

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

**TRABAJO FINAL DE GRADUACIÓN PARA OPTAR POR EL GRADO DE
BACHILLERATO EN INGENIERÍA EN SISTEMAS**

**Prototipo funcional para la gestión del recurso humano en la empresa Nutrimax, ubicada
en Curridabat**

MARVIN RODRIGO NÚÑEZ SANDOVAL

LIC. SANTIAGO ASTORGA PEREIRA

TUTOR

Sede Central

DICIEMBRE, 2025

Dedicatoria

Quisiera dedicar este logro principalmente a mi madre Fanny y a mi abuela María Irene por el enorme esfuerzo que han hecho al apoyarme durante mi etapa de formación académica, por el cariño que me brindaron y por su apoyo emocional en momentos difíciles que me ayudaron a continuar en el camino.

También, a toda mi familia porque con sus oraciones, palabras de aliento y consejos hicieron de mí una mejor persona y me dieron las fuerzas necesarias para salir adelante. Además de todas las personas especiales tanto amigos como profesores que me acompañaron en esta etapa, contribuyendo a mi formación tanto profesional y como ser humano.

Agradecimientos

En primera instancia, quisiera agradecer a Dios por bendecirme en todo momento, por darme las fuerzas y ser mi guía en cada paso que doy, dándome la fortaleza para continuar y por dejarme cumplir con una de mis metas y sueños.

También, quiero dar gracias a mi madre, la mujer más fuerte y valiente que conozco, por darme tantos consejos de vida y enseñanzas increíbles que me han servido y me servirán en un futuro. Agradezco a las personas que me apoyaron durante mi proceso estudiantil, tanto profesores que me enseñaron sus valiosos conocimientos y me retaron a superarme, como amistades que estuvieron en el proceso.

CONTENIDO

Dedicatoria	2
Agradecimientos	3
CARTA DE APROBACIÓN DEL TUTOR.....	4
CARTA DE APROBACIÓN DEL LECTOR	14
DECLARACIÓN JURADA	15
CARTA DEL FILÓLOGO	16
CARTA DE SOLICITUD DE DEFENSA	17
CONTENIDO	18
Tablas	22
Figura	26
Resumen.....	28
CAPÍTULO I: INTRODUCCIÓN	29
Planteamiento del problema	29
Objetivo general	30
Objetivos específicos.....	30
Justificación.....	30
Viabilidad operativa	30
Viabilidad técnica.....	31
Viabilidad Económica	32
Viabilidad legal	35
Proyecciones.....	35
Alcance funcional	36
Alcance metodológico	37

Ciclo de vida del software.....	38
MVC	38
Alcance tecnológico.....	38
CAPÍTULO II: MARCO REFERENCIAL	40
CAPÍTULO III. MARCO METODOLÓGICO.....	61
Enfoques de investigación.....	61
Enfoque cuantitativo	61
Enfoque cualitativo	61
Enfoque mixto.....	62
Enfoque de investigación seleccionado.....	62
Tipos de investigación.....	62
Tipo de investigación seleccionado.....	63
Fuentes de información	64
Fuente de información primaria.....	64
Fuente de información secundaria	64
Fuentes de información terciaria.....	64
Variables	65
Definición conceptual	65
Definición operacional.....	65
Definición instrumental	65
Población.....	68
Muestra.....	68
Instrumentos de recolección de datos.....	69
Observación	70
Encuesta	70

	20
Entrevista	70
Proceso para análisis y recolección de datos.....	71
CAPÍTULO IV: ANÁLISIS DE RESULTADOS	72
CAPÍTULO V: PROPUESTA	82
Análisis.....	82
Análisis de software	82
Módulo de Calcular Planilla:	82
Módulo de Calcular Extras:	82
Módulo de Calcular Deducciones:.....	83
Módulo de Calcular Liquidaciones:.....	83
Módulo de Calcular Permisos:.....	83
Módulo de Calcular Aguinaldo:.....	83
Módulo de Calcular Incapacidades:.....	83
Módulo de Solicitar Vacaciones:.....	83
Módulo de Evaluar Empleados:.....	84
Módulo de Mantenimientos:.....	84
Módulo de Consultas:	84
Módulo de Reportes:.....	84
Módulo de Seguridad:.....	84
Análisis del hardware	84
Análisis de telecomunicaciones	85
Análisis de herramientas técnicas	86
Análisis del conocimiento del personal.....	88
Casos de uso	89
Descripción de casos de uso.....	90

Diseño	184
Arquitectura del sistema.....	184
.....	184
Arquitectura de software	185
Diseño de entradas	186
Diseño de base de datos	190
Diccionario de datos.....	190
Diseño de procesos.....	203
Diseño de salidas.....	209
Diagramas UML.....	212
Diagrama de secuencia.....	213
Programación	219
Entradas y salidas	219
Procesos.....	222
Validaciones	225
Pruebas	229
CAPÍTULO VI: CONCLUSIONES Y RECOMENDACIONES	235
Conclusiones.....	235
Recomendaciones	236
REFERENCIAS.....	238
APÉNDICES.....	245

Tablas

Tabla 1: Descripción y costos por hardware	32
Tabla 2: Descripción y costos de software.....	33
Tabla 3: Descripción y costos de servicios de base de datos y host	33
Tabla 4: Descripción y costos de mano de obra.....	34
Tabla 4. Cuadro de variables.....	66
Tabla 5: Nivel de confianza para el calculo de la muestra.....	69
Tabla 6: Hardware implementado para el desarrollo del sistema	85
Tabla 7: Descripción y costos de software (desarrollo).....	87
Tabla 8: Servicios gestionados (base de datos y hosting en la nube).....	87
Tabla 9: Caso de uso “Crear empleado”	90
Tabla 10: Caso de uso “Editar empleado”	92
Tabla 11: Caso de uso “Eliminar empleado”	94
Tabla 12: Caso de uso “Leer empleado”	95
Tabla 13: Caso de uso “Crear departamento”	96
Tabla 14: Caso de uso “Editar departamento”	98
Tabla 15: Caso de uso “Eliminar departamento”	100
Tabla 16: Caso de uso “Leer departamento”	102
Tabla 17: Caso de uso “Crear puesto”	104
Tabla 18: Caso de uso “Editar puesto”	106
Tabla 19: Caso de uso “Eliminar puesto”	108
Tabla 20: Caso de uso “Leer puesto”	110
Tabla 21: Caso de uso “Crear Horario”	112
Tabla 22: Caso de uso “Editar Horario”	114
Tabla 23: Caso de uso “Eliminar Horario”	116
Tabla 24: Caso de uso “Leer Horario”	118
Tabla 25: Caso de uso “Crear Rol”	120
Tabla 26: Caso de uso “Editar Rol”	122
Tabla 27: Caso de uso “Eliminar Rol”	124
Tabla 28: Caso de uso “Leer Rol”	126
Tabla 29: Caso de uso “Crear Usuario”	127

Tabla 30: Caso de uso “Editar Usuario”	129
Tabla 31: Caso de uso “Eliminar Usuario”	131
Tabla 32: Caso de uso “Leer Usuario”	133
Tabla 33: Caso de uso “Crear Vacaciones”	135
Tabla 34: Caso de uso “Editar Vacaciones”	137
Tabla 35: Caso de uso “Cancelar Vacaciones”	139
Tabla 36: Caso de uso “Aprobar Vacaciones”	141
Tabla 37: Caso de uso “Rechazar Vacaciones”	143
Tabla 38: Caso de uso “Crear Permiso”	145
Tabla 39: Caso de uso “Aprobar Permiso”	147
Tabla 40: Caso de uso “Rechazar Permiso”	149
Tabla 41: Caso de uso “Leer Permiso”	151
Tabla 42: Caso de uso “Crear Horas Extras”	153
Tabla 43: Caso de uso “Aprobar Horas Extras”	155
Tabla 44: Caso de uso “Rechazar Horas Extras”	157
Tabla 45: Caso de uso “Leer Horas Extras”	159
Tabla 46: Caso de uso “Crear Planilla”	161
Tabla 47: Caso de uso “Generar Deducciones”	163
Tabla 48: Caso de uso “Previsualizar Aguinaldo”	165
Tabla 49: Caso de uso “Generar Aguinaldo”	167
Tabla 50: Caso de uso “Crear Liquidación”	169
Tabla 51: Caso de uso “Leer Liquidaciones”	171
Tabla 52: Caso de uso “Consultar Empleados”	173
Tabla 53: Caso de uso “Consultar Vacaciones”	174
Tabla 54: Caso de uso “Consultar Permisos”	175
Tabla 55: Caso de uso “Consultar Deducciones”	176
Tabla 56: Caso de uso “Reporte de Planilla”	177
Tabla 57: Caso de uso “Reporte de Deducciones”	178
Tabla 58: Caso de uso “Reporte de Vacaciones”	179
Tabla 59: Caso de uso “Reporte de Horas Extras”	180
Tabla 60: Caso de uso “Crear Plantilla de Evaluación”	181

Tabla 61: Caso de uso “Iniciar Sesión	182
Tabla 62: Estructura tabla Aguinaldo.....	191
Tabla 63: Estructura tabla Asistencia.....	191
Tabla 64: Estructura tabla AuditLog.....	191
Tabla 65: Estructura tabla Canton.....	192
Tabla 66: Estructura tabla Correo	192
Tabla 67: Estructura tabla Criterio.....	193
Tabla 68: Estructura tabla Deducciones.....	193
Tabla 69: Estructura tabla Departamento.....	194
Tabla 70: Estructura tabla DiasFeriados	194
Tabla 72: Estructura tabla Empleado	195
Tabla 73: Estructura tabla Evaluación	195
Tabla 74: Estructura tabla EvaluacionDetalle.....	196
Tabla 76: Estructura tabla HorasExtras.....	197
Tabla 77: Estructura tabla ImpuestosRenta.....	197
Tabla 78: Estructura tabla Incapacidad	198
Tabla 79: Estructura tabla Liquidaciones.....	198
Tabla 80: Estructura tabla Permisos.....	199
Tabla 81: Estructura tabla Planilla	199
Tabla 82: Estructura tabla PlantillaCriterio.....	200
Tabla 83: Estructura tabla PlantillaEvaluacion	201
Tabla 84: Estructura tabla Provincia	201
Tabla 85: Estructura tabla Puestos	201
Tabla 86: Estructura tabla Roles	202
Tabla 87: Estructura tabla Teléfono	202
Tabla 88: Estructura tabla Usuario.....	202
Tabla 89: Estructura tabla Vacaciones	203
Tabla 90: Caso de prueba de inicio de sesión	229
Tabla 91: Caso de prueba de verificación de campos del colaborador	231
Tabla 92: Caso de prueba de validación de la identificación del empleado	232
Tabla 93: Caso de prueba de verificación de solicitud de horas extra	233

Tabla 94: Caso de prueba de verificación de solicitud de vacaciones	234
---	-----

Figura

Figura 1: Aplicaciones de software.....	42
Figura 2: Herramienta de Microsoft Azure.....	44
Figura 3: Demostración de C# como lenguaje multiplataforma.....	47
Figura 4: Proceso de Scrum	52
Figura 5: Resultado encuesta colaboradores – Pregunta 1.....	72
Figura 6: Resultado encuesta colaboradores – Pregunta 2.....	73
Figura 7: Resultado encuesta colaboradores – Pregunta 3.....	74
Figura 8: Resultado encuesta colaboradores – Pregunta 4.....	75
Figura 9: Resultado encuesta colaboradores – Pregunta 5.....	76
Figura 10: Resultado encuesta colaboradores – Pregunta 6.....	76
Figura 11: Resultado encuesta colaboradores – Pregunta 7.....	77
Figura 12: Diagrama de red	86
Figura 13: Arquitectura del sistema	184
Figura 14: Arquitecta de software.....	186
Figura 15: Inicio de sesión	187
Figura 16: Menú principal	188
Figura 17: Agregar empleado	188
Figura 18: Solicitar vacaciones.....	189
Figura 19: Planilla	189
Figura 20. Diseño de la base de datos.....	190
Figura 21: Proceso de inicio de sesión.....	204
Figura 22: Agregar empleado.....	205
Figura 23: Solicitud de vacaciones	206
Figura 24: Generar planilla.....	207
Figura 26: Detalle de planillas	209
Figura 27: Lista de empleados	210
Figura 28: Vacaciones solicitadas	210
Figura 29: Asistencias	211
Figura 30: Permisos	211
Figura 31: Diagrama de clases.....	212

Figura 32: Solicitud de horas extra	213
Figura 33: Solicitud de vacaciones	214
Figura 34: Solicitud de incapacidades	215
Figura 35: Generación de planilla.....	216
Figura 36: Generación de liquidaciones	217
Figura 37: Agregar empleado.....	219
Figura 38: Agregar empleado 2.....	220
Figura 39: Agregar empleado 3.....	220
Figura 40: salida de empleados.....	221
Figura 41: salida de empleados 2.....	221
Figura 42: Controlador GET agregar	222
Figura 43: Controlador POST agregar	222
Figura 44: Servicio AgregarEmpleado.....	223
Figura 45: Servicio AgregarEmpleado 2.....	224
Figura 46: Controlador Index	224
Figura 47: Servicio ObtenerEmpleados	225
Figura 48: Validaciones.....	226
Figura 49: Validaciones 2.....	226
Figura 50: Validaciones 3.....	227
Figura 51: Validaciones 4.....	227
Figura 52: Validaciones 5.....	227
Figura 53: Módulos alcanzados	228

Resumen

El tema del presente trabajo de grado consiste en la elaboración de un prototipo funcional para la gestión del talento humano, así como el cálculo correcto de la nómina salarial y liquidaciones de los empleados, entre otros, para la empresa Nutrimax, la cual es una macrobiótica que se encarga de la venta de gran variedad de productos que promueven la alimentación saludable y el bienestar integral. Es importante destacar que la empresa no cuenta con un sistema automatizado y todos los procesos los hacen de manera manual o con cálculos en hojas de Excel propensos a manipulación y errores humanos. Este documento está organizado en las siguientes secciones: introducción, marco referencial, marco metodológico, desarrollo, conclusiones y recomendaciones.

En la sección correspondiente de la introducción, se incluye el planteamiento del problema que sustenta el desarrollo de este proyecto, así como los objetivos tanto general como específicos, justificación, los estudios de viabilidad de la propuesta y proyecciones.

En el apartado del marco referencial, se detallan los principales conceptos para el entendimiento de tema elaborado.

En el capítulo del marco metodológico utilizado durante este trabajo, se especifican el tipo de investigación por realizar, el cuadro de variables que corresponde a los objetivos planteados, los instrumentos de recolección de datos utilizados y también el análisis de los datos obtenidos por medio de los instrumentos usados como las encuestas, entrevistas realizados a los usuarios de la compañía.

En el capítulo de desarrollo, se realiza el análisis de la arquitectura del sistema y el *software*, así como el diagrama de la base de datos. También se muestran los diagramas de flujo relevantes para la creación del prototipo, los casos de uso, pantallas del diseño de entrada y salida del sistema, parte de la programación realizada y las pruebas del prototipo.

