El usuario podrá realizar reportes en el sistema
Descripción	Se requiere que el sistema tenga la capacidad de realizar reportes en cada módulo, que le permita al usuario exportar la información en PDF o Excel para tener un registro físico si así lo desea.
Prioridad	Alta

Fuente: Elaboración propia, 2024.

Tabla 15:
Requerimiento 12.

Identificador Del Requerimiento Funcional	RF-12
Nombre	Seguridad
Características	El usuario podrá realizar un login dentro del sistema
Descripción	Se requiere que el sistema tenga un login para que no puedan ingresar personas externas, solamente podrán ingresar los empleados que estén registrados. Esto ayudará a mantener seguro los datos de la empresa, solo el usuario que sea administrador podrá modificar el programa, los demás solo podrán hacer solicitudes.
Prioridad	Alta

Fuente: Elaboración propia, 2024.

Tabla 16:*Matriz de requerimientos.*

Indicador	Módulo	Requerimientos
1	Calcular planilla	RF-01, RF-04, RF-07, RF-08, RF-09, RF-10, RF-11
2	Cálculo de Aguinaldo	RF-01, RF-02, RF-04, RF-07, RF-08, RF-09, RF-10, RF-11
3	Cálculo Vacaciones	RF-03, RF-09, RF-10, RF-11
4	Cálculo de horas extras	RF-01, RF-04, RF-08, RF-09, RF-10, RF-11
5	Control de Permisos del Colaborador.	RF-01, RF-05, RF-09, RF-10, RF-11
6	Cálculo de Liquidación	RF-02, RF-03, RF-04, RF-06, RF-09, RF-10, RF-11
7	Control de asistencia	RF-08, RF-09, RF-10, RF-11, RF-12
8	Mantenimientos	RF-09
9	Consultas	RF-10
10	Reportes	RF-11
11	Seguridad	RF-12

Fuente: Elaboración propia, 2024.

Análisis

Análisis del Software

El análisis de *software* es una parte crucial dentro del desarrollo del sistema, en este se describe detalladamente la funcionalidad de los módulos que componen el sistema de recursos humanos. También se describe el propósito y cómo estos interactúan entre ellos para cumplir los objetivos del proyecto. El sistema está diseñado para gestionar de manera eficiente los datos y una mejor precisión en los cálculos.

Módulo de calcular planilla.

Este módulo se encarga de calcular la remuneración de los colaboradores, considerando las deducciones aplicables en Costa Rica, así como el pago correspondiente por las horas extras trabajadas. el cálculo de planilla se generará para todos los colaboradores de forma automática y en caso de alguna inconsistencia, contará con la opción de recalculiar la planilla. una vez la planilla esté emitida el colaborador podrá tener acceso a esta y ver de forma detallada cada uno los rubros de planilla.

Módulo de calcular aguinaldo.

Este módulo será administrado con el propósito de efectuar el pago anual del aguinaldo a los colaboradores. Este cálculo se realizará considerando la acumulación de los doce salarios mensuales y su distribución en los doce meses del año trabajados. El cálculo se generará de manera masiva mejorando los tiempos, una vez sea emitido el aguinaldo, el usuario podrá ingresar al sistema y el aguinaldo correspondiente.

Módulo de cálculo de vacaciones.

Este módulo se encargará de gestionar las solicitudes que corresponden a las vacaciones tomando en cuenta los días disponibles del colaborador, deberá ingresar al sistema y realizar la solicitud de los días de vacaciones. El empleado realizará la solicitud vía sistema a la jefatura

inmediata, cuando esta apruebe o repruebe, se le notificará vía sistema a RR.HH. y este le notificará al empleado también vía sistema.

Módulo de calcular horas extras.

Este módulo se encargará de llevar el control de las horas extras, el empleado realizará la solicitud vía sistema a la jefatura inmediata; cuando esta apruebe o repruebe, se le notificará vía sistema a RR.HH. y este le notificará al empleado también vía sistema. Una vez las horas extras sean aprobadas serán validadas con base en las marcas del colaborador debido a que cuentan con un horario que deben cumplir. Una vez terminada la hora de salida, el colaborador realiza las horas extras, las cuales se agregan de forma automática al sistema dependiendo de la cantidad de horas que realice.

Módulo de control de permisos del colaborador.

Este módulo se encargará de las solicitudes correspondientes a los permisos solicitados por los empleados, ya sean con goce o sin goce salarial, para realizar la solicitud el empleado deberá ingresar al sistema y seleccionar el tipo de permiso que desea solicitar. Al realizar la solicitud esta llega al administrador, el cual aprueba o reprueba las solicitudes, una vez aprobado el permiso, el colaborador podrá hacer uso del permiso, en caso de ser denegado no podrá gozar del permiso.

Módulo de calcular liquidación.

La incorporación de un módulo de cálculo de liquidación optimizará la precisión y eficiencia en el proceso de finalización laboral, asegurando pagos exactos y evitando posibles demoras. Se seleccionará el tipo de liquidación, debido a que es con o sin goce al salario, en caso de ser sin goce, el cálculo es equivalente a 0; en caso de ser con goce, se realizará el pago conforme a la ley costarricense. El empleado podrá ingresar al módulo y ver los montos exactos.

Módulo de mantenimiento.

El módulo de mantenimiento es esencial en el sistema, ya que se encarga de las operaciones fundamentales de gestión de datos en todos los módulos. Permite la eliminación de registros obsoletos o incorrectos, la inserción de nueva información, la modificación de datos existentes para corregir errores o actualizar información, y la actualización de la base de datos para adaptarse a nuevas necesidades que puedan darse en adelante.

Módulo de consultas.

Este módulo se encargará de facilitar la información de cada uno de los módulos, permitiendo filtrar la información importante o ayudando a encontrar solamente el registro deseado permitiendo una optimización de tiempo. Además, al permitir filtrar la información importante, ayuda a los usuarios a enfocarse en los datos relevantes para sus tareas.

Módulo de reportes.

Este módulo se encargará de generar la información proporcionada en diferentes tablas, permitiendo generar un PDF de la información. Adicional a esto, se podrá realizar reportes filtrando la información y obtener datos que ayuden a la administración a la toma de decisiones. Esta funcionalidad es de suma importancia, ya que permite optimizar el proceso de análisis y representación de datos.

Módulo de seguridad.

Este módulo se encargará de realizar la autenticación de contraseñas y definición de perfiles permitiendo tener una mayor seguridad, ya que, al realizar la definición de perfiles, se restringen los permisos, permitiendo que solo los administradores puedan aprobar solicitudes, agregar información, editar o eliminar. Los usuarios que no sean administradores podrán realizar solicitudes y ver los cálculos respectivos, pero no podrán alterar información, esto brinda una capa de seguridad al sistema.

Módulo de control de asistencia.

Este módulo es de suma importancia para el sistema, debido a que permite llevar un control del horario de los colaboradores. Además, permite asegurar que se cumplan los horarios establecidos, también permite tener un mejor control de horas extras. Este módulo es crucial para el cálculo de planilla y para garantizar el cumplimiento de las normativas laborales relacionadas con los horarios de trabajo.

Análisis del Hardware

El análisis del *hardware* es parte crucial para el desarrollo de cualquier sistema. En este análisis, se muestran los componentes requeridos para el desarrollo del prototipo. Cabe mencionar que, para la elaboración de este prototipo, la empresa no incurrió en gastos, ya que este fue desarrollado con fines universitarios. En la siguiente tabla, se detallan los componentes utilizados para el desarrollo del prototipo

Tabla 17:
Hardware requerido

Rubros	Características	Costos
Laptop	“Laptop MSI” con el sistema operativo Windows 11 pro-64 bits, procesador AMD Ryzen 7 7730U Memoria 16384MB RAM	₡ 0
Mouse	Model M991-RGB	₡ 0

Fuente: Elaboración propia, 2024.

Análisis Detallado de los Elementos de Telecomunicaciones

En este análisis se detallan los elementos necesarios para asegurar la conectividad y comunicación eficiente con el sistema. Estos componentes son cruciales para el funcionamiento del sistema y permitirán la transferencia de datos, así como la interacción entre los módulos y el usuario. Los componentes se detallan en la siguiente tabla.

Tabla 18:

Análisis de telecomunicaciones

Tipo de red	Internet	Equipo	Costos
Se necesita una conexión LAN para garantizar la conexión del equipo a internet entre el <i>router</i> y la laptop.	Se necesita un acceso a internet para permitir la conexión entre el sistema con una velocidad mínima de 25MB para establecer la conexión con la laptop y el servidor web.	<i>Router</i>	₪ 0

Fuente: Elaboración propia, 2024.

Descripción Detallada de las Herramientas Técnicas Utilizadas

Para el desarrollo del prototipo, se han implementado diversas herramientas técnicas que permiten la creación y funcionamiento eficiente del sistema. Dentro de estas herramientas se incluye lenguaje de programación, *frameworks* y sistema de gestión de bases de datos. Cabe mencionar que todo lo que se utiliza no incurre en ningún gasto por parte de la empresa, ya que todas las herramientas son gratis y permiten que el sistema sea eficiente.

La fusión de estas herramientas no solo se centra en la estética, sino también en la usabilidad, ya que resultará en una interfaz intuitiva y fácil de navegar mejorando la experiencia

del usuario y haciéndola amigable con este, lo cual permite que impacte de manera positiva en la empresa. Las herramientas utilizadas en la creación del sistema se detallan en la siguiente tabla.

Tabla 19:

Software requerido para la programación

Rubros	Características	Costos
Motor de base de datos	SQL Server 2022	€ 0
Entorno de desarrollo integrado (IDE)	Microsoft Visual Studio Community 2022	€ 0
Herramientas para el diseño de bases de datos	MySQL Workbench 8.0	€ 0

Fuente: Elaboración propia, 2024.

En el desarrollo del prototipo funcional, se utiliza una aplicación web basada en la tecnología de Microsoft. Se hace uso de Visual Studio 2022 como el entorno de desarrollo principal y se usa el lenguaje de programación C# con el .NET Framework; específicamente en su versión 4.8.1, que incluye ASP.NET. Esta fusión de herramientas ofrece una plataforma completa y sólida para el desarrollo de aplicaciones web.

Para complementar el uso de este lenguaje, se utiliza Microsoft SQL Server 2022 con una licencia estándar; lo que representa una estrategia sólida para asegurar funcionalidades esenciales en el almacenamiento y gestión de datos, debido a que esta plataforma ofrece una base confiable que respalda los requisitos fundamentales de la empresa en el manejo de la información. Adicionalmente, cuenta con capacidades avanzadas para la generación de informes y análisis, consolidando una solución integral y robusta para la construcción del prototipo funcional.

Para el diseño de las vistas del prototipo funcional, se implementa HTML con plantillas de Bootstrap a fin de generar una interfaz amigable con el usuario, complementando con CSS para darle los estilos a todo el prototipo. También se hace uso de JavaScript con *DataTable* para generar vistas más amigables cuando se deba dar mantenimientos. Esta fusión de herramientas genera una interfaz de usuario atractiva y fácil de usar en la aplicación web.

La elección de HTML brinda una estructura sólida para las páginas web y al complementarlo con Bootstrap, se asegura de tener un buen diseño mejorando la experiencia de usuario.

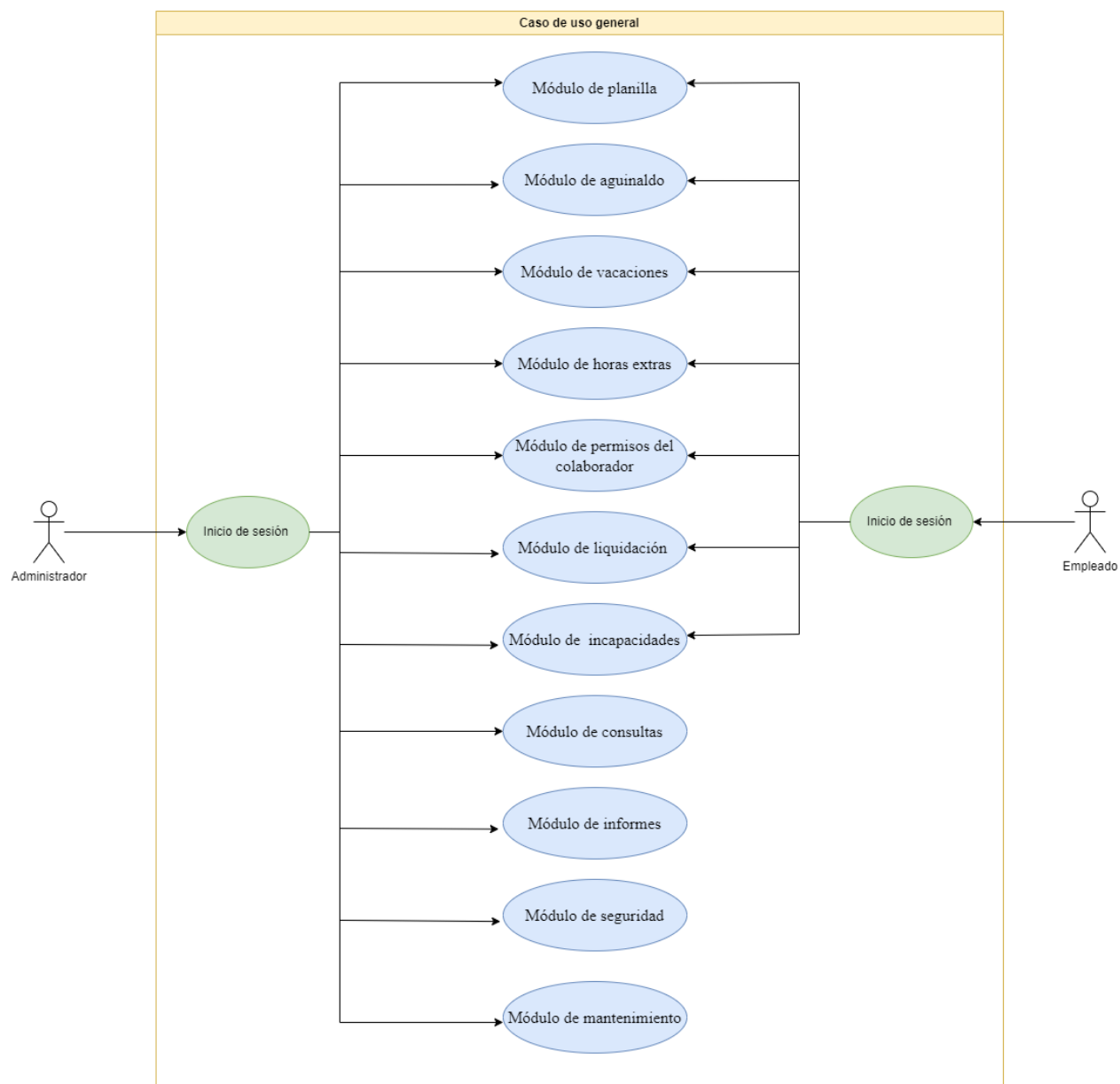
En cuanto al uso de los estilos, la implementación de CSS desempeñará un papel crucial para la personalización de cada una de las vistas del prototipo. Cabe mencionar que todas llevarán el mismo estilo a fin de hacerlo atractivo para el usuario. Además, al aplicar estilos cohesivos, se logrará una apariencia profesional y alineada con la identidad visual de la empresa.

La inclusión de JavaScript, en particular con el uso de la librería *DataTable*, potenciará las capacidades interactivas del prototipo. *DataTable* ofrece funcionalidades avanzadas para la presentación de datos tabulares, como ordenamiento, filtrado y paginación, lo que facilita la navegación y manipulación de grandes conjuntos de datos. Además, permite al usuario buscar de manera rápida algún registro debido a que cuenta con funciones para buscar.

Descripción Detallada del Conocimiento Básico que Debe Tener el Recurso Humano que Operará el Sistema

Para garantizar una operación eficiente del sistema, es fundamental que los usuarios tengan conocimiento básico de computación. Adicional a esto, deberá tener conocimiento acerca de recursos humanos, conocimiento básico en registro y mantenimiento de datos, comprender cómo registrar y actualizar información de empleados. Aunado a esto, deberá tener conocimiento acerca de cómo se aplican las leyes de Costa Rica, y deberá contar con conocimiento básico acerca de navegadores web, ya que el sistema está en este formato.

Figura 19:
Diagrama de casos de uso



Fuente: Elaboración propia, 2024.

Casos de uso

Tabla 20:

Caso de uso de planilla

Prototipo: Propuesta funcional para la gestión del recurso humano para la Farmacia D'Angelo ubicada en Cartago	
Número de caso de uso:	Cálculo de planilla
Fecha elaboración:	17/06/2024
Descripción caso de uso:	Este módulo permitirá realizar el pago de planilla para todos los colaboradores de manera automatizada tomando en cuenta las deducciones según la ley costarricense.
Autor caso de uso:	Deyvin Polanco Espinoza
Actores relacionados:	Administrador
Precondiciones:	El actor debe iniciar sesión para lograr el ingreso exitoso al módulo
Flujo básico del caso de uso	
<ol style="list-style-type: none"> 1. El caso de uso inicia cuando el actor decide ingresa al módulo de cálculo de planilla 2. El sistema mostrará el cálculo de planilla, al presionar el botón “Calcular planilla” (SF-01), si desea generar un reporte en PDF deberá presionar el botón de “Generar PDF” (SF-02), en caso de generar la planilla en Excel debería presionar “Generar Excel” (SF-03) 3. El sistema mostrará el botón de “Editar planilla” donde este permitirá recalculer la planilla en caso de encontrar inconsistencias en el cálculo (SF-04) 	
Subflujos	
SF-01	Una vez el usuario presiona el botón de “Calcular planilla”, se mostrará la información correspondiente al pago de planilla (FA-01) donde se mostrará el total de horas extras laboradas, las deducciones correspondientes, el monto de incapacidades, la fecha en la cual se emitió y la fecha fin y el monto total de la planilla.
SF-02	Una vez el usuario presiona el botón de “Generar PDF”, se mostrará la información correspondiente al módulo de planilla (FA-02)
SF-03	Una vez el usuario presiona el botón de “Generar Excel”, se mostrará la información correspondiente al módulo de planilla (FA-02)
SF-04	Una vez el usuario presiona el botón de “Editar planilla”, se vuelve a calcular la planilla en caso de tener algún error (FA-03)
Flujos alternos	
FA-01 Validar información	Se valida que existan empleados activos en la base de datos.
FA-02 Validar planilla	Se valida que exista información en el módulo de planilla para poder generar el reporte.
FA-03 Validar información	Se valida que exista ya un registro de planilla para poder recalcularla.
Requerimientos especiales	
No se posee requerimientos especiales.	
Postcondiciones	
Visualizar, descargar reporte en formato PDF y Excel.	

Fuente: Elaboración propia, 2024.

Tabla 21:
Caso de uso de cálculo de planilla

Prototipo: Propuesta funcional para la gestión del recurso humano para la Farmacia D'Angelo ubicada en Cartago	
Número de caso de uso:	Cálculo de planilla
Fecha elaboración:	17/06/2024
Descripción caso de uso:	Este módulo permitirá visualizar el pago de
Autor caso de uso:	Deyvin Polanco Espinoza
Actores relacionados:	Empleado
Precondiciones:	El actor debe iniciar sesión para lograr el ingreso exitoso al módulo
Flujo básico del caso de uso	
<ol style="list-style-type: none"> 1. El caso de uso inicia cuando el actor decide ingresar al módulo de cálculo de planilla 2. El sistema mostrará el módulo de planilla donde se mostrará el total de horas extras laboradas, las deducciones correspondientes, el monto de incapacidades, la fecha en la cual se emitió y la fecha fin y el monto total de la planilla. 3. El sistema mostrará un botón “Generar PDF” (SF-01), en caso de generar la planilla en Excel debería presionar “Generar Excel” (SF-02) 	
Subflujos	
SF-01	Una vez el usuario presiona el botón de “Generar PDF” , se mostrará la información correspondiente al pago de planilla (FA-01) donde se mostrará el total de horas extras laboradas, las deducciones correspondientes, el monto de incapacidades, la fecha en la cual se emitió y la fecha fin y el monto total de la planilla.
SF-02	Una vez el usuario presiona el botón de “Generar Excel” , se mostrará la información correspondiente al módulo de planilla (FA-01)
Flujos alternos	
FA-01 Validar planilla	Se valida que exista información en el módulo de planilla para poder generar el reporte.
Requerimientos especiales	
No se posee requerimientos especiales.	
Postcondiciones	
Visualizar, descargar reporte en formato PDF y Excel.	

Fuente: Elaboración propia, 2024.

Tabla 22:
Caso de uso de cálculo de aguinaldo

Prototipo: Propuesta funcional para la gestión del recurso humano para la Farmacia D'Angelo ubicada en Cartago	
Número de caso de uso:	Cálculo de Aguinaldo
Fecha elaboración:	17/06/2024
Descripción caso de uso:	Este módulo permitirá realizar el pago de aguinaldo para todos los colaboradores de manera automatizada.
Autor caso de uso:	Deyvin Polanco Espinoza
Actores relacionados:	Administrador
Precondiciones:	El actor debe iniciar sesión para lograr el ingreso exitoso al módulo
Flujo básico del caso de uso	
<ol style="list-style-type: none"> 1. El caso de uso inicia cuando el actor decide ingresa al módulo de cálculo de aguinaldo. 2. El sistema mostrará el cálculo de aguinaldo, al presionar el presionar el botón “Calcular aguinaldo” (SF-01), si desea generar un reporte en PDF deberá presionar el botón de “Generar PDF” (SF-02). 3. El sistema mostrará el botón de “Detalles” donde mostrará la información del aguinaldo del colaborador (SF-04). 4. El sistema mostrará el botón de “Eliminar” (SF-05). 	
Subflujos	
SF-01	Una vez el usuario presiona el botón de “Calcular aguinaldo”, se mostrará la información correspondiente al cálculo de aguinaldo (FA-01) donde se mostrará el monto total del aguinaldo.
SF-02	Una vez el usuario presiona el botón de “Generar PDF”, se mostrará la información correspondiente al módulo de planilla (FA-02).
SF-03	Una vez el usuario presiona el botón de “Detalles”, se muestra la información individual de cada colaborador. (FA-03).
SF-04	Una vez el usuario presiona el botón de “Eliminar”, se eliminará ese registro en caso de tener un cálculo incorrecto del cálculo (FA-04).
Flujos alternos	
FA-01 Validar información	Se valida que existan empleados activos en la base de datos.
FA-02 Validar planilla	Se valida que exista información en el módulo de planilla para poder generar el reporte.
FA-03 Validar información	Se valida que exista ya un registro de aguinaldo para mostrar los detalles de aguinaldo del colaborador.
FA-04 Validar aguinaldo	Se valida que exista registro para poder hacer la eliminación en caso de ser requerido.
Requerimientos especiales	
No se posee requerimientos especiales.	
Postcondiciones	
Visualizar, detallar, eliminar y descargar reporte en formato PDF y Excel.	

Fuente: Elaboración propia, 2024.

Tabla 23:
Caso de uso de cálculo de aguinaldo

Prototipo: Propuesta funcional para la gestión del recurso humano para la Farmacia D'Angelo ubicada en Cartago	
Número de caso de uso:	Cálculo de Aguinaldo
Fecha elaboración:	17/06/2024
Descripción caso de uso:	Este módulo permitirá realizar el pago de aguinaldo para todos los colaboradores de manera automatizada.
Autor caso de uso:	Deyvin Polanco Espinoza
Actores relacionados:	Empleado
Precondiciones:	El actor debe iniciar sesión para lograr el ingreso exitoso al módulo
Flujo básico del caso de uso	
<ol style="list-style-type: none"> 1. El caso de uso inicia cuando el actor decide ingresar al módulo de cálculo de aguinaldo. 2. El sistema mostrará la información relacionada a Aguinaldo la cual es el monto de aguinaldo. 3. El sistema contará con un botón el cual es "Generar PDF" (SF-01). 	
Subflujos	
SF-01	Una vez el usuario presiona el botón de "Generar PDF", se mostrará la información correspondiente al cálculo de aguinaldo (FA-01) donde se mostrará el monto total del aguinaldo.
Flujos alternos	
FA-01 Validar información	Se valida que existan datos dentro de este módulo
Requerimientos especiales	
No se posee requerimientos especiales.	
Postcondiciones	
Visualizar y descargar reporte en formato PDF.	

Fuente: Elaboración propia, 2024.

Tabla 24 :
Caso de uso de vacaciones

Prototipo: Propuesta funcional para la gestión del recurso humano para la Farmacia D'Angelo ubicada en Cartago	
Número de caso de uso:	Cálculo de vacaciones
Fecha elaboración:	17/06/2024
Descripción caso de uso:	Este módulo permitirá realizar el cálculo de vacaciones de los colaboradores
Autor caso de uso:	Deyvin Polanco Espinoza
Actores relacionados:	Administrador
Precondiciones:	El actor debe iniciar sesión para lograr el ingreso exitoso al módulo
Flujo básico del caso de uso	
<ol style="list-style-type: none"> 1. El caso de uso inicia cuando el actor decide ingresar al módulo de cálculo de aguinaldo. 2. El sistema mostrará las solicitudes de los colaboradores para que sean aprobadas las vacaciones, el sistema contará con un botón de “Aprobar” donde las vacaciones podrán ser aprobadas de manera exitosa. (SF-01). 3. El sistema contará con un botón el cual es “Generar PDF”. (SF-02). 4. El sistema mostrará el botón de “Detalles” donde mostrará la información de vacaciones del colaborador (SF-03). 	
Subflujos	
SF-01	Una vez el usuario presiona el botón de “Aprobar” (FA-01), el colaborador será notificado y podrá gozar de los días de vacaciones.
SF-02	Una vez el usuario presiona el botón de “Generar PDF”, se mostrará la información correspondiente al módulo de vacaciones
SF-03	Una vez el usuario presiona el botón de “Detalles”, se muestra la información individual de cada colaborador. (FA-03).
Flujos alternos	
FA-01 Validar solicitudes	Se valida que existan solicitudes de vacaciones.
FA-02 Validar vacaciones	Se valida que exista información en el módulo de vacaciones para poder generar el reporte.
FA-03 Validar información	Se valida que exista ya un registro de vacaciones para mostrar los detalles y generar el reporte.
Requerimientos especiales	
No se posee requerimientos especiales	
Postcondiciones	
Visualizar, Aprobar y descargar reporte en formato PDF.	