En el capítulo de las conclusiones, se exponen los resultados logrados a partir del desarrollo de la investigación y del prototipo. Además, se agregan las recomendaciones pertinentes a partir del desarrollo del documento, así como la elaboración del prototipo y de su posible implementación.

CAPÍTULO I: INTRODUCCIÓN

Planteamiento del problema

El proyecto del sistema de recursos humanos se desarrollará en una macrobiótica llamada NutriMax, y es una iniciativa que busca fortalecer y optimizar la gestión del talento humano dentro de la macrobiótica. Desde su fundación el 26 de marzo del 2004 en Curridabat, NutriMax ha sido un ejemplo de emprendedurismo y compromiso con la calidad y el bienestar.

En sus primeros años, NutriMax se destacó por la producción y distribución de un hidratante natural que gozó de gran aceptación en la zona norte del país. Con el tiempo, la empresa se enfocó en la venta de productos alimenticios de alta calidad, suplementos alimenticios y bebidas, atendiendo a un público específico entre los 30 y 65 años con un nivel socioeconómico de medio a medio alto. A lo largo de sus aproximadamente 20 años de trayectoria en el mercado, NutriMax ha enfrentado desafíos y altibajos, lo que ha llevado a la necesidad de implementar estrategias innovadoras para mantenerse competitivos y adaptarse a las cambiantes demandas del entorno empresarial.

En la actualidad, NutriMax cuenta con una tienda física ubicada en Curridabat, donde ofrece una amplia variedad de productos que promueven la alimentación saludable y el bienestar integral. El equipo de la empresa está compuesto por siete empleados, entre los cuales se encuentran la dueña, jefatura de tienda, vendedores, encargada de planillas, mensajeros, entre otros, quienes trabajan en conjunto para ofrecer un servicio de calidad y una experiencia satisfactoria a los clientes.

Uno de los principales problemas es que la macrobiótica no dispone de un método automatizado para el cálculo de las planillas, lo cual implica que se lleve a cabo de forma manual. Esta situación conlleva el riesgo de que se produzcan errores en los pagos, lo cual resulta en remuneraciones superiores o inferiores a las correspondientes para los empleados. Además, la macrobiótica lleva el control de las solicitudes de manera manual, lo que puede provocar que se manipulen las horas extra que se les da a los empleados, esto puede ser más o menos cantidad.

Aunado a ello, actualmente, no hay un orden establecido en la administración y el manejo de los permisos solicitados por los colaboradores, lo cual genera problemas de comunicación a lo interno de la macrobiótica.

Objetivo general

Desarrollar un prototipo funcional de un sistema de gestión de recursos humanos que facilite la gestión del personal y realice el cálculo preciso de la nómina salarial en la empresa NutriMax.

Objetivos específicos

Analizar los requerimientos y los procesos existentes de la empresa NutriMax.

Diseñar un prototipo funcional que permita simplificar el proceso de cálculo de nómina para los trabajadores de la empresa NutriMax.

Programar el prototipo empleando las herramientas para el desarrollo del aplicativo web y manejador de base de datos.

Probar el prototipo funcional para garantizar su funcionalidad, eficiencia y seguridad en la gestión del personal.

Justificación

En un entorno empresarial en constante cambio, la gestión eficaz de los recursos humanos se ha convertido en un factor clave para el éxito de las empresas. Sin una gestión adecuada de los recursos humanos, las organizaciones corren un alto riesgo de utilizar procesos obsoletos e ineficientes, como la gestión de la nómina mediante hojas de Excel, que no solo son propensas a errores humanos sino también a manipulación.

Crear un sistema automatizado de recursos humanos no es solo una necesidad, sino una inversión que beneficiaría a Nutrimax en muchas áreas, incluida la puntualidad y seguridad de los pagos de nómina, evitando conflictos y errores humanos. También la gestión de las vacaciones y permisos de forma eficiente ya que los empleados podrán realizar solicitudes de forma sencilla y los responsables de recursos humanos y jefaturas podrán aprobarlas de forma eficiente.

Viabilidad operativa

A nivel organizacional, Nutrimax es una empresa establecida en el mercado costarricense, con casi 20 años desde su fundación. Adicionalmente, cada empleado conoce con claridad sus

ocupaciones diarias, alcances y responsabilidades de su cargo. Además, la forma en la cual se desarrollará el prototipo no representa un posible inconveniente en la usabilidad del usuario. Los lenguajes y las arquitecturas utilizadas permiten crear una aplicación sencilla, fácil de entender y le facilita al usuario adaptarse rápidamente.

El personal actual cuenta con conocimientos suficientes en el uso de computadoras, esto significa que están familiarizados con operaciones básicas y herramientas tecnológicas. Aprovechando estos conocimientos, el personal, con una breve capacitación podrán hacer uso de la nueva herramienta, dando viabilidad operativa a la implementación del sistema. La fase de capacitación no forma parte de esta práctica profesional.

No se requieren cambios en la estructura organizativa de la empresa, por lo que no será requerido contratar nuevo personal, así como el prototipo no sustituirá las actividades de ningún empleado.

Viabilidad técnica

Para la elaboración de este prototipo funcional se requerirá una computadora con procesador Intel i9 con 16 GB de memoria RAM, 1 TB de disco duro y un disco SSD de 256 GB para el arranque del sistema operativo, incorporando Windows 11 como el sistema operativo del computador. Dicho computador ya se encuentra a disposición.

La empresa posee con una red por cable Ethernet (LAN) e Internet inalámbrico local (WLAN), de una conexión de 200 MB de descarga y 200 MB de subida, que para efectos de una posible implementación es suficiente. Con respecto a los recursos técnicos, la empresa cuenta con computadores con el acceso a Internet.

En lo que se refiere a las herramientas de *software* necesarias para la elaboración del prototipo, se puntualizan las aplicaciones que se van a utilizar:

- Visual Studio 2022.
- ASP.NET Core 8.0
- C#
- JavaScript
- JQuery 3.4.1
- Bootstrap v5.2.3
- CSS 3

- HTML 5
- MySQL Workbench 8.0.43 Community

Las herramientas mencionadas anteriormente son de uso libre, por lo tanto, no genera costos económicos. A su vez, cabe resaltar que, para el almacenamiento de la información y del aplicativo, se planea realizar un contrato con Microsoft Azure, la cual proveerán los servicios necesarios para las bases de datos, ISS y servicios en la nube.

Dado lo indicado anteriormente, y teniendo en cuenta el uso de la tecnología de Microsoft Azure, no será necesario en la empresa instalar servidores web, bases de datos, para que el sistema funcione correctamente. Se puede concluir que el prototipo cumple con los requisitos de viabilidad técnica establecidos.

Viabilidad Económica

Para la elaboración de este proyecto, se detallan la descripción, la cantidad y el valor económico del *hardware*, *software*, servicios y costos del recurso humano:

Tabla 1: Descripción y costos por hardware

Rubro	Costo unitario	Cantidad	Costo total (tipo de moneda)
Computador con procesador Intel Core i7-1355U, con 32 GB de memoria RAM, 1 TB memoria SSD.	€509 000	1	€509 000
Gran total			€509 000

Fuente: Costos extraídos del sitio web Amazon (2023).

El computador descrito ha sido previamente adquirido para un proyecto anterior, por lo que está disponible y no representa ningún costo adicional para el desarrollo de este proyecto.

Tabla 2: Descripción y costos de software

Rubro	Costo unitario	Cantidad	Costo total (tipo de moneda)
Visual Studio 2022	€0	1	€0
ASP.NET Core 8.0	€0	1	€0
C#	€0	1	€0
JavaScript - Jquery 3.4.1	€0	1	€0
Bootstrap v5.2.3 - CSS 3 – HTML 5	€0	1	€0
MySQL Workbench 8.0.43 Community	€0	1	€0
Gran total			€0

Fuente: Elaboración propia.

Los programas utilizados, *frameworks*, IDEs y versiones, son productos de licencia libre, por lo que no genera gastos respectivos.

Tabla 3: Descripción y costos de servicios de base de datos y host

Rubro	Descripción	Costo/Estimado (mes) Unitario	Cantidad	Costo/Estimado (mes) Total
Sitio Web – Azure App Service (Linux, Plan básico B1)	1 instancia, 1 vCPU / 1.75 GB RAM, 10 GB	€6630	1	€6630
Base de datos – Azure Database for MySQL (Flexible Server, SKU de entrada)	1 vCore, 2–4 GB RAM, 20–50 GB almacenamiento, backup 30 días	€6262	1	€6262

Almacenamiento MySQL	20 GB en disco	₡1370	1	₡1370
Backups/archivos (Blob Storage)	50–100 GB/mes	₡454	1	₡454
Gran total				₡14716

Fuente: Costos extraídos del sitio web de Microsoft Azure (2025)

Los productos de almacenamiento de datos y el *hosting* se planean efectuar a través de Microsoft Azure, los costos preliminares pueden verse ligeramente alterados con relación al uso y administración de datos. Dichos costos son tomados de Microsoft Azure y serán asumidos en su totalidad por la empresa Nutrimax.

Tabla 4: Descripción y costos de mano de obra

Rubro	Costo unitario por hora	Cantidad de horas	Costo total (tipo de moneda)
Análisis	₡1600	60	₡96 000
Diseño	₡1600	120	₡192 000
Desarrollo	₡1600	480	₡768 000
Pruebas	₡1600	100	₡160 000
Gran total			₡1 216 000

Fuente: Costos extraídos del sitio web Ministerio del trabajo Costa Rica [MTSS] (2023).

El salario que se detalla para el programador es un promedio mensual de horas netas y no va a representar una inversión para la empresa, ya que se le va a desarrollar al cliente de manera gratuita debido a que es un proyecto de graduación.

Los valores son calculados según la lista de salarios mínimos para el año 2023, provisto por el MTSS (2023).

Considerando el desglose de montos preliminar, la compañía está en la capacidad económica para la ejecución del proyecto, por lo tanto, se concluye que el prototipo cuenta con la viabilidad económica para su desarrollo.

Viabilidad legal

Como parte de la investigación legal, se ha realizado una búsqueda de las licencias de las herramientas a utilizar para el proyecto. Todas ellas son licencia libre, por lo que no habrá ninguna limitación en el inicio del desarrollo.

Con respecto a las leyes de nuestro país, se respetará la Ley de Protección de la persona frente al tratamiento de sus datos personales (Ley N.º 8968). Los datos personales de los empleados y del personal serán mantenidos únicamente a lo interno de la empresa y no serán compartidos con ninguna otra entidad.

La Ley de Derechos de Autor y Derechos Conexos (Ley N.º 6683) también será observada, y no se utilizará código propiedad de terceros y se mencionará en las respectivas citas bibliográficas y referencias cualquier texto de terceros. Al mismo tiempo, el prototipo elaborado no será distribuido sin el consentimiento del desarrollador y será utilizado únicamente para los fines por los cuales será diseñado. Además, se acatarán las disposiciones del código penal (Ley N.º 4573) y sus reformas introducidas por la Ley N.º 8148, que adiciona, entre otros, los artículos 196 bis, 217 bis y 229 bis, vinculados a conductas como acceso indebido, fraude y sabotaje informático.

Dado lo anterior, el prototipo cumple con la viabilidad legal requerida.

Proyecciones

Con este proyecto, se pretende optimizar la administración de los recursos humanos de la empresa, incluyendo la gestión de los datos de los trabajadores, procesos de nómina, evaluaciones de desempeño. Se pretende facilitar procesos reduciendo la carga administrativa y reduciendo errores en las tareas diarias.

Unos de los objetivos es la de mejorar la experiencia de los empleados brindando información y recursos de una manera eficiente. La organización se beneficia automatizando labores administrativas lo cual liberará tiempo y reducirá errores, además el prototipo cumplirá las leyes y las regulaciones laborales. A largo plazo, la reducción de errores y el manejo correcto de los recursos humanos pueden llevar a una disminución de los costos. Entre los alcances del sistema, es que se llevaran un registro completo y seguro de los datos de los trabajadores, así como

historiales laborales. También, por medio del sistema, se podrán realizar análisis de datos estratégicos para la toma de decisiones futuras.

Alcance funcional

El alcance funcional del sistema de recursos humanos de Nutrimax será permitir el almacenamiento de grandes cantidades de datos, incluidos datos personales, registros de empleados e historial de nómina. Además, tiene disponibilidad continua porque es un sistema web que debe estar disponible las 24 horas del día, los 7 días de la semana. También permite la creación y gestión de perfiles personales de empleados, lo que facilita el seguimiento de información personal como el historial laboral. Además, este será un sistema escalable ya que podrá expandirse si aumenta la fuerza laboral. Entre los módulos que tendrá el sistema serán:

- **Calcular planilla:** Este módulo tomará los datos de los empleados, tales como las deducciones que se les deben realizar, las horas extra que han realizado y las horas naturales laboradas por el empleado para realizar un cálculo automático de la planilla.
- **Calcular extras:** Este módulo realizará una solicitud de horas extra electrónica por medio un formulario mostrado en pantalla que el usuario deberá llenar y enviar por medio de un botón para que sea aprobada o denegada la solicitud.
- **Calcular deducciones:** Este módulo se encargará tomar parámetros como permisos de ausencia, llegadas tardías, monto del seguro social, entre otros, para que se pueda tener la certeza del monto de deducciones que se le debe aplicar a determinado empleado.
- **Calcular liquidaciones:** Este módulo se encargará de tomar datos de la base de datos que le corresponden al usuario que se va a liquidar para hacer el cálculo de manera automática y mostrar el resultado en un campo de la pantalla.
- **Calcular permisos:** Este módulo se encargará de llevar el control de los permisos que toman los empleados, enviar una solicitud para autorizar el permiso del empleado y descontar la cantidad de días tomados de la cantidad de permisos disponibles.
- **Calcular aguinaldo:** Este módulo se encargará de realizar de manera automática los cálculos necesarios para determinar el aguinaldo de los colaboradores. Para el

cálculo, incluirá las horas ordinarias, horas extra, así como cualquier otro pago salarial que se haya realizado en el período.

- Calcular incapacidades: Este módulo se encargará de gestionar el cálculo de incapacidades laborales de los colaboradores, tomando como base los datos registrados en la base de datos y las normativas vigentes del Código de Trabajo. El sistema permitirá registrar la fecha de inicio y finalización de la incapacidad, el tipo de incapacidad (por enfermedad común, accidente laboral, maternidad, entre otros), y calcular automáticamente el monto correspondiente según las políticas internas y las obligaciones patronales.
- Solicitar vacaciones: Este módulo se encargará del cálculo y administración de las vacaciones de los empleados, enviando a jefatura una solicitud para aprobar las vacaciones, luego de aprobada la solicitud, se toman las vacaciones disponibles y se descuentan las vacaciones que tomó el empleado, manejando estos datos por medio de la base de datos.
- Evaluar empleados: Este módulo se encargará de tomar criterios estandarizados por la empresa para realizar una evaluación más concreta.
- Mantenimientos: Este módulo se encargará de realizar el borrado, inserción, modificación, actualización de datos.
- Consultas: Este módulo se encargará de generar información proporcionada de las diferentes tablas.
- Reportes: Este módulo se encargará 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
- Seguridad: Este módulo se encargará de realizar la autenticación de contraseñas y definición de perfiles.

Alcance metodológico

El método que se usará para el desarrollo del *software* será la metodología ágil, en este caso, Scrum. “Scrum es un marco de gestión de proyectos de metodología ágil que ayuda a los equipos a estructurar y gestionar el trabajo mediante un conjunto de valores, principios y prácticas” (Proyectos Ágiles, 2021).

Scrum es un marco de trabajo ágil el cual se utiliza para el desarrollo de proyectos, en el cual se van realizando entregables del producto final. El equipo de trabajo de Scrum consiste en los Equipos Scrum, roles, eventos, artefactos y reglas. Cada componente de trabajo sirve para un propósito.

Ciclo de vida del software

El ciclo de vida del desarrollo de *software* (en inglés: SDLC – Systems Development Life Cycle) es la estructura que contiene los procesos, actividades y tareas relacionadas con el desarrollo y mantenimiento de un producto de *software*, abarcando la vida completa del sistema, desde la definición de los requisitos hasta la finalización de su uso (Ungoti, 2021).

MVC

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 (García, 2021).

La vista es el componente que se encarga de generar la interfaz de usuario. El modelo se encarga de manejar los datos y la información. El controlador es un intermediario entre los anteriores componentes, que se encarga de indicar la vista y al modelo como deben de actuar (García, 2021).

Alcance tecnológico

La tecnología que se va a usar en el desarrollo será un desarrollo web, a través de la programación en visual studio, con C#, el cual permite crear aplicativos web.

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* (Anandmeg, 2023).

Entre las múltiples ventajas que ofrece la herramienta, se podrían describir:

Visual Studio proporciona a los desarrolladores un entorno de desarrollo enriquecido para desarrollar código de alta calidad de forma eficaz y colaborativa.

- Instalador basado en cargas de trabajo: instale solo lo que necesita.
- Herramientas y características de codificación eficaces: todo lo que necesita para compilar sus aplicaciones en un solo lugar.
- Compatibilidad con varios lenguajes: código en C++, C#, JavaScript, TypeScript, Python, etc.
- Desarrollo multiplataforma: compilación de aplicaciones para cualquier plataforma.
- Integración del control de versiones: colaboración en el código con compañeros de equipo (Anandmeg, 2023).

CAPÍTULO II: MARCO REFERENCIAL

En este capítulo se tiene como principal propósito presentar y definir los diferentes conceptos que permitirán una mayor comprensión del desarrollo del prototipo realizado, creando un marco de referencia de las definiciones dadas por diferentes autores.

Para ayudar a la empresa Nutrimax con la problemática actual que poseen, se pretende la creación de un prototipo, el cual podría definirse como una versión inicial de un programa de *software* el cual nos ayudara a encontrar posibles soluciones al problema que se desea resolver, definir conceptos, entre otros. Según (Sommerville, 2011):

Un prototipo es una versión preliminar de un sistema de software que se usa para demostrar conceptos, tratar opciones de diseño y encontrar más sobre el problema y sus posibles soluciones. Los prototipos del sistema permiten a los usuarios ver qué tan bien el sistema apoya su trabajo. Pueden obtener nuevas ideas para requerimientos y descubrir áreas de fortalezas y debilidades en el software. (p. 45)

Con el concepto anterior, se observa que es un prototipo de un sistema, pero se tiene que encontrar la forma de tener acceso al sistema todo el día, a todas horas, esto se puede lograr implementando el aplicativo en un navegador web o una aplicación web, lo cual es un programa o aplicativo que se ejecuta por medio de un navegador web y permite ejecutar diferentes funciones en Internet, como por ejemplo realizar compras, enviar mensajes, realizar videollamadas, editar imágenes, realizar pagos. Las aplicaciones web se comunican con un servidor web alojado en la cual se almacenan los datos y se procesan. Una de las ventajas de un aplicativo web es que no se necesitan que el programa este alojado en las computadoras de los usuarios, ya que puede ser accedido en cualquier lugar con una conexión de Internet. Leandro (2023) define una aplicación web como:

Una aplicación web es cualquier aplicación que es accedida vía web por una red como Internet o una intranet. En general, el término también se utiliza para designar aquellos programas informáticos que son ejecutados en el entorno del navegador (por ejemplo, un applet de Java) o codificado con algún lenguaje soportado por el navegador (como JavaScript, combinado con HTML); confiándose en el navegador web para que reproduzca (renderice) la aplicación. (párrs. 1-2)

Algunas de las ventajas que ofrece la solución de implementar el sistema en la web, según Londoño (2023), son las siguientes:

Las aplicaciones web son accesibles desde todos los navegadores web, por medio de diferentes dispositivos y desde diferentes ubicaciones. La mayor parte de los datos de las aplicaciones web se guardan en la nube, por lo que las empresas no requieren invertir en almacenamiento adicional para ejecutarlas. Por su parte, los usuarios no tienen que descargar las aplicaciones y no se requiere capacidad en el disco duro para usarlas. (párr. 8)

De forma breve, las aplicaciones web permiten acceder por medio de cualquier dispositivo conectado a Internet, ya que se encuentran almacenados en la nube, con lo cual las empresas no tienen que invertir en infraestructura. También, una aplicación web puede permitir ser utilizada por una empresa para la gestión de la información, como lo son los clientes, la información de la empresa, los inventarios o la contabilidad, a través de un navegador web.

Si bien el aplicativo web brinda una gran flexibilidad al admitir el acceso desde múltiples dispositivos, no se debe pasar por alto la importancia de la accesibilidad y el rendimiento cuando se utiliza desde una computadora. Las computadoras son dispositivos los cuales sirven para procesar datos y convertirla en información útil, y estas se controlan mediante programas en las cuales se les da instrucciones de lo que se quiere hacer. Según Marker (2023), la computadora es “un dispositivo informático que es capaz de recibir, procesar, almacenar y devolver información. Una computadora está preparada, para llevar a cabo operaciones lógicas o aritméticas de forma automática cuando el usuario lo indique, mediante comandos a través del software” (párrs. 4-6).

De acuerdo con la definición anterior, una computadora es un dispositivo que manipula los datos recolectados y los convierte en información valiosa para las personas. Las computadoras son manejadas por programas y por el *hardware*, que es la parte física de una computadora, el cual está conformado por componentes electrónicos, mecánicos, memorias, pantallas y otros materiales físicos para el funcionamiento del computador. El *hardware* se define como “el hardware es material real” que se puede ver con los ojos y tocar con los dedos. Es el conjunto de piezas que componen un ordenador y que en conjunto permiten el funcionamiento del equipo al agregar el software” (Figueiras, 2021, párrs. 2-3).

El *hardware* se divide en varias secciones, entre ellas: de procesamiento, los cuales se pueden mencionar el CPU. También de almacenamiento como lo son los discos duros de memoria,

memoria RAM. De gráficos como las tarjetas de video y los periféricos como llaves mayas, teclados, *mouse*. Mientras que el *hardware* constituye la base física de una computadora, es el *software* el que guía y controla las operaciones, este incluye programas, sistemas operativos y aplicaciones. Al respecto, Figueiras (2021) define el *software* como “el conjunto de códigos, programas, aplicaciones y demás factores no físicos presentes en nuestros ordenadores. Algunos ejemplos de *software* son el BIOS, el sistema operativo, los programas o los videojuegos” (párr. 8). Dada la definición anterior, el *software* es la parte no física o lógica que le dan las instrucciones al *hardware*, sistema operativo o aplicación para realizar tarea realizar.

Figura 1: Aplicaciones de software



Fuente: (Buho agencia digital, 2022).

El *software* engloba una gran variedad de programas y aplicaciones, pero el sistema o cerebro que manda y ordena todas las instrucciones a la computadora sería el sistema operativo. El sistema operativo es la parte más importante de una computadora ya que controla todo lo que hace el computador entre el *hardware* y *software* de manera eficiente. Este se encarga de controlar la memoria, el CPU, los archivos y toda la información del sistema. De acuerdo con Bigelow (2021):

Un sistema operativo (SO) es el programa que, después de ser cargado inicialmente en la computadora por un programa de arranque, administra todos los demás programas de aplicación en una computadora. Los programas de aplicación hacen uso del sistema

operativo al realizar solicitudes de servicios a través de una interfaz de programa de aplicación (API) definida. (párr. 1)

Mientras que el sistema operativo es la columna vertebral que gestiona y coordina todas las operaciones de las computadoras, un programa está diseñado para llevar a cabo una tarea específica.

Los programas se definen como una serie de instrucciones que guardan y manipulan los datos en la memoria de una computadora y les dice qué hacer con ellos. Estos se desarrollan en un lenguaje de programación entendible para las personas. Según Marker (2022), “un programa de computadora, aplicación o *software*, son un conjunto de instrucciones en forma secuencial, llamado código, que a través de su interpretación por el sistema operativo o *hardware*, le permiten desarrollar una acción específica a una computadora” (párr. 8). De acuerdo con lo anterior, los programas tradicionales se ejecutan en dispositivos locales, pero actualmente existe una tecnología más novedosa, en la cual muchas compañías están implementando sus programas o sistemas desde la famosa nube, esto permite utilizar los aplicativos en cualquier dispositivo conectado a Internet.

En la actualidad, la computación en la nube es una alternativa que ha tomado mucha fuerza debido a las grandes ventajas que esta nos ofrece, como el uso de servidores potentes, infraestructura alquilada el cual se paga y se alojan los aplicativos, así como el mantenimiento es de parte del proveedor. De acuerdo con Grapsas (2021):

La computación en la nube (o cloud computing, en inglés) es una tecnología que permite acceder remotamente, de cualquier lugar del mundo y en cualquier momento, a softwares, almacenamiento de archivos y procesamiento de datos a través de Internet, sin la necesidad de conectarse a un ordenador personal o servidor local. (párr. 3)

Se puede definir como una manera de acceder a aplicaciones, servidores, recursos informáticos por medio de la Internet. Son servicios de pago brindado por proveedores en la cual se pueden alojar las aplicaciones creadas sin tener que instalarlas en las máquinas.

Para el desarrollo de este proyecto, se ha seleccionado la plataforma de Microsoft Azure para implementar el aplicativo en la nube sin tener que incurrir en los grandes gastos de compra se servidores, equipo, redes, entre otros. Microsoft Azure es un servicio en la nube creado por Microsoft, en el cual se pueden implementar, probar y administrar aplicaciones mediante sus plataformas. Azure ofrece gran variedad de servicios como infraestructura, procesos, bases de datos, redes en la cual la organización paga por lo que se utiliza. De acuerdo con Conzultek (s.f.):

“Microsoft Azure es una plataforma de pago por uso que integra servicios en la nube pública para que desarrolladores y equipos de TI administren e implementen aplicaciones y otros recursos a través de un gran centro de datos mundial” (párr. 1).

Figura 2: Herramienta de Microsoft Azure



Fuente: (Conzultek, s.f.).

Un concepto clave dentro de la computación en la nube y de Microsoft Azure es PaaS, que significa Plataforma como servicio, y es un tipo de computación en la nube, el cual ofrece una plataforma completa para desarrollar y gestionar las aplicaciones sin tener que mantener la infraestructura necesaria. El proveedor del servicio es el que se encarga del soporte y de proporcionar el *hardware* y el *software*. El cliente únicamente paga por los servicios que va a utilizar. Según Grapsas (2021), en este enfoque de computación en la nube, se obtiene un entorno de desarrollo completo según la necesidad, que permite la creación y modificación de *software* y aplicaciones. La principal ventaja del modelo PaaS es que cubre desde sistemas operativos hasta herramientas de desarrollo, sistemas de gestión de bases de datos, servicios de inteligencia empresarial y otros recursos, así como la infraestructura básica para ejecutar y mejorar aplicaciones web.

Esta PaaS proporciona un entorno para el desarrollo e implementación de aplicaciones web, pero previamente se debe poseer un componente principal y muy importante, las bases de datos, el cual se podrá integrar en el aplicativo web.

Una base de datos es un conjunto de datos ordenados y estructurados que representan una realidad y están ordenados independientemente de las aplicaciones, lo cual pueden ser manipulados por usuarios y diferentes aplicaciones. Una base de datos se compone de tablas, que son una colección de filas que representan objetos, y columnas que representan los atributos. De acuerdo con Gómez (2017): “Una base de datos es un conjunto ordenado y estructurado de datos que representan una realidad objetiva y que están organizados independientemente de las aplicaciones, significa que puedan ser utilizadas y compartidas por usuarios y aplicaciones diferentes” (p. 11).

Para simplificar la interacción con las bases de datos, se emplean herramientas determinadas conocidas como Sistemas de Gestión de Bases de Datos. Gómez (2017) explica que:

Un sistema de gestión de bases de datos (SGBD) consiste en una colección de datos interrelacionados y un conjunto de programas para acceder a dichos datos. El objetivo principal de un SGBD es proporcionar una forma de almacenar y recuperar la información de una base de datos de manera que sea tanto práctica como eficiente. (p. 11)

De acuerdo con la definición anterior, los sistemas de gestión de bases de datos son un conjunto de programas diseñados para acceder a los datos. Como menciona Gómez (2017):

Los sistemas de bases de datos se diseñan para gestionar grandes cantidades de información. La gestión de los datos implica tanto la definición de estructuras para almacenar la información como la provisión de mecanismos para la manipulación de la información. Además, los sistemas de bases de datos deben proporcionar la fiabilidad de la información almacenada, a pesar de las caídas del sistema o los intentos de acceso sin autorización. Si los datos van a ser compartidos entre diversos usuarios, el sistema debe evitar posibles resultados anómalos. (p. 11)

Un Sistema Gestor de Base de Datos (SGBD) es una recopilación de datos y programas para gestionar, almacenar y acceder a dichos datos. El objetivo es proporcionar una forma de almacenar y recuperar información de una base de datos de manera sencilla. Estos sistemas se diseñan para recuperar grandes cantidades de información.

Entre los diferentes sistemas disponibles, se ha seleccionado MySQL como el gestor de base de datos para el desarrollo del prototipo funcional. MySQL es un sistema de gestión de bases de datos relacional de código abierto, ampliamente utilizado en aplicaciones web y empresariales por su rendimiento, escalabilidad y compatibilidad con múltiples plataformas. Permite almacenar,

consultar y manipular datos de manera eficiente, utilizando el lenguaje SQL como estándar de interacción.

Para la administración visual de la base de datos, se emplea MySQL Workbench, una herramienta oficial que proporciona una interfaz gráfica para modelar, diseñar, consultar y administrar bases de datos MySQL. Según Informattec Digital (s.f.), “MySQL Workbench reúne herramientas visuales avanzadas para diseño, modelado, administración y migración de bases de datos MySQL, disponible tanto en versión gratuita open-source como comercial”. Además, Informattec Digital (s.f.) destaca que “es software disponible ampliamente (sin restricciones) y compatible con Windows y Mac”.