Fuente: Elaboración propia, 2024.

Tabla 25:
Caso de uso de vacaciones

Prototipo: Propuesta funcional para la gestión del recurso humano para la Farmacia D'Angelo ubicada en Cartago	
Número de caso de uso:	Cálculo de vacaciones
Fecha elaboración:	17/06/2024
Descripción caso de uso:	Este módulo permitirá realizar el cálculo de vacaciones de los colaboradores
Autor caso de uso:	Deyvin Polanco Espinoza
Actores relacionados:	Empleado
Precondiciones:	El actor debe iniciar sesión para lograr el ingreso exitoso al módulo
Flujo básico del caso de uso	
<ol style="list-style-type: none"> 1. El caso de uso inicia cuando el actor decide ingresar al módulo de vacaciones. 2. El sistema mostrará las vacaciones disponibles y las vacaciones consumidas, contará con un botón de “Solicitar vacaciones” (SF-01). 3. El sistema contará con un botón el cual es “Generar PDF”. (SF-02). 	
Subflujos	
SF-01	Una vez el usuario presiona el botón de solicitar, se mostrará un calendario para que seleccione las fechas en las que desea solicitar las vacaciones (FA-01) (FA-02)
SF-02	Una vez el usuario presiona el botón de “Generar PDF”, se mostrará la información correspondiente al módulo de vacaciones. (FA-03)
Flujos alternos	
FA-01 Validar días de vacaciones	Se valida que cuente con días disponibles para realizar la solicitud.
FA-02 Validar fechas	Se valida que las fechas ingresadas no sea mayor que la cantidad de días con las que dispone el colaborador.
FA-03 Validar información	Se valida que exista ya un registro de vacaciones para poder generar un PDF.
Requerimientos especiales	
No se posee requerimientos especiales	
Postcondiciones	
Visualizar, solicitar y descargar reporte en formato PDF.	

Fuente: Elaboración propia, 2024.

Tabla 26:*Caso de uso de horas extras*

Prototipo: Propuesta funcional para la gestión del recurso humano para la Farmacia D'Angelo ubicada en Cartago	
Número de caso de uso:	Cálculo de horas extras
Fecha elaboración:	17/06/2024
Descripción caso de uso:	Este módulo permitirá realizar el cálculo de horas extras para los colaboradores
Autor caso de uso:	Deyvin Polanco Espinoza
Actores relacionados:	Administrador
Precondiciones:	El actor debe iniciar sesión para lograr el ingreso exitoso al módulo
Flujo básico del caso de uso	
<ol style="list-style-type: none"> 1. El caso de uso inicia cuando el actor decide ingresar al módulo de solicitudes. 2. El sistema mostrará las solicitudes de los colaboradores para aprobar las horas extras, mostrará el botón de “Aprobar” (SF-01) se mostrará la fecha en la que se solicitaron las horas extras, el estado y el colaborador que las solicita. 3. El sistema contará con un botón el cual es “Generar PDF”. (SF-02) 4. Una vez las horas extras estén aprobadas y laboradas se mostrarán en el módulo de horas extras. (SF-03). 	
Subflujos	
SF-01	Una vez el usuario presiona el botón de “Aprobar”, se le notificará al usuario que está autorizado a realizar las horas extras (FA-01)
SF-02	Una vez el usuario presiona el botón de “Generar PDF”, se mostrará la información correspondiente al módulo de vacaciones. (FA-02)
SF-03	Una vez las horas extras estén aprobadas y laboradas por el colaborador se mostrarán en el módulo de horas extras.
Flujos alternos	
FA-01 Validar solicitudes	Se valida que exista una solicitud para realizar horas extras.
FA-02 Validar información	Se valida que exista ya un registro de horas extras para poder generar un PDF,
FA-03 Validar extras	Se valida que exista ya una solicitud previa y que se cumpliera con el horario para mostrar las horas correspondientes.
Requerimientos especiales	
No se posee requerimientos especiales,	
Postcondiciones	
Visualizar, solicitar y descargar reporte en formato PDF.	

Fuente: Elaboración propia, 2024.

Tabla 27:*Caso de uso de horas extras*

Prototipo: Propuesta funcional para la gestión del recurso humano para la Farmacia D'Angelo ubicada en Cartago	
Número de caso de uso:	Cálculo de horas extras
Fecha elaboración:	17/06/2024
Descripción caso de uso:	Este módulo permitirá solicitar horas extras
Autor caso de uso:	Deyvin Polanco Espinoza
Actores relacionados:	Empleado
Precondiciones:	El actor debe iniciar sesión para lograr el ingreso exitoso al módulo
Flujo básico del caso de uso	
<ol style="list-style-type: none"> 1. El caso de uso inicia cuando el actor decide ingresar al módulo de solicitudes personales. 2. El sistema mostrará contará con un botón de “Solicitar hora extra” (SF-01). 	
Subflujos	
SF-01	Una vez el usuario presiona el botón de “Solicitar hora extra”, se mostrará una vista donde le permitirá seleccionar la fecha de la solicitud y agregar un comentario acerca de la descripción de las horas extras que se realizarán (FA-01) (FA-02)
Flujos alternos	
FA-01 Validar información	Se valida que el usuario exista en la base de datos para poder hacer la solicitud.
FA-02 Validar solicitud	Se valida que la solicitud se realice de manera correcta.
Requerimientos especiales	
No se posee requerimientos especiales.	
Postcondiciones	
Visualizar, solicitar y descargar reporte en formato PDF.	

Fuente: Elaboración propia, 2024.

Tabla 28:*Caso de uso de control de permisos del colaborador*

Prototipo: Propuesta funcional para la gestión del recurso humano para la Farmacia D'Angelo ubicada en Cartago	
Número de caso de uso:	Control de permisos del colaborador
Fecha elaboración:	17/06/2024
Descripción caso de uso:	Este módulo permitirá realizar el control de permiso de los colaboradores
Autor caso de uso:	Deyvin Polanco Espinoza
Actores relacionados:	Administrador
Precondiciones:	El actor debe iniciar sesión para lograr el ingreso exitoso al módulo
Flujo básico del caso de uso	
<ol style="list-style-type: none"> 1. El caso de uso inicia cuando el actor decide ingresar al módulo de control de permisos del colaborador 2. El sistema mostrará las solicitudes de los colaboradores para aprobar las horas extras, mostrará el botón de “Aprobar” (SF-01) se mostrará la fecha en la que se solicitó el permiso, el estado y el colaborador que las solicita. 3. Contará con un botón de “Detalle” (SF-02) que mostrará la información de cada usuario. 4. Contará con un botón de “Eliminar” (SF-03) que permite eliminar. 	
Subflujos	
SF-01	Una vez el usuario presiona el botón de “Aprobar”, se le notificará al usuario que está autorizado para tomar el permiso. (FA-01).
SF-02	Una vez el usuario presiona el botón de “Detalles”, se mostrará la información detallada de cada usuario individual. (FA-02).
SF-03	Una vez el usuario presiona el botón de “Eliminar”, se eliminará el registro del empleado. (FA-03).
Flujos alternos	
FA-01 Validar solicitudes	Se valida que exista una solicitud para aprobar los permisos.
FA-02 Validar información	Se valida que exista un registro para poder detallar la información.
FA-03 Validar permisos	Se valida que exista ya una solicitud para poder realizar la eliminación de los datos
Requerimientos especiales	
No se posee requerimientos especiales	
Postcondiciones	
Visualizar, aprobar y eliminar.	

Fuente: Elaboración propia, 2024.

Tabla 29:
Caso de uso de control de permisos del colaborador

Prototipo: Propuesta funcional para la gestión del recurso humano para la Farmacia D'Angelo ubicada en Cartago	
Número de caso de uso:	Control de permisos del colaborador
Fecha elaboración:	17/06/2024
Descripción caso de uso:	Este módulo permitirá realizar la solicitud de los permisos.
Autor caso de uso:	Deyvin Polanco Espinoza
Actores relacionados:	Empleado
Precondiciones:	El actor debe iniciar sesión para lograr el ingreso exitoso al módulo
Flujo básico del caso de uso	
<ol style="list-style-type: none"> 1. El caso de uso inicia cuando el actor decide ingresar al módulo de solicitudes personales. 2. El sistema mostrará las solicitudes del colaborador, mostrara si están aprobadas o rechazadas y la fecha de solicitud, el sistema mostrara un botón de “Solicitar permisos”. (SF-01) 	
Subflujos	
SF-01	Una vez el usuario presiona el botón de “Solicitar permiso”, se mostrara una vista donde el usuario seleccionara el tipo de permiso y la fecha en la que solicitara el permiso, una vez creado se mostrará el estado de la solicitud cuando se apruebe podrá gozar del permiso. (FA-01) (FA-02)
Flujos alternos	
FA-01 Validar permisos	Se valida que el empleado este registrado en el sistema para que pueda realizar la solicitud de permisos.
FA-02 Validar información	Se valida que la información se inserte de manera correcta.
Requerimientos especiales	
No se posee requerimientos especiales.	
Postcondiciones	
Visualizar y solicitar.	

Fuente: Elaboración propia, 2024.

Tabla 30:*Caso de uso de cálculo de liquidación*

Prototipo: Propuesta funcional para la gestión del recurso humano para la Farmacia D'Angelo ubicada en Cartago	
Número de caso de uso:	Calcular liquidación
Fecha elaboración:	17/06/2024
Descripción caso de uso:	Este módulo permitirá realizar el cálculo de liquidación de los colaboradores
Autor caso de uso:	Deyvin Polanco Espinoza
Actores relacionados:	Administrador
Precondiciones:	El actor debe iniciar sesión para lograr el ingreso exitoso al módulo
Flujo básico del caso de uso	
<ol style="list-style-type: none"> 1. El caso de uso inicia cuando el actor decide ingresar al módulo de control de permisos del colaborador 2. El sistema mostrará las solicitudes de los colaboradores para aprobar las horas extras, mostrará el botón de “Calcular liquidación” (SF-01) donde se mostrará el tipo de liquidación si es con goce o sin goce salarial 3. Contará con un botón de “Detalle” (SF-02) que mostrará la información de cada usuario. 4. Contará con un botón de “Eliminar” (SF-03) que permite eliminar. 	
Subflujos	
SF-01	Una vez el usuario presiona el botón de crear, se mostrará una vista donde se ingresará la fecha de salida del colaborador, el tipo de liquidación donde se valida si es con o sin goce salarial, adicional a esto mostrara los cálculos correspondientes como, monto preaviso, monto cesantía, monto de vacaciones, monto de aguinaldo y el monto total, (FA-01)
SF-02	Una vez el usuario presiona el botón de “Detalles”, se mostrará la información detallada de cada usuario individual. (FA-02)
SF-03	Una vez el usuario presiona el botón de “Eliminar”, se eliminará el registro del empleado. (FA-03)
Flujos alternos	
FA-01 Validar liquidación	Se valida que exista información registrada en la base de datos para poder realizar el cálculo de manera correcta.
FA-02 Validar información	Se valida que exista un registro para poder detallar la información.
FA-03 Validar información de liquidación	Se valida que exista un registro para realizar la eliminación.
Requerimientos especiales	
No se posee requerimientos especiales.	
Postcondiciones	
Visualizar, aprobar y eliminar.	

Fuente: Elaboración propia, 2024.

Tabla 31:*Caso de uso de cálculo de liquidación*

Prototipo: Propuesta funcional para la gestión del recurso humano para la Farmacia D'Angelo ubicada en Cartago	
Número de caso de uso:	Calcular liquidación
Fecha elaboración:	17/06/2024
Descripción caso de uso:	Este módulo permitirá realizar el cálculo de liquidación de los colaboradores
Autor caso de uso:	Deyvin Polanco Espinoza
Actores relacionados:	Empleado
Precondiciones:	El actor debe iniciar sesión para lograr el ingreso exitoso al módulo
Flujo básico del caso de uso	
<ol style="list-style-type: none"> 1. El caso de uso inicia cuando el actor decide ingresar al módulo de control de permisos del colaborador. 2. El sistema mostrará el cálculo respectivo de la liquidación. (SF-01). 	
Subflujos	
SF-01	Una vez el usuario entra al módulo cálculo de liquidación mostrará los cálculos correspondientes como, monto preaviso, monto cesantía, monto de vacaciones, monto de aguinaldo y el monto total. (FA-01)
Flujos alternos	
FA-01 Validar liquidación	Se valida que exista información registrada en la base de datos para poder realizar el cálculo de manera correcta.
Requerimientos especiales	
No se posee requerimientos especiales	
Postcondiciones	
Visualizar	

Fuente: Elaboración propia, 2024.

Tabla 32:
Caso de uso cálculo de liquidación

Prototipo: Propuesta funcional para la gestión del recurso humano para la Farmacia D'Angelo ubicada en Cartago	
Número de caso de uso:	Calcular incapacidades
Fecha elaboración:	17/06/2024
Descripción caso de uso:	Este módulo permitirá realizar el cálculo de liquidación de los colaboradores
Autor caso de uso:	Deyvin Polanco Espinoza
Actores relacionados:	Administrador
Precondiciones:	El actor debe iniciar sesión para lograr el ingreso exitoso al módulo
Flujo básico del caso de uso	
<ol style="list-style-type: none"> 1. El caso de uso inicia cuando el actor decide ingresar al módulo de incapacidades 2. El sistema mostrará un botón llamado “Crear incapacidades” (SF-01) 3. El sistema mostrará un botón llamado “Editar” (SF-02) 4. Contará con un botón de “Detalle” (SF-03) que mostrará la información de cada usuario. 5. Contará con un botón de “Eliminar” (SF-04) que permite eliminar. 	
Subflujos	
SF-01	Una vez el usuario presiona el botón de crear, se mostrará una vista donde se ingresará la fecha de inicio, fecha fin, el colaborador al que se le aplica la incapacidad y el tipo de incapacidad luego se crea la incapacidad. (FA-01) con su cálculo correspondiente.
SF-02	Se mostrará el botón de editar el tipo de incapacidad. (FA-02)
SF-03	Una vez el usuario presiona el botón de “Detalles” se mostrará la información detallada de cada usuario individual. (FA-02).
SF-04	Una vez el usuario presiona el botón de “Eliminar”, se eliminará el registro del empleado. (FA-03).
Flujos alternos	
FA-01 Validar liquidación	Se valida que exista información registrada en la base de datos para poder realizar el cálculo de manera correcta.
FA-02 Validar registro	Se valida que exista un registro de incapacidades para poder editarlo.
FA-03 Validar información	Se valida que exista un registro para poder detallar la información.
FA-03 Validar información de incapacidades	Se valida que exista un registro para realizar la eliminación
Requerimientos especiales	
No se posee requerimientos especiales.	
Postcondiciones	
Visualizar, aprobar y eliminar.	

Fuente: Elaboración propia, 2024.

Tabla 33:
Caso de uso cálculo de incapacidad

Prototipo: Propuesta funcional para la gestión del recurso humano para la Farmacia D'Angelo ubicada en Cartago	
Número de caso de uso:	Calcular incapacidad
Fecha elaboración:	17/06/2024
Descripción caso de uso:	Este módulo permitirá visualizar las incapacidades de los colaboradores
Autor caso de uso:	Deyvin Polanco Espinoza
Actores relacionados:	Empleado
Precondiciones:	El actor debe iniciar sesión para lograr el ingreso exitoso al módulo
Flujo básico del caso de uso	
<ol style="list-style-type: none"> 1. El caso de uso inicia cuando el actor decide ingresar al módulo de incapacidad. 2. El sistema mostrará el cálculo respectivo de incapacidad. (SF-01) 	
Subflujos	
SF-01	Una vez el usuario entra al módulo incapacidad, se le mostrará el tipo de incapacidad, monto la fecha de inicio y la fecha final de la incapacidad. (FA-01)
Flujos alternos	
FA-01 Validar incapacidad	Se valida que exista información registrada en la base de datos para poder realizar el cálculo de manera correcta.
Requerimientos especiales	
No se posee requerimientos especiales.	
Postcondiciones	
Visualizar.	

Fuente: Elaboración propia, 2024.

Tabla 34:*Caso de uso de mantenimiento*

Prototipo: Propuesta funcional para la gestión del recurso humano para la Farmacia D'Angelo ubicada en Cartago	
Número de caso de uso:	Mantenimiento
Fecha elaboración:	17/06/2024
Descripción caso de uso:	Este módulo permitirá realizar el mantenimiento a todos los registros del sistema.
Autor caso de uso:	Deyvin Polanco Espinoza
Actores relacionados:	Administrador
Precondiciones:	El actor debe iniciar sesión para lograr el ingreso exitoso al módulo
Flujo básico del caso de uso	
<ol style="list-style-type: none"> 1. El caso de uso inicia cuando el actor decide ingresar al módulo de mantenimiento. 2. El sistema mostrará un botón llamado “Crear” (SF-01) que mantendrá la misma estructura en todas las tablas ya que este cumple con un estándar para que mantenga el mismo formato. 3. El sistema mostrará un botón llamado “Editar” (SF-02). 4. Contará con un botón de “Detalle” (SF-03) que mostrará la información de cada usuario. 5. Contará con un botón de “Eliminar” (SF-04) que permite eliminar. 	
Subflujos	
SF-01	Una vez el usuario presiona el botón de “Crear”, se mostrará una vista donde se ingresará los campos necesarios para realizar las inserciones. (FA-01).
SF-02	Se mostrará el botón de editar para realizar modificaciones. (FA-02)
SF-03	Una vez el usuario presiona el botón de “Detalles”, se mostrará la información detallada individual. (FA-02)
SF-04	Una vez el usuario presiona el botón de “Eliminar”, se eliminará el registro. (FA-03)
Flujos alternos	
FA-01 Validar liquidación	Se valida que la información se agregue de manera correcta para poder agregar los datos de forma correcta en la base de datos.
FA-02 Validar registro	Se valida que exista un registro para poder editarlo.
FA-03 Validar información	Se valida que exista un registro para poder detallar la información.
FA-03 Validar información de incapacidades	Se valida que exista un registro para realizar la eliminación.
Requerimientos especiales	
No se posee requerimientos especiales.	
Postcondiciones	
Crear, modificar, visualizar y eliminar.	

Fuente: Elaboración propia, 2024.

Tabla 35:
Caso de uso de consultas

Prototipo: Propuesta funcional para la gestión del recurso humano para la Farmacia D'Angelo ubicada en Cartago	
Número de caso de uso:	Consulta
Fecha elaboración:	17/06/2024
Descripción caso de uso:	Este módulo permitirá realizar consultas de manera rápida y poder acceder más rápido a la información
Autor caso de uso:	Deyvin Polanco Espinoza
Actores relacionados:	Administrador
Precondiciones:	El actor debe iniciar sesión para lograr el ingreso exitoso al módulo
Flujo básico del caso de uso	
<ol style="list-style-type: none"> 1. El caso de uso inicia cuando el actor decide ingresa a cualquier modulo del sistema. 2. El sistema mostrara una barra que le permite acceder más rápido a la información solicitada. (SF-01) 	
Subflujos	
SF-01	Una vez el usuario ingresa datos para filtrar una búsqueda, se mostrarán los resultados (FA-01)
Flujos alternos	
FA-01 Validar liquidación	Se valida que exista información que pueda ser filtrada para agilizar la búsqueda.
Requerimientos especiales	
No se posee requerimientos especiales.	
Postcondiciones	
Buscar información.	

Fuente: Elaboración propia, 2024.

Tabla 36:
Caso de uso de reportes

Prototipo: Propuesta funcional para la gestión del recurso humano para la Farmacia D'Angelo ubicada en Cartago	
Número de caso de uso:	Reporte
Fecha elaboración:	17/06/2024
Descripción caso de uso:	Este módulo permitirá realizar reportes de los módulos.
Autor caso de uso:	Deyvin Polanco Espinoza
Actores relacionados:	Administrador
Precondiciones:	El actor debe iniciar sesión para lograr el ingreso exitoso al módulo
Flujo básico del caso de uso	
<ol style="list-style-type: none"> 1. El caso de uso inicia cuando el actor decide ingresa a cualquier módulo del sistema. 2. El sistema mostrará un botón “Generar PDF” (SF-01) que les permitirá exportar a PDF el contenido del módulo. 3. El sistema mostrará un módulo de generar reportes, donde se seleccionará el tipo de reporte que se desea realizar, una vez seleccionado se presiona el botón “Generar reporte” (SF-02) 	
Subflujos	
SF-01	Una vez el usuario presiona el botón “Generar PDF” , se genera un PDF con la información del módulo (FA-01)
SF-02	Al ingresar al módulo de reportes, se podrán generar reportes administrativos que le ayudara a la toma de decisiones. (FA-01)
Flujos alternos	
FA-01 Validar información	Se valida que exista información que pueda ser filtrada para realizar reportes
Requerimientos especiales	
No se posee requerimientos especiales.	
Postcondiciones	
Generar reportes.	

Fuente: Elaboración propia, 2024.

Tabla 37:
Caso de uso de seguridad

Prototipo: Propuesta funcional para la gestión del recurso humano para la Farmacia D'Angelo ubicada en Cartago	
Número de caso de uso:	Seguridad
Fecha elaboración:	17/06/2024
Descripción caso de uso:	Este módulo permitirá realizar reportes de los módulos.
Autor caso de uso:	Deyvin Polanco Espinoza
Actores relacionados:	Administrador
Precondiciones:	El actor debe iniciar sesión para lograr el ingreso exitoso al módulo
Flujo básico del caso de uso	
<ol style="list-style-type: none"> 1. El caso de uso inicia cuando el actor decide ingresa a cualquier módulo de seguridad. 2. El sistema contara con un módulo de seguridad que permite ingresar usuarios y clasificarlos en administrador y usuario (SF-01). 3. Ingresa al sistema, se mostrará el botón “Iniciar sesión” (SF-02) 	
Subflujos	
SF-01	Se clasifica el usuario para que pueda realizar el ingreso al sistema, una vez creado si es administrador este deberá poder ingresar a todos los módulos. (FA-01)
SF-02	Una vez el usuario sea creado a la hora de ingresar al sistema, se muestra la información del usuario dependiendo de su tipo de roll.
Flujos alternos	
FA-01 Validar usuarios	Se valida que se registre de forma correcta en el sistema.
FA-02 Validar información	Se valida que el roll de usuario sea creado de manera correcta para poder permitir el acceso dependiendo de su roll.
Requerimientos especiales	
No se posee requerimientos especiales.	
Postcondiciones	
Buscar información.	

Fuente: Elaboración propia, 2024.

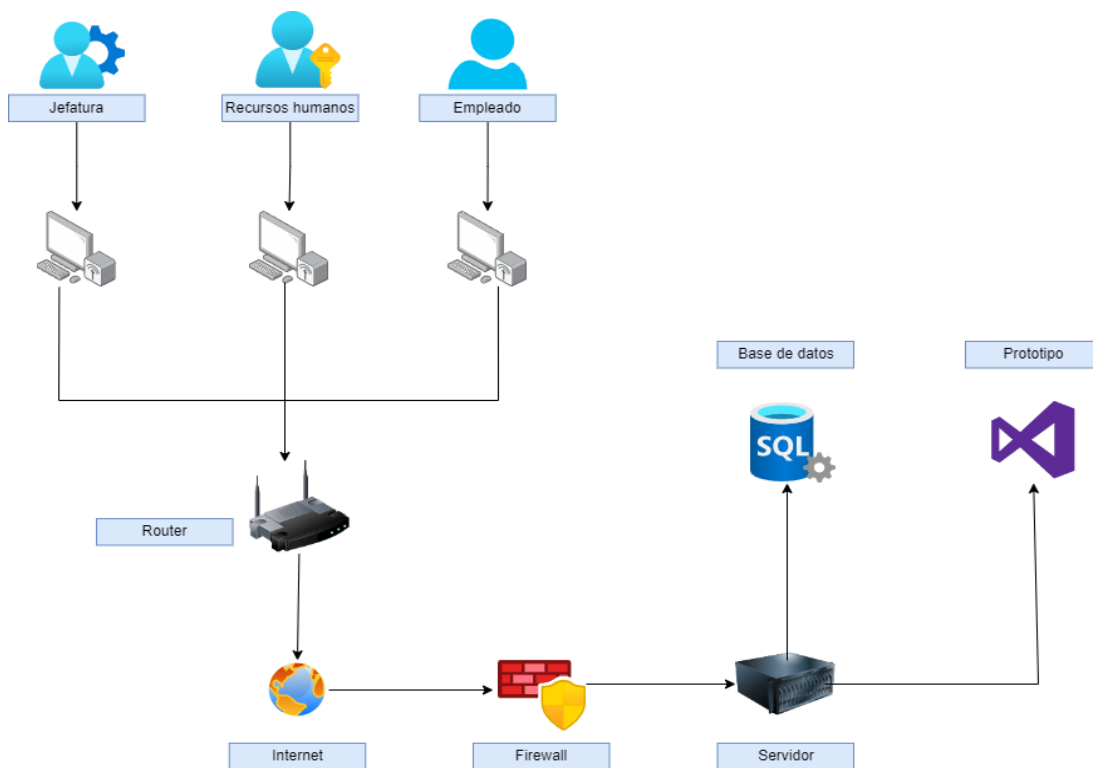
Diseño

El diseño del *software* es un aspecto fundamental en el desarrollo de sistemas de información. En esta sección se muestra la arquitectura del *software*, incluyendo la estructura del sistema, los componentes principales y cómo interactúan entre sí. Se busca dar una visión detallada de cómo se organiza y funciona el sistema, facilitando la comprensión del funcionamiento del sistema

Arquitectura del Sistema

En esta sección, se analiza la arquitectura del sistema proporcionando una visión general del prototipo. Este proporciona una comprensión general del prototipo y cómo se relaciona con el entorno organizativo, además, contribuye a sus operaciones. En la figura 20, se detalla la arquitectura del sistema.