El uso de MySQL Workbench se alinea con el entorno de desarrollo basado en ASP.NET Core y C#, ya que permite una integración fluida mediante conectores compatibles y facilita la implementación de operaciones CRUD dentro de la lógica del sistema. Esta elección responde a criterios de eficiencia, accesibilidad y adecuación al contexto técnico del proyecto. MySQL Workbench es un componente esencial cuando se hace referencia a entornos de desarrollo, y .NET es uno de los marcos de desarrollo más utilizados para crear aplicaciones.

Es una plataforma para los desarrolladores que permite crear gran cantidad de aplicaciones y en el cual se pueden utilizar varios lenguajes de programación como C#. Según Microsoft (s.f.): “.NET es una plataforma para desarrolladores de código abierto, multiplataforma y gratuita para crear muchos tipos de aplicaciones diferentes. Con .NET, puede usar varios lenguajes, editores y bibliotecas para crear aplicaciones web, móviles, de escritorio, juegos, IoT, etc”.

Para el desarrollo de este prototipo, se plantea utilizar el marco de desarrollo de ASP.NET puesto que, que permite la creación de páginas web dinámicas a través de un lenguaje de programación integrado. Permite el desarrollo en varios lenguajes como C# y Visual Basic.NET.

De acuerdo con Rodríguez (2023):

Es un marco de trabajo (framework) desarrollado por Microsoft para construir aplicaciones web y servicios en línea. El nombre “ASP.NET” se refiere a “Active Server Pages .NET“, lo que hace referencia a la evolución de la tecnología ASP (Active Server Pages) hacia el entorno de desarrollo .NET. ASP.NET permite a los desarrolladores construir aplicaciones web dinámicas, robustas y escalables utilizando varios lenguajes de programación, como C# (C Sharp) y Visual Basic.NET. (párr. 2-3)

Al utilizar la plataforma de ASP.NET, se puede escribir el programa que se va a diseñar por medio de varios lenguajes de programación, entre ellos C#. De acuerdo con Microsoft (2023):

C# 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# 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*. (párr. 2-3)

Figura 3: Demostración de C# como lenguaje multiplataforma



Fuente: (Microsoft.com, s.f.).

C# es un lenguaje moderno y multiplataforma orientado a objetos que se usa para crear aplicaciones en la plataforma de .NET. C# se inspiró en lenguajes como C, C++ y Java. Con este lenguaje se pueden desarrollar aplicaciones web, de escritorio y juegos. Es uno de los lenguajes más populares. Entre las características de C#, es que se puede implementar un patrón de diseño llamado MVC para desarrollar aplicaciones estructuradas y escalables.

El patrón de diseño MVC se define como una guía para el desarrollo de aplicaciones en la cual la aplicación se divide en 3 modelos, el primer es un modelo que representa los datos del aplicativo, el segundo son las vistas que muestra información, y el tercer modelo son los controladores que procesan las peticiones entre los usuarios del aplicativo y el cual controla la ejecución del sistema. El patrón de diseño Modelo-Vista-Controlador, según Hernández (2020):

[...] es un patrón de diseño que separa una aplicación en tres componentes lógicos principales: Modelo, Vista y Controlador, de ahí la abreviatura. Cada componente de la arquitectura está construido para manejar un aspecto de desarrollo específico de una aplicación. La arquitectura MVC separa la lógica empresarial y la capa de presentación

entre sí. Se utilizaba tradicionalmente para interfaces gráficas de usuario (GUI) de escritorio, pero hoy en día se ha vuelto popular para diseñar aplicaciones web y aplicaciones móviles. (párr. 2-3)

Para este prototipo de recursos humanos para la macrobiótica, se decide utilizar el modelo MVC para organizar la información y separar el aplicativo en tres modelos separados, el cual el primer modelo representa la información relacionada como datos personales, las vistas representan los formularios y el aplicativo web que se va a acceder y los controladores procesan las solicitudes de los usuarios. La arquitectura MVC fracciona los objetos o componentes implicados en una aplicación en 3 tipos: modelo, vista y controlador. De acuerdo con Pezua (2023), el modelo del diseño MVC:

Es la capa responsable de manejar los datos y la lógica de negocio de la aplicación. En esta capa se define cómo se almacenan y se manipulan los datos, y cómo se lleva a cabo la lógica de negocio necesaria para que la aplicación funcione correctamente. (párr. 4)

También Pezua (2023) define la vista como “la capa que se encarga de mostrar los datos del modelo al usuario y de permitir la interacción con la aplicación. Esta capa se centra en la presentación y la interfaz de usuario de la aplicación” (párr. 7).

Por último, el controlador es:

La capa que se encarga de gestionar las solicitudes del usuario, procesar la entrada y realizar las acciones necesarias en el modelo y la vista para responder a la solicitud del usuario. Esta capa actúa como intermediario entre la Vista y el Modelo. (párr. 11)

También, es esencial explorar cómo esta estructura se integra con la interfaz de usuario, y aquí es donde entra la definición de HTML, que es un lenguaje para la creación de páginas web, este lenguaje se utiliza para definir el diseño de las páginas, y funciona mediante etiquetas para definir encabezados, párrafos e imágenes y videos dentro de la página. Estas etiquetas son interpretadas por los navegadores web.

Según menciona Coppola (2023), “HTML es la abreviatura de hypertext markup language, lenguaje central de la World Wide Web. Originalmente diseñado como un lenguaje para explicar documentos científicos, se ha adaptado para describir la estructura básica de páginas web y aplicaciones en línea” (párr. 4).

HTML es el lenguaje de marcas o etiquetas para dar el formato a los documentos de la página web, pero es deficiente a la hora de abordar dinamismo e interactividad en la web. Una

solución sería la implementación de JavaScript, que es un lenguaje de programación interpretado, utilizado para crear páginas webs responsivas y el cual es uno de los más populares actualmente. De acuerdo con Gonzalez (2023), “JavaScript es un lenguaje de programación de alto nivel, interpretado y orientado a objetos, diseñado principalmente para agregar interactividad y dinamismo a las páginas web (párr. 2)”. Algunas de las características que ofrece Javascript, según González (2023), son las siguientes:

JavaScript se caracteriza por ser un lenguaje de programación versátil que se ejecuta en el navegador del usuario, lo que permite la creación de páginas web interactivas. Es conocido por su capacidad para modificar el contenido de una página en tiempo real, responder a eventos del usuario y comunicarse con servidores web. (párr. 4)

Así como JavaScript brinda dinamismo a las páginas web, también existe una solución para el aspecto visual y de diseño y el cual será implementado para el desarrollo de este sistema, y es el lenguaje CSS que se usa para la presentación y el aspecto de las páginas web en el cual se pueden controlar opciones como los colores, el diseño y animaciones. Según Santos (2023):

CSS son las siglas en inglés para «hojas de estilo en cascada» (Cascading Style Sheets). Básicamente, es un lenguaje que maneja el diseño y presentación de las páginas web, es decir, cómo lucen cuando un usuario las visita. Funciona junto con el lenguaje HTML que se encarga del contenido básico de los sitios. (párr. 4)

Mientras que CSS controla el aspecto visual, AJAX permite cargar y enviar datos de manera asincrónica en segundo plano, lo que brinda una experiencia más dinámica a los usuarios. AJAX es un conjunto de técnicas de desarrollo web el cual permite que los aplicativos webs se actualicen de forma tal que no se tenga que recargar toda la página, sino que lo hacer de forma asíncrona, en segundo plano. De acuerdo con Saavedra (2023), AJAX es una abreviación de Asynchronous JavaScript and XML que se traduce como “JavaScript asíncrono y XML”. Este término describe un grupo de tecnologías que se usan para obtener datos del servidor en segundo plano y usarlos para actualizar la página sin recargarla. El objetivo principal de AJAX es hacer que los sitios y las aplicaciones web sean más fáciles de usar, más rápidos y con mayor capacidad de respuesta.

Para simplificar la programación en el lenguaje de JavaScript, el uso de la biblioteca JQuery ayudará a simplificar la programación de ese lenguaje. Funciona para el manejo de eventos y manipulación de documentos HTML.

jQuery es una biblioteca JavaScript rápida, pequeña y rica en funciones. Hace cosas como recorrido y manipulación de documentos HTML, manejo de eventos, animación, y Ajax mucho más simple con una API fácil de usar que funciona en una multitud de navegadores. Con una combinación de versatilidad y extensibilidad, jQuery ha cambiado la forma en que millones de personas escriben JavaScript. (jQuery.com, s.f.)

Para ofrecer una experiencia de desarrollo web más completa y atractiva, se utilizará Bootstrap. Este es un *framework* web de código abierto que facilita el desarrollo y la creación de sitios web responsivos y que se adaptan a dispositivos móviles. Es decir, el sitio o aplicación se adaptará al teléfono o *smartphone* de la persona que acceda a la página. Según el sitio Hostinger tutoriales (2023), Bootstrap es un *framework* de desarrollo web y de código abierto. Está diseñado para proporcionar el proceso de desarrollo de los sitios web responsivos y orientados a los dispositivos móviles, proporcionando una colección de sintaxis para diseños de plantillas. Bootstrap ayuda a los programadores a construir sitios web más rápidamente, ya que no tienen que preocuparse por los comandos y funciones básicos. Consta de scripts basados en HTML, CSS y JS para diversas funciones y componentes relacionados con el diseño web.

Como parte de las definiciones de herramientas que serán utilizadas para el desarrollo del programa, también es importante mencionar las técnicas o metodologías que existen para el desarrollo de *software*. Las metodologías de desarrollo de *software* son un conjunto de técnicas, prácticas y herramientas que se utilizan para diseñar, probar y entregar *software* de una alta calidad de manera eficiente. Se aplican en diferentes etapas del ciclo de vida del *software*. Según Maida et al. (2015):

Una metodología de desarrollo de *software* es un marco de trabajo que se usa para estructurar, planificar y controlar el proceso de desarrollo de sistemas de información. Una gran variedad de estos marcos de trabajo han evolucionado durante los años, cada uno con sus propias fortalezas y debilidades. Una metodología de desarrollo de sistemas no tiene que ser necesariamente adecuada para usarla en todos los proyectos. Cada una de las metodologías disponibles es más adecuada para tipos específicos de proyectos, basados en consideraciones técnicas, organizacionales, de proyecto y de equipo. (p. 14)

En la actualidad, existen diversas metodologías para el desarrollo de *software*, como se describe enseguida:

Existen las metodologías tradiciones que son un conjunto de prácticas y técnicas para construir y entregar *software* de alta calidad y se aplican en diferentes etapas del ciclo de vida del *software*. Algunos ejemplos de son el modelo en cascada, espiral. Estas son adecuadas para para proyectos con requisitos bien definidos. De acuerdo con Maida et al. (2015):

Las metodologías tradicionales son denominadas, a veces, de forma despectiva, como metodologías pesadas. Centran su atención en llevar una documentación exhaustiva de todo el proyecto, la planificación y control del mismo, en especificaciones precisas de requisitos y modelado y en cumplir con un plan de trabajo, definido todo esto, en la fase inicial del desarrollo del proyecto. Estas metodologías tradicionales imponen una disciplina rigurosa de trabajo sobre el proceso de desarrollo del software, con el fin de conseguir un software más eficiente. Para ello, se hace énfasis en la planificación total de todo el trabajo a realizar y una vez que está todo detallado, comienza el ciclo de desarrollo del producto *software*. (p. 17)

Las metodologías tradicionales en el desarrollo de *software* enfrentan limitaciones cuando se trata de lidiar con la incertidumbre de los requisitos. Como, por ejemplo, en entornos donde es difícil predecir con precisión qué se necesita o donde los requisitos tienden a cambiar. Además de ello, los costes para implementar son altos. Algunas de las metodologías tradicionales que existen son:

- Espiral
- Cascada

Las metodologías ágiles son un conjunto de metodologías de desarrollo de *software* que se centralizan en la colaboración, la adaptabilidad y la entrega continua. Estas metodologías buscan entregar pequeñas piezas de *software* funcional en un corto período de tiempo para mejorar la satisfacción del cliente. Algunas metodologías populares incluyen Scrum, Kanban y Programación Extrema (XP). Según Maida et al. (2015):

Este enfoque nace como respuesta a los problemas que puedan ocasionar las metodologías tradicionales y se basa en dos aspectos fundamentales, retrasar las decisiones y la planificación adaptativa. Basan su fundamento en la adaptabilidad de los procesos de desarrollo. Un modelo de desarrollo ágil, generalmente es un proceso Incremental (entregas frecuentes con ciclos rápidos), también Cooperativo (clientes y desarrolladores trabajan constantemente con una comunicación muy fina y constante), Sencillo (el método es fácil

de aprender y modificar para el equipo) y finalmente Adaptativo (capaz de permitir cambios de último momento). (p. 18)

Las metodologías ágiles proporcionan un conjunto de directrices y enfoques prácticos que simplifican la entrega de proyectos, resultando en una experiencia más satisfactoria tanto para los clientes como para los equipos de trabajo. Se alejan de los procesos de las metodologías tradicionales, reduciendo la documentación y evitando el apego a métodos formales. Estas metodologías hacen empeño en que la capacidad de adaptación a los cambios es más valiosa que la rigidez de seguir un plan de manera estricta. Algunas de las metodologías ágiles que existen son:

- Scrum
- XP (*Extreme Programming*)

Dentro del amplio espectro de metodologías ágiles, una de las más destacadas y ampliamente utilizadas es Scrum. Scrum es un marco de desarrollo de *software* ágil basado en la colaboración, la adaptabilidad y la entrega continua de productos funcionales.

Figura 4: Proceso de Scrum



Fuente: (Revista Escuela de Negocios y Dirección, 2022)

Scrum consta de roles, eventos y artefactos que ayudan a los equipos a estructurar y gestionar su trabajo. Según Martins (2023):

Scrum es un marco de trabajo ágil a través del cual las personas pueden abordar problemas complejos adaptativos a la vez que se entregan productos de forma eficiente y creativa con el máximo valor. Así, Scrum es una metodología que ayuda a los equipos a colaborar y

realizar un trabajo de alto impacto. La metodología Scrum proporciona un plan de valores, roles y pautas para ayudar a tu equipo a concentrarse en la iteración y la mejora continua en proyectos complejos. Por otra parte, deberás saber que en Scrum se trabaja con equipos pequeños multidisciplinares en ciclos iterativos centrados en el cliente y se crea un producto de forma incremental.

Scrum es un enfoque ágil que permite a las personas abordar problemas complejos, al mismo tiempo que se entregan productos de manera eficiente y creativa, aumentando su valor. Se trata de una metodología que fomenta la colaboración en equipos, permitiéndoles lograr un alto impacto. Scrum proporciona un conjunto de principios y roles que ayudan a los equipos a enfocarse en la iteración y la mejora constante en proyectos complejos. En este marco de trabajo, se trabaja con equipos pequeños y se siguen ciclos iterativos centrados en las necesidades del cliente y se construye el producto de manera incremental.

Se ha hablado acerca de todos los conceptos relacionados con las tecnologías que se van a aplicar en el desarrollo de este proyecto, sin embargo, cabe resaltar y explicar de todos los otros tópicos relacionados con los recursos humanos y de su manejo aquí en Costa Rica, de cómo se realizan los cálculos de salarios y de deducciones de ley, así como horas extra, vacaciones y liquidaciones correspondientes de los colaboradores.

Primeramente, hay que empezar por definir el concepto principal del desarrollo de este proyecto, como lo es el concepto de recursos humanos, que según Martín (2023) “los recursos humanos hacen referencia a todo el conjunto de empleados, trabajadores, que posea una vinculación laboral con una organización, economía, o sector en concreto” (párr. 2). Esta definición hace referencia a todos los empleados actuales con los que cuenta la compañía y que comparten una relación laboral con la organización, los cuales contribuyen al funcionamiento y logros de los objetivos de la empresa.

Cabe mencionar que dentro de la compañía existe un departamento de Recursos Humanos que juega un papel de suma importancia en la administración y gestión del talento humano disponible, con el objetivo de conseguir los objetivos propuestos. Este departamento se encarga de diversas funciones relacionadas con la gestión de empleados, entre ellas la contratación del personal por medio de procesos de selección, entrevistas, entre otros. También participa en la capacitación de empleados, en la evaluación del desempeño de los colaboradores, encargado de salarios, beneficios y en la gestión de conflictos para resolver disputas o conflictos laborales.

Además de su papel fundamental en la gestión del talento humano, el departamento de Recursos Humanos desempeña un papel crítico en la gestión financiera de la compañía. Una de las responsabilidades es la administración y cálculo de salarios y beneficios de los empleados. Este proceso se realiza por medio de la elaboración y ejecución de las planillas de pago, el cual certifica que cada colaborador reciba su salario de manera correcta y oportuna. La planilla de pago es un documento por el cual, según Álvaro (2023), “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).

De acuerdo con la cita anterior, se procesan los detalles para el salario de una persona, y en términos generales se puede describir el salario de un empleado como la remuneración que un empleado recibe por sus servicios. Sin embargo, cuando se describen los salarios de los colaboradores, puede hablarse de dos conceptos importantes: el salario bruto y el salario neto. El salario bruto es, según el sitio web Consejo Nacional de Rectores (2021):

Es el monto económico total devengado mensualmente por el funcionario; es decir sin ser sujeto a las deducciones establecidas por la diversa normativa nacional, ni deducciones personales. Para su cálculo se consideran todos los rubros salariales ordinarios que recibe en forma permanente el funcionario. (párr. 1)

Se puede describir el salario bruto como el monto total ganado mensualmente por un empleado, incluyendo horas extra y otros montos, antes de rebajos y cargas sociales obligatorias de ley. Por su parte, el salario neto, según el Consejo Nacional de Rectores (2021):

Es el monto económico devengado mensualmente por el funcionario al cual se le deducen las cargas sociales y pólizas establecidas en la normativa aplicable a nivel nacional e institucional, según corresponda (Deducción por concepto del régimen de Invalidez, Vejez y Muerte de la Caja Costarricense de Seguro Social (CCSS), el régimen de Enfermedad y Maternidad de la CCSS, régimen de pensiones de la Junta de Pensiones del Magisterio Nacional, póliza obligatoria del Magisterio Nacional, Banco Popular, etc...). (párr. 1)

Según el texto anterior, el salario neto corresponde al salario bruto de que recibe el empleado, el cual resulta afectado por las deducciones relacionadas con las cargas sociales y pólizas establecidas a nivel nacional.

A medida que se detallan los conceptos de relacionados con la compensación salarial y las deducciones asociadas, es importante reconocer acerca de los elementos que componen la jornada laboral o jornada ordinaria de trabajo. De acuerdo con MTSS (s.f.), “es el número máximo de horas ordinarias, permitido por ley, en las que la persona trabajadora se encuentra sujeto a las órdenes de la persona empleadora” (párr. 1).

Existen varios tipos de jornadas laborales en Costa Rica, entre ellas se detallan, según las horas laboradas por día, jornadas ordinarias normales o jornadas ordinarias especiales. Además, si se trabaja de día o de noche, se mencionan las jornadas diurnas, nocturnas y mixtas. En el caso de la empresa Nutrimax, solo laboran con jornada diurna.

Los tipos de jornadas diurnas son aquellas en la cual se trabaja en un periodo comprendido de cinco de la mañana a las siete de la noche. Según el MTSS (s.f.), la jornada diurna “es aquella en la cual se trabaja en el período comprendido entre las cinco de la mañana y las siete de la noche. Está compuesta por ocho (8) horas por día y cuarenta y ocho (48) horas por semana” (párr. 3).

En el caso de las jornadas nocturna, son las que los colaboradores realizan sus responsabilidades laborales en la noche. Según MTSS (s.f.):

Es la que se realiza entre las siete de la noche y las cinco de la mañana del día siguiente. Es de seis (6) horas por día y treinta y seis (36) horas semanales. No se permite la jornada acumulativa nocturna. (párr. 4)

Cabe mencionar que, en ocasiones, las organizaciones necesitan apoyo para el cumplimiento de sus objetivos y aquí es donde entran en juego las horas extra, que es el trabajo realizado más allá de las jornadas ordinarias realizadas por los empleados, el cual el cálculo para el pago de estas horas es regulado por el MTSS. Si el pago para un empleado es mensual o de manera quincenal, estas horas laboradas se pagan, tal como señala el sitio web Toptrabajos (2022), “si tu pago es mensual o quincenal, lo haces dividiendo tu sueldo entre 30 días y luego entre las horas que dura tu jornada. Luego multiplicas por 1.5” (párr. 3). Es importante mencionar que para el pago de horas extra en días festivos el pago o el valor equivale se multiplica por dos.

Al explorar las horas extra y su impacto en los ingresos, es esencial considerar cómo estas horas adicionales también influyen en las deducciones de ley que afecta el salario neto del trabajador. Mientras que las horas extra representan un esfuerzo adicional y se ven reflejadas en el aumento de los ingresos brutos, las deducciones de ley entran en juego para garantizar el cumplimiento de las responsabilidades fiscales y del seguro social. Entre las deducciones que se

le aplican a cada trabajador se pueden encontrar la correspondiente a la Caja Costarricense de Seguro Social el cual corresponde a un 10,67 % del total del salario bruto del colaborador mensualmente. Según el sitio web BG&A Abogados Corporativos (2023), “a partir del 01 de enero de 2023, por concepto de aporte de seguridad social se debe retener del salario del trabajador un 10,67%” (párr. 4). También caben mencionar las deducciones del Régimen de Invalidez, Vejez y Muerte (IVM), el Impuesto sobre la renta (ISR), entre otros.

Otro de los puntos que hay que considerar es el del cálculo del aguinaldo, el cual se define como un salario adicional que se le debe de pagar a los empleados, dentro de los primeros veinte días del mes de diciembre de cada año. Según el MTSS (s.f.):

El aguinaldo se calcula con base en todos los salarios ordinarios y extraordinarios, devengados por la persona trabajadora durante los doce meses que van del 1° diciembre del año anterior al 30 de noviembre del año que se trate. Se obtiene de la suma de dichos salarios y se divide entre doce. (párr. 3)

De acuerdo con el MTSS (s.f.), se suman todos los salarios mensuales de la persona, incluyendo horas ordinarias y horas extra, ósea el salario bruto de todos los meses y después se divide la cantidad entre doce. Si la persona empleada estuvo incapacitada o sin goce de salario, se calcula el aguinaldo con base al tiempo en el que recibió salarios y no se toma en cuenta el periodo en que la persona estuvo incapacitada por enfermedad, puesto que recibió un subsidio y no recibió un salario como tal.

Otro de los aspectos cruciales en los derechos laborales en Costa Rica es el tiempo de vacaciones al que tienen derecho los trabajadores. Así como el aguinaldo representa un beneficio anual, las vacaciones son una pausa que permite el descanso a los empleados. Las vacaciones son una parte integral para el bienestar de los empleados, el cual ofrece un tiempo libre remunerado para descansar. Según el MTSS (s.f.), “las vacaciones son un derecho y una necesidad biológica de toda persona trabajadora. Consisten en un descanso anual remunerado, que tiene como propósito permitir a la persona trabajadora reponer el desgaste de energías realizado durante el año de labores” (párr. 1).

De acuerdo con el MTSS (s.f.), la cantidad de días a la cual tiene derecho una persona se describe a continuación:

Toda persona trabajadora tiene derecho a disfrutar dos semanas de vacaciones que equivalen a doce días más dos días de descanso 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. 2)

El pago de las vacaciones de los empleados se calcula con base en el promedio de los salarios ordinarios y extraordinarios recibidos durante las últimas cincuenta semanas en los centros de trabajo, a partir del momento en el que la persona trabajadora adquiere el derecho al descanso y también el pago dependerá del tipo de forma de pago que aplica en la organización. Este cálculo se realiza de la siguiente manera, según el MTSS (s.f.):

Pago semanal: se le deben pagar doce (12) días, porque en esta modalidad no se paga el día de descanso. El cálculo se realiza tomando el salario semanal dividiéndolo entre 6 y multiplicando el resultado por 12.

Pago mensual o quincenal: le corresponden catorce (14) días, porque incluye en el pago los dos días de descanso semanal. El cálculo se realiza tomando el salario mensual dividiéndolo entre 30 y multiplicando el resultado por 14.

En el caso de actividades comerciales, aunque la forma de pago sea semanal, para efectos de vacaciones se aplica la siguiente fórmula: se toma el salario semanal dividiéndolo entre 7 y multiplicando el resultado por 14. (párrs. 4-6)

Las vacaciones representan un tiempo retribuido durante el cual los empleados pueden desconectarse y recuperar energías. Sin embargo, cuando la relación laboral llega a su fin, ya sea por decisión del empleado o del empleador, entra en juego la liquidación, y en el cual surge la necesidad de calcular y proporcionar pagos y beneficios pendientes. La liquidación, según Toptrabajos (2022), “en Costa Rica, como en el resto de países de la región, cuando un trabajador es despedido o renuncia, tiene derecho al pago de las prestaciones acumuladas junto con sus salarios pendientes. A esto se le conoce como liquidación” (párr. 21).

Cabe mencionar que el Código de Trabajo de Costa Rica establece varios motivos por los cuales un contrato por tiempo indefinido puede terminar. De acuerdo con Toptrabajos (2022):

Despido injustificado: Esta se da cuando el empleador es quien finalizar el contrato sin causas atribuibles al trabajador. En estos casos, el trabajador tiene derecho al pago del auxilio de cesantía, que es una indemnización por despido injustificado. Además, debe recibir el pago de prestaciones pendientes. Si el empleador no preavisó con el tiempo de ley, también.

Despido justificado: Si el trabajador ha cometido faltas graves a la normativa de la empresa, el empleador puede despedirlo. En este caso, se pierde el derecho al auxilio de cesantía. Sin embargo, el trabajador sí recibirá el pago de prestaciones pendientes.

Renuncia: Esta se da cuando el trabajador decide retirarse voluntariamente de la empresa. En dicho caso, se pagan las prestaciones, pero no la cesantía. La única excepción es si la renuncia está motivada por razones atribuibles al empleador. Por ejemplo, en casos de acoso, hostigamiento, engaño al trabajador, entre otros. (párr. 26)

Para el cálculo de las liquidaciones, se toman en cuenta diversos aspectos, desde salarios no pagados hasta beneficios acumulados como las vacaciones no disfrutadas y aguinaldo. Además, se debe tener en consideración otros elementos como lo son el preaviso y cesantía, esto depende de la forma en cómo se dio la terminación del contrato. Según Ruiz Jiménez (2022), el preaviso “es una obligación que tiene toda persona trabajadora de avisarle al patrono que está renunciando o en su caso si el patrono está dando por terminada la relación con responsabilidad patronal” (párr. 2).

Según el texto anterior, el preaviso puede describirse como la notificación anticipada ya sea de cualquiera de las partes para la finalización a un contrato laboral. El tiempo del preaviso depende de la duración de la relación laboral, de acuerdo con Ruiz Jiménez (2022):

Trabajo continuo menor de 3 meses no hay preaviso (no se debe pagar nada). Trabajo continuo entre 3 meses a 6 meses sería mínimo 1 semana de anticipación de preaviso. Trabajo continuo que exceda de seis meses y no sea mayor de un año, sería mínimo 15 días de preaviso. Cuando el trabajo continuo y exceda un año entonces sería un mes de preaviso. (párrs. 5-8)

El preaviso se calcula tomando en consideración el promedio de los salarios ordinarios y extraordinarios devengados durante un lapso de los últimos seis meses, básicamente se suman los salarios de estos meses y se divide entre seis para obtener el monto mensual.

El preaviso es una notificación por la cual se da por finalizada el contrato laboral en Costa Rica, que tiene un papel importante a la hora de realizar las liquidaciones de los empleados, pero también es crucial considerar otro aspecto fundamental como lo es la cesantía. La cesantía es un derecho que tienen los trabajadores cuando culmina la relación laboral con la empresa, y funciona como una especie de respaldo económico que tendrá el empleado mientras encuentra otro trabajo. Según el MTSS (s.f.), “es un derecho que tienen las personas trabajadoras a ser indemnizados en

caso de terminación de la relación laboral con responsabilidad patronal” (párr. 12). El cálculo de la cesantía en Costa Rica se define por las siguientes reglas:

1. Después de un trabajo continuo no menor de tres meses ni mayor de seis, un importe igual a siete días de salario.
2. Después de un trabajo continuo mayor de seis meses, pero menor de un año, un importe igual a catorce días de salario.
3. Después de un trabajo continuo mayor de un año, con el importe de días de salario indicado en la siguiente tabla:
 - a) AÑO 1 19,5 días por año laborado.
 - b) AÑO 2 20 días por año laborado o fracción superior a seis meses.
 - c) AÑO 3 20,5 días por año laborado o fracción superior a seis meses.
 - d) AÑO 4 21 días por año laborado o fracción superior a seis meses.
 - e) AÑO 5 21,24 días por año laborado o fracción superior a seis meses.
 - f) AÑO 6 21,5 días por año laborado o fracción superior a seis meses.
 - g) AÑO 7 22 días por año laborado o fracción superior a seis meses.
 - h) AÑO 8 22 días por año laborado o fracción superior a seis meses.
 - i) AÑO 9 22 días por año laborado o fracción superior a seis meses.
 - j) AÑO 10 21,5 días por año laborado o fracción superior a seis meses.
 - k) AÑO 11 21 días por año laborado o fracción superior a seis meses.
 - l) AÑO 12 20,5 días por año laborado o fracción superior a seis meses.
 - m) AÑO 13 y siguientes: 20 días por año laborado o fracción superior a seis meses.
4. En ningún caso podrá indemnizar dicho auxilio de cesantía más que los últimos ocho años de relación laboral.
5. El auxilio de cesantía deberá pagarse aunque el trabajador pase inmediatamente a servir a las órdenes de otro patrono.

Al pago de la cesantía no se le aplica ningún tipo de deducción por cargas sociales. (MTSS, s.f.)