Figura 20:
Arquitectura de sistema



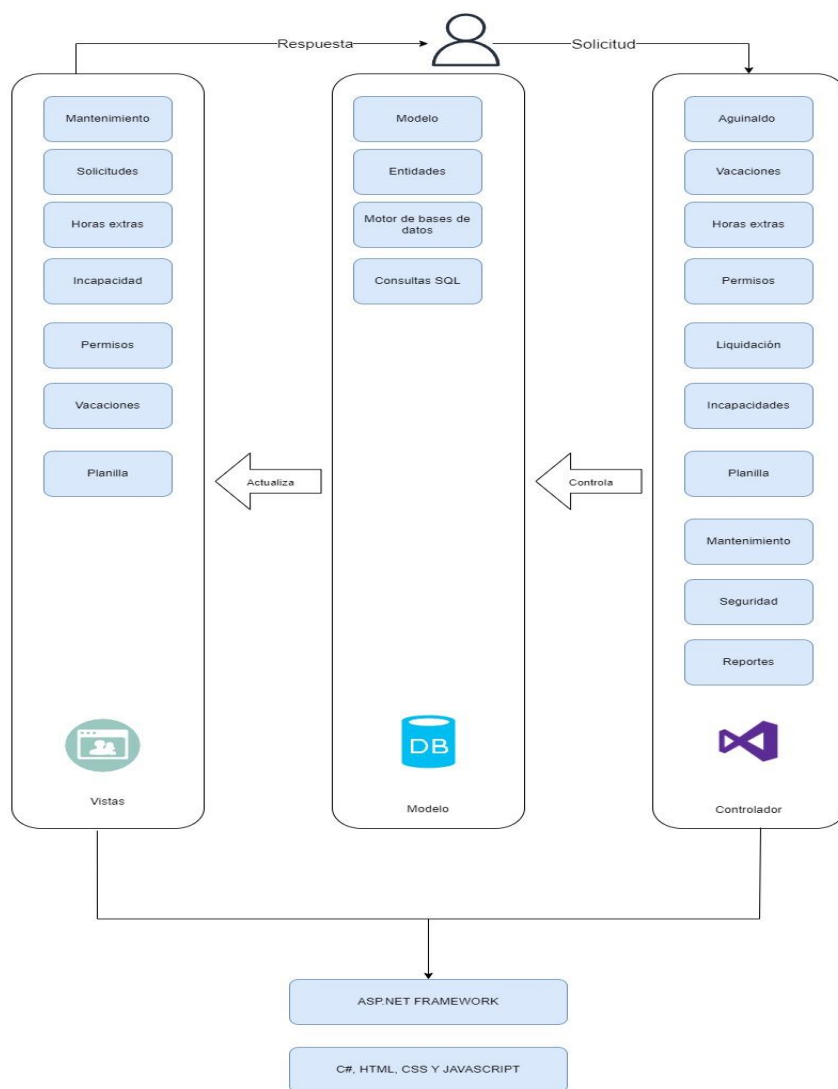
Fuente: Elaboración propia, 2024.

En la figura 20, se puede visualizar cómo está establecida la red LAN mediante la cual los usuarios pueden ingresar al sistema, solamente estos tres usuarios podrán ingresar al sistema, ya que son los únicos tres roles definidos dentro del sistema. El servidor mantendrá el sitio web y la relación a bases de datos. Adicional a esto, el servidor está de forma local, lo que permitirá agregar equipos personales y facilitar el ingreso al sistema.

Arquitectura de Software

En esta sección, se muestra la arquitectura interna del prototipo, donde se detallan sus principales módulos y procesos que conforman su estructura. Se analiza la organización en capas utilizadas en el diseño del *software*, la arquitectura interna del *software* se diseñó con un enfoque modular y en capas para garantizar la escalabilidad y flexibilidad del sistema, en la siguiente imagen se detalla la arquitectura del *software*.

Figura 21:
Arquitectura del software



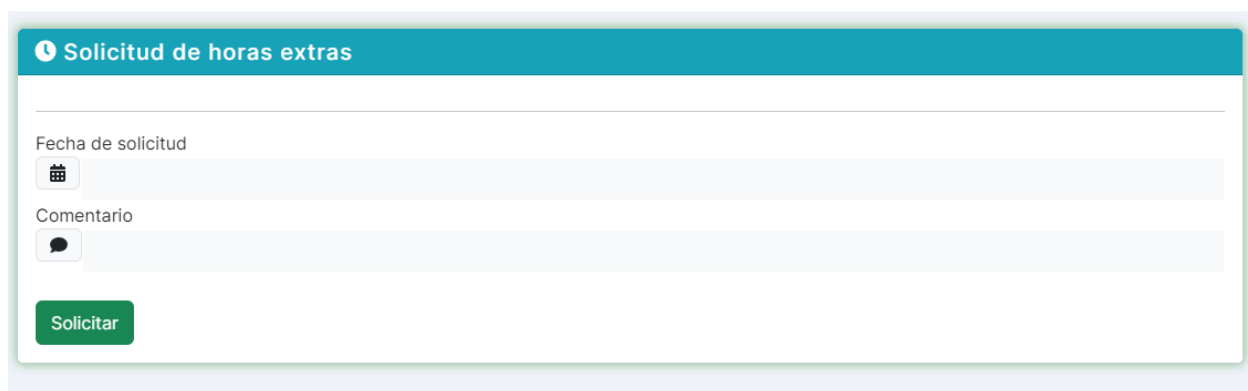
Fuente: Elaboración propia, 2024.

Diseño de Entradas

En este apartado, se detallan las entradas de datos del sistema, especificando los formularios y campos, los cuales el usuario debe completar para interactuar con los diferentes módulos del sistema. El diseño de entradas se ha realizado de manera minuciosa para que no se puedan ingresar datos erróneos en la solicitud, en las siguientes imágenes se muestran pantallas del diseño de entradas.

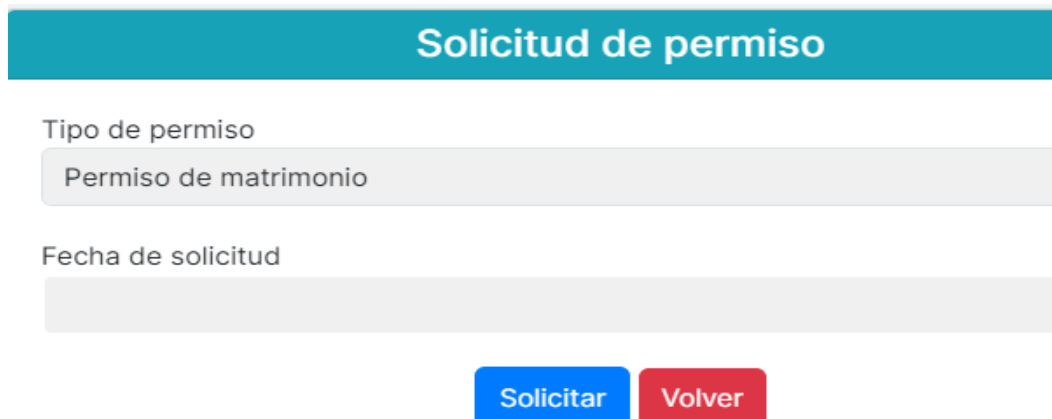
Figura 22:

Diseño de entradas solicitud de extras

The image shows a web form titled "Solicitud de horas extras" (Request for extra hours). The form has a teal header with the title and a back arrow icon. Below the header, there are two input fields: "Fecha de solicitud" (Request date) with a calendar icon and "Comentario" (Comment) with a speech bubble icon. At the bottom of the form is a green button labeled "Solicitar" (Request).

Fuente: Elaboración propia, 2024.

En esta pantalla se realiza la solicitud de horas extras, el usuario deberá ingresar la fecha en la cual desea realizar la solicitud y un comentario, en la parte de la inserción de fechas está validado para que simplemente se pueda ingresar una fecha. Se maneja de esta manera porque esto evita que el usuario inserte datos incorrectos.

Figura 23:*Diseño de entradas solicitud de permisos*

Formulario de solicitud de permisos. El encabezado es un banner azul con el texto "Solicitud de permiso". Debajo, hay un campo de texto con el label "Tipo de permiso" que contiene el texto "Permiso de matrimonio". A continuación, hay un campo de texto con el label "Fecha de solicitud". En la parte inferior, hay dos botones: "Solicitar" (azul) y "Volver" (rojo).

Fuente: Elaboración propia, 2024.

En la figura 23, se muestra la solicitud de permisos, el usuario deberá insertar el tipo de permiso que desea solicitar y la fecha de solicitud, en los permisos se muestra un *viewbag* con los permisos existentes en la empresa para que el colaborador no solicite un permiso que no existe. En el caso de la fecha se muestra como calendario, para que el usuario no pueda ingresar números o letras y, de esta manera, se realiza una inserción de datos de forma correcta.

Figura 24:*Diseño de entradas solicitud de vacaciones*

Formulario de solicitud de vacaciones. El encabezado es un banner azul con el texto "+ Crear vacaciones". Debajo, hay un campo de texto con el label "Fecha de inicio" y un icono de calendario. A continuación, hay un campo de texto con el label "Fecha fin" y un icono de calendario. En la parte inferior, hay dos botones: "Solicitar" (azul) y "Volver" (rojo).

Fuente: Elaboración propia, 2024.

En la figura 24, se muestra la solicitud de vacaciones. El usuario deberá insertar la fecha de inicio y la fecha fin de las vacaciones, cada uno de los campos a la hora de insertar muestra un calendario, lo que no permite que el usuario inserte número o letras, ya que el formato es de tipo fecha. Se valida de esta manera para que se puedan insertar los datos de manera correcta.

Figura 25:
Diseño de entradas crear incapacidad



El formulario 'Crear incapacidad' tiene un encabezado azul con el texto '+ Crear incapacidad'. Contiene cuatro campos de entrada con iconos de calendario: 'Fecha de inicio', 'Fecha fin', 'ID de Empleado' (con el valor 'Deyvin') y 'Tipo de Incapacidad' (con el valor 'CCSS'). En la parte inferior hay dos botones: 'Crear' (azul) y 'Volver' (rojo).

Fuente: Elaboración propia, 2024.

En la figura 25, se muestra la creación de incapacidades, lo que es el empleado y el tipo de incapacidad son un *viewbag* porque es información que tiene que estar creada para generar una incapacidad. Se valida de esta forma para que el usuario no pueda ingresar datos que no existen, las fechas son tipo calendario para que no puedan insertar datos o números.

Figura 26:
Diseño de entradas crear horario



El formulario 'Crear horario' tiene un encabezado azul con el texto '+ Crear horario'. Contiene cinco campos de entrada: 'Hora de Entrada', 'Hora de Salida', 'Día' (con el valor 'Jueves'), 'Tipo horario' (con el valor 'Diurno') y 'ID de Empleado' (con el valor 'Deyvin'). En la parte inferior hay dos botones: 'Crear' (azul) y 'Volver' (rojo).

Fuente: Elaboración propia, 2024.

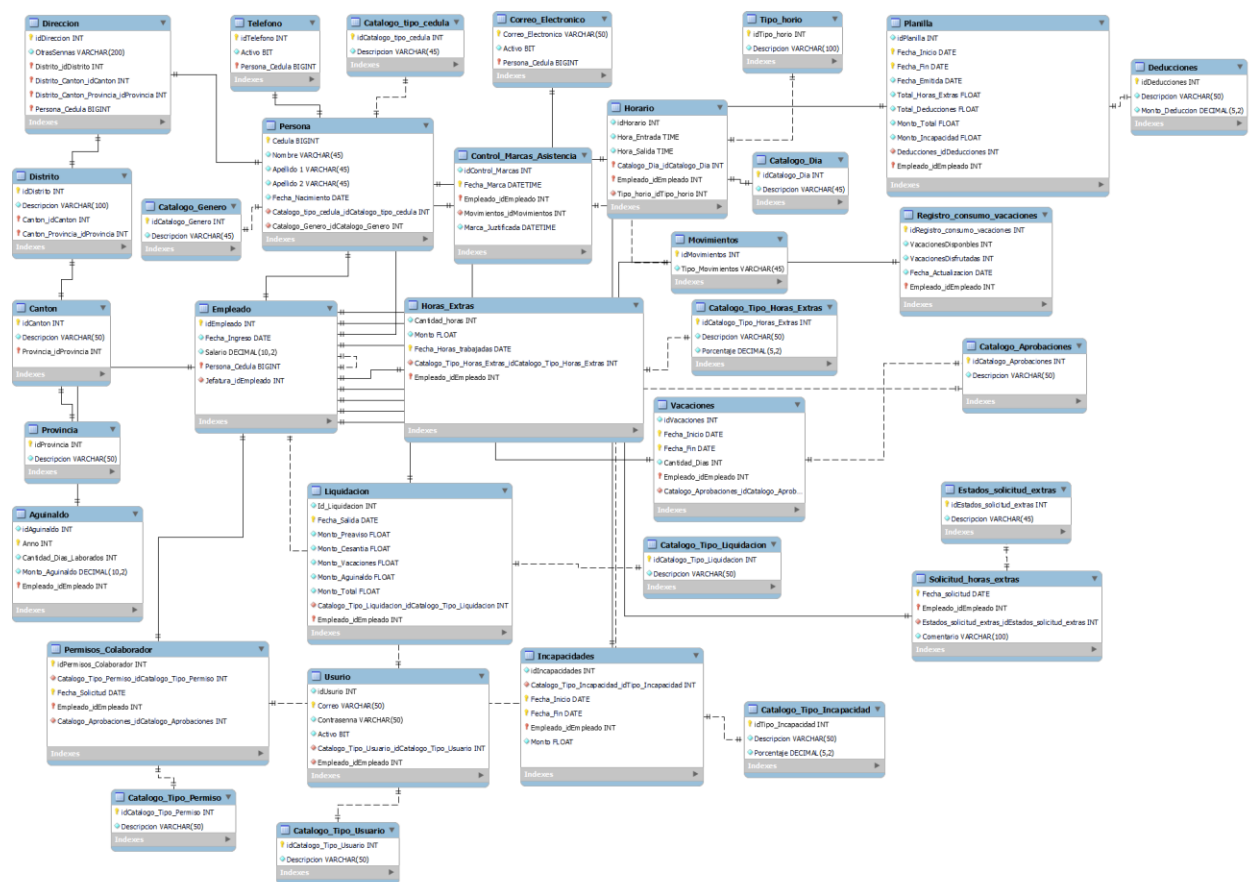
En la figura 26, se muestra la creación de horarios, lo que es el empleado y el tipo turno y día son un *viewbag* porque es información que tiene que estar creada para generar un horario. La

hora de entrada y salida no se pueden insertar números, ya que es un formato de tipo hora. Esto se restringe para que se pueda insertar la información de forma correcta y no existan inconsistencias.

Diseño Físico de la Base de Datos

En esta sección, se muestra el diseño de la base de datos implementado para el desarrollo del prototipo. Se detallan todas las tablas o entidades utilizadas, así como sus relaciones y estructuras. Para la elaboración de la base de datos, se respetaron las formas normales con el fin de garantizar una base de datos de calidad y optimizada. En la siguiente imagen se muestra el modelo entidad relación.

Figura 27:
Diseño físico de base de datos



Fuente: Elaboración propia, 2024.

En esta sección, se muestra el diseño de la base de datos implementada para el desarrollo del prototipo. Se detallan todas las tablas o entidades utilizadas, se aprecian las relaciones que se implementaron con el fin de que el prototipo sea desarrollado de manera eficiente. Se muestran la información de las llaves primarias y en algunas tablas existen relaciones compuestas que permiten que no exista duplicidad de datos. Además, para el desarrollo de la base de datos, se tomaron en cuenta las formas normales de datos; la normalización es esencial para garantizar la organización eficiente de la base de datos.

Diccionario de Datos

Tabla 38:

Diccionario de datos aguinaldo

Nombre de columna	Tipo dato	PK	NN	Descripción
idAguinaldo	INT		✓	
Anno	DATE	✓	✓	
Cantidad_Dias_Laborados	INT		✓	
Monto_Aguinaldo	DECIMAL(10,2)		✓	
Empleado_idEmpleado	INT	✓	✓	

Fuente: Elaboración propia, 2024

Tabla 39:

Diccionario de datos cantón

Nombre de columna	Tipo dato	PK	NN	Descripción
idCanton	INT	✓	✓	
Descripcion	VARCHAR(50)		✓	
Provincia_idProvincia	INT	✓	✓	

Fuente: Elaboración propia, 2024

Tabla 40:

Diccionario de datos catálogo aprobaciones.

Nombre de columna	Tipo dato	PK	NN	Descripción
idCatalogo_Aprobaciones	INT	✓	✓	
Descripcion	VARCHAR(50)		✓	

Fuente: Elaboración propia, 2024

Tabla 41:*Diccionario de datos catálogo día*

Nombre de columna	Tipo dato	PK	NN	Descripción
idCatalogo_Dia	INT	✓	✓	
Descripcion	VARCHAR(45)		✓	

Fuente: Elaboración propia, 2024

Tabla 42:*Diccionario de datos catálogo género*

Nombre de columna	Tipo dato	PK	NN	Descripción
idCatalogo_Genero	INT	✓	✓	
Descripcion	VARCHAR(45)		✓	

Fuente: Elaboración propia, 2024

Tabla 43:*Diccionario de datos catálogo tipo horas extras.*

Nombre de columna	Tipo dato	PK	NN	Descripción
idCatalogo_Tipo_Horas_Extras	INT	✓	✓	
Descripcion	VARCHAR(50)		✓	
Porcentaje	DECIMAL(5,2)		✓	

Fuente: Elaboración propia, 2024

Tabla 44:*Diccionario de datos Catálogo tipo incapacidad*

Nombre de columna	Tipo dato	PK	NN	Descripción
idTipo_Incapacidad	INT	✓	✓	
Descripcion	VARCHAR(50)		✓	
Porcentaje	DECIMAL (5,2)		✓	

Fuente: Elaboración propia, 2024

Tabla 45:*Diccionario de datos Catálogo tipo liquidación*

Nombre de columna	Tipo dato	PK	NN	Descripción
idCatalogo_Tipo_Liquidacion	INT	✓	✓	
Descripcion	VARCHAR (50)		✓	

Fuente: Elaboración propia, 2024

Tabla 46:*Diccionario de datos catálogo tipo permiso*

Nombre de columna	Tipo dato	PK	NN	Descripción
idCatalogo_Tipo_Permission	INT	✓	✓	
Descripcion	VARCHAR(50)		✓	

Fuente: Elaboración propia, 2024

Tabla 47:*Diccionario de datos catálogo tipo usuario*

Nombre de columna	Tipo dato	PK	NN	Descripción
idCatalogo_Tipo_Usuario	INT	✓	✓	
Descripcion	VARCHAR(50)		✓	

Fuente: Elaboración propia, 2024

Tabla 48:*Diccionario de datos tipo cédula*

Nombre de columna	Tipo dato	PK	NN	Descripción
idCatalogo_tipo_cedula	INT	✓	✓	
Descripcion	VARCHAR(45)		✓	

Fuente: Elaboración propia, 2024

Tabla 49:*Diccionario de datos control de marcas de asistencia*

Nombre de columna	Tipo dato	PK	NN	Descripción
idControl_Marcas	INT		✓	
Fecha_Marca	DATETIME	✓	✓	
Empleado_idEmpleado	INT	✓	✓	
Movimientos_idMovimientos	INT		✓	
Marca_Justificada	DATETIME		✓	

Fuente: Elaboración propia, 2024

Tabla 50:*Diccionario de datos correo electrónico*

Nombre de columna	Tipo dato	PK	NN	Descripción
Correo_Electronico	VARCHAR(50)	✓	✓	
Activo	BIT		✓	
Persona_Cedula	BIGINT	✓	✓	

Fuente: Elaboración propia, 2024

Tabla 51:*Diccionario de bases de datos deducciones.*

Nombre de columna	Tipo dato	PK	NN	Descripción
idDeducciones	INT	✓	✓	
Descripcion	VARCHAR(50)		✓	
Monto_Deducccion	DECIMAL(5,2)		✓	

Fuente: Elaboración propia, 2024

Tabla 52:*Diccionario de datos dirección*

Nombre de columna	Tipo dato	PK	NN	Descripción
idDireccion	INT	✓	✓	
OtrasSennas	VARCHAR(200)		✓	
Distrito_idDistrito	INT	✓	✓	
Distrito_Canton_idCanton	INT	✓	✓	
Distrito_Canton_Provincia_idProvincia	INT	✓	✓	
Persona_Cedula	BIGINT	✓	✓	

Fuente: Elaboración propia, 2024

Tabla 53:*Diccionario de datos distrito*

Nombre de columna	Tipo dato	PK	NN	Descripción
idDistrito	INT	✓	✓	
Descripcion	VARCHAR(100)		✓	
Canton_idCanton	INT	✓	✓	
Canton_Provincia_idProvincia	INT	✓	✓	

Fuente: Elaboración propia, 2024

Tabla 54:*Diccionario de datos empleado*

Nombre de columna	Tipo dato	PK	NN	Descripción
idEmpleado	INT	✓	✓	
Fecha_Ingreso	DATE		✓	
Salario	DECIMAL(10,2)		✓	
Persona_Cedula	BIGINT	✓	✓	
Jefatura_idEmpleado	INT		✓	

Fuente: Elaboración propia, 2024

Tabla 55:*Diccionario de datos horario*

Nombre de columna	Tipo dato	PK	NN	Descripción
idHorario	INT		✓	
Hora_Entrada	TIME		✓	
Hora_Salida	TIME		✓	
Catalogo_Dia_idCatalogo_Dia	INT	✓	✓	
Empleado_idEmpleado	INT	✓	✓	
Tipo_horio_idTipo_horio	INT		✓	

Fuente: Elaboración propia, 2024

Tabla 56:*Diccionario de datos horas extras*

Nombre de columna	Tipo dato	PK	NN	Descripción
Cantidad_horas	INT		✓	
Monto	FLOAT		✓	
Fecha_Horas_trabajadas	DATE	✓	✓	
Catalogo_Tipo_Horas_Extras_idCatalogo_Tipo_Horas_Extras	INT		✓	
Empleado_idEmpleado	INT	✓	✓	

Fuente: Elaboración propia, 2024

Tabla 57:*Diccionario de datos incapacidades*

Nombre de columna	Tipo dato	PK	NN	Descripción
idIncapacidades	INT		✓	
Catalogo_Tipo_Incapacidad_id Tipo_Incapacidad	INT		✓	
Fecha_Inicio	DATE	✓	✓	
Fecha_Fin	DATE	✓	✓	
Empleado_idEmpleado	INT	✓	✓	
Monto	FLOAT		✓	

Fuente: Elaboración propia, 2024

Tabla 58:*Diccionario de datos liquidación*

Nombre de columna	Tipo dato	PK	NN	Descripción
Id_Liquidacion	INT		✓	
Fecha_Salida	DATE	✓	✓	
Monto_Preaviso	FLOAT		✓	
Monto_Cesantia	FLOAT		✓	
Monto_Vacaciones	FLOAT		✓	
Monto_Aguinaldo	FLOAT		✓	
Monto_Total	FLOAT		✓	
Catalogo_Tipo_Liquidacion_id Catalogo_Tipo_Liquidacion	INT		✓	
Empleado_idEmpleado	INT	✓	✓	

Fuente: Elaboración propia, 2024

Tabla 59:*Diccionario de datos movimiento*

Nombre de columna	Tipo dato	PK	NN	Descripción
idMovimientos	INT	✓	✓	
Tipo_Movimientos	VARCHAR(45)		✓	

Fuente: Elaboración propia, 2024

Tabla 60:*Diccionario de datos permisos del colaborador*

Nombre de columna	Tipo dato	PK	NN	Descripción
idPermisos_Colaborador	INT	✓	✓	
Catalogo_Tipo_Permission_idCatalogo_Tipo_Permission	INT		✓	
Fecha_Solicitud	DATE	✓	✓	
Empleado_idEmpleado	INT	✓	✓	
Catalogo_Aprobaciones_idCatalogo_Aprobaciones	INT		✓	

Fuente: Elaboración propia, 2024

Tabla 61:*Diccionario de datos persona*

Nombre de columna	Tipo dato	PK	NN	Descripción
Cedula	BIGINT	✓	✓	
Nombre	VARCHAR(45)		✓	
Apellido 1	VARCHAR(45)		✓	
Apellido 2	VARCHAR(45)		✓	
Fecha_Nacimiento	DATE		✓	
Catalogo_tipo_cedula_idCatalogo_tipo_cedula	INT		✓	
Catalogo_Genero_idCatalogo_Genero	INT		✓	

Fuente: Elaboración propia, 2024

Tabla 62:*diccionario de datos de planilla*

Nombre de columna	Tipo dato	PK	NN	Descripción
idPlanilla	INT		✓	
Fecha_Inicio	DATE	✓	✓	
Fecha_Fin	DATE	✓	✓	
Fecha_Emitida	DATE		✓	
Total_Horas_Extras	FLOAT		✓	
Total_Deducciones	FLOAT		✓	
Monto_Total	FLOAT		✓	
Monto_Incapacidad	FLOAT		✓	
Deducciones_idDeducciones	INT		✓	
Empleado_idEmpleado	INT	✓	✓	

Fuente: Elaboración propia, 2024

Tabla 63:*Diccionario de datos de provincia*

Nombre de columna	Tipo dato	PK	NN	Descripción
idProvincia	INT	✓	✓	
Descripcion	VARCHAR(50)		✓	

Fuente: Elaboración propia, 2024

Tabla 64:*Diccionario de datos solicitud de extras*

Nombre de columna	Tipo dato	PK	NN	Descripción
Fecha_solicitud	DATE	✓	✓	
Empleado_idEmpleado	INT	✓	✓	
Estados_solicitud_extras_idEstados_solicitud_extras	INT		✓	
Comentario	VARCHAR(100)		✓	

Fuente: Elaboración propia, 2024

Tabla 65:*Diccionario de datos teléfono*

Nombre de columna	Tipo dato	PK	NN	Descripción
idTelefono	INT	✓	✓	
Activo	BIT		✓	
Persona_Cedula	BIGINT	✓	✓	

Fuente: Elaboración propia, 2024

Tabla 66:*Diccionario de datos tipo horario*

Nombre de columna	Tipo dato	PK	NN	Descripción
idTipo_horio	INT	✓	✓	
Descripcion	VARCHAR(100)		✓	

Fuente: Elaboración propia, 2024

Tabla 67:*Diccionario de datos usuario*

Nombre de columna	Tipo dato	PK	NN	Descripción
idUsuario	INT		✓	
Correo	VARCHAR(50)	✓	✓	
Contraseña	VARCHAR(50)		✓	
Activo	BIT		✓	
Catalogo_Tipo_Usuario_idCatalogo_Tipo_Usuario	INT		✓	
Empleado_idEmpleado	INT		✓	

Fuente: Elaboración propia, 2024

Tabla 68:*Diccionario de datos vacaciones*

Nombre de columna	Tipo dato	PK	NN	Descripción
idVacaciones	INT		✓	
Fecha_Inicio	DATE	✓	✓	
Fecha_Fin	DATE	✓	✓	
Cantidad_Dias	INT		✓	
Empleado_idEmpleado	INT	✓	✓	
Catalogo_Aprobaciones_idCatalogo_Aprobaciones	INT		✓	

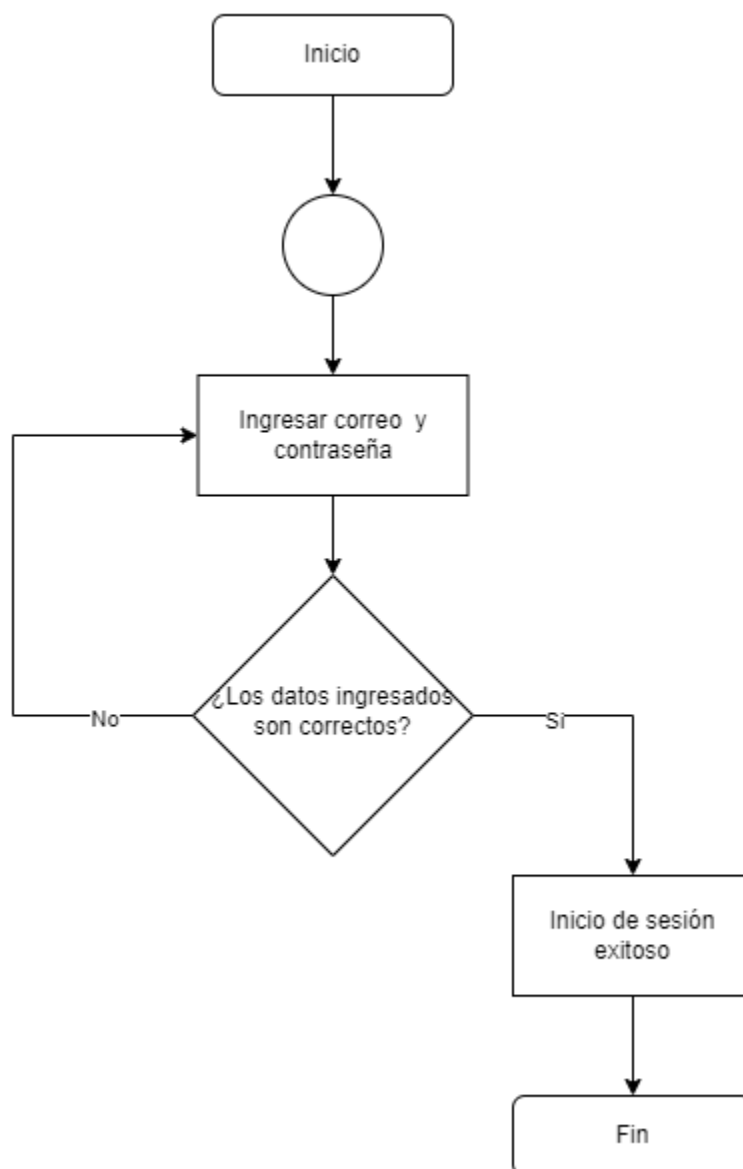
Fuente: Elaboración propia, 2024

Diseño de Procesos

En esta sección, se describe el diseño de procesos que conforman el sistema de recursos humanos. Se destacan los flujos de trabajo que son la clave para la interacción entre los módulos del sistema. El objetivo de estos diagramas es garantizar que las operaciones del sistema sean eficientes.

Figura 28:

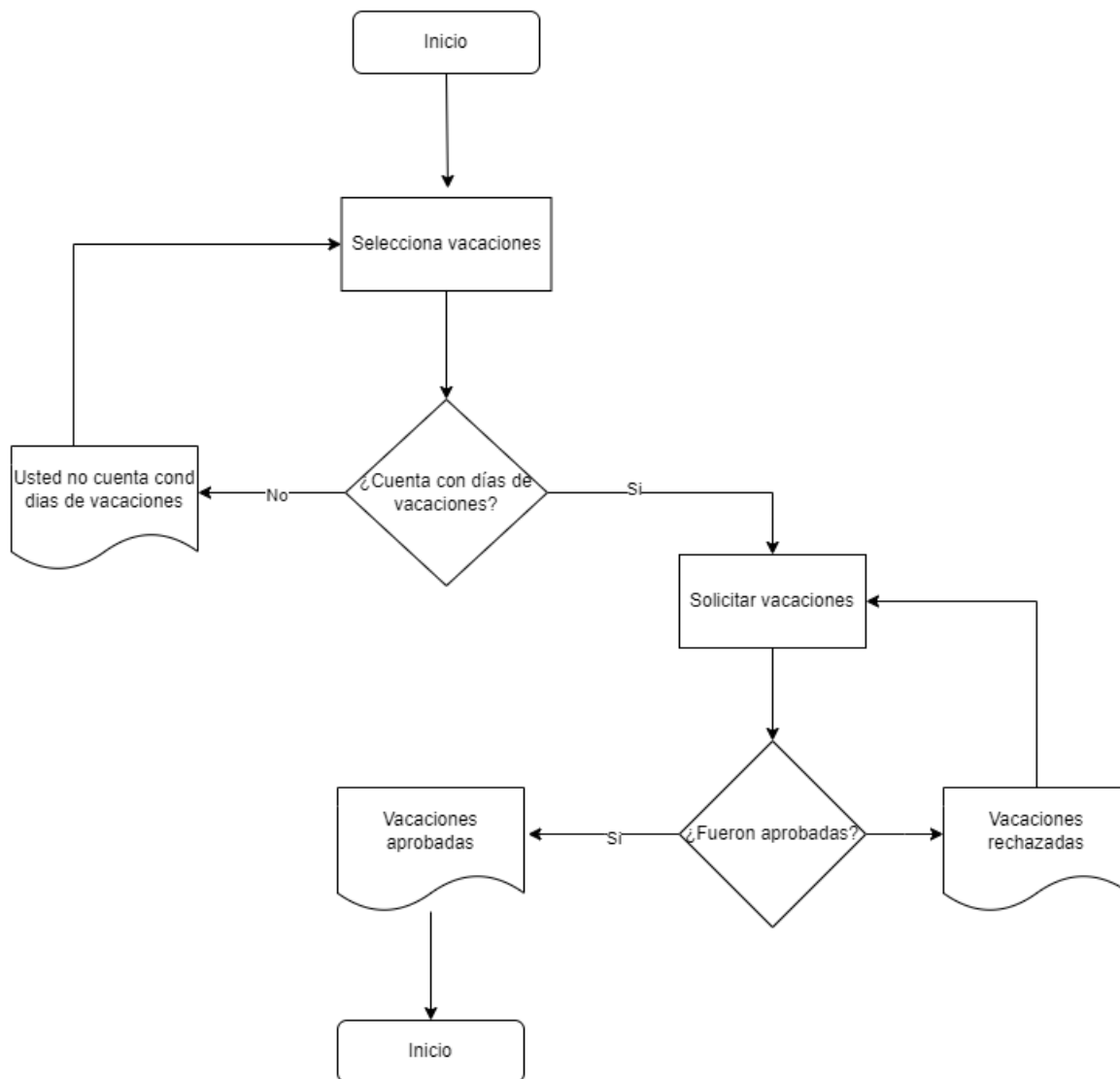
Diagramas de flujos inicio de sesión



Fuente: Elaboración propia, 2024.

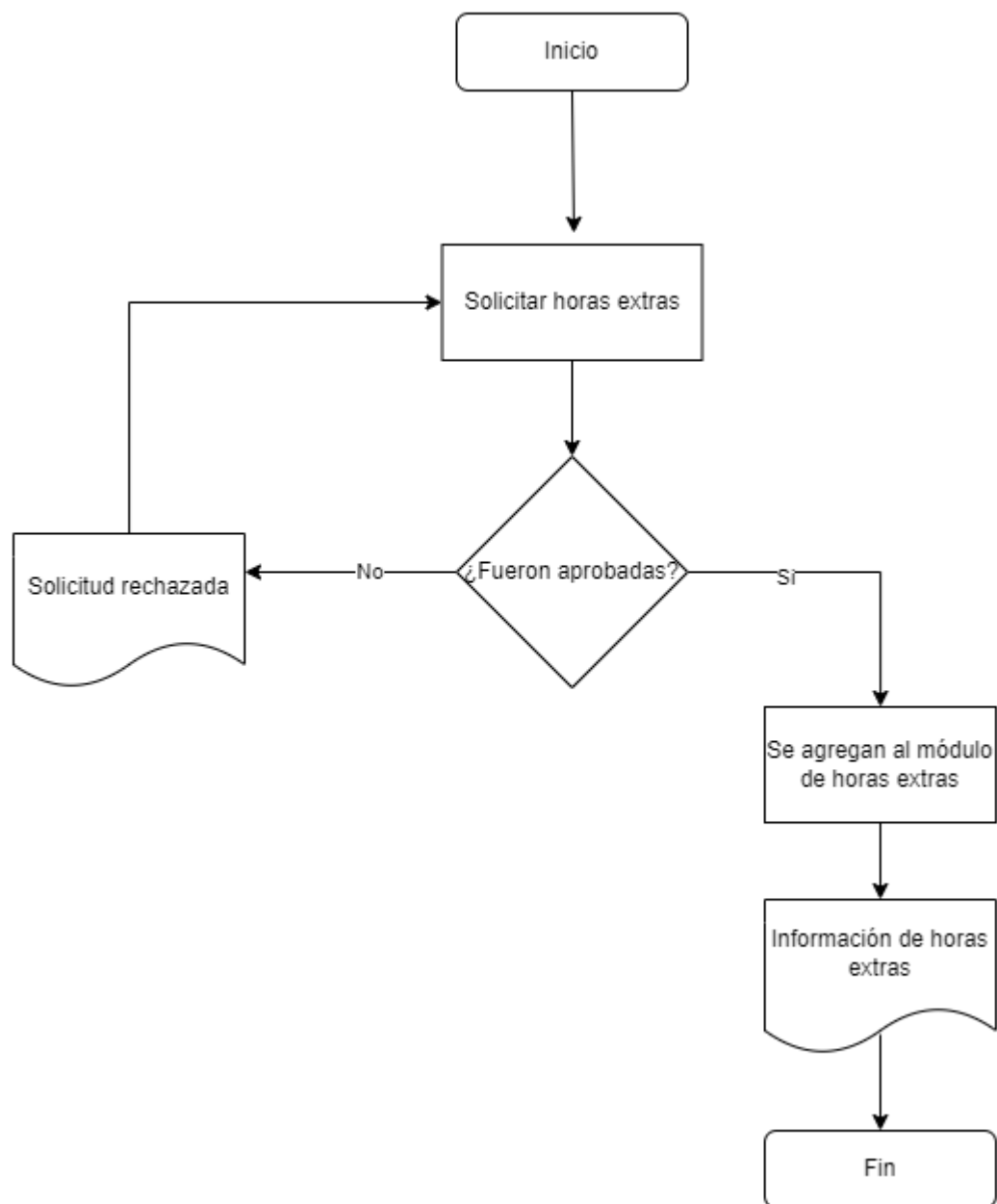
Figura 29:

Diagrama de flujo de vacaciones



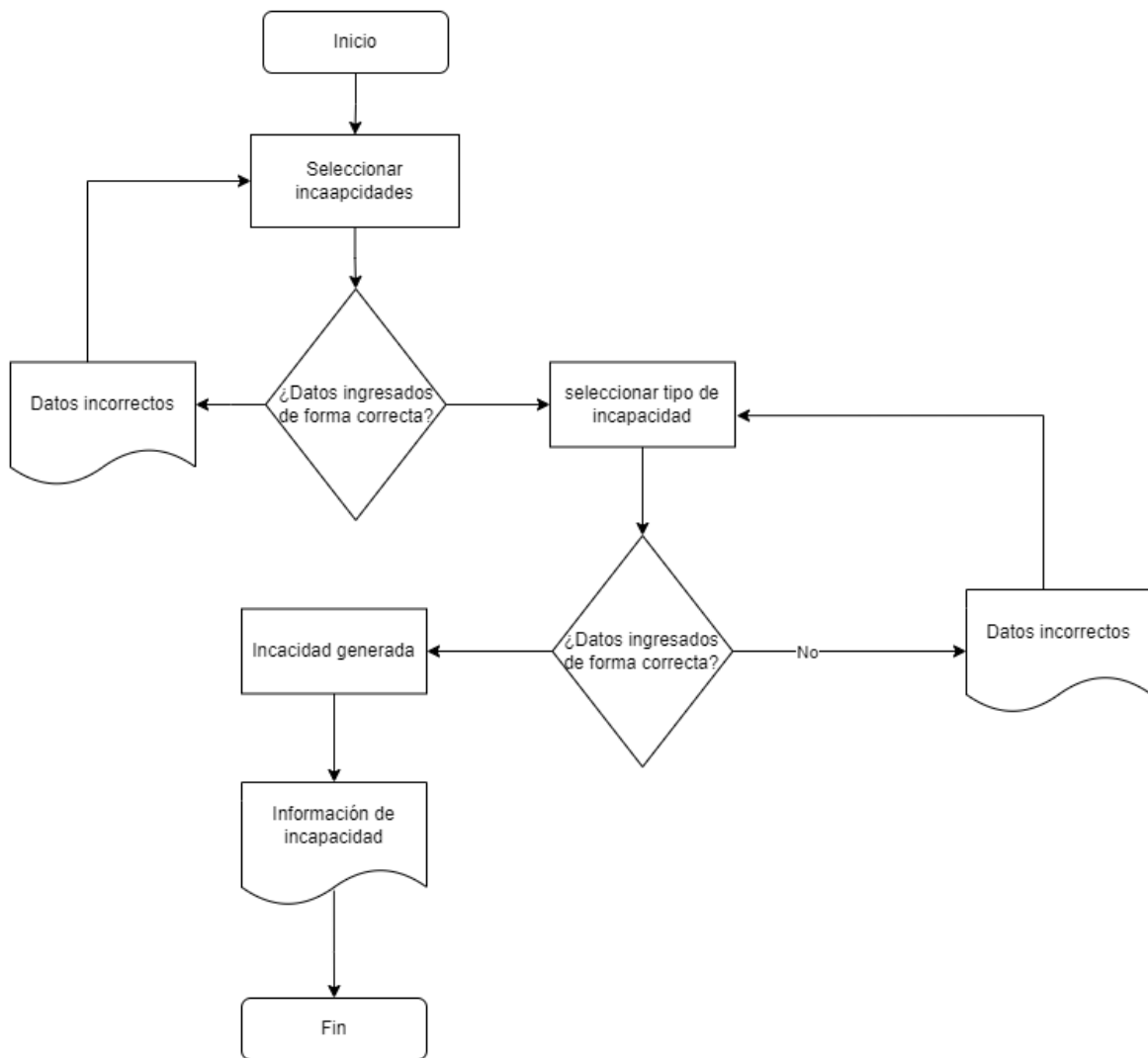
Fuente: Elaboración propia, 2024.

Figura 30:
Diagrama de flujos de horas extras.



Fuente: Elaboración propia, 2024.

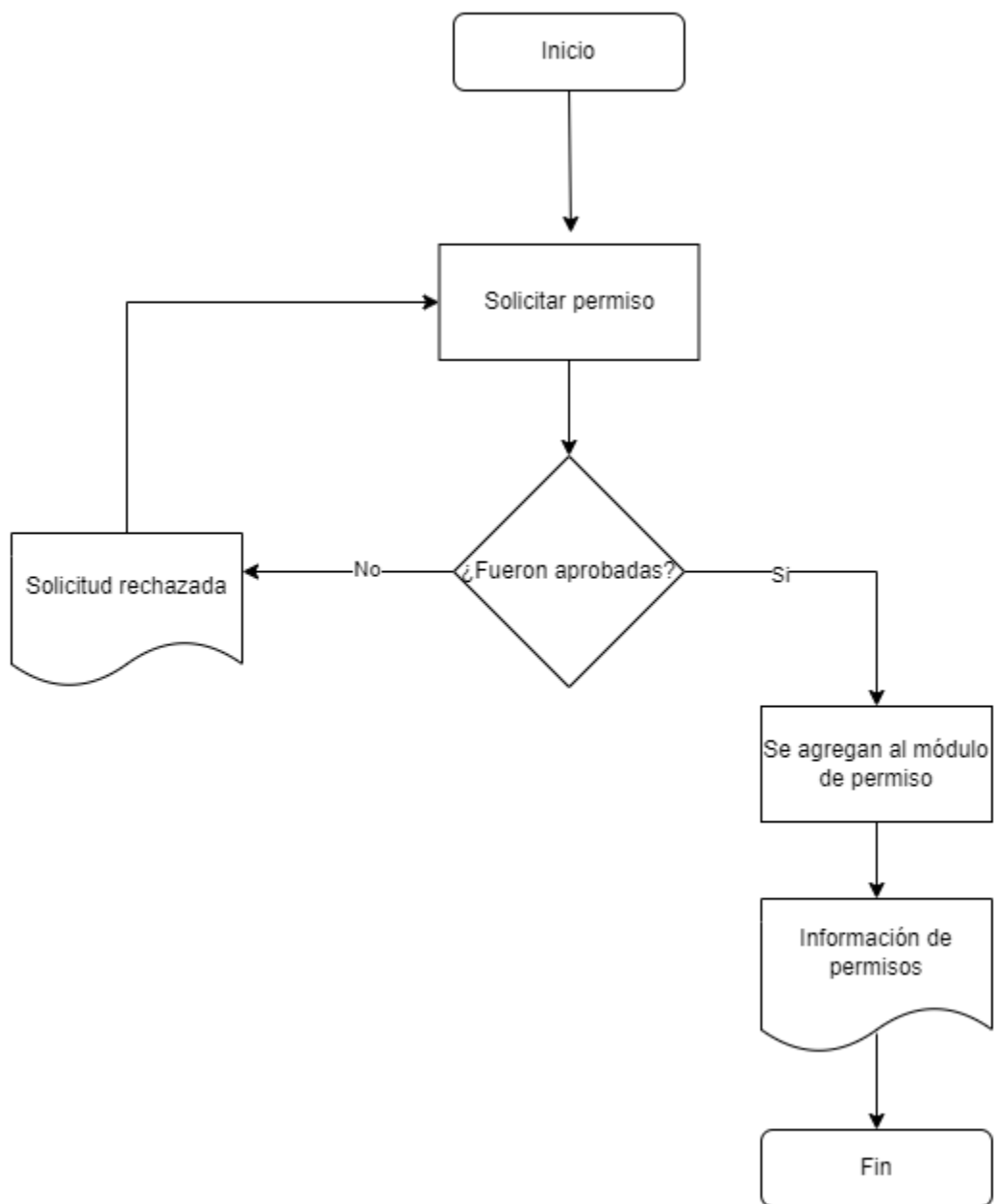
Figura 31:
Diagrama de flujos de incapacidad



Fuente: Elaboración propia, 2024.

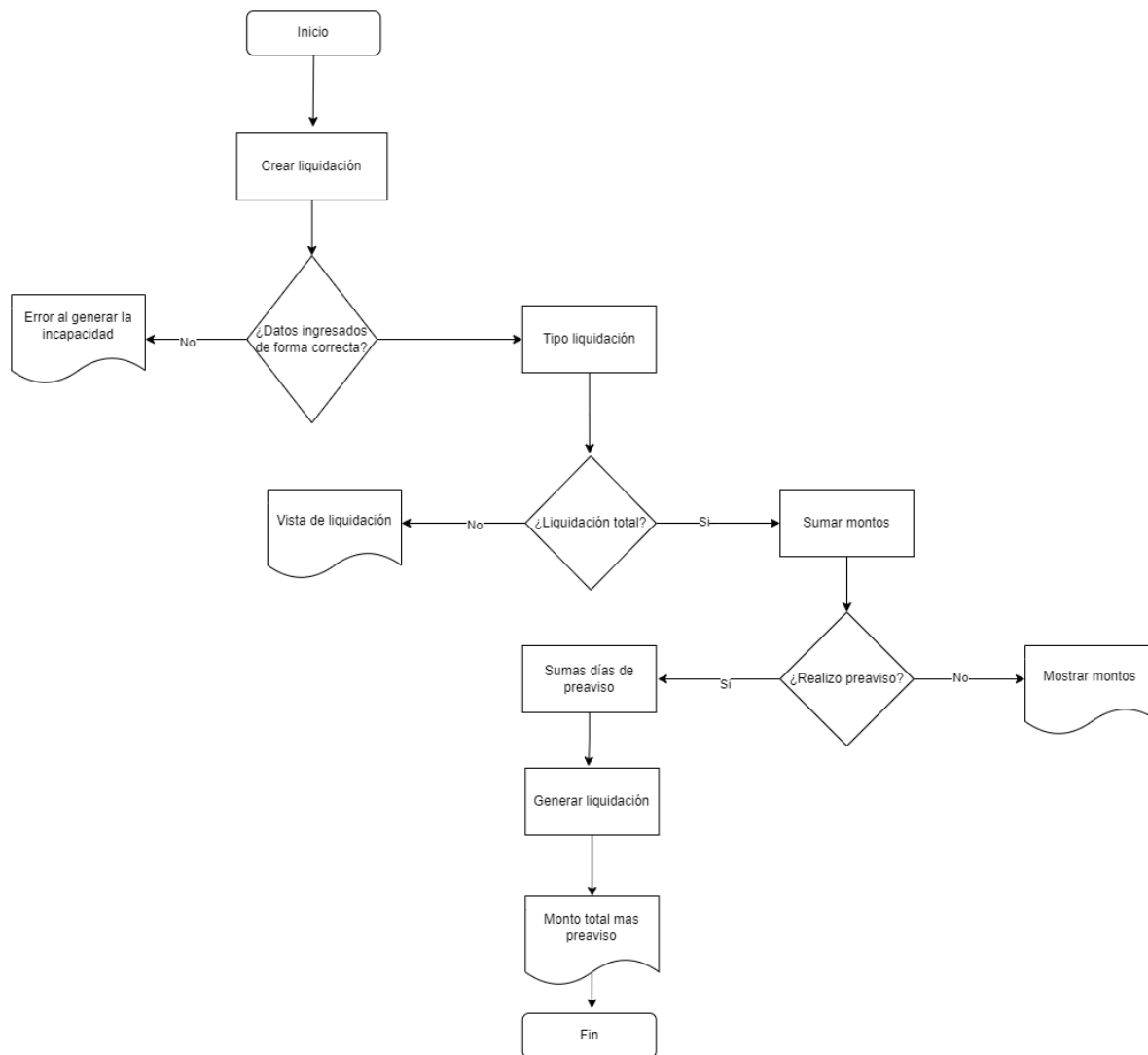
Figura 32:

Diagrama de flujo de permisos del colaborador.



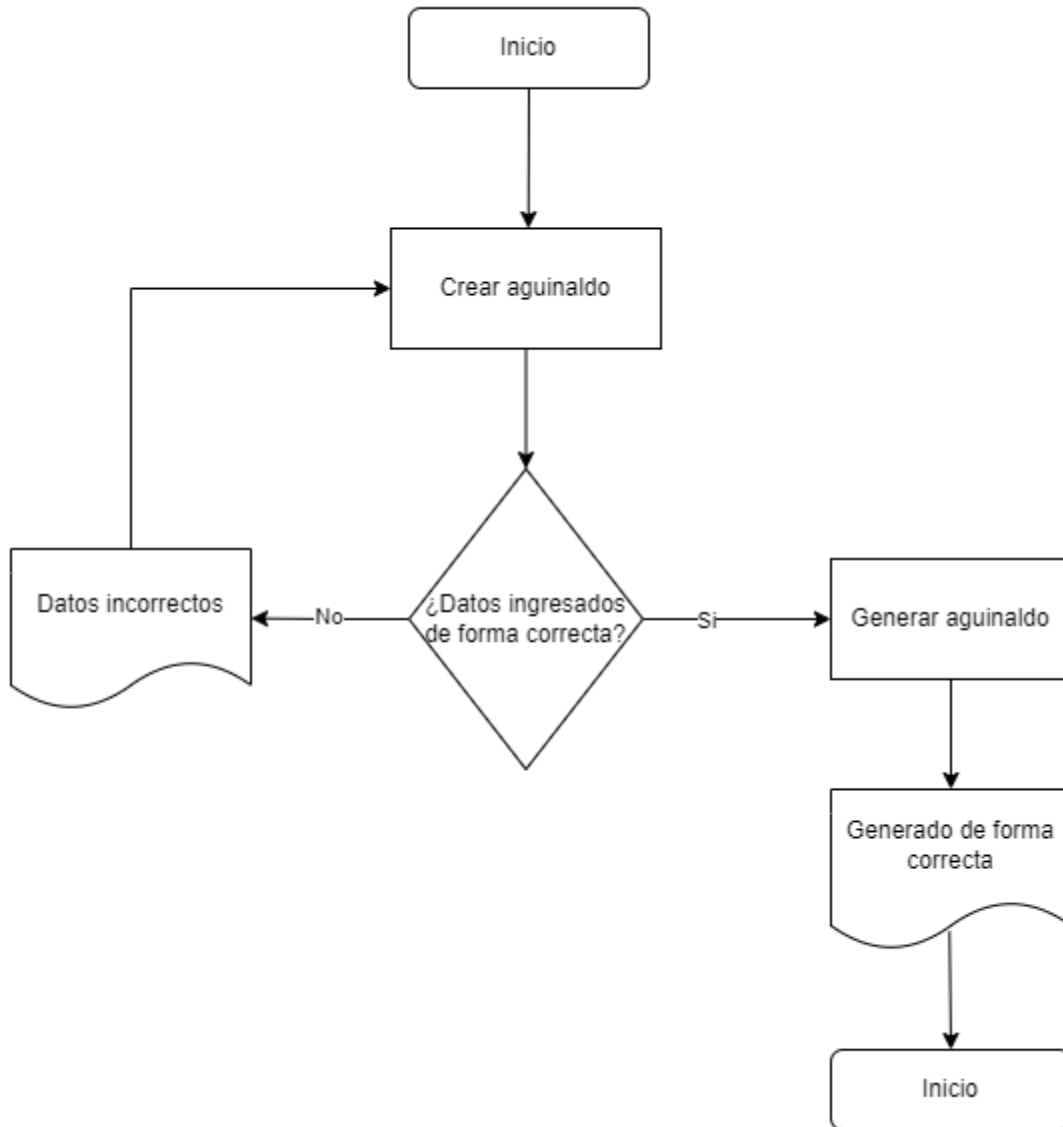
Fuente: Elaboración propia, 2024.