El cálculo de la cesantía se basa en diversos factores, como la duración del empleado y la razón de la culminación laboral, y es importante conocer los detalles específicos, ya que impacta directamente la estabilidad financiera del empleado y las obligaciones económicas del empleador.

Una vez explorado la importación del cálculo de la cesantía en Costa Rica, es importante tener en cuenta que estos derechos económicos no solo se aplican durante la vigencia del empleo, sino también en momentos específicos como lo son los días feriados, los cuales son fechas establecidas por la ley en las cuales las actividades laborales y educativas se suspende, permitiendo a las personas disfrutar de tiempo libre y participar en actividades recreativas, religiosas y culturales. Estos días están establecidos para conmemorar eventos históricos, cívicos o religiosos. Cabe mencionar que algunos días son de pago obligatorio; sin embargo, si se labora ese día, el pago será doble.

CAPÍTULO III. MARCO METODOLÓGICO

En el siguiente capítulo, se describen los conceptos referentes a la metodología de investigación, así como las técnicas, instrumentos, herramientas y procedimientos utilizados en la elaboración de este trabajo.

Enfoques de investigación

La mayoría de las investigaciones realizadas en la actualidad emplean principalmente dos enfoques: el cuantitativo y el cualitativo. A continuación, se describirá detalladamente cada uno de estos enfoques.

Enfoque cuantitativo

El método cuantitativo “utiliza la recolección de datos para probar hipótesis con base en la medición numérica y el análisis estadístico, con el fin establecer pautas de comportamiento y probar teorías” (Hernández et al., 2014, p. 4).

Los métodos de investigación cuantitativa son métodos que utilizan datos numéricos y análisis estadístico para probar hipótesis, establecer relaciones entre variables y obtener resultados generales de una población. Es decir, los métodos cuantitativos se basan en los datos históricos de una entidad con el fin de establecer patrones que indiquen tendencias o comportamientos a lo largo del tiempo.

Enfoque cualitativo

El método cualitativo “utiliza la recolección y análisis de los datos para afinar las preguntas de investigación o revelar nuevas interrogantes en el proceso de interpretación” (Hernández et al., 2014, p. 7). Por otra parte, los métodos cualitativos se basan en contextos estructurales y situacionales, y se consideran técnicas como entrevistas y discusiones grupales.

A diferencia de los métodos cuantitativos, que tienen como objetivo relacionar resultados con métodos estadísticos, los métodos cualitativos tienen como objetivo recopilar comportamientos y declaraciones que, posteriormente, se interpretan y analizan para generar resultados.

Enfoque mixto

Los métodos de investigación mixtos combinan tanto la investigación cuantitativa como la cualitativa para una comprensión más completa del tema a desarrollar. Este método aprovecha las bondades y las fortalezas de cada uno de los métodos al mezclar los dos enfoques en la misma investigación. Según Hernández et al. (2014):

Los métodos mixtos representan un conjunto de procesos sistemáticos, empíricos y críticos de investigación e implican la recolección y el análisis de datos cuantitativos y cualitativos, así como su integración y discusión conjunta, para realizar inferencias producto de toda la información recabada (meta inferencias) y lograr un mayor entendimiento del fenómeno bajo estudio. (p. 534)

Un enfoque mixto puede ser una combinación equilibrada o no, según el área de interés, sin que alguno de los dos esté erróneo. Se diferencian en que pueden estudiarse combinando características de ambos métodos.

Enfoque de investigación seleccionado

El método utilizado para el desarrollo de este proyecto será una investigación mixta, ya que el método cualitativo ayudará a comprender detalles, percepciones personales por medio de entrevistas y otras técnicas para entender las necesidades de los colaboradores. Por otra parte, se compone de elementos cuantitativos recopilados por medio de análisis de fuentes de información de los colaboradores como una encuesta para conocer los datos numéricos relevante para la generación y levantamiento de requerimientos.

Tipos de investigación

Es importante mencionar los tipos de investigación que se pueden aplicar para abordar diferentes aspectos de manera efectiva. Entre los tipos de investigación que se consideran aplicables se incluyen: exploratorio, descriptivo y explicativo. El tipo de investigación exploratorio, según mencionan Hernández et al. (2014):

Los estudios exploratorios sirven para preparar el terreno y, por lo común, anteceden a investigaciones con alcances descriptivos, correlacionales o explicativos. Por lo general, los estudios descriptivos son la base de las investigaciones correlacionales, las cuales a su

vez proporcionan información para llevar a cabo estudios explicativos que generan un sentido de entendimiento y están muy estructurados. Las investigaciones que se realizan en un campo de conocimiento específico pueden incluir diferentes alcances en las distintas etapas de su desarrollo. (p. 91)

Basado en el punto anterior, el objetivo principal es adquirir una comprensión general del tema de investigación, identificar áreas de interés y definir preguntas de investigación más específicas. Los estudios exploratorios son útiles para explorar territorios desconocidos y generan hipótesis preliminares.

Por otro lado, el tipo de investigación descriptivo, de acuerdo con Hernández et al. (2014): [...] busca especificar las propiedades, las características y los perfiles de personas, grupos, comunidades, procesos, objetos o cualquier otro fenómeno que se someta a un análisis. Es decir, únicamente pretenden medir o recoger información de manera independiente o conjunta sobre los conceptos o las variables a las que se refieren, esto es, su objetivo no es indicar cómo se relacionan éstas. (p. 92)

Estos estudios se enfocan en recopilar información de manera independiente o combinada sobre conceptos o variables determinadas, sin buscar establecer relaciones entre ellas. Su función principal es prestar una imagen clara y detallada de lo que se está estudiando.

Por su parte, el tipo explicativo o correlacional, según Hernández et al. (2014):

Este tipo de estudios tiene como finalidad conocer la relación o grado de asociación que exista entre dos o más conceptos, categorías o variables en una muestra o contexto en particular. En ocasiones solo se analiza la relación entre dos variables, pero con frecuencia se ubican en el estudio vínculos entre tres, cuatro o más variables. (p. 93)

Los estudios correlacionales o explicativos tienen como objetivo principal explorar la relación que existe entre dos o más conceptos, categorías o variables en un contexto o muestra específica.

Tipo de investigación seleccionado

Para efectos de este proyecto, de acuerdo con la descripción de cada uno de los tipos de investigación definidos, se determina que el tipo descriptivo se utilizará para el fin de esta

investigación. Esto se determina, debido a que el desarrollo de este estudio demanda conocer con la presente problemática y, a partir de estos resultados, mejorar los procedimientos de la empresa.

Fuentes de información

Las fuentes de información son recursos utilizados para obtener datos sobre un tema, principalmente para trabajos de investigación. Se pueden considerar tres tipos: primario, secundario y terciario. Cada tipo se define a continuación.

Fuente de información primaria

Las fuentes de información primarias son recursos que proporcionan datos originales de primera mano, y no ha sido interpretado por otros actores. Según Morales (2021) “contiene información que no ha sido alterada, interpretada o analizada por otros autores, sino que es del propio autor. En otras palabras, información que se mantiene intacta desde su elaboración” (párr. 1). Ejemplos de estas fuentes son libros, artículos de revistas científicas, tesis, documentos oficiales.

Fuente de información secundaria

Las fuentes de información secundaria son aquellas que recopilan o analizan datos de fuentes primarias. Estas fuentes proporcionan una revisión o análisis de la información original. De acuerdo con Morales (2021), “la fuente secundaria ofrece información que hace referencia a una fuente primaria, pero no ofrece información como tal, sino que expone una información que ya recoge una fuente primaria” (párr. 5).

Fuentes de información terciaria

Las fuentes de información terciarias son recursos que se basan de fuentes secundarias y primarias. Son formadas normalmente por bibliografías, catálogos de bibliotecas, directorios. Según la Universidad de Guadalajara (s.f.), las fuentes de información terciaria “son guías físicas o virtuales que contienen información sobre las fuentes secundarias. Forman parte de la colección de referencia de una biblioteca” (párr. 3).

Variables

Una variable es una característica o factor que se estudia y puede variar o cambiar en diferentes situaciones o momentos. Las variables son elementos que puede ser manipulado, controlado o medido y se utilizan para analizar y comparar efectos de un fenómeno o problema de estudio. Según Hernández et al. (2014), “una variable es una propiedad que puede fluctuar y cuya variación es susceptible de medirse u observarse (p. 105).

Definición conceptual

La definición conceptual de una variable se precisa como definiciones de libros o diccionarios que sustentan las variables de estudio y representan las características de un objeto. Según Moreno (2018), la definición conceptual de las variables “son definiciones de diccionario o de libros especializados y cuando describen la esencia o las características reales de un objeto o fenómeno se les denomina ‘definiciones reales’” (párr. 1).

Definición operacional

La definición operacional especifica las actividades u operaciones se deben de realizar para medir una variable. Esta nos dice que, para recoger datos con respecto a una variable, se deben de realizar ciertos procedimientos necesarios para identificar una variable. De acuerdo con Moreno (2018), son un “conjunto de procedimientos que describen las actividades que un observador debe realizar para recibir las impresiones sensoriales, las cuales indican la existencia de un concepto teórico en mayor o menor grado” (párr. 1).

Definición instrumental

Por medio de las variables instrumentales, se aclaran como se va a estudiar la definición de las variables que se acaban de definir y como se recolectara la información, ya sea por medio del uso de los instrumentos como la observación, encuestas, registros o cualquier otra técnica que sea de utilidad. De acuerdo con Moreno (2018):

La definición instrumental de las variables es también conocido como la tercera variable, de modo que es aquello en la que se aclara como se estudiará la variable que se acaba de definir, los medios o instrumentos para recoger la información. En mérito de ello, se deben definirse y elaborarse los instrumentos y medios con que se recolectará la información. (párr. 1)

Tabla 4. Cuadro de variables

Objetivo específico	Variable	Variable conceptual	Variable operacional	Variable instrumental
Analizar los requerimientos y los procesos existentes en la empresa NutriMax.	Definición de los requerimientos.	Requerimientos de <i>software</i> son simplemente una descripción de lo que un programa de <i>software</i> en particular debe hacer. Actúan como pautas para que los desarrolladores creen un producto funcional que satisfaga las necesidades de los usuarios (Northware, 2022, párr. 2).	Entrevistas Encuestas	Guía de entrevistas Guía de encuestas
Diseñar un prototipo funcional que permita simplificar el proceso de cálculo de nómina para los trabajadores de la empresa NutriMax.	Prototipo funcional.	Un modelo prototipo o modelo de desarrollo evolutivo es utilizado principalmente en el desarrollo de <i>software</i> para ofrecer al usuario una visión previa de cómo será el programa o	Diagramas de flujo. Diagrama de bases de datos. Diseño de clases	Microsoft Visual Studio. MySQL Workbench. Editores UML.

		sistema (Hosting Plus, 2021, párr. 2)		
Programar el prototipo empleando las herramientas para el desarrollo del aplicativo web y manejador de base de datos.	Programación.	La programación es el proceso que se lleva a cabo para desarrollar algún tipo de aplicación, web o <i>software</i> , para representar un concepto o un proyecto que requiere del uso de un lenguaje informático para realizarse (Assembler Institute, 2022, párr. 2).	Programar los módulos necesarios y los procesos para la creación del prototipo funcional	Microsoft Visual Studio. MySQL Workbench. ASP.NET. C#.
Probar el prototipo funcional para garantizar su funcionalidad, eficiencia y seguridad en la gestión del personal.	Proceso de pruebas.	La prueba de <i>software</i> es el proceso de evaluar y verificar que un producto o aplicación de <i>software</i> hace lo que se supone que debe hacer (IBM, s.f., párr. 1)	Plan de pruebas de aprobación. Pruebas Web.	Guía de casos de prueba.

Población

La población es el grupo completo de los elementos o individuos sujetos a estudiar con un número y características determinadas. Pueden incluir personas, objetos, eventos o cualquier otro tipo de identidad que sea relevante para el tema en estudio. En muchos casos, debido a limitaciones en los recursos, tiempo o costos, se selecciona la muestra que es una parte del total de la población. Organizadores Gráficos (2022) menciona que la población:

Es el conjunto, totalidad o generalización de unidades, individuos, objetos o sujetos que tienen ciertas cantidades y características para ser estudiados, que pueden ser personas, objetos, instituciones, eventos, y otros en los que se puede obtener o puede proporcionar información (datos) investigación que luego pueden ser concluidos. (párr. 1)

La empresa Nutrimax se encuentra conformada por varios departamentos entre ellos se encuentran contabilidad, recursos humanos, vendedores y en total suman alrededor de ocho empleados.

Muestra

La muestra es un subconjunto seleccionado de la totalidad de la población en estudio. La muestra se elige para representar a la población de interés y se utiliza para realizar las observaciones, recopilar datos relevantes o llevar a cabo análisis. De acuerdo con Organizadores Gráficos (2022), la muestra es:

Es una pequeña parte de los miembros de la población tomada de acuerdo con ciertos procedimientos que pueden representar a la población. La muestra se utiliza si la población en estudio es grande y es imposible que el investigador estudie a toda la población. (párr. 6)

Para calcular el tamaño exacto de la muestra, se utilizará la siguiente fórmula:

$$n = \frac{k^2 N p q}{e^2 (N - 1) + K^2 p q}$$

Donde:

n= tamaño de la muestra.

N= tamaño de la población.

K= nivel de confianza.

p= proporción esperada

q= probabilidad de fracaso

e= precisión (margen de error)

Tabla 5: Nivel de confianza para el cálculo de la muestra

Valor de k	1,28	1,44	1,65	1,96	2,58
Nivel de confianza deseado	80 %	85 %	90 %	95 %	99 %

Fuente: Elaboración propia.

Se sustituyen los valores en la fórmula:

$$N= 7$$

$$K= 2,58 \text{ es } 99 \%$$

$$E= 1 \%$$

$$p= 0,3$$

$$q= 0,7$$

$$n= \frac{2,58^2 \times 7 \times 0,3 \times 0,7}{0,1^2(7 - 1) + 2,58^2 \times 0,3 \times 0,7}$$

$$n= 5,87$$

El tamaño total de la muestra a tomar, redondeando el número hacia arriba, es de seis personas. Para efectos de esta investigación, no se hará una muestra, puesto que la población de la organización es de siete personas.

Instrumentos de recolección de datos

Según la página Tesis y Másters (2023), los instrumentos de recolección de datos son “una herramienta de la cual se vale un investigador para obtener información que le permita desarrollar su proyecto. Su principal función es extraer o construir datos de primera mano de la población o los fenómenos que se desean conocer” (párrs. 3-4) Existen diferentes tipos de recolección de datos, los cuales se proceden a definir:

Observación

La observación una técnica de investigación por la cual se recopilan datos por medio de la observación directa de los fenómenos a estudiar. Es una técnica que permite la recolección de datos de un fenómeno o situación, sin intervenir en ella. La observación puede ser afectada por varios factores como por ejemplo prejuicios del investigador, el ambiente o características del objeto o sujeto observado. De acuerdo con Ortega (2023), la observación es uno de los “instrumentos para recopilar información que consiste en observar a los individuos en su entorno natural o en una situación que se produce de forma natural”. (párr. 13)

Encuesta

Una encuesta es una técnica de recopilación de datos que se utiliza para obtener información haciendo una serie de preguntas estandarizadas a una muestra de individuos de estudio. Las encuestas proporcionan datos estructurados sobre opiniones, aptitudes y comportamientos relevantes para el tema de investigación.