Figura 33:
Diagrama de flujo de liquidación.



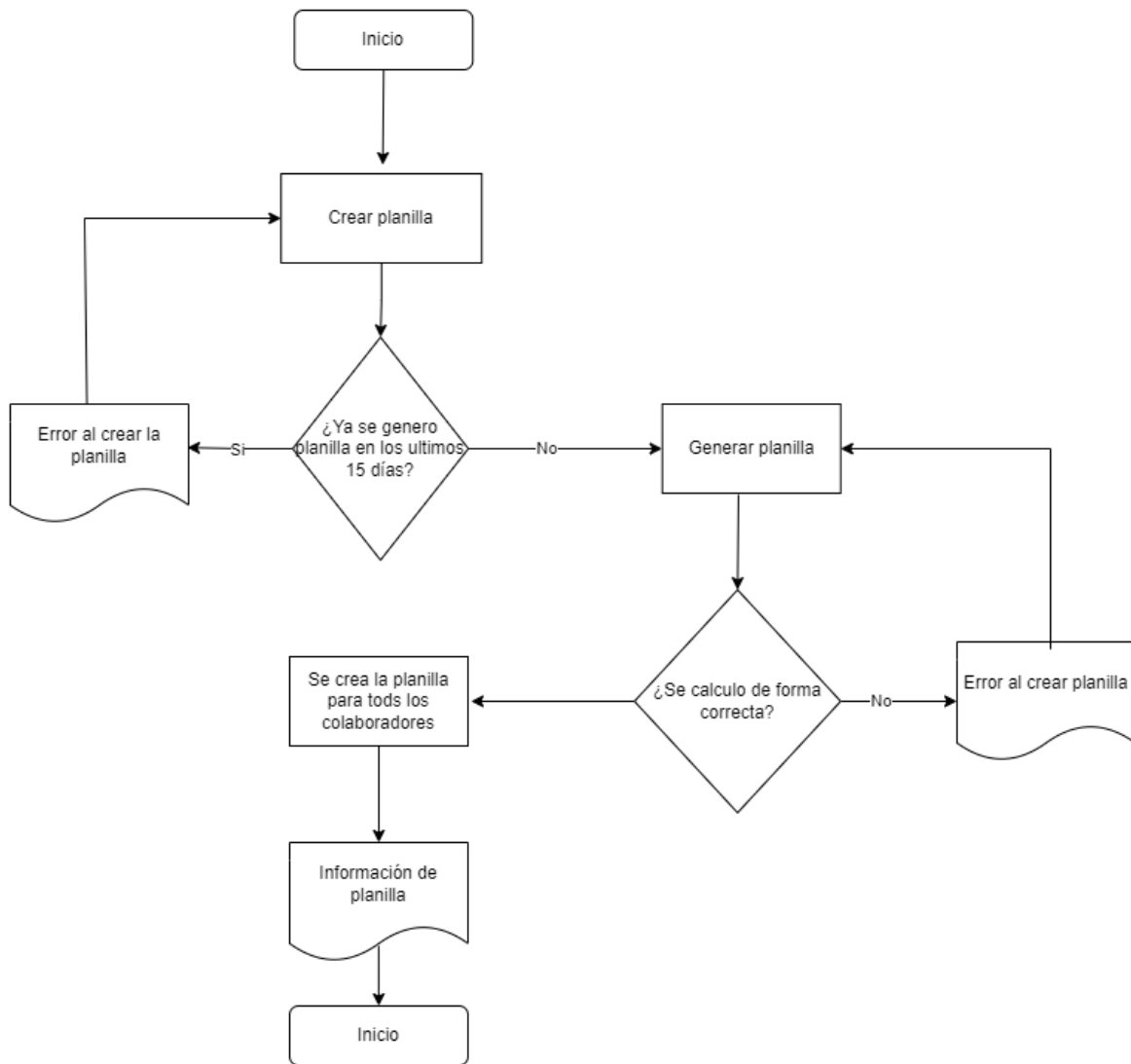
Fuente: Elaboración propia, 2024.

Figura 34:
Diagrama de flujo de aguinaldo.



Fuente: Elaboración propia, 2024.

Figura 35:
Diagrama de flujo de planilla



Fuente: Elaboración propia, 2024.

Diseño de Salidas

En esta sección, se presenta las diferentes salidas generadas por el sistema, proporciona información relevante para la gestión de recursos humanos. Además, brinda una información detallada y organizada del sistema que ayuda a la toma de decisiones de la jefatura.

Figura 36:

Diseño de salidas para horarios.

ID DE HORARIO	HORA DE ENTRADA	HORA DE SALIDA	EMPLEADO	DÍA	TIPO HORARIO	ACCIONES
1	07:00 a. m.	06:00 p. m.	Deyvin Polanco	Jueves	Diurno	Editar Detalles Eliminar

Mostrando página 1 de 1

Anterior 1 Siguiente

Fuente: Elaboración propia, 2024.

Se muestra la creación de horarios de manera exitosa, asignando un horario para un colaborador.

Figura 37:

Diseño de salidas de planilla

EMPLEADO	FECHA DE INICIO	FECHA DE FIN	FECHA EMITIDA	TOTAL HORAS EXTRAS	TOTAL DEDUCCIONES	MONTO TOTAL	MONTO INCAPACIDAD	DESCRIPCIÓN DEDUCCIONES	ACCIONES
1	01/06/2024	15/06/2024	19/06/2024	0	3500	31500	0	Rebajo de seguro	Editar Detalles Eliminar
2	01/06/2024	15/06/2024	19/06/2024	0	4300	38700	0	Rebajo de seguro	Editar Detalles Eliminar

Mostrando página 1 de 1

Anterior 1 Siguiente

Fuente: Elaboración propia, 2024.

Figura 38:*Diseño de salidas de incapacidades*

ID DEL COLABORADOR	TIPO DE INCAPACIDAD	FECHA DE INICIO	FECHA DE FIN	MONTO	ACCIONES
Deyvin Polanco	CCSS	18/06/2024	21/06/2024	12000	Editar Detalles Eliminar

Fuente: Elaboración propia, 2024.

Figura 39:*Diseño de salida para vacaciones.*

Empleado	Cantidad de Días	Estado	Fecha de Inicio	Fecha de Fin	Acción
2	4	Aprobado	14/06/2024	20/06/2024	Aprobar
1	4	Aprobado	17/06/2024	21/06/2024	Aprobar

Fuente: Elaboración propia, 2024.

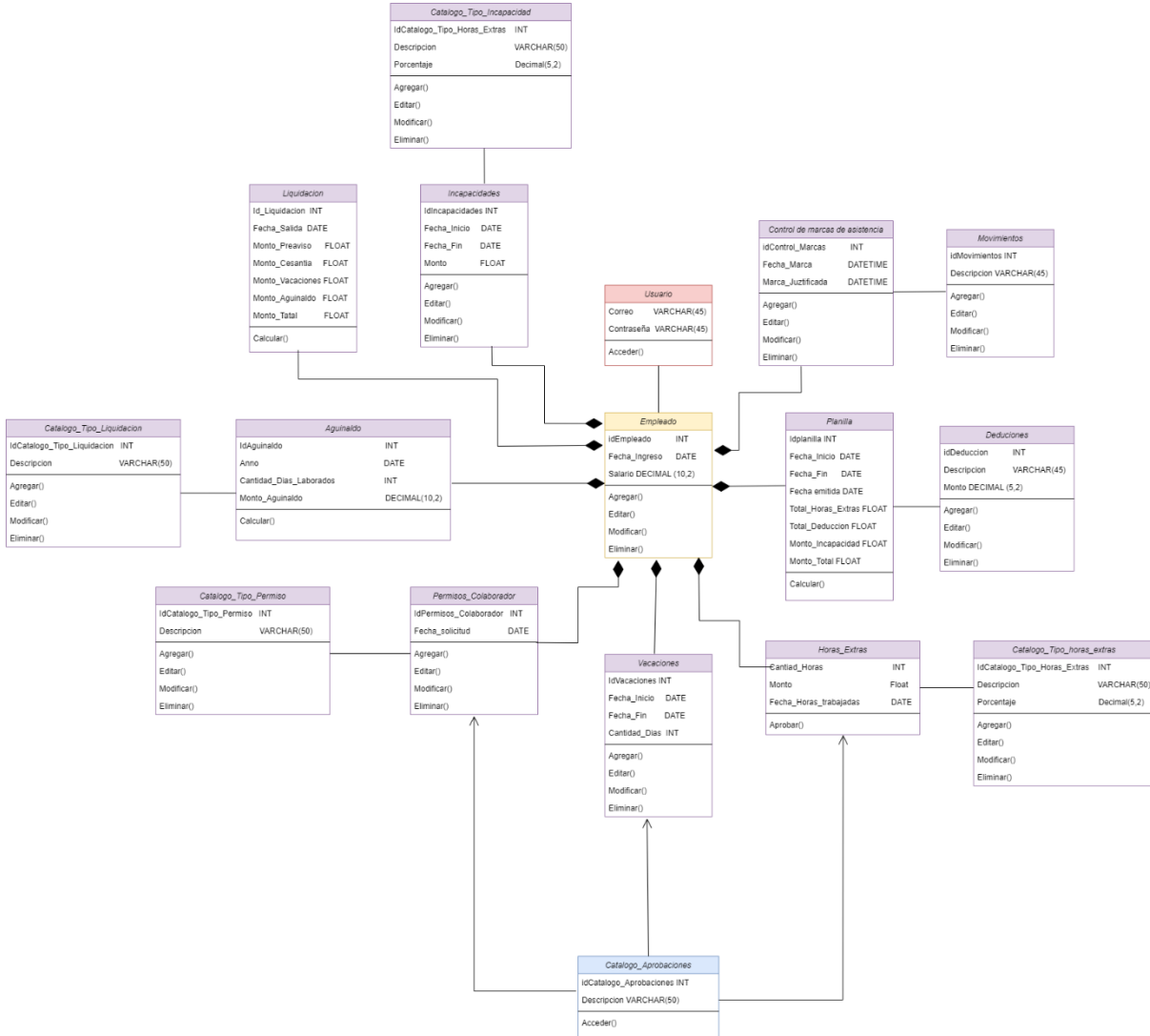
Figura 40:*Diseño de salida de solicitudes*

Comentario	Estado	Fecha de Solicitud	Empleado	Acciones
Solicitud de extras	En espera	18/06/2024	1	Aprobar

Fuente: Elaboración propia, 2024.

Diagrama de Clases

Figura 41:
Diagrama de clases.



Fuente: Elaboración propia, 2024.

Programación

Dentro del ámbito de este sistema, se muestran fragmentos detallados tanto en las entradas como en las salidas. Las entradas se refieren a procesos de inserción de datos, donde se capturan y procesan los datos ingresados por el usuario. Por otro lado, las salidas muestran fragmentos de código que representan resultados o respuestas del sistema. Estas salidas pueden incluir datos procesados, visualización de resultados, generación de informes o cualquier otro tipo de interacción que el sistema tenga con el usuarios.

Entradas

Figura 42:

Entrada del módulo de planilla

```

@using (Html.BeginForm())
{
    @Html.AntiForgeryToken()

    <div class="container">
        <div class="row justify-content-center">
            <div class="col-md-6">
                <div class="card">
                    <div class="card-header">
                        <h4 class="text-center"><i class="fas fa-plus"></i> Crear planilla</h4>
                    </div>
                    <div class="card-body">
                        @Html.ValidationSummary(true, "", new { @class = "text-danger" })

                        <div class="form-group">
                            <label for="Fecha_Inicio" class="control-label"><i class="fas fa-calendar-alt"></i> Fecha de inicio</label>
                            @Html.EditorFor(model => model.Fecha_Inicio, new { htmlAttributes = new { @class = "form-control", id = "FechaInicio" } })
                            @Html.ValidationMessageFor(model => model.Fecha_Inicio, "", new { @class = "text-danger" })
                        </div>

                        <div class="form-group">
                            <label for="Fecha_Fin" class="control-label"><i class="fas fa-calendar-alt"></i> Fecha fin</label>
                            @Html.EditorFor(model => model.Fecha_Fin, new { htmlAttributes = new { @class = "form-control", id = "FechaFin" } })
                            @Html.ValidationMessageFor(model => model.Fecha_Fin, "", new { @class = "text-danger" })
                        </div>

                        <div class="form-group text-center">
                            <button type="submit" class="btn btn-primary">
                                <i class="fas fa-save"></i> Crear
                            </button>

                            @Html.ActionLink("Volver", "Index", null, new { @class = "btn btn-secondary" })
                        </div>
                    </div>
                </div>
            </div>
        </div>
    </div>
}

```

Fuente: Elaboración propia, 2024.

En la figura 42, se aprecia la entrada de planilla donde se realiza la inserción de la fecha de inicio y fin del periodo de la planilla, esto con el fin de establecer las fechas; debido a que, muchas veces, las planillas se terminan antes de que culmine la quincena, de esta manera, se puede realizar de forma correcta.

Figura 43:
Entrada del módulo de aguinaldo

```

<div class="form-horizontal">
  <h4>aguinaldo</h4>
  <hr />
  @Html.ValidationSummary(true, "", new { @class = "text-danger" })
  <div class="form-group">
    @Html.LabelFor(model => model.idAguinaldo, htmlAttributes: new { @class = "control-label col-md-2" })
    <div class="col-md-10">
      @Html.EditorFor(model => model.idAguinaldo, new { htmlAttributes = new { @class = "form-control" } })
      @Html.ValidationMessageFor(model => model.idAguinaldo, "", new { @class = "text-danger" })
    </div>
  </div>

  <div class="form-group">
    @Html.LabelFor(model => model.Anno, htmlAttributes: new { @class = "control-label col-md-2" })
    <div class="col-md-10">
      @Html.EditorFor(model => model.Anno, new { htmlAttributes = new { @class = "form-control" } })
      @Html.ValidationMessageFor(model => model.Anno, "", new { @class = "text-danger" })
    </div>
  </div>

  <div class="form-group">
    @Html.LabelFor(model => model.Monto_Aguinaldo, htmlAttributes: new { @class = "control-label col-md-2" })
    <div class="col-md-10">
      @Html.EditorFor(model => model.Monto_Aguinaldo, new { htmlAttributes = new { @class = "form-control" } })
      @Html.ValidationMessageFor(model => model.Monto_Aguinaldo, "", new { @class = "text-danger" })
    </div>
  </div>

  <div class="form-group">
    @Html.LabelFor(model => model.Empleado_idEmpleado, htmlAttributes: new { @class = "control-label col-md-2" })
    <div class="col-md-10">
      @Html.EditorFor(model => model.Empleado_idEmpleado, new { htmlAttributes = new { @class = "form-control" } })
      @Html.ValidationMessageFor(model => model.Empleado_idEmpleado, "", new { @class = "text-danger" })
    </div>
  </div>

  <div class="form-group">
    <div class="col-md-offset-2 col-md-10">
      <input type="submit" value="Create" class="btn btn-default" />
    </div>
  </div>
</div>

```

Fuente: Elaboración propia, 2024.

En la figura 43, se muestran las entradas del módulo de aguinaldo. Se toman todos los pagos del salario bruto en planilla y se dividen entre 12 para realizar el cálculo correcto de aguinaldo.

Figura 44:
Entrada del módulo de vacaciones

```
2 @using (Html.BeginForm())
3
4 @Html.AntiForgeryToken()
5
6 <div class="container">
7   <div class="row justify-content-center">
8     <div class="col-md-6">
9       <div class="card">
10        <div class="card-header">
11          <h4 class="text-center"><i class="fas fa-plus"></i> Crear vacaciones</h4>
12        </div>
13        <div class="card-body">
14          @Html.ValidationSummary(true, "", new { @class = "text-danger" })
15
16          <div class="form-group">
17            <label for="Fecha_Inicio" class="control-label"><i class="fas fa-calendar"></i> Fecha de inicio</label>
18            @Html.EditorFor(model => model.Fecha_Inicio, new { htmlAttributes = new { @class = "form-control", id = "FechaInicio" } })
19            @Html.ValidationMessageFor(model => model.Fecha_Inicio, "", new { @class = "text-danger" })
20          </div>
21
22          <div class="form-group">
23            <label for="Fecha_Fin" class="control-label"><i class="fas fa-calendar"></i> Fecha fin</label>
24            @Html.EditorFor(model => model.Fecha_Fin, new { htmlAttributes = new { @class = "form-control", id = "FechaFin" } })
25            @Html.ValidationMessageFor(model => model.Fecha_Fin, "", new { @class = "text-danger" })
26          </div>
27
28          <div class="form-group text-center">
29            <button type="submit" class="btn btn-primary">
30              <i class="fas fa-save"></i> Crear
31            </button>
32
33            @Html.ActionLink("Volver", "IndexPersonal", null, new { @class = "btn btn-secondary" })
34          </div>
35        </div>
36      </div>
37    </div>
38  </div>
39</div>
```

Fuente: Elaboración propia, 2024.

En el módulo de cálculo de vacaciones, el colaborador inserta la fecha de inicio y la fecha fin en la cual él desea solicitar las vacaciones y luego hace una solicitud a jefatura directa para que se puedan aprobar o rechazar.

Figura 45:
Entrada del módulo de horas extras

```

<div class="card">
  <div class="card-header">
    <h2 class="card-title"><i class="fas fa-clock"></i> Solicitud de horas extras</h2>
  </div>
  <div class="card-body">
    <using (Html.BeginForm())
    {
      <Html.AntiForgeryToken()
      <hr />
      <Html.ValidationSummary(true, "", new { @class = "text-danger" })
      <div class="form-group">
        <Html.LabelFor(model => model.Fecha_solicitud, "Fecha de solicitud", htmlAttributes: new { @class = "control-label" })
        <div>
          <div class="input-group">
            <div class="input-group-prepend">
              <span class="input-group-text"><i class="fas fa-calendar-alt"></i></span>
            </div>
            <div>
              <Html.EditorFor(model => model.Fecha_solicitud, new { htmlAttributes = new { @class = "form-control", id = "FechaSolicitud", min = DateTime.Today } })
              <Html.ValidationMessageFor(model => model.Fecha_solicitud, "", new { @class = "text-danger" })
            </div>
          </div>
        </div>
      </div>
      <div class="form-group">
        <Html.LabelFor(model => model.Comentario, htmlAttributes: new { @class = "control-label" })
        <div>
          <div class="input-group">
            <div class="input-group-prepend">
              <span class="input-group-text"><i class="fas fa-comment"></i></span>
            </div>
            <div>
              <Html.EditorFor(model => model.Comentario, new { htmlAttributes = new { @class = "form-control" } })
              <Html.ValidationMessageFor(model => model.Comentario, "", new { @class = "text-danger" })
            </div>
          </div>
        </div>
      </div>
      <br />
      <div class="form-group">
        <div>
          <input type="submit" value="Create" class="btn btn-success" />
        </div>
      </div>
    }
  </div>
</div>

```

Fuente: Elaboración propia, 2024.

En la figura 45, se aprecia las entradas de horas extras donde el empleado solicita la fecha en las que realizará las horas extras y un comentario. Esta solicitud llegará a jefatura directa para ser aprobada o rechazada.

Figura 46:
Entrada del módulo de permisos del colaborador.

```

<div class="col-md-6">
  <div class="card">
    <div class="card-header">
      <h4 class="text-center">Solicitud de permiso</h4>
    </div>
    <div class="card-body">
      @using (Html.BeginForm())
      {
        @Html.AntiForgeryToken()

        @Html.ValidationSummary(true, "", new { @class = "text-danger" })

        <div class="form-group">
          @Html.LabelFor(model => model.Fecha_Solicitud, "Fecha de solicitud", htmlAttributes: new { @class = "control-label" })
          <div>
            @Html.EditorFor(model => model.Fecha_Solicitud, new { htmlAttributes = new { @class = "form-control", id = "FechaSolicitud" } })
            @Html.ValidationMessageFor(model => model.Fecha_Solicitud, "", new { @class = "text-danger" })
          </div>
        </div>

        <div class="form-group">
          @Html.LabelFor(model => model.Cantidad_horas, "Cantidad de horas", htmlAttributes: new { @class = "control-label" })
          <div>
            @Html.EditorFor(model => model.Cantidad_horas, new { htmlAttributes = new { @class = "form-control" } })
            @Html.ValidationMessageFor(model => model.Cantidad_horas, "", new { @class = "text-danger" })
          </div>
        </div>

        <div class="form-group">
          @Html.LabelFor(model => model.Asunto, "Asunto", htmlAttributes: new { @class = "control-label" })
          <div>
            @Html.EditorFor(model => model.Asunto, new { htmlAttributes = new { @class = "form-control" } })
            @Html.ValidationMessageFor(model => model.Asunto, "", new { @class = "text-danger" })
          </div>
        </div>

        <div class="form-group">
          <div class="text-center">
            <input type="submit" value="Create" class="btn btn-primary" />
            <a href="@Url.Action("IndexPersonal")" class="btn btn-secondary">Volver</a>
          </div>
        </div>
      }
    </div>
  </div>

```

Fuente: Elaboración propia, 2024.

En la figura 46, se muestran las entradas del módulo de permisos, donde se solicita la fecha de la solicitud del permiso, la cantidad de horas y el asunto. Luego de esto se envía a jefatura donde se definirá si el permiso es con goce de salario y si este se aprueba o rechaza.

Figura 47:
Entrada del módulo de liquidación

```

<div class="row justify-content-center">
  <div class="col-md-6">
    <div class="card">
      <div class="card-header">
        <h4 class="text-center"><i class="fas fa-plus"></i> Crear liquidación</h4>
      </div>
      <div class="card-body">
        @using (Html.BeginForm())
        {
          @Html.AntiForgeryToken()

          @Html.ValidationSummary(true, "", new { @class = "text-danger" })

          <div class="form-group">
            @Html.LabelFor(model => model.Fecha_Salida, "Fecha de salida", htmlAttributes: new { @class = "control-label" })
            @Html.TextBoxFor(model => model.Fecha_Salida, "{0:yyyy-MM-dd}", new { @class = "form-control", type = "date", id = "FechaSalida", placeholder = "Fecha de salida" })
            @Html.ValidationMessageFor(model => model.Fecha_Salida, "", new { @class = "text-danger" })
          </div>

          <div class="form-group">
            @Html.LabelFor(model => model.Dias_preaviso, "Días de preaviso", htmlAttributes: new { @class = "control-label" })
            @Html.EditorFor(model => model.Dias_preaviso, new { htmlAttributes = new { @class = "form-control", placeholder = "Días de preaviso" } })
            @Html.ValidationMessageFor(model => model.Dias_preaviso, "", new { @class = "text-danger" })
          </div>

          <div class="form-group">
            @Html.LabelFor(model => model.Realizo_preaviso, "¿Realizó preaviso?", htmlAttributes: new { @class = "control-label" })
            @Html.DropDownList("HizoPreaviso", new List<SelectListItem>
            {
              new SelectListItem { Text = "Sí", Value = "1" },
              new SelectListItem { Text = "No", Value = "0" }
            }, "Selecciones", new { @class = "form-control" })
          </div>

          <div class="form-group">
            @Html.LabelFor(model => model.Catalogo_Tipo_Liquidacion_idCatalogo_Tipo_Liquidacion, "Tipo de liquidación", htmlAttributes: new { @class = "control-label" })
            @Html.DropDownListFor(model => model.Catalogo_Tipo_Liquidacion_idCatalogo_Tipo_Liquidacion, (SelectList)ViewBag.Catalogo_Tipo_Liquidacion_idCatalogo_Tipo_Liquidacion, "Selecione u")
            @Html.ValidationMessageFor(model => model.Catalogo_Tipo_Liquidacion_idCatalogo_Tipo_Liquidacion, "", new { @class = "text-danger" })
          </div>
        }
      </div>
    </div>
  </div>
</div>

```

Fuente: Elaboración propia, 2024.

En la figura 47, se muestran las entradas de la tabla de liquidación donde se ingresan la fecha de salida del colaborador, si realizó preaviso, cantidad de preaviso, tipo de liquidación y al empleado al cual le está aplicando la liquidación.

Figura 48:
Entrada del módulo de incapacidades

```

<div class="card-body">
    @using (Html.BeginForm())
    @Html.AntiForgeryToken()
    @Html.ValidationSummary(true, "", new { @class = "text-danger" })
    <div class="form-group">
        <label for="Fecha_Inicio" class="control-label"><i class="fas fa-calendar"></i> Fecha de inicio</label>
        @Html.EditorFor(model => model.Fecha_Inicio, new { htmlAttributes = new { @class = "form-control", id = "FechaSolicitud" } })
        @Html.ValidationMessageFor(model => model.Fecha_Inicio, "", new { @class = "text-danger" })
    </div>
    <div class="form-group">
        <label for="Fecha_Fin" class="control-label"><i class="fas fa-calendar"></i> Fecha fin</label>
        @Html.EditorFor(model => model.Fecha_Fin, new { htmlAttributes = new { @class = "form-control", id = "FechaFin" } })
        @Html.ValidationMessageFor(model => model.Fecha_Fin, "", new { @class = "text-danger" })
    </div>
    <div class="form-group">
        <label for="Empleado_idEmpleado" class="control-label"><i class="fas fa-user"></i> ID de Empleado</label>
        @Html.DropDownList("Empleado_idEmpleado", null, htmlAttributes: new { @class = "form-control" })
        @Html.ValidationMessageFor(model => model.Empleado_idEmpleado, "", new { @class = "text-danger" })
    </div>
    <div class="form-group">
        <label for="Catalogo_Tipo_Incapacidad_idTipo_Incapacidad" class="control-label"><i class="fas fa-file-medical"></i> Tipo de Incapacidad</label>
        @Html.DropDownList("Catalogo_Tipo_Incapacidad_idTipo_Incapacidad", null, htmlAttributes: new { @class = "form-control" })
        @Html.ValidationMessageFor(model => model.Catalogo_Tipo_Incapacidad_idTipo_Incapacidad, "", new { @class = "text-danger" })
    </div>
    <div class="form-group text-center">
        <button type="submit" class="btn btn-primary">
            <i class="fas fa-save"></i> Crear
        </button>
        <a href="@Url.Action("Index")" class="btn btn-secondary">
            <i class="fas fa-arrow-left"></i> Volver
        </a>
    </div>
</div>

```

Fuente: Elaboración propia, 2024.

En la figura 48, se aprecian las entradas de incapacidades, donde se inserta la fecha inicio, fecha de fin, el id del colaborador y el tipo de incapacidad. Estos datos permiten realizar la inserción correcta de la incapacidad.

Salidas

Figura 49:

Salida del módulo de planilla

```

</div>
<div class="card-body">
  <@if (Model.Any())>
    <table id="productoTable" class="table table-bordered">
      <thead>
        <tr>
          <th>Empleado</th>
          <th>Fecha de inicio</th>
          <th>Fecha fin</th>
          <th>Fecha emitida</th>
          <th>Monto horas extras</th>
          <th>Monto deducciones</th>
          <th>Monto incapacidades</th>
          <th>Tipo deducción</th>
          <th>Impuesto de renta</th>
          <th>Salario bruto</th>
          <th>Monto total</th>
        </tr>
      </thead>
      <tbody>
        <@foreach (var item in Model)>
          <tr>
            <td>@Html.DisplayFor(modelItem => item.empleado.persona.Nombre) @Html.DisplayFor(modelItem => item.empleado.persona.Apellido_1)</td>
            <td>@item.planilla.Fecha_Inicio.ToString("dd/MM/yyyy")</td>
            <td>@item.planilla.Fecha_Fin.ToString("dd/MM/yyyy")</td>
            <td>@item.planilla.Fecha_Emitida.ToString("dd/MM/yyyy")</td>
            <td>@Html.DisplayFor(modelItem => item.planilla.Total_Horas_Extras)</td>
            <td>@Html.DisplayFor(modelItem => item.planilla.Total_Deducciones)</td>
            <td>@Html.DisplayFor(modelItem => item.planilla.Monto_Incapacidad)</td>
            <td>@Html.DisplayFor(modelItem => item.planilla.deducciones.Descripcion)</td>
            <td>@Html.DisplayFor(modelItem => item.planilla.ImpuestoRenta)</td>
            <td>@Html.DisplayFor(modelItem => item.planilla.Salario_bruto)</td>
            <td>@Html.DisplayFor(modelItem => item.planilla.Monto_Total)</td>
          </tr>
        </tbody>
      </table>
    </div>
  </div>

```

Fuente: Elaboración propia, 2024.

En la figura 49, se detalla la planilla mediante un *viewmodel* que permite mostrar toda la información de planilla, como empleado, fecha inicio, fecha fin, fecha emitida, monto horas extras, monto deducciones, monto incapacidad, tipo de deducción, impuesto de renta, salario bruto y monto total.

Figura 52:
Salida del módulo de horas extras

```

<th>Comentario</th>
<th>Estado</th>
<th>Fecha de Solicitud</th>
<th>Empleado</th>
<th>Acciones</th>
</thead>
<tbody>
<tr>
<td>{{item.solicitud_horas_extras.Comentario}}</td>
<td>
<div style="display: flex; justify-content: space-between; align-items: center;">
<div style="display: flex; gap: 5px;">
<div style="border: 1px solid #ccc; padding: 2px 5px; border-radius: 3px;">
{{item.solicitud_horas_extras.estados_solicitud_extras.Descripcion == "Aprobado"}}
</div>
<div style="border: 1px solid #ccc; padding: 2px 5px; border-radius: 3px;">
{{item.solicitud_horas_extras.estados_solicitud_extras.Descripcion == "Denegado"}}
</div>
<div style="border: 1px solid #ccc; padding: 2px 5px; border-radius: 3px;">
{{item.solicitud_horas_extras.estados_solicitud_extras.Descripcion == "En espera"}}
</div>
</div>
<div style="margin-left: auto; text-align: right;">
<button type="button" class="btn btn-success" style="font-size: 0.8em; padding: 2px 5px;">
{{item.solicitud_horas_extras.estados_solicitud_extras.Descripcion}}
</button>
</div>
</div>
</td>
<td>{{item.solicitud_horas_extras.Fecha_solicitud.ToString("dd/MM/yyyy")}}</td>
<td>{{Html.DisplayFor(modelItem => item.empleado.persona.Nombre)}} {{Html.DisplayFor(modelItem => item.empleado.persona.Apellido_1)}}</td>
<td>
<div style="display: flex; align-items: center; gap: 10px;">
<div style="display: flex; gap: 5px;">
<div style="border: 1px solid #ccc; padding: 2px 5px; border-radius: 3px;">
{{item.solicitud_horas_extras.estados_solicitud_extras.Descripcion}}
</div>
</div>
<div style="margin-left: auto; text-align: right;">
<button type="button" class="btn btn-success" style="font-size: 0.8em; padding: 2px 5px;">
{{item.solicitud_horas_extras.estados_solicitud_extras.Descripcion}}
</button>
</div>
</div>
</td>
</tr>
</tbody>

```

Fuente: Elaboración propia, 2024.

En la figura 52, se pueden apreciar los datos de la solicitud de horas extras, donde se mostrará la solicitud del colaborador y si está aprobado o rechazado por parte de jefatura directa.

Figura 55:
Salida del módulo de incapacidades

```

<div class="col">
  <table id="productoTable" class="table table-bordered table-striped">
    <thead>
      <tr>
        <th>ID del colaborador</th>
        <th>Tipo de Incapacidad</th>
        <th>Fecha de Inicio</th>
        <th>Fecha de Fin</th>
        <th>Monto</th>
        <th>Acciones</th>
      </tr>
    </thead>
    <tbody>
      <tr>
        <td>
          <div class="btn-group" role="group">
            <a href="#">
              <i class="fas fa-edit"></i> Editar
            </a>
            <a href="#">
              <i class="fas fa-info-circle"></i> Detalles
            </a>
            <a href="#">
              <i class="fas fa-trash-alt"></i> Eliminar
            </a>
          </div>
        </td>
      </tr>
    </tbody>
  </table>

```

Fuente: Elaboración propia, 2024.

En la figura 55, se aprecia el módulo de incapacidades donde se detalla todo lo relacionado a la incapacidad de los colaboradores de la empresa.

Procesos

Figura 56:
Proceso del cálculo de planilla.

```

[HttpPost]
[ValidateAntiForgeryToken]
public ActionResult Create(planilla planilla)
{
    try
    {
        // Verificar si ya existe una planilla con las mismas fechas de inicio y fin
        bool planillaExistente = db.planilla.Any(p =>
            p.Fecha_Inicio == planilla.Fecha_Inicio &&
            p.Fecha_Fin == planilla.Fecha_Fin);

        if (planillaExistente)
        {
            TempData["ErrorMessage"] = "Ya existe una planilla con las mismas fechas de inicio y fin.";
            return View(planilla); // Retorna la vista con el mensaje de error
        }

        int idDeducciones = 1;
        var empleados = db.empleado.ToList();

        foreach (var empleado in empleados)
        {
            DataSet ds = new DataSet();
            var usuario = db.usuario.FirstOrDefault(u => u.Empleado_idEmpleado == empleado.idEmpleado);

            // Verifica si el usuario no es null, está activo y no está en liquidación
            if (usuario != null && usuario.Activo)
            {
                bool enLiquidacion = db.liquidacion.Any(l => l.Empleado_idEmpleado == empleado.idEmpleado);

                if (!enLiquidacion)
                {
                    var result = db.SP_Calculo_de_planilla(empleado.Persona_Cedula, idDeducciones, planilla.Fecha_Inicio, planilla.Fecha_Fin);
                }
            }
        }

        // Guardar la nueva planilla solo si no hay errores de validación
        if (ModelState.IsValid)
        {
            db.planilla.Add(planilla);
            db.SaveChanges();
            return RedirectToAction("Index"); // Redirige al Index después de guardar exitosamente
        }
    }
    catch (Exception)
    {
        TempData["ErrorMessage"] = "Verificar la planilla.";
    }

    // En caso de error o ModelState no válido, recargar la vista con el modelo de planilla
    return RedirectToAction("Index");
}

```

Fuente: Elaboración propia, 2024.

Figura 57:
Proceso del cálculo de aguinaldo.

```
[HttpPost]
[ValidateAntiForgeryToken]
0 referencias
public ActionResult CalcularAguinaldo()
{
    try
    {
        var empleados = db.empleado.ToList();

        foreach (var empleado in empleados)
        {
            var usuario = db.usuario.FirstOrDefault(u => u.Empleado_idEmpleado == empleado.idEmpleado);

            if (usuario != null && usuario.Activo)
            {
                // Llamado al procedimiento almacenado SP_Calculo_Aguinaldo
                var result = db.SP_Calculo_Aguinaldo();
            }
        }

        return RedirectToAction("Index");
    }
    catch (Exception)
    {
        Session["Error"] = "No se pudo calcular el aguinaldo";
        return RedirectToAction("Index");
    }
}
```

Fuente: Elaboración propia, 2024.

Figura 58:
Proceso del cálculo de vacaciones.

```
// more details see https://go.microsoft.com/fwlink/?linkid=317598
[HttpPost]
[ValidateAntiForgeryToken]
0 referencias
public ActionResult Create([Bind(Include = "IdVacaciones, Fecha_Inicio, Fecha_Fin, Cantidad_Dias, Empleado_idEmpleado, Catalogo_Aprobaciones_idCatalogo_Aprobaciones")] vacaciones vacaciones)
{
    // Validación para asegurarse de que la fecha de inicio sea estrictamente menor a la fecha de fin
    if (vacaciones.Fecha_Inicio >= vacaciones.Fecha_Fin)
    {
        ModelState.AddModelError("", "La fecha de finalización de las vacaciones no puede ser menor a la fecha de inicio.");
        ViewBag.Catalogo_Aprobaciones_idCatalogo_Aprobaciones = new SelectList(db.catalogo_aprobaciones, "idCatalogo_Aprobaciones", "Descripcion", vacaciones.Catalogo_Aprobaciones_idCatalogo_Aprobaciones);
        return View(vacaciones);
    }

    if (ModelState.IsValid)
    {
        UserInfo user = (UserInfo)Session["UserInfo"];

        // Verificar si hay registros en Control_Marcas con Movimientos_idMovimientos igual a 1
        bool hasAttendance = db.control_marcas_asistencia
            .Any(cm => cm.Empleado_idEmpleado == user.IdEmpleado && cm.Movimientos_idMovimientos == 1);

        if (!hasAttendance)
        {
            TempData["ErrorMessage"] = "No puedes solicitar si no marcas asistencia.";
            ViewBag.Catalogo_Aprobaciones_idCatalogo_Aprobaciones = new SelectList(db.catalogo_aprobaciones, "idCatalogo_Aprobaciones", "Descripcion", vacaciones.Catalogo_Aprobaciones_idCatalogo_Aprobaciones);
            return View(vacaciones);
        }

        // Verificar si ya existe una solicitud de vacaciones en las fechas solicitadas
        bool fechasSolapadas = db.vacaciones.Any(v =>
            v.Empleado_idEmpleado == user.IdEmpleado &&
            ((vacaciones.Fecha_Inicio >= v.Fecha_Inicio && vacaciones.Fecha_Inicio <= v.Fecha_Fin) ||
            (vacaciones.Fecha_Fin >= v.Fecha_Inicio && vacaciones.Fecha_Fin <= v.Fecha_Fin)));

        if (fechasSolapadas)
        {
            TempData["ErrorMessage"] = "Ya has solicitado vacaciones en estas fechas.";
            ViewBag.Catalogo_Aprobaciones_idCatalogo_Aprobaciones = new SelectList(db.catalogo_aprobaciones, "idCatalogo_Aprobaciones", "Descripcion", vacaciones.Catalogo_Aprobaciones_idCatalogo_Aprobaciones);
            return View(vacaciones);
        }

        TimeSpan diferencia = vacaciones.Fecha_Fin - vacaciones.Fecha_Inicio;
        int dias = diferencia.Days;
        int finesDeSemana = 0;

        for (DateTime fecha = vacaciones.Fecha_Inicio; fecha <= vacaciones.Fecha_Fin; fecha = fecha.AddDays(1))
        {
            if (fecha.DayOfWeek == DayOfWeek.Saturday || fecha.DayOfWeek == DayOfWeek.Sunday)
            {
                finesDeSemana++;
            }
        }

        // Obtener todos los feriados dentro del rango de Fecha_Inicio y Fecha_Fin
        var feriadosEnRango = db.feriados
            .Where(f => f.Fecha_Feriado >= vacaciones.Fecha_Inicio && f.Fecha_Feriado <= vacaciones.Fecha_Fin && f.Activo)
            .Select(f => f.Fecha_Feriado)
            .ToList();
    }
}
```

Fuente: Elaboración propia, 2024.

Figura 59:
Proceso para solicitar permisos.

```

[HttpPost]
[ValidateAntiForgeryToken]
0 referencias
public ActionResult Create([Bind(Include = "Catalogo_Tipo_Permission_idCatalogo_Tipo_Permission, Fecha_Solicitud, Cantidad_horas, Asunto")] permisos_colaborador permisos_colaborador)
{
    UserInfo user = (UserInfo)Session["UserInfo"];

    permisos_colaborador.Empleado_idEmpleado = user.IdEmpleado;
    permisos_colaborador.Catalogo_Aprobaciones_idCatalogo_Aprobaciones = 3;
    permisos_colaborador.Catalogo_Tipo_Permission_idCatalogo_Tipo_Permission = 3;

    // Comprobar si la fecha de solicitud coincide con alguna fecha de feriado activa
    bool esFeriado = db.feriados.Any(f => f.Fecha_Feriado == permisos_colaborador.Fecha_Solicitud && f.Activo);
    if (esFeriado)
    {
        TempData["ErrorMessage"] = "No puedes solicitar permisos en un día feriado.";
        return RedirectToAction("Create");
    }

    // Comprobar si ya existe una solicitud de permiso para la misma fecha y el mismo empleado
    bool solicitudExiste = db.permisos_colaborador.Any(p => p.Fecha_Solicitud == permisos_colaborador.Fecha_Solicitud && p.Empleado_idEmpleado == permisos_colaborador.Empleado_idEmpleado);
    if (solicitudExiste)
    {
        TempData["ErrorMessage"] = "Ya tienes una solicitud de permiso para esta fecha.";
        return RedirectToAction("Create");
    }

    // Verificar si hay registros en Control_Marcas con Movimientos_idMovimientos igual a 1
    bool hasAttendance = db.control_marcas_asistencia
        .Any(cm => cm.Empleado_idEmpleado == user.IdEmpleado && cm.Movimientos_idMovimientos == 1);

    if (!hasAttendance)
    {
        TempData["ErrorMessage"] = "No puedes solicitar permiso si no marcas asistencia.";
        return RedirectToAction("Create");
    }

    // Calcular la diferencia de horas entre Hora_Entrada y Hora_Salida del horario del empleado para el día solicitado
    DateTime fechaSolicitud = permisos_colaborador.Fecha_Solicitud.Date;

    horario horarioEmpleado = db.horario
        .FirstOrDefault(h => h.Empleado_idEmpleado == user.IdEmpleado
            && h.Catalogo_Dia_idCatalogo_Dia == (int)fechaSolicitud.DayOfWeek);

    if (horarioEmpleado == null)
    {
        TempData["ErrorMessage"] = "No se encontró un horario registrado para este día.";
        return RedirectToAction("Create");
    }

    TimeSpan diferenciaHoras = horarioEmpleado.Hora_Salida - horarioEmpleado.Hora_Entrada;
    TimeSpan horasSolicitadas = TimeSpan.FromHours(permisos_colaborador.Cantidad_horas);

    // Verificar si la cantidad de horas solicitadas es mayor que la diferencia de horas del horario

```

Fuente: Elaboración propia, 2024.

Figura 60:
Proceso para calcular la liquidación.

```

0 referencias
public ActionResult Create(Liquidacion _objLiquidacion, int HizoPreaviso)
{
    try
    {
        // Verificar si el empleado ya tiene una liquidación existente
        var liquidacionExistente = db.liquidacion.FirstOrDefault(l => l.Empleado_idEmpleado == _objLiquidacion.Empleado_idEmpleado);

        if (liquidacionExistente != null)
        {
            // Agregar un mensaje de error que indique que el empleado ya tiene una liquidación.
            TempData["ErrorMessage"] = "El empleado ya tiene una liquidación existente.";
            ViewBag.Catalogo_Tipo_Liquidacion_idCatalogo_Tipo_Liquidacion = new SelectList(db.catalogo_tipo_liquidacion, "idCatalogo_Tipo_Liquidacion", "Descripcion", _objLiquidacion.Catalogo_Tipo_Liquidacion_id);
            ViewBag.Empleado_idEmpleado = new SelectList(db.empleado.Include(e => e.persona).ToList(), "idEmpleado", "persona.Nombre", _objLiquidacion.Empleado_idEmpleado);
            return View(_objLiquidacion);
        }

        if (ModelState.IsValid)
        {
            // Obtener los días disponibles para preaviso
            int diasDisponiblesParaPreaviso = CalcularDiasPreaviso(_objLiquidacion.Empleado_idEmpleado);

            // Validar que los días de preaviso ingresados no excedan los disponibles
            if (_objLiquidacion.Dias_preaviso > diasDisponiblesParaPreaviso)
            {
                ModelState.AddModelError("Dias_preaviso", "Los días de preaviso ingresados exceden los días disponibles.");
                ViewBag.Catalogo_Tipo_Liquidacion_idCatalogo_Tipo_Liquidacion = new SelectList(db.catalogo_tipo_liquidacion, "idCatalogo_Tipo_Liquidacion", "Descripcion", _objLiquidacion.Catalogo_Tipo_Liquidacion_id);
                ViewBag.Empleado_idEmpleado = new SelectList(db.empleado.Include(e => e.persona).ToList(), "idEmpleado", "persona.Nombre", _objLiquidacion.Empleado_idEmpleado);
                return View(_objLiquidacion);
            }

            // Si pasa la validación, continuar con la creación de la liquidación
            int ultimoIdLiquidacion = db.liquidacion.Any() ? db.liquidacion.Max(l => l.Id_Liquidacion) : 0;
            int nuevoIdLiquidacion = ultimoIdLiquidacion + 1;

            // Crear el objeto liquidacion
            liquidacion _liquidacion = new liquidacion
            {
                Id_Liquidacion = nuevoIdLiquidacion,
                Fecha_Salida = _objLiquidacion.Fecha_Salida,
                Dias_preaviso = _objLiquidacion.Dias_preaviso,
                Catalogo_Tipo_Liquidacion_idCatalogo_Tipo_Liquidacion = _objLiquidacion.Catalogo_Tipo_Liquidacion_idCatalogo_Tipo_Liquidacion,
                Empleado_idEmpleado = _objLiquidacion.Empleado_idEmpleado,
                Monto_Prevaviso = 0, // Se calculará más adelante
                Monto_Cesantia = 0, // Se calculará más adelante
                Monto_Vacaciones = (float)CalcularMontoVacaciones(_objLiquidacion.Empleado_idEmpleado),
                Monto_Aguinaldo = (float)CalcularMontoAguinaldo(_objLiquidacion.Empleado_idEmpleado),
                Monto_Total = 0 // Se calculará más adelante
            };

            float montoPreaviso = 0;

            // Calcular el monto del preaviso si corresponde
            if (HizoPreaviso == 1)
            {

```

Fuente: Elaboración propia, 2024.

Figura 61:
Proceso para calcular la incapacidad.

```
[HttpPost]
[ValidateAntiForgeryToken]
0 referencias
public ActionResult Create([Bind(Include = "idIncapacidades,Catalogo_Tipo_Incapacidad_idTipo_Incapacidad,Fecha_Inicio,Fecha_Fin,Empleado_idEmpleado")] incapacidades incapacidad)
{
    if (ModelState.IsValid)
    {
        // Validar que la fecha de fin no sea menor que la fecha de inicio
        if ( incapacidad.Fecha_Fin < incapacidad.Fecha_Inicio)
        {
            ModelState.AddModelError("", "La fecha de finalización de la incapacidad no puede ser menor a la fecha de inicio.");
            ViewBag_Catalogo_Tipo_Incapacidad_idTipo_Incapacidad = new SelectList(db.catalogo_tipo_incapacidad, "idTipo_Incapacidad", "Descripcion", incapacidad.Catalogo_Tipo_Incapacidad_idTipo_Incapacidad);
            ViewBag_Empleado_idEmpleado = new SelectList(db.empleado.Include(e => e.persona).ToList(), "idEmpleado", "persona.Nombre", incapacidad.Empleado_idEmpleado);
            return View(incapacidad);
        }

        // Verificar si ya existe una incapacidad para el mismo colaborador con las mismas fechas
        bool incapacidadExistente = db.incapacidades
            .Any(i => i.Empleado_idEmpleado == incapacidad.Empleado_idEmpleado
                && i.Fecha_Inicio == incapacidad.Fecha_Inicio
                && i.Fecha_Fin == incapacidad.Fecha_Fin);

        if (incapacidadExistente)
        {
            TempData["ErrorMessage"] = "Ya existe una incapacidad para este colaborador en la misma fecha.";
            ViewBag_Catalogo_Tipo_Incapacidad_idTipo_Incapacidad = new SelectList(db.catalogo_tipo_incapacidad, "idTipo_Incapacidad", "Descripcion", incapacidad.Catalogo_Tipo_Incapacidad_idTipo_Incapacidad);
            ViewBag_Empleado_idEmpleado = new SelectList(db.empleado.Include(e => e.persona).ToList(), "idEmpleado", "persona.Nombre", incapacidad.Empleado_idEmpleado);
            return View(incapacidad);
        }

        // Verificar si la fecha de inicio es un día feriado
        bool esFeriadoFechaInicio = db.feriados.Any(f => f.Fecha_Feriado == incapacidad.Fecha_Inicio && f.Activo);

        if (esFeriadoFechaInicio)
        {
            TempData["ErrorMessage"] = "La fecha de inicio no puede ser un día feriado.";
            ViewBag_Catalogo_Tipo_Incapacidad_idTipo_Incapacidad = new SelectList(db.catalogo_tipo_incapacidad, "idTipo_Incapacidad", "Descripcion", incapacidad.Catalogo_Tipo_Incapacidad_idTipo_Incapacidad);
            ViewBag_Empleado_idEmpleado = new SelectList(db.empleado.Include(e => e.persona).ToList(), "idEmpleado", "persona.Nombre", incapacidad.Empleado_idEmpleado);
            return View(incapacidad);
        }

        // Verificar si la fecha de fin es un día feriado
        bool esFeriadoFechaFin = db.feriados.Any(f => f.Fecha_Feriado == incapacidad.Fecha_Fin && f.Activo);

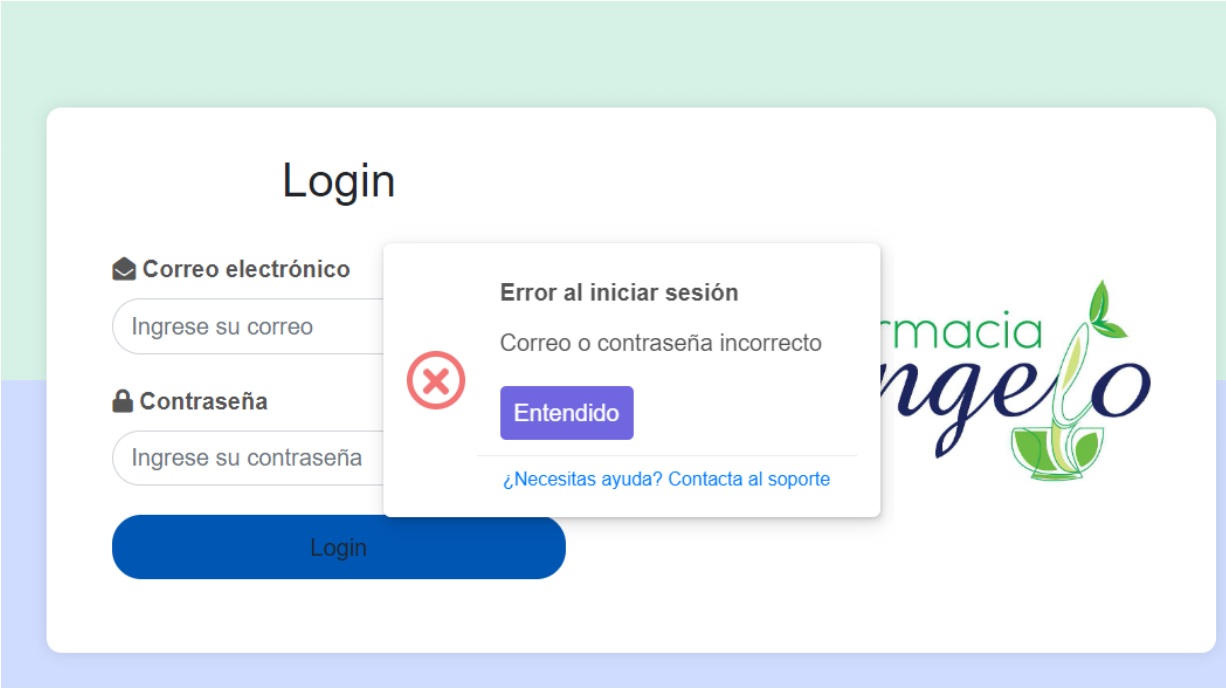
        if (esFeriadoFechaFin)
        {
            TempData["ErrorMessage"] = "La fecha de fin no puede ser un día feriado.";
            ViewBag_Catalogo_Tipo_Incapacidad_idTipo_Incapacidad = new SelectList(db.catalogo_tipo_incapacidad, "idTipo_Incapacidad", "Descripcion", incapacidad.Catalogo_Tipo_Incapacidad_idTipo_Incapacidad);
            ViewBag_Empleado_idEmpleado = new SelectList(db.empleado.Include(e => e.persona).ToList(), "idEmpleado", "persona.Nombre", incapacidad.Empleado_idEmpleado);
            return View(incapacidad);
        }
    }
}
```

Fuente: Elaboración propia, 2024.