Según menciona Gómez (2023), una encuesta “es un método de investigación que recopila información, datos y comentarios por medio de una serie de preguntas específicas. La mayoría de las encuestas se realizan con la intención de hacer suposiciones sobre una población, grupo o muestra representativa” (párr. 5).

Se aplicarán encuestas en la elaboración de este proyecto como una de las herramientas clave para recuperar información valiosa y comprender más a fondo los procesos actuales de la organización.

Entrevista

La entrevista es un método para la recolección de datos que consiste en preguntar a una persona o a varias acerca de su opinión referente a la empresa o algún tema en específico. El objetivo es el de conocer opiniones o comportamientos de las personas encuestadas.

Según indica Muguira (2023), la entrevista “es un método de recolección de datos que consiste en preguntar a una o varias personas su opinión sobre una empresa, un producto o un tema. El objetivo principal es conocer los comportamientos, actitudes y opiniones de las personas” (párrs. 4-5).

De la misma forma, se aplicará una entrevista en el desarrollo de este trabajo para la recolección de datos y opiniones.

Proceso para análisis y recolección de datos

Para el análisis y la recolección de datos, se realizará una entrevista a la encargada de planillas para conocer los procesos actuales que se realizan en el cálculo de la planilla de los trabajadores que tiene la macrobiótica, qué parámetros utilizan para el cálculo de las horas totales, aguinaldos y horas extra laboradas, con el fin de concretar cuales son los procedimientos y puntos de mejora que se pueden aplicar al sistema.

También se utilizarán las encuestas para conocer la opinión del personal de la macrobiótica acerca de los procesos actuales que manejan las labores de recursos humanos dentro de la compañía, así como cálculos de nómina salarial, procesos actuales, permisos solicitados, para conocer sus opiniones y aptitudes.

CAPÍTULO IV: ANÁLISIS DE RESULTADOS

En esta sección se detallan los resultados obtenidos mediante la aplicación de los instrumentos a los miembros de la empresa Nutrimax. La recopilación de estos datos es fundamental para comprender las posibles causas de las problemáticas existentes que afectan a la organización. El propósito fundamental es el de identificar áreas de mejora y de proponer soluciones efectivas.

Los medios de recolección de datos seleccionados en la elaboración de este prototipo, como la encuesta, permite recolectar información de datos cuantitativos por medio de una serie de preguntas estratégicamente diseñadas y aplicados al personal de la compañía. El objetivo de la ejecución de la encuesta es la de encontrar datos valiosos, que permitan ser graficados y tabulados y que proporcionen una base sólida para la interpretación y análisis de los resultados.

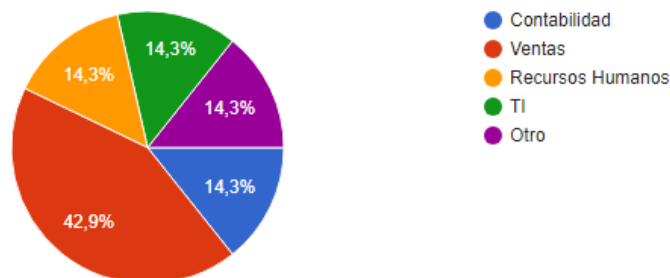
Otro aspecto que se tomó en cuenta dentro del proceso de recolección de la información fue la de emplear una entrevista como herramienta complementaria a la encuesta, aunque con un alcance más específico. El propósito de la entrevista es de capturar perspectivas y opiniones individuales de manera más detallada. Este enfoque cualitativo busca revelar lo que los participantes piensan.

Estas entrevistas proporcionan un contexto cualitativo valioso que, junto con los datos cuantitativos de la encuesta, ofrecen una imagen más completa del panorama de la compañía. Los resultados obtenidos de la encuesta aplicada fueron:

Figura 5: Resultado encuesta colaboradores – Pregunta 1

¿A qué departamento perteneces dentro de la macrobiótica?

7 respuestas



Fuente: Elaboración propia.

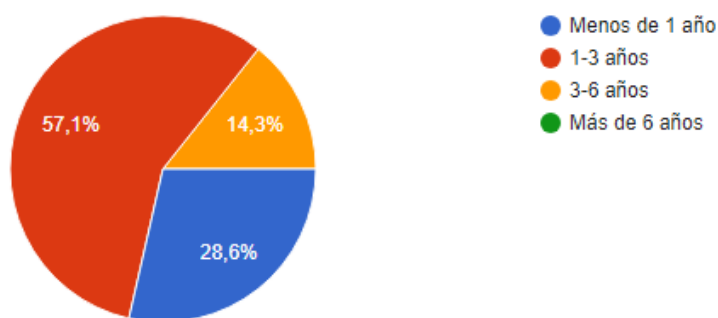
En cuanto a la conformación de la fuerza laboral de la macrobiótica, el 42,9 % pertenece al departamento de ventas, y los demás departamentos como el área de contabilidad, recursos humanos, departamento de sistemas, entre otros, conforman un 14,3 % de la fuerza laboral.

Con esta pregunta, lo que se pretende es identificar la distribución de la fuerza laboral de la compañía, el cual se identifican las necesidades de los departamentos y ajustar estrategias de recursos humanos a la compañía, así como retener y motivar a los empleados.

Figura 6: Resultado encuesta colaboradores – Pregunta 2

¿Cuánto tiempo llevas laborando en la macrobiótica?

7 respuestas



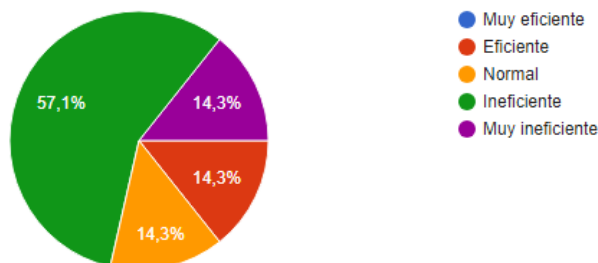
Fuente: Elaboración propia.

De acuerdo con la totalidad de la población encuestada, se observa que más de la mitad de los empleados con un 57,1 % llevan laborando de un rango de entre 1 a 3 años, también que el 28,6 % lleva un año en la compañía, y que solo el 14,3 % de la compañía poseen de 3 a 6 años. Con esta pregunta lo que se busca es obtener un promedio de la cantidad de años que poseen laborando para la compañía, además de brindar información valiosa que ayuda a identificar y diseñar estrategias precisas para la retención y el desarrollo de los empleados en las diferentes etapas de su tiempo en la compañía.

Figura 7: Resultado encuesta colaboradores – Pregunta 3

¿Qué percepción posee acerca de la eficacia de los procesos actuales en el área de Recursos Humanos?

7 respuestas



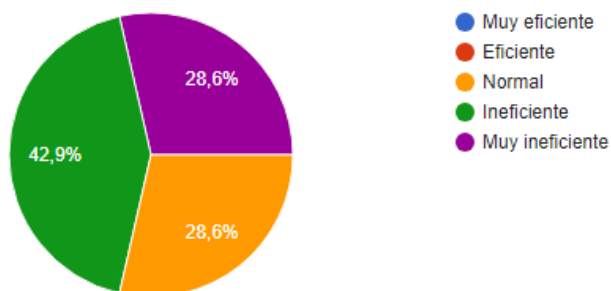
Fuente: Elaboración propia.

Con esta pregunta, se buscó conocer acerca del nivel de percepción sobre la eficacia de los actuales procesos en el área de recursos humanos en la compañía, del cual los resultados fueron en su mayoría de una percepción negativa, con un 57,1 % lo consideran ineficiente, más un 14,3 % que lo consideran como muy ineficiente, lo cual significa que la población encuestada en su mayoría no está conforme con los procesos actuales del departamento. Una pequeña parte de los encuestados ven los procesos como normales o eficientes, pero esta cifra es relativamente menor en comparación con los que piensan que los procesos son relativamente ineficientes, lo cual indica que existen aspectos específicos que funcionan bien, pero también existen áreas críticas que requieren de atención.

Figura 8: Resultado encuesta colaboradores – Pregunta 4

¿Cómo calificaría la eficiencia de los procesos de cálculo de nomina salarial en la empresa?

7 respuestas



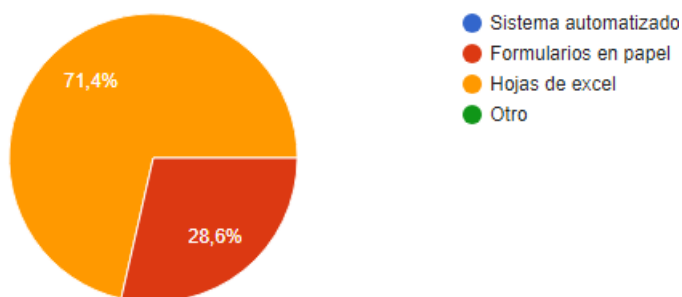
Fuente: Elaboración propia.

Con esta pregunta, lo que se busca es conocer la eficiencia que perciben los empleados de la macrobiótica acerca de los procesos de cálculo a la hora de realizar los salarios de la compañía, lo cual demuestra en su mayoría el descontento existente con un 42,9 % de los empleados encuestados, además de que un 28,6 % indica que el proceso es muy ineficiente, y un 28,6 % piensa que es un proceso normal. Este descontento podría deberse a diversos factores como por ejemplo problemas con la precisión de los cálculos, la comunicación de información salarial, descuidos de la encargada, entre otros. Con el desarrollo del prototipo, el cálculo de las planillas se haría de manera automatizada y evitaría errores humanos, con lo cual, se evita el descontento y desconfianza de los empleados.

Figura 9: Resultado encuesta colaboradores – Pregunta 5

¿Cómo se manejan actualmente las solicitudes de permisos de la compañía?

7 respuestas



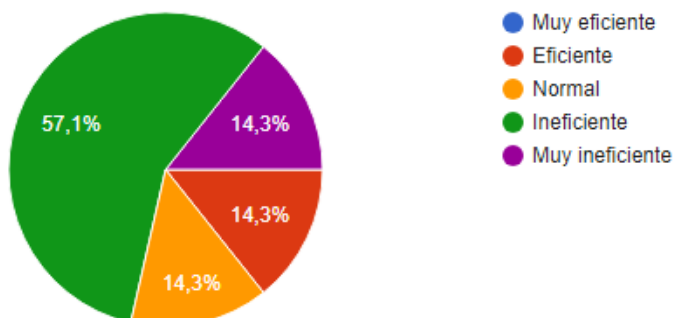
Fuente: Elaboración propia.

La empresa actualmente gestiona las solicitudes de permisos de los colaboradores, en su mayoría (un 71,4 %) mediante hojas de Excel, y en un 28,6 % a través de formularios en papel. Este hecho indica que el proceso actual presenta ciertas limitaciones, como la ineficiencia en los procedimientos (que pueden ser propensos a errores), la dificultad para dar seguimiento al estado de las solicitudes y el riesgo de duplicar o cometer errores. El prototipo brindará un proceso simple por el cual la solicitud se realizará de manera sencilla por medio de los módulos y pantallas creadas para los usuarios.

Figura 10: Resultado encuesta colaboradores – Pregunta 6

¿Cómo calificaría la eficiencia del proceso de manejo de permisos?

7 respuestas



Fuente: Elaboración propia.

La calificación general del proceso en el manejo de permisos, según los resultados obtenidos, refleja una percepción en su mayoría negativa. Un 57,1 % de las respuestas califican los procesos como ineficientes, lo que indica que una mayoría significativa de los encuestados consideran que el manejo de los permisos no funciona de una manera óptima. Además de que un 14,3 % consideran estos procesos como un problema significativo. Un 14,3 % lo califica como eficiente y otro 14,3 % como normal, lo cual es una cifra relativamente pequeña en comparación con la percepción negativa. Dada la mayoría de las respuestas negativas, estos resultados sugieren que existen áreas críticas que requieren mejoras en el manejo de permisos.

Figura 11: Resultado encuesta colaboradores – Pregunta 7



Fuente: Elaboración propia.

Con base a esta pregunta, lo que se busca es analizar cuáles áreas son las que los empleados consideran importante de mejorar, lo cual da como resultado que un 57,1 % considera que el área de gestión de planilla, un 14,3 % considera que la gestión de solicitudes, un 14,3 % el área de gestión de vacaciones y el resto piensa que debe de mejorarse los procesos de gestión de liquidaciones.

Además, seguidamente, se muestran los resultados obtenidos por la entrevista aplicada a la encargada de recursos humanos; de este modo, se detalla la pregunta realizada y, luego, la respuesta brindada, tal como se muestra enseguida:

- a. ¿Podría detallar el rol y las responsabilidades como encargada de planillas de la Macrobiótica?

Elaborar todo el proceso de planillas del personal (quincenales), para que la información contable sea procesadas y aplicada, y se paguen los salarios oportunamente. Envío de comprobantes de pago por medio de correo a todo el personal. Elaborar planillas externas CCSS – INS. Realizar todas aquellas actividades necesarias para el cumplimiento de las funciones, las que estén estipuladas y las que sean expresamente encomendadas por el jefe inmediato. Asesorar a la administración de la empresa en cuanto a la aplicación de mejores prácticas en el área de recursos humanos.

La encargada de planillas es responsable de gestionar los procesos relacionados con las planillas de los empleados, cumpliendo con las obligaciones legales ante la CCSS y el INS, y posteriormente remitiendo la información al área contable para su procesamiento y aplicación. Asimismo, se encarga del envío masivo de los comprobantes de pago a los colaboradores.

Esta pregunta ofrece una guía detallada sobre las responsabilidades y el alcance del puesto de encargada de planillas, cuya información puede utilizarse para optimizar y fortalecer el sistema de planillas, con lo cual se garantiza la puntualidad y el cumplimiento de las obligaciones patronales.

b. ¿Posee experiencia en la evaluación y generación de planillas?

Sí, alrededor de 3 años en la generación y pago de planillas de las empresas en las que he laborado.

La encargada posee experiencia y preparación en el área en la que se desempeña, el cual es un punto valioso para los intereses de la compañía. La combinación de estos conocimientos junto con la experiencia previa recalca la versatilidad de la persona, y estas cualidades son fundamentales para la implementación y uso del prototipo.

c. ¿Podría detallar el proceso actual para el cálculo de planilla de los trabajadores de la macrobiótica?

En caso de tener trabajadores asalariados, divide el salario anual por 12. Tendrás el costo laboral mensual, y se verifica empleado por empleado si tiene horas extra, vacaciones, entre

otros y se divide el salario mensual entre 2. Después, suma todas esas cantidades para encontrar tus costos laborales totales del mes. Si quieres conocer los costos laborales anuales, solamente suma los costos de cada mes para tener el resultado total.