Pruebas Funcionales

Tabla 69:

Prueba funcional inicio de sesión.

Prueba funcional 01			
Nombre:	Inicio de Sesión		
Módulo de prueba:	Módulo de seguridad		
Fecha de prueba realizada:	11/7/2024		
Realizado por:	Deyvin Polanco Espinoza		
Caso a probar	Resultado deseado	Resultado obtenido	Estado de prueba
Validar que el sistema no permita al usuario ingresar con credenciales inválidas.	El sistema no debe permitir al usuario ingresar con las credenciales inválidas. El sistema debe mostrar error "Correo o contraseña incorrecto".	El sistema no permite al usuario ingresar al sistema. El sistema muestra un error "Correo o contraseña incorrecto".	Exitoso
Evidencia de prueba			
			

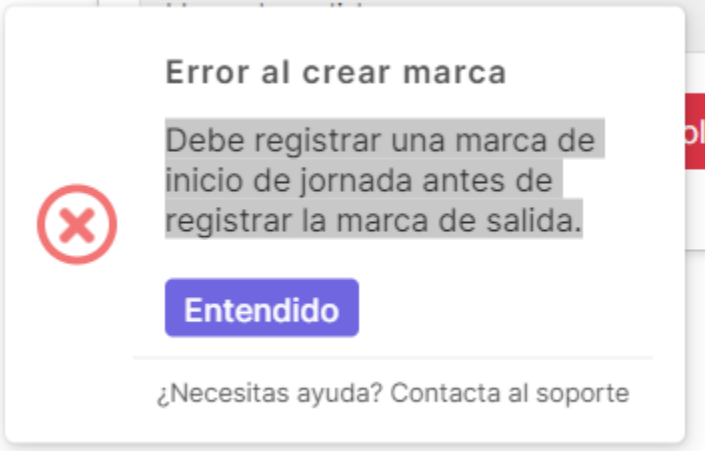
Fuente: Elaboración propia, 2024.

Tabla 70:
Prueba funcional

Prueba funcional 02			
Nombre:	Validación de usuarios inactivos		
Módulo de prueba:	Módulo de seguridad		
Fecha de prueba realizada:	11/7/2024		
Realizado por:	Deyvin Polanco Espinoza		
Caso a probar	Resultado deseado	Resultado obtenido	Estado de prueba
Validar que el sistema no permita al que esta inactivo dentro del sistema.	El sistema no debe permitir al usuario ingresar, si este está inactivo, lo que significa que ya no tiene permisos para acceder al sistema.	El sistema no permite al usuario ingresar al sistema. El aplicativo le muestra la etiqueta de usuario inactivo.	Satisfactoria
Evidencia de prueba			
			

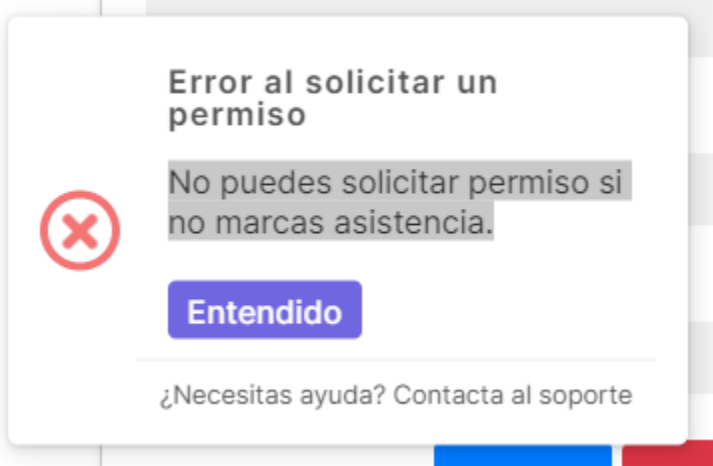
Fuente: Elaboración propia, 2024.

Tabla 71:
Prueba funcional

Prueba funcional 03			
Nombre:	Validar marcas de asistencia		
Módulo de prueba:	Control de marcas de asistencia		
Fecha de prueba realizada:	11/7/2024		
Realizado por:	Deyvin Polanco Espinoza		
Caso a probar	Resultado deseado	Resultado obtenido	Estado de prueba
Validar que no el usuario no pueda marcar una marca de salida antes de una marca de inicio.	El sistema no debe permitir al usuario marcar fin de jornada antes de iniciar, ya que causará inconsistencias dentro del sistema	El sistema no permite al usuario ingresar a la una marca de fin de jornada antes de iniciar. El aplicativo le muestra la etiqueta “Debe registrar una marca de inicio de jornada antes de registrar la marca de salida.”	Satisfactoria
Evidencia de prueba			
			

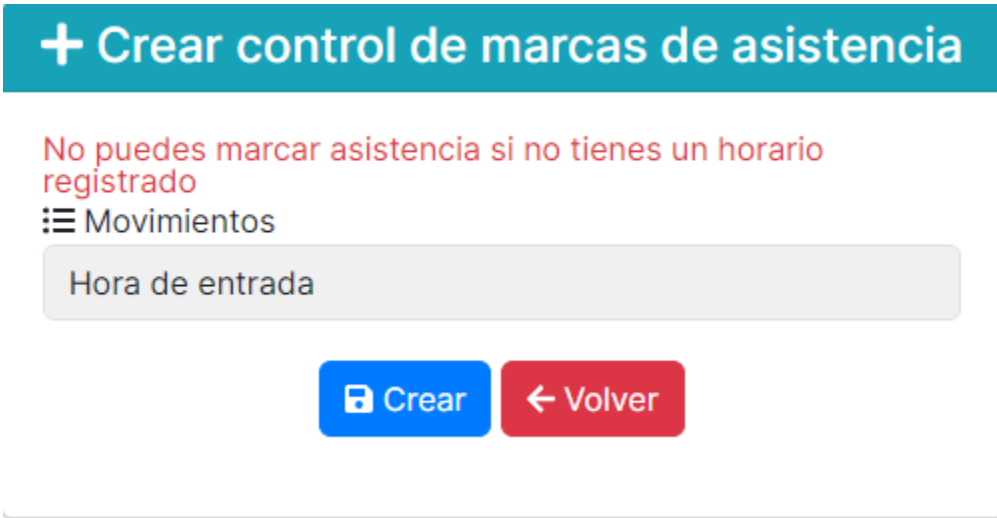
Fuente: Elaboración propia, 2024.

Tabla 72:
Prueba funcional.

Prueba funcional 04			
Nombre:	Utilización del sistema		
Módulo de prueba:	Control de marcas de asistencia		
Fecha de prueba realizada:	11/7/2024		
Realizado por:	Deyvin Polanco Espinoza		
Caso a probar	Resultado deseado	Resultado obtenido	Estado de prueba
Validar que el sistema no permita al usuario realizar peticiones si este no realiza una marca de inicio de jornada.	El sistema no debe permitir al usuario solicitar cualquier petición si no realiza una marca. El sistema debe mostrar una etiqueta de error “No puedes solicitar permiso si no marcas asistencia.”.	El sistema no permite al usuario realizar solicitudes. El aplicativo le muestra la etiqueta de error “No puedes solicitar permiso si no marcas asistencia”.	Satisfactoria
Evidencia de prueba			
			

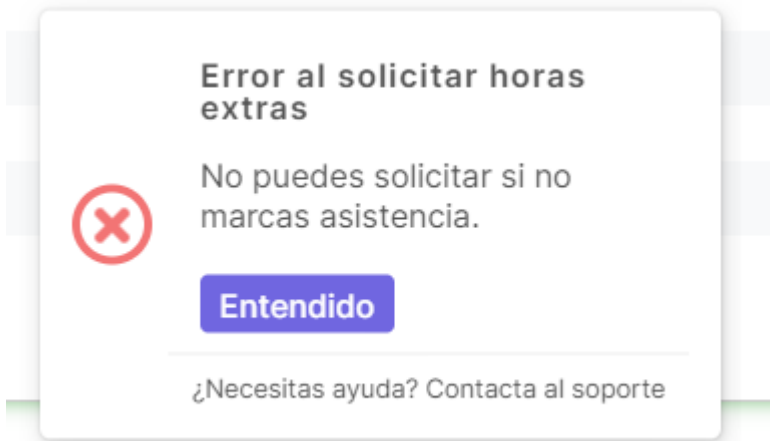
Fuente: Elaboración propia, 2024.

Tabla 73:
Prueba funcional.

Prueba funcional 05			
Nombre:	Utilización del sistema		
Módulo de prueba:	Control de marcas de asistencia		
Fecha de prueba realizada:	11/7/2024		
Realizado por:	Deyvin Polanco Espinoza		
Caso a probar	Resultado deseado	Resultado obtenido	Estado de prueba
Validar que el sistema no permita al usuario realizar una marca si no cuenta con horario asignado.	El sistema no debe permitir al usuario realizar una marca si no cuenta con un horario. El sistema debe mostrar una etiqueta de error “No puedes marcar asistencia si no tienes un horario registrado”.	El sistema no permite al usuario realizar marcas si no cuenta con un horario. El aplicativo le muestra la etiqueta de error “No puedes marcar asistencia si no tienes un horario registrado”.	Satisfactoria
Evidencia de prueba			
			

Fuente: Elaboración propia, 2024.

Tabla 74:
Pruebas funcional

Prueba funcional 06			
Nombre:	Validación de horas extras		
Módulo de prueba:	Horas extras		
Fecha de prueba realizada:	11/7/2024		
Realizado por:	Deyvin Polanco Espinoza		
Caso a probar	Resultado deseado	Resultado obtenido	Estado de prueba
Validar que el sistema no permita al usuario realizar una solicitud si no cuenta con una marca de inicio de asistencia.	El sistema no debe permitir al usuario realizar una solicitud si no cuenta con una marca de inicio de asistencia. El sistema debe mostrar una etiqueta de error "No puedes solicitar si no marcas asistencia.".	El sistema no permite al usuario una solicitud si no cuenta con una marca de inicio de asistencia. El aplicativo le muestra la etiqueta de error "No puedes solicitar si no marcas asistencia".	Satisfactoria
Evidencia de prueba			
			

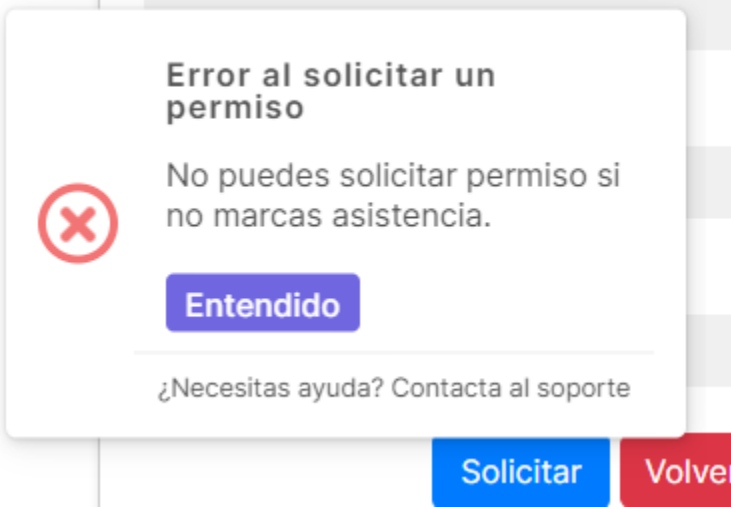
Fuente: Elaboración propia, 2024.

Tabla 75:
Pruebas funcionales

Prueba funcional 07			
Nombre:	Solicitud de extras		
Módulo de prueba:	Horas extras		
Fecha de prueba realizada:	11/7/2024		
Realizado por:	Deyvin Polanco Espinoza		
Caso a probar	Resultado deseado	Resultado obtenido	Estado de prueba
Validar que el sistema no permita al usuario realizar una solicitud de horas extras si ya cuenta con una.	El sistema no debe permitir al usuario realizar una solicitud de horas extras si ya cuenta con una. El sistema debe mostrar una etiqueta de error "Ya has solicitado horas extras para este día."	El sistema no permite al usuario realizar una solicitud de horas extras si ya cuenta con una. El aplicativo le muestra la etiqueta de error "Ya has solicitado horas extras para este día".	Satisfactoria
Evidencia de prueba			
			

Fuente: Elaboración propia, 2024.

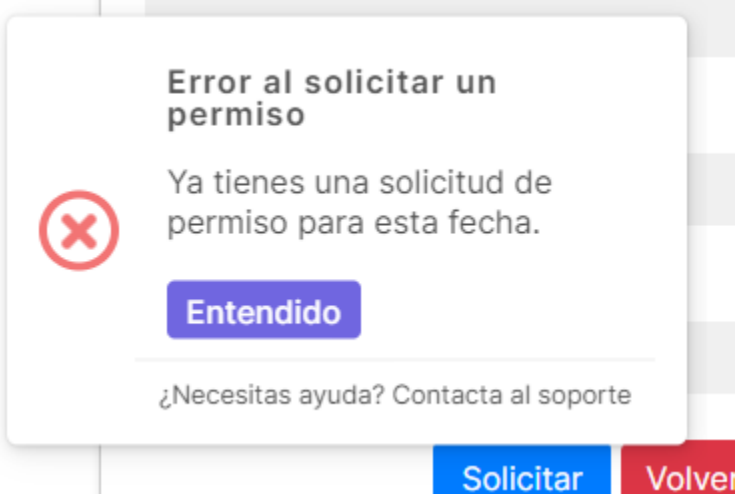
Tabla 76:
Pruebas funcionales

Prueba funcional 08			
Nombre:	Verificar marcas para solicitud de permisos.		
Módulo de prueba:	Permisos del colaborador.		
Fecha de prueba realizada:	11/7/2024		
Realizado por:	Deyvin Polanco Espinoza		
Caso a probar	Resultado deseado	Resultado obtenido	Estado de prueba
Validar que el sistema no permita al usuario realizar una solicitud si no cuenta con una marca de inicio de asistencia.	El sistema no debe permitir al usuario realizar una solicitud de permisos si no cuenta con una marca. El sistema debe mostrar una etiqueta de error "No puedes solicitar permiso si no marcas asistencia."	El sistema no permite al usuario realizar una solicitud de permisos si no cuenta con una marca. El aplicativo le muestra la etiqueta de error "No puedes solicitar permiso si no marcas asistencia".	Satisfactoria
Evidencia de prueba			
			

Fuente: Elaboración propia, 2024.

Tabla 77:
Pruebas funcionales

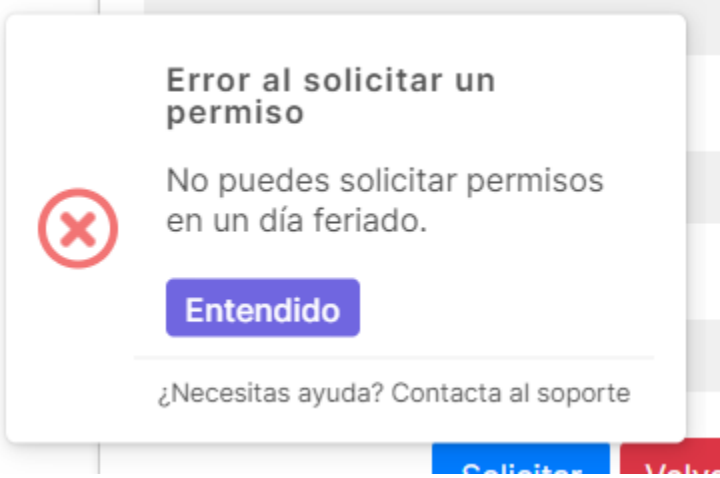
Prueba funcional 09	
Nombre:	Verificar si existen solicitud entre fechas para solicitud de permisos.
Módulo de prueba:	Permisos del colaborador.

Fecha de prueba realizada:	11/7/2024		
Realizado por:	Deyvin Polanco Espinoza		
Caso a probar	Resultado deseado	Resultado obtenido	Estado de prueba
Validar que el sistema no permita al usuario realizar una solicitud si ya cuenta con una el mismo día.	El sistema no debe permitir al usuario realizar una solicitud si ya cuenta con una el mismo día. El sistema debe mostrar una etiqueta de error “Ya tienes una solicitud de permiso para esta fecha”.	El sistema no permite al usuario realizar una solicitud si ya cuenta con una el mismo día. El aplicativo le muestra la etiqueta de error “Ya tienes una solicitud de permiso para esta fecha”.	Satisfactoria
Evidencia de prueba			
			

Fuente: Elaboración propia, 2024.

Tabla 78:
Pruebas funcionales

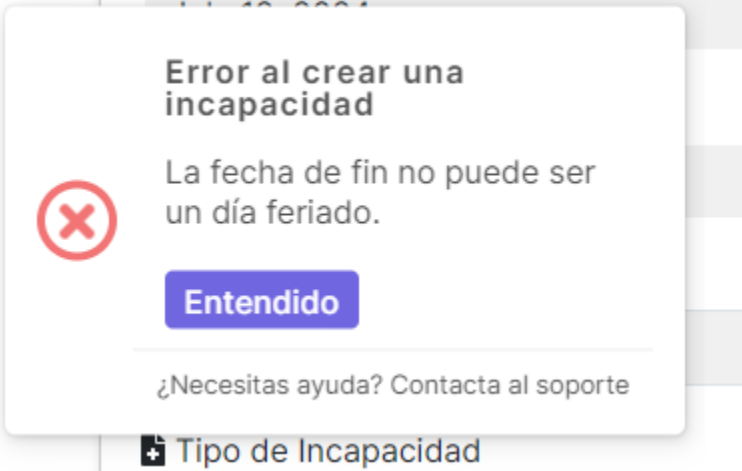
Prueba funcional 10			
Nombre:	Verificar que no sea un día feriado para solicitud de permisos.		
Módulo de prueba:	Permisos del colaborador.		
Fecha de prueba realizada:	11/7/2024		
Realizado por:	Deyvin Polanco Espinoza		
Caso a probar	Resultado deseado	Resultado obtenido	Estado de prueba

Validar que el sistema no permita al usuario realizar una solicitud de permisos si es un día feriado.	El sistema no debe permitir al usuario realizar una solicitud de permisos si es un día feriado. El sistema debe mostrar una etiqueta de error "No puedes solicitar permisos en un día feriado."	El sistema no permite al usuario realizar una solicitud de permisos si es un día feriado. El aplicativo le muestra la etiqueta de error "No puedes solicitar permisos en un día feriado."	Satisfactoria
Evidencia de prueba			
			

Fuente: Elaboración propia, 2024.

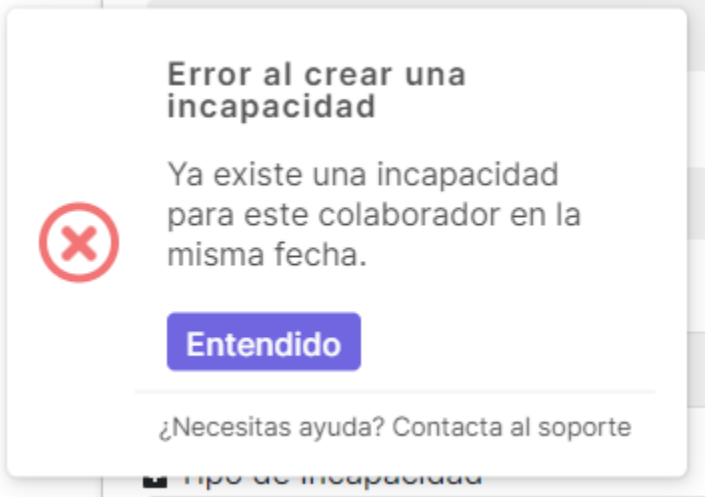
Tabla 79:
Pruebas funcionales

Prueba funcional 11			
Nombre:	Verificar que no sea un día feriado para crear incapacidades.		
Módulo de prueba:	Incapacidades.		
Fecha de prueba realizada:	11/7/2024		
Realizado por:	Deyvin Polanco Espinoza		
Caso a probar	Resultado deseado	Resultado obtenido	Estado de prueba

<p>Validar que el sistema no permita al usuario realizar una incapacidad si es un día feriado.</p>	<p>El sistema no debe permitir al usuario realizar una incapacidad si es un día feriado. El sistema debe mostrar una etiqueta de error “No puedes solicitar permisos en un día feriado.”.</p>	<p>El sistema no permita al usuario realizar una incapacidad si es un día feriado. El aplicativo le muestra la etiqueta de error “No puedes solicitar permisos en un día feriado.”.</p>	<p>Satisfactoria</p>
Evidencia de prueba			
			

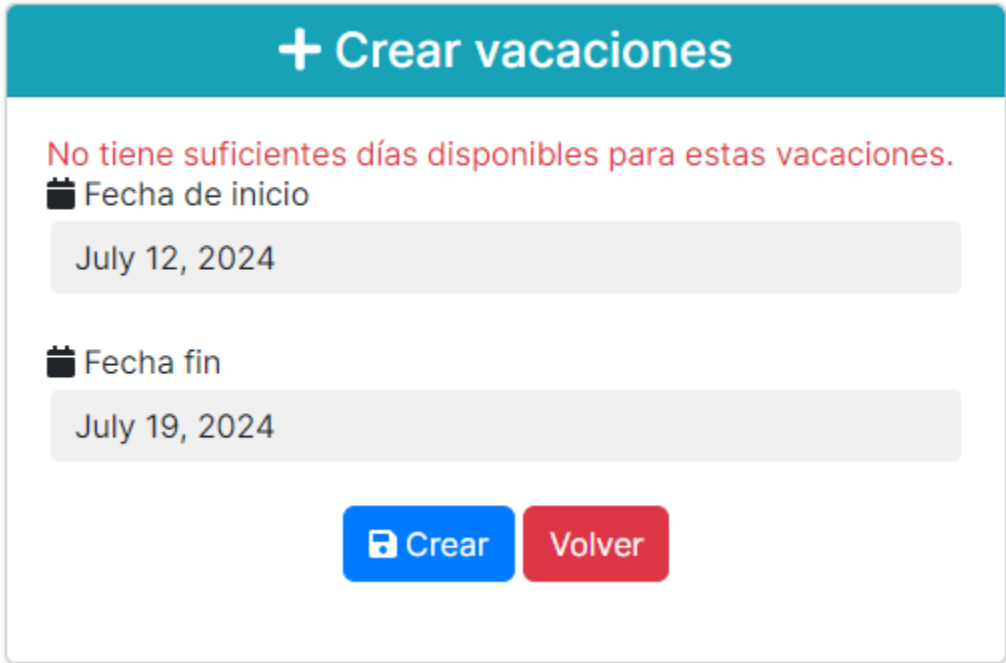
Fuente: Elaboración propia, 2024.

Tabla 80:
Pruebas funcionales

Prueba funcional 12			
Nombre:	Verificar que no se dupliquen incapacidades para el mismo empleado en las mismas fechas.		
Módulo de prueba:	Incapacidades.		
Fecha de prueba realizada:	11/7/2024		
Realizado por:	Deyvin Polanco Espinoza		
Caso a probar	Resultado deseado	Resultado obtenido	Estado de prueba
Validar que el sistema no permita al usuario realizar una incapacidad para el mismo empleado en las mismas fechas.	El sistema no debe permitir al usuario realizar una incapacidad para el mismo empleado en las mismas fechas. El sistema debe mostrar una etiqueta de error "No puedes solicitar permisos en un día feriado."	El sistema no permita al usuario realizar una incapacidad para el mismo empleado en las mismas fechas. El aplicativo le muestra la etiqueta de error "No puedes solicitar permisos en un día feriado."	Satisfactoria
Evidencia de prueba			
 <p>The screenshot shows a white error dialog box with a red 'X' icon. The text reads: "Error al crear una incapacidad", "Ya existe una incapacidad para este colaborador en la misma fecha.", and a blue "Entendido" button. At the bottom, there is a link: "¿Necesitas ayuda? Contacta al soporte".</p>			

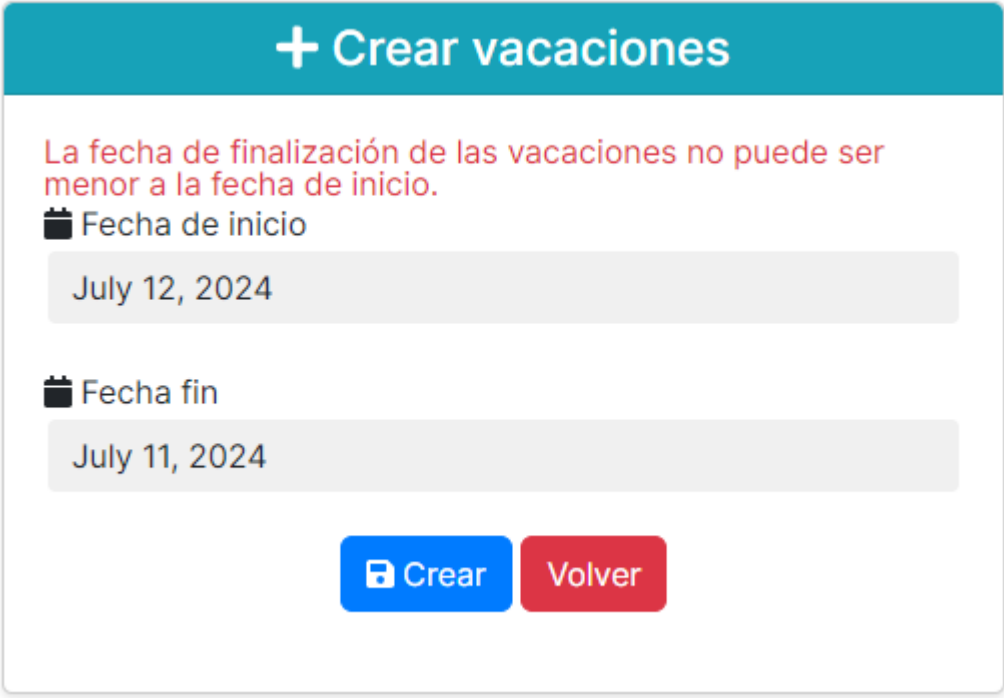
Fuente: Elaboración propia, 2024.

Tabla 81:
Pruebas funcionales

Prueba funcional 13			
Nombre:	Verificar que un colaborador que no cuenta con días de vacaciones no lo deje solicitar		
Módulo de prueba:	Control de vacaciones.		
Fecha de prueba realizada:	11/7/2024		
Realizado por:	Deyvin Polanco Espinoza		
Caso a probar	Resultado deseado	Resultado obtenido	Estado de prueba
Validar que el sistema no permita al usuario realizar una solicitud de días de vacaciones si no cuenta con días disponibles.	El sistema no debe permitir al usuario realizar una solicitud de días de vacaciones si no cuenta con días disponibles. El sistema debe mostrar una etiqueta de error “No tiene suficientes días disponibles para estas vacaciones”.	El sistema no permita al usuario realizar una solicitud de días de vacaciones si no cuenta con días disponibles. El aplicativo le muestra la etiqueta de error “No tiene suficientes días disponibles para estas vacaciones”.	Satisfactoria
Evidencia de prueba			
			

Fuente: Elaboración propia, 2024.

Tabla 82:
Pruebas funcionales

Prueba funcional 14			
Nombre:	Verificar que la fecha de inicio no sea igual a la fecha de fin de las vacaciones		
Módulo de prueba:	Control de vacaciones.		
Fecha de prueba realizada:	11/7/2024		
Realizado por:	Deyvin Polanco Espinoza		
Caso a probar	Resultado deseado	Resultado obtenido	Estado de prueba
Validar que el sistema no permita al usuario realizar una solicitud de días de vacaciones la fecha de fin es mayor a la fecha de inicio.	El sistema no debe permitir al usuario realizar una solicitud de días de vacaciones la fecha de fin es mayor a la fecha de inicio. El sistema debe mostrar una etiqueta de error “La fecha de finalización de las vacaciones no puede ser menor a la fecha de inicio”.	El sistema no permita al usuario realizar una solicitud de días de vacaciones la fecha de fin es mayor a la fecha de inicio. El aplicativo le muestra la etiqueta de error “La fecha de finalización de las vacaciones no puede ser menor a la fecha de inicio”.	Satisfactoria
Evidencia de prueba			
			

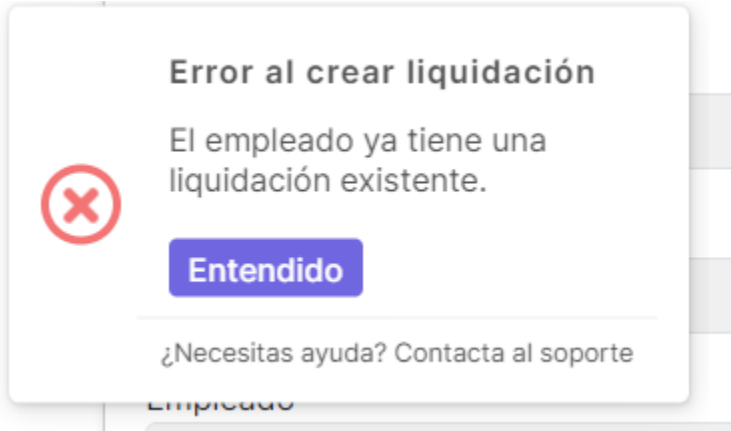
Fuente: Elaboración propia, 2024.

Tabla 83:
Pruebas funcionales

Prueba funcional 15			
Nombre:	Verificar que al liquidar a un colaborador se apliquen los montos correctos		
Módulo de prueba:	Liquidación.		
Fecha de prueba realizada:	11/7/2024		
Realizado por:	Deyvin Polanco Espinoza		
Caso a probar	Resultado deseado	Resultado obtenido	Estado de prueba
Validar que el sistema permita al usuario realizar una liquidación con responsabilidad laboral de forma correcta.	El sistema debe permitir al usuario realizar una liquidación con responsabilidad laboral de forma correcta. El sistema debe mostrar la liquidación del colaborador.	El sistema permite al usuario realizar una liquidación con responsabilidad laboral de forma correcta. El sistema debe mostrar la liquidación del colaborador.	Satisfactoria
Evidencia de prueba			


Fuente: Elaboración propia, 2024.

Tabla 84:
Pruebas funcionales

Prueba funcional 16			
Nombre:	Verificar que un colaborador que no se pueda aplicar una liquidación a un colaborador que ya cuenta con una		
Módulo de prueba:	liquidación.		
Fecha de prueba realizada:	11/7/2024		
Realizado por:	Deyvin Polanco Espinoza		
Caso a probar	Resultado deseado	Resultado obtenido	Estado de prueba
Validar que el sistema no permita al usuario realizar liquidación si el colaborador ya cuenta con una.	El sistema no permita al usuario realizar liquidación si el colaborador ya cuenta con una. El sistema debe mostrar una etiqueta de error “El empleado ya tiene una liquidación existente”.	El sistema no permita al usuario realizar liquidación si el colaborador ya cuenta con una. El aplicativo le muestra la etiqueta de error “El empleado ya tiene una liquidación existente”.	Satisfactoria
Evidencia de prueba			
			

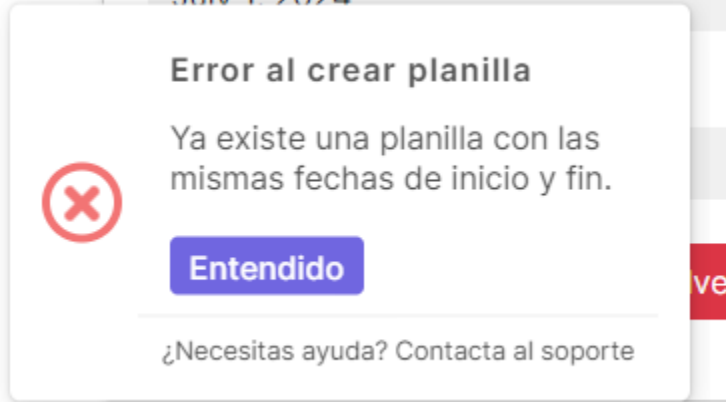
Fuente: Elaboración propia, 2024.

Tabla 85:
Pruebas funcionales

Prueba funcional 17																																				
Nombre:	Realizar la planilla de manera masiva																																			
Módulo de prueba:	Planilla.																																			
Fecha de prueba realizada:	11/7/2024																																			
Realizado por:	Deyvin Polanco Espinoza																																			
Caso a probar	Resultado deseado	Resultado obtenido	Estado de prueba																																	
Validar que el sistema permita al usuario realizar la planilla de manera masiva.	El sistema debe permitir al usuario realizar la planilla de manera masiva de para todos los colaboradores.	El sistema permita al usuario realizar la planilla de manera masiva de para todos los colaboradores.	Satisfactoria																																	
Evidencia de prueba																																				
 <table border="1"> <thead> <tr> <th>Empleado</th> <th>Fecha de inicio</th> <th>Fecha fin</th> <th>Fecha emitida</th> <th>Monto horas extras</th> <th>Monto deducciones</th> <th>Monto incapacidades</th> <th>Tipo deducción</th> <th>Impuesto de renta</th> <th>Salario bruto</th> <th>Monto total</th> </tr> </thead> <tbody> <tr> <td>Gimena Munoz</td> <td>01/07/2024</td> <td>15/07/2024</td> <td>12/07/2024</td> <td>0</td> <td>8272</td> <td>0</td> <td>CCSS</td> <td>0</td> <td>160000</td> <td>151728</td> </tr> <tr> <td>Pith Polanco</td> <td>01/07/2024</td> <td>15/07/2024</td> <td>12/07/2024</td> <td>1500</td> <td>1059,85</td> <td>4000</td> <td>CCSS</td> <td>0</td> <td>24500</td> <td>23440,15</td> </tr> </tbody> </table>				Empleado	Fecha de inicio	Fecha fin	Fecha emitida	Monto horas extras	Monto deducciones	Monto incapacidades	Tipo deducción	Impuesto de renta	Salario bruto	Monto total	Gimena Munoz	01/07/2024	15/07/2024	12/07/2024	0	8272	0	CCSS	0	160000	151728	Pith Polanco	01/07/2024	15/07/2024	12/07/2024	1500	1059,85	4000	CCSS	0	24500	23440,15
Empleado	Fecha de inicio	Fecha fin	Fecha emitida	Monto horas extras	Monto deducciones	Monto incapacidades	Tipo deducción	Impuesto de renta	Salario bruto	Monto total																										
Gimena Munoz	01/07/2024	15/07/2024	12/07/2024	0	8272	0	CCSS	0	160000	151728																										
Pith Polanco	01/07/2024	15/07/2024	12/07/2024	1500	1059,85	4000	CCSS	0	24500	23440,15																										


Fuente: Elaboración propia, 2024.

Tabla 86:
Pruebas funcionales

Prueba funcional 18			
Nombre:	Verificar que la planilla no se pueda crear en las mismas fechas		
Módulo de prueba:	Planilla.		
Fecha de prueba realizada:	11/7/2024		
Realizado por:	Deyvin Polanco Espinoza		
Caso a probar	Resultado deseado	Resultado obtenido	Estado de prueba
Validar que el sistema no permita al usuario realizar una planilla en las mismas fechas.	El sistema no debe permitir al usuario realizar una planilla en las mismas fechas. El sistema debe mostrar una etiqueta de error “Ya existe una planilla con las mismas fechas de inicio y fin.”.	El sistema no permite al usuario realizar una planilla en las mismas fechas. El aplicativo le muestra la etiqueta de error “Ya existe una planilla con las mismas fechas de inicio y fin.”.	Satisfactoria
Evidencia de prueba			
			

Fuente: Elaboración propia, 2024.

Tabla 87:
Pruebas funcionales

Prueba funcional 19												
Nombre:	Verificar que el cálculo de aguinaldo se realice de manera masiva											
Módulo de prueba:	Aguinaldo.											
Fecha de prueba realizada:	11/7/2024											
Realizado por:	Deyvin Polanco Espinoza											
Caso a probar	Resultado deseado	Resultado obtenido	Estado de prueba									
Validar que el sistema permita al usuario realizar el aguinaldo de manera masiva.	El sistema debe permitir al usuario realizar el cálculo de aguinaldo de manera masiva para todos los colaboradores y mostrarlo en la vista.	El sistema permite al usuario al usuario realizar el cálculo de aguinaldo de manera masiva para todos los colaboradores y mostrarlo en la vista.	Satisfactoria									
Evidencia de prueba												
 <table border="1"> <thead> <tr> <th>Empleado</th> <th>Año</th> <th>Monto de aguinaldo</th> </tr> </thead> <tbody> <tr> <td>Gimena Munoz</td> <td>2024</td> <td>13333,33</td> </tr> <tr> <td>Pith Polanco</td> <td>2024</td> <td>2041,67</td> </tr> </tbody> </table>				Empleado	Año	Monto de aguinaldo	Gimena Munoz	2024	13333,33	Pith Polanco	2024	2041,67
Empleado	Año	Monto de aguinaldo										
Gimena Munoz	2024	13333,33										
Pith Polanco	2024	2041,67										

Fuente: Elaboración propia, 2024.

Prueba funcional 20									
Nombre:	Aplicar reportes filtrados por colaborador								
Módulo de prueba:	Reportes.								
Fecha de prueba realizada:	11/7/2024								
Realizado por:	Deyvin Polanco Espinoza								
Caso a probar	Resultado deseado	Resultado obtenido	Estado de prueba						
Validar que el sistema permita filtrar reportes	El sistema debe permitir generar reportes con filtros	El sistema genera reporta sistemas con filtros.	Satisfactoria						
Evidencia de prueba									
<div style="border: 1px solid #ccc; padding: 10px;"> <div style="display: flex; justify-content: space-between; align-items: center;"> Generar PDF </div> <div style="display: flex; justify-content: space-between; margin-top: 10px;"> <div style="width: 45%;"> <p>Filtrar por Empleado:</p> <input type="text" value="Selecciona un empleado"/> </div> <div style="width: 45%;"> <p>Filtrar por Mes:</p> <input type="text" value="Selecciona un mes"/> <ul style="list-style-type: none"> <li style="background-color: #007bff; color: white; padding: 2px;">Selecciona un mes enero febrero marzo abril mayo junio julio agosto septiembre octubre noviembre diciembre </div> </div> <div style="margin-top: 10px;"> <p>Mostrar <input type="text" value="10"/> registros por página</p> <table border="1" style="width: 100%; border-collapse: collapse; text-align: center;"> <thead> <tr style="background-color: #009688; color: white;"> <th>Tipo de Movimiento</th> <th>Empleado</th> </tr> </thead> <tbody> <tr> <td>Hora de entrada</td> <td>Gimena Munoz</td> </tr> <tr> <td>Hora de entrada</td> <td>Pith Polanco</td> </tr> </tbody> </table> <p>Mostrando página 1 de 1</p> </div> </div>				Tipo de Movimiento	Empleado	Hora de entrada	Gimena Munoz	Hora de entrada	Pith Polanco
Tipo de Movimiento	Empleado								
Hora de entrada	Gimena Munoz								
Hora de entrada	Pith Polanco								

REFERENCIAS

- AltexSoft. (2021, 24 de diciembre). The good and the bad of .NET framework programming. <https://www.altexsoft.com/blog/the-good-and-the-bad-of-net-framework-programming/>
- Anandmeg. (2023, 31 de octubre). *¿Qué es el IDE de Visual Studio?* Microsoft Learn. <https://learn.microsoft.com/es-es/visualstudio/get-started/visual-studio-ide?view=vs-2022>
- Apen. (2022, 4 agosto). *Hardware - Apen Informática*. Apen Informática. <https://apen.es/glosario-de-informatica/hardware/>
- ASOBANCARIA. (2021, 22 de julio). *¿Qué debe saber para calcular correctamente su liquidación?* Saber Más, Ser Más. <https://www.sabermassermas.com/que-debe-saber-para-calcular-correctamente-liquidacion/>
- Atlassian. (s.f.). *Las pruebas de software en la entrega continua | Atlassian*. <https://www.atlassian.com/es/continuous-delivery/software-testing>
- Barbara. (2024b, 2 de febrero). *Fuentes de Información: Guía para la Investigación Académica Efectiva*. Gabinete De Estudios. <https://gabinetedeestudios.com/las-fuentes-de-informacion-que-son-las-fuentes-de-informacion-y-cuantos-tipos-existen>
- Bello, E. (2023, 9 de marzo). *Desarrollo de software: Todo lo que tienes que saber*. IEBSchool. <https://www.iebschool.com/blog/desarrollo-de-software-tecnologia/>
- BG&A. (2023, 13 de diciembre). *Aguinaldo (en Costa Rica) Todo Lo Que Debes Saber*. BGA Abogados Corporativos. <https://bgacorp.com/aguinaldo-costa-rica/>
- BillWagner. (2023, 15 febrero). *Un paseo por C#: Información general*. Microsoft Learn. <https://learn.microsoft.com/es-es/dotnet/csharp/tour-of-csharp/>
- Caronte Studio. (2021, 23 diciembre). *Qué es Modelo-Vista-Controlador*. <https://carontestudio.com/blog/que-es-modelo-vista-controlador/>

- Cimec. (2023, 20 de julio). *La investigación descriptiva y sus características*. <https://www.cimec.es/investigacion-descriptiva-caracteristicas/>
- Coppola, M. (2023, 4 de mayo). Desarrollo web: qué es, etapas y principales lenguajes. *HubSpot*. <https://blog.hubspot.es/website/que-es-desarrollo-web>
- Equipo editorial Etecé. (2023, 19 de noviembre). *Lenguaje de programación - concepto, tipos y ejemplos*. Concepto.de. <https://concepto.de/lenguaje-de-programacion/>
- Estrada, J. (2024, 9 enero). *Las vacaciones en Costa Rica*. Legal Aid Costa Rica. <https://www.lawyerincostarica.net/blog/vacaciones>
- Ética Legal Costa Rica. (2023). *Horas extra en Costa Rica*. LinkedIn. <https://es.linkedin.com/pulse/horas-extra-en-costa-rica-%C3%A9tica-legal-costa-rica->
- Euroinnova Business School. (2023, 24 de octubre). *Descubre qué es el mecanismo biela manivela y cómo funciona*. <https://www.euroinnova.cr/blog/que-es-un-sistema-informatico>
- Gamarra, G. (2023, 11 de diciembre). *¿Qué son los recursos humanos? Repasamos conceptos*. Factorial. <https://factorialhr.es/blog/que-son-recursos-humanos-definicion/#definicion-de-recursos-humanos>
- GoDaddy. (2023, 29 de junio). *Qué es base de datos – definición, significado y ejemplos*. <https://co.godaddy.com/blog/que-es-base-de-datos/>
- Hernández-Sampieri, R., Méndez-Valencia, S., Mendoza-Torres, C., y Cuevas-Romo, A. (2017). *Fundamentos de Investigación*. McGraw-Hill
- Hughes, A., y Stedman, C. (2021, 23 de abril). *Microsoft SQL Server*. ComputerWeekly.es. <https://www.computerweekly.com/es/definicion/Microsoft-SQL-Server>
- Investigadores. (2022, 21 de abril). *Fuentes de información primarias, secundarias y terciarias*. Técnicas de Investigación. <https://tecnicasdeinvestigacion.com/fuentes-de-informacion-primaria-y-secundaria-y-terciaria/>

- Jiménez, S. R. (2024, 26 de marzo). *¿Qué es un prototipo y para qué sirve?: Ejemplos y Tipos*. Agencia SEO En Lima. <https://saulromanjimenez.com/que-prototipo-sirve-ejemplos/>
- León, F. R. (2022, 17 de julio). Incapacidad laboral: Seis preguntas y respuestas sobre el trámite en Costa Rica. *El Financiero*. <https://www.elfinancierocr.com/economia-y-politica/incapacidad-laboral-seis-preguntas-y-respuestas/OQU6H23RZJAYRJYLK46PYBDZXU/story/>
- Mendoza, M. L. (2023, 13 abril). *Qué es un lenguaje de programación*. *OpenWebinars.net*. <https://openwebinars.net/blog/que-es-un-lenguaje-de-programacion/>
- Ministerio de Educación Pública de Costa Rica. (2024, 9 febrero). Permiso con goce de salario. <https://dgth.mep.go.cr/permisos-con-goce-de-salario-autorizados-por-jefes-inmediatos/>
- NeTEC Global Knowledge. (2022). *¿Qué es programación?* <https://www.netec.com/que-es-programacion>
- Next U. (2022, 31 agosto). *CSS: ¿Qué es y para qué sirve?* Blog | NextU LATAM. <https://www.nextu.com/blog/que-es-css-rc22/>
- Northware. (2022, 29 de julio). *Requerimientos en el desarrollo de software y aplicaciones*. <https://www.northware.mx/blog/requerimientos-en-el-desarrollo-de-software-y-aplicaciones/>
- Poder Ejecutivo de Costa Rica. (1954). *Decreto 21: Reglamento del Estatuto de Servicio Civil*. Sistema Costarricense de Información Jurídica. http://www.pgrweb.go.cr/scij/Busqueda/Normativa/Normas/nrm_texto_completo.aspx?nValor1=1&nValor2=8975
- Proavance. (2023, 15 de junio). *¿Sabes cuál es la importancia de la planilla en una empresa?* <https://proavance.pe/2021/10/16/sabes-cual-es-la-importancia-de-la-planilla-en-una-empresa/>

- Rick-Anderson. (2024, 19 marzo). *Introducción a MVC de ASP.NET*. Microsoft Learn. <https://learn.microsoft.com/es-es/aspnet/mvc/overview/older-versions-1/overview/asp-net-mvc-overview>
- Risso, I. (2022, 1 de abril). *Domina el modelo en cascada y potencia al máximo tus proyectos de software*. Crehana. <https://www.crehana.com/blog/transformacion-digital/modelo-en-cascada/>
- SalusPlay. (s.f.). *Las variables de investigación*. <https://www.salusplay.com/apuntes/apuntes-metodologia-de-la-investigacion/tema-2-las-variables-de-investigacion>
- Suárez, E. (2024, 22 de febrero). *Fuentes primarias y secundarias: todo lo que necesitas saber*. Experto Universitario. <https://expertouniversitario.es/blog/fuentes-primarias-y-secundarias/>
- Urrutia, D. (2023, 17 de octubre). *Qué es el JavaScript - definición, significado y ejemplos*. Arimetrics. <https://www.arimetrics.com/glosario-digital/javascript>
- Vadavo. (2023, 6 de julio). *HTML: Qué es y para qué sirve – VADAVO*. Blog de VADAVO. <https://www.vadavo.com/blog/html-que-es-y-para-que-sirve/>

APÉNDICES

Apéndice 1

Guía de entrevista

En el marco de una investigación sobre el cálculo de planilla de la empresa D'Angelo, le invitamos a completar este cuestionario. Su participación es de gran importancia para comprender cómo el tema en estudio influye en la actividad de la organización. Este cuestionario es confidencial. Sus respuestas solo se utilizarán con fines académicos y no serán compartidas con otras personas. Completar el cuestionario tomará aproximadamente 10 minutos.

1. ¿Con qué frecuencia se presentan errores en el cálculo de la planilla?
 - Nunca
 - Ocasionalmente
 - Frecuentemente
 - Siempre
2. ¿Considera que la falta de un sistema ágil para el cálculo de planilla ha afectado la precisión en el pago de los empleados?
 - Sí
 - No
 - No estoy seguro/a
3. ¿Cómo calificaría la eficacia del proceso actual de pago del aguinaldo?
 - Muy eficaz
 - Eficaz
 - Poco eficaz
 - Nada eficaz
4. ¿Cuál de los siguientes aspectos del proceso de pago del aguinaldo considera que necesita mayor mejora?
 - Precisión en los cálculos
 - Rapidez en el proceso
 - Facilidad de acceso a la información

- ¿Cómo evaluarías la precisión del registro de tus días de vacaciones en la empresa?
Muy preciso
 - Preciso
 - Poco preciso
 - Nada preciso
5. ¿Qué método o herramienta utiliza actualmente la empresa para llevar el registro de los días de vacaciones de los colaboradores?
- Excel
 - Papel y lápiz
6. ¿Con qué frecuencia realiza horas extras en su trabajo?
- Nunca
 - Ocasionalmente
 - Frecuentemente
 - Siempre
 - ¿Qué método o herramienta utiliza actualmente la empresa para llevar el registro de las horas extras?
 - Excel
 - Papel y lápiz
 - No sabe/no responde
7. ¿Cómo calificaría el control de sus horas extras en la empresa?
- Muy preciso
 - Preciso
 - Poco preciso
 - Nada preciso
8. ¿Ha experimentado problemas con el registro o la reducción correspondiente en la remuneración de sus permisos laborales en la empresa?
- Sí
 - No
9. ¿Cómo calificaría la eficacia del proceso actual de gestión de permisos laborales en la empresa?

- Muy eficaz
 - Eficaz
 - Poco eficaz
 - Nada eficaz
10. ¿Ha experimentado problemas con los cálculos de liquidación de empleados que hayan resultado en errores en los pagos finales?
- Sí
 - No
11. ¿Cómo calificaría la eficacia del proceso actual de cálculo de liquidación de empleados en la empresa?
- Muy eficaz
 - Eficaz
 - Poco eficaz
 - Nada eficaz
12. ¿Con qué frecuencia ha experimentado problemas en el registro de incapacidades en la empresa?
- Nunca
 - Ocasionalmente
 - Frecuentemente
 - Siempre
13. ¿Qué método o herramienta utiliza actualmente la empresa para llevar el registro de incapacidades?
- Excel
 - Papel y lápiz
14. ¿Estaría de acuerdo en que se utilice un sistema automatizado para el cálculo de planilla que mejore la precisión y claridad en los montos a pagar?
- Sí
 - No
 - No estoy seguro/a

Apéndice 2

Guía de observación

Entidad:	Farmacia D'Angelo
Dirección física de la entidad:	Cartago central occidental 50 sur de la esquina suroeste de la escuela ascensión Esquivel
Fecha de la actividad de observación:	1/3/2023
Nombre del estudiante:	Deyvin Francisco Polanco Espinoza

Tabla de control de aspectos observados:

No	Aspectos por observar	Cumple	No Cumple	Oportunidad de mejora	Detalle de Observación
1	Proceso de registro y manejo de la información para el cálculo de la planilla, incluyendo la precisión de los datos, la utilización de fórmulas correctas según la ley costarricense.		X		
2	Proceso actual de pago del aguinaldo, incluyendo el registro y manejo de la información, la precisión de los cálculos, y la satisfacción de los colaboradores.				
3	Proceso actual de seguimiento de los días de vacaciones de los colaboradores, incluyendo la precisión de los registros, la comunicación con los colaboradores sobre sus días disponibles, y la prevención de pagos incorrectos.				

No	Aspectos por observar	Cumple	No Cumple	Oportunidad de mejora	Detalle de Observación
4	Proceso actual de registro de las horas extras de los colaboradores, la precisión de los registros, y la comunicación de las horas extras trabajadas.				
5	Proceso actual de gestión de permisos laborales en la empresa, incluyendo la identificación y registro de los tipos de permisos, la consideración de la reducción correspondiente en la remuneración, y la precisión de los registros.				
6	Proceso actual de cálculo de liquidación en la empresa, la precisión de los cálculos, y la eficiencia del proceso.				
7	Proceso actual de registro de incapacidades en la empresa la precisión de los registros, y la claridad en los montos a pagar por los diferentes tipos de incapacidad.				

Fuente: Elaboración propia, 2024.