Para el cálculo mensual del salario, ella divide el monto anual por 12, de esta manera obtiene el costo mensual, agregando todos los cálculos correspondientes como lo son horas extra y salarios, y seguidamente realiza la suma de todas esas cantidades para encontrar los costos laborales totales del mes. Esta pregunta permite una comprensión más clara de cómo se operan las planillas y los costos laborales de la compañía, de manera que se pueden evaluar la eficiencia de los procedimientos actuales e identificar puntos de mejora.

d. ¿Podrías detallar como se realiza el cálculo de horas extra de los empleados?

Se revisa en los documentos y hojas de excel si el empleado posee horas extra durante la quincena. Si posee, se pagan con un recargo del 50 % del valor de la hora ordinaria o a 1,5 del valor de la hora ordinaria, que es la diurna (artículo 139 del Código de Trabajo).

El proceso inicia con la revisión de los documentos y las hojas de Excel en las que se lleva el registro correspondiente. Si se identifica que existen horas extra asignadas a un colaborador, se procede al cálculo del salario, aplicando un valor de 1,5 veces el de la hora ordinaria. Además, esta pregunta permite una comprensión más clara del manejo de este componente específico dentro de la planilla, siendo dicha información fundamental para garantizar la precisión en el pago de las horas extra.

e. ¿Puedes hablarme acerca del proceso para verificar la cantidad de días de vacaciones que tiene cada empleado?

Se revisa la fecha de ingreso del trabajador y se le suma un día cuando cumple el mes.

El proceso para verificar la cantidad de días de vacaciones que tiene cada empleado de la Macrobiótica es sencillo y se basa en la fecha de ingreso de cada trabajador, y cada revisión implica sumar un día al total de días de vacaciones cuando el empleado cumple cada mes de servicio. Este es un proceso simple, pero genera pérdida de tiempo y en ocasiones errores humanos, y el prototipo

lo realizaría automáticamente, lo cual facilita las gestiones de solicitudes para la encargada y para los empleados.

f. ¿Cómo calcula el monto que tiene que pagar de aguinaldo a finalizar el año?

El aguinaldo se calcula con base en todos los salarios ordinarios y extraordinarios, devengados por la persona trabajadora durante los doce meses que van del 1º diciembre del año anterior al 30 de noviembre del año que se trate. Se obtiene de la suma de dichos salarios y se divide entre doce.

El cálculo del aguinaldo sigue un proceso claro y específico. El monto del aguinaldo se establece tomando en consideración todos los salarios ordinarios y extraordinarios devengados por el trabajador durante los doce meses que comprenden desde el 1 de diciembre del año anterior hasta el 30 de noviembre del año en curso. Se calcula con la suma total de estos salarios dividida entre doce. Con base en esta pregunta, se obtiene información clara y sencilla de cómo se podría aplicar y calcular el aguinaldo de manera sencilla en el módulo del prototipo.

g. ¿Ha sufrido manipulación o extravío de los documentos formales donde se lleva el control de estas horas extra?

Sí.

Esta afirmación sugiere que la empresa ha experimentado problemas relacionados con la integridad o la pérdida de documentos cruciales que registran las horas extraordinarias trabajadas por los empleados. Esta información es esencial para evaluar la seguridad que tendrá el prototipo y así evitar pérdidas de documentos.

h. ¿Ha sufrido atrasos al momento de generar los pagos correspondientes de los empleados?

Sí.

Esto implica que ha habido ocasiones en que la empresa no ha cumplido con los plazos establecidos para la emisión de los pagos salariales. Los atrasos en los pagos pueden tener impactos negativos tanto en la moral de los empleados como en su bienestar financiero. Estas situaciones a menudo generan preocupación y frustración entre los trabajadores, lo que podría afectar la productividad y la satisfacción laboral. Además de que puede tener implicaciones legales. Con el prototipo funcional, se evitarían este tipo de errores y se mejoraría la puntualidad con la que se realizan los pagos de las quincenas a los empleados.

- i. ¿Cuáles son los desafíos más importantes que afronta el departamento de recursos humanos?

La Renuncia Silenciosa. Centrarse en el desarrollo del talento interno. Centrarse en las nuevas tecnologías.

Se identifican varios desafíos clave que el departamento de recursos humanos enfrenta en la macrobiótica. La renuncia silenciosa constituye un reto que evidencia la necesidad de mantener un ambiente laboral que promueva el compromiso y prevenga la pérdida de talento. Asimismo, el enfoque en el desarrollo del talento interno resalta la importancia de retener y fortalecer las habilidades dentro de la organización. Finalmente, la adopción de nuevas tecnologías refleja la conciencia de que la innovación tecnológica desempeña un papel fundamental en la gestión de los recursos humanos.

CAPÍTULO V: PROPUESTA

En el siguiente capítulo se desarrollan todos los aspectos generales acerca de la propuesta para la solución de las problemáticas que afectan a Nutrimax con respecto a la gestión de recursos humanos. Se detallan el diseño del prototipo, como lo son: Casos de uso, Arquitecturas del sistema, Diseño de interfaces, Diagramas UML. Así como el funcionamiento del sistema, procesos, extractos del código fuente. El desarrollo del prototipo de *software* contiene las fases de: análisis, diseño, programación y pruebas.

Análisis

Esta es la primera fase del desarrollo del prototipo. Este punto contiene temas como los análisis detallados del *software* y *hardware*, análisis de telecomunicaciones, infraestructura, bases de datos.

Análisis de software

En esta fase se ve un panorama general del *software* propuesto para la gestión del recurso humano en Nutrimax, y en el cual se desglosan los módulos funcionales, y se describen cada uno de ellos.

Módulo de Calcular Planilla:

Este módulo tomara los datos de los empleados tales como las deducciones que se les deben realizar, las horas extra que han realizado y las horas naturales laboradas por el empleado para realizar un cálculo automático de la planilla.

Módulo de Calcular Extras:

Este módulo realizará una solicitud de horas extra electrónica por medio un formulario mostrado en pantalla que el usuario deberá llenar y enviar por medio de un botón para que sea aprobada o denegada la solicitud.

Módulo de Calcular Deducciones:

Este módulo se encargará tomar parámetros como permisos de ausencia, llegadas tardías, monto del seguro social, entre otros, para que se pueda tener la certeza del monto de deducciones que se le debe aplicar a determinado empleado.

Módulo de Calcular Liquidaciones:

Este módulo se encargará de tomar datos de la base de datos que le corresponden al usuario que se va a liquidar para hacer el cálculo de manera automática y mostrar el resultado en un campo de la pantalla.

Módulo de Calcular Permisos:

Este módulo se encargará de llevar el control de los permisos que toman los empleados, enviar una solicitud para autorizar el permiso del empleado y descontar la cantidad de días tomados de la cantidad de permisos disponibles.

Módulo de Calcular Aguinaldo:

Este módulo se encargará de generar el cálculo del aguinaldo que debe incluir las horas ordinarias y las horas extraordinarias, así como cualquier otro pago salarial que se haya realizado en el período.

Módulo de Calcular Incapacidades:

Este módulo se encargará de gestionar el cálculo de incapacidades laborales de los colaboradores, tomando como base los datos registrados en la base de datos y las normativas vigentes del Código de Trabajo. El sistema permitirá registrar la fecha de inicio y finalización de la incapacidad, el tipo de incapacidad (por enfermedad común, accidente laboral, maternidad, entre otros), y calcular automáticamente el monto correspondiente según las políticas internas y las obligaciones patronales.

Módulo de Solicitar Vacaciones:

Este módulo se encargará del cálculo y administración de las vacaciones de los empleados, enviando a jefatura una solicitud para aprobar las vacaciones, luego de aprobada la solicitud, se toman las vacaciones disponibles y se descuentan las vacaciones que tomó el empleado, manejando estos datos por medio de la base de datos.

Módulo de Evaluar Empleados:

Este módulo se encargará de tomar criterios estandarizados por la empresa para realizar una evaluación más concreta.

Módulo de Mantenimientos:

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

Módulo de Consultas:

Este módulo se encargará de generar información proporcionada de las diferentes tablas.

Módulo de Reportes:

Este módulo se encargará 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 de Seguridad:

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

Análisis del hardware

En esta sección se presenta el *hardware* necesario tanto para la programación y pruebas del prototipo como para la puesta en producción del sistema. Dado que la solución se implementará en la nube (PaaS), la carga principal de cómputo en producción no recae en equipos locales, por lo que el *hardware* se limita a estaciones de trabajo de desarrollo. El análisis detalla especificaciones técnicas mínimas y recomendadas.

La Tabla 6 presenta el equipo de desarrollo propuesto, suficiente para compilar y trabajar con múltiples proyectos. En el entorno de producción, los usuarios finales únicamente requieren computadoras con un navegador moderno y una conexión estable a Internet. Los montos aquí consignados son estimativos y pueden ajustarse según el proveedor local y la disponibilidad del mercado.

Tabla 6: Hardware implementado para el desarrollo del sistema

Rubro	Costo unitario	Cantidad	Costo total (tipo de moneda)
Computador con procesador Intel Core i7-1355U, con 32 GB de memoria RAM, 1 TB memoria SSD.	€509 000	1	€509 000
Gran total			€509 000

Fuente: Costos extraídos del sitio web Amazon (2025)

Análisis de telecomunicaciones

El sistema de recursos humanos de NutriMax se alojará en la nube mediante un servicio de aplicación web (Azure App Service) y una base de datos gestionada (Azure Database for MySQL), accesibles únicamente a través de Internet bajo protocolo HTTPS (TLS 1.2/1.3). Los usuarios, desde la tienda o ubicaciones remotas, ingresarán mediante navegador con autenticación y autorización basada en roles (Empleado, Jefatura, Recursos Humanos, Administrador).

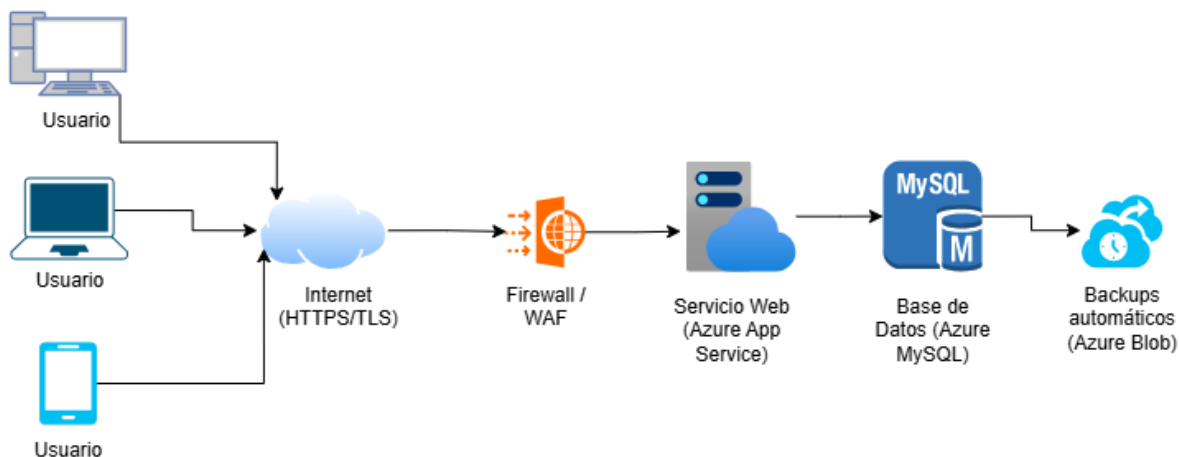
La topología propuesta sitúa un WAF/Firewall *front-end* protegiendo la aplicación, y reglas de *firewall* de base de datos que restringen el acceso a la BD únicamente desde la aplicación. No existe acceso directo a la base de datos desde las redes de los usuarios.

En términos de seguridad, se aplicará cifrado en tránsito (HTTPS), y *backups* automáticos diarios con retención de 30 días en almacenamiento redundante (Azure Blob). Se realizará una prueba de restauración mensual.

El objetivo de disponibilidad es 99,9 % mensual, suficiente para la operación de una tienda. La solución puede escalar horizontalmente (plan superior) ante aumento de carga. Para el consumo de ancho de banda, las transacciones son ligeras; con 7–10 usuarios y 2 concurrentes en hora pico, una conexión de 10 Mbps simétricos por sitio es holgada; el enlace actual de 200/200 Mbps supera ampliamente los requerimientos.

Esta arquitectura garantiza seguridad, escalabilidad y mantenibilidad, reduce la complejidad de operación local (sin servidores en sitio) y permite crecimiento futuro (más sucursales/usuarios) con mínimos cambios de infraestructura.

Figura 12: Diagrama de red



Fuente: Elaboración propia.

Análisis de herramientas técnicas

En este apartado se detalla el *stack* tecnológico utilizado en el desarrollo del prototipo y su operación en producción, así como el esquema de licenciamiento, estimación de costos y consideraciones de capacidad (usuarios, almacenamiento y rendimiento).

Para el *backend* y la parte web se utiliza .NET 8 (ASP.NET CORE MVC) como *framework* principal para la aplicación web. C# como lenguaje de programación. Para el *frontend* se utiliza HTML5, CSS3, Bootstrap 5, JavaScript (jQuery 3.4.1) para la interfaz gráfica. Los programas utilizados, *frameworks*, IDEs y versiones, son productos de licencia libre, por lo que no genera gastos respectivos. Para la base de datos en producción se usará MySQL (servicio gestionado en la nube) como motor transaccional y en producción se utiliza MySQL Workbench 8.0 CE para modelado/administración.

Tabla 7: Descripción y costos de software (desarrollo)

Rubro	Costo unitario	Cantidad	Costo total (tipo de moneda)
Visual Studio 2022	€0	1	€0
.NET Core ASP.NET 8.0	€0	1	€0
C#	€0	1	€0
JavaScript - JQuery 3.4.1	€0	1	€0
Bootstrap v5.2.3 - CSS 3 – HTML 5	€0	1	€0
MySQL Workbench 8.0 CE	€0	1	€0
Gran total			€0

Fuente: Elaboración propia.

Tabla 8: Servicios gestionados (base de datos y hosting en la nube)

Rubro	Descripción	Costo/Estimación (mes) Unitario	Cantidad	Costo/Estimación (mes) Total
Sitio Web – Azure App Service (Linux, Plan básico B1)	1 instancia, 1 vCPU / 1.75 GB RAM, 10 GB	€6630	1	€6630
Base de datos – Azure Database for MySQL (Flexible Server, SKU de entrada)	1 vCore, 2–4 GB RAM, 20–50 GB almacenamiento, backup 30 días	€6262	1	€6262
Almacenamiento MySQL	20 GB en disco	€1370	1	€1370
Backups/archivos	50–100 GB/mes	€454	1	€454

(Blob Storage)				
Gran total				€14 716

Fuente: Costos extraídos del sitio web de Microsoft Azure (2025).

Estos costos son aproximaciones mensuales. Estas herramientas permiten un entorno de desarrollo moderno, estable y escalable, reduciendo costos de licenciamiento y facilitando la integración con servicios en la nube de Microsoft Azure.

Análisis del conocimiento del personal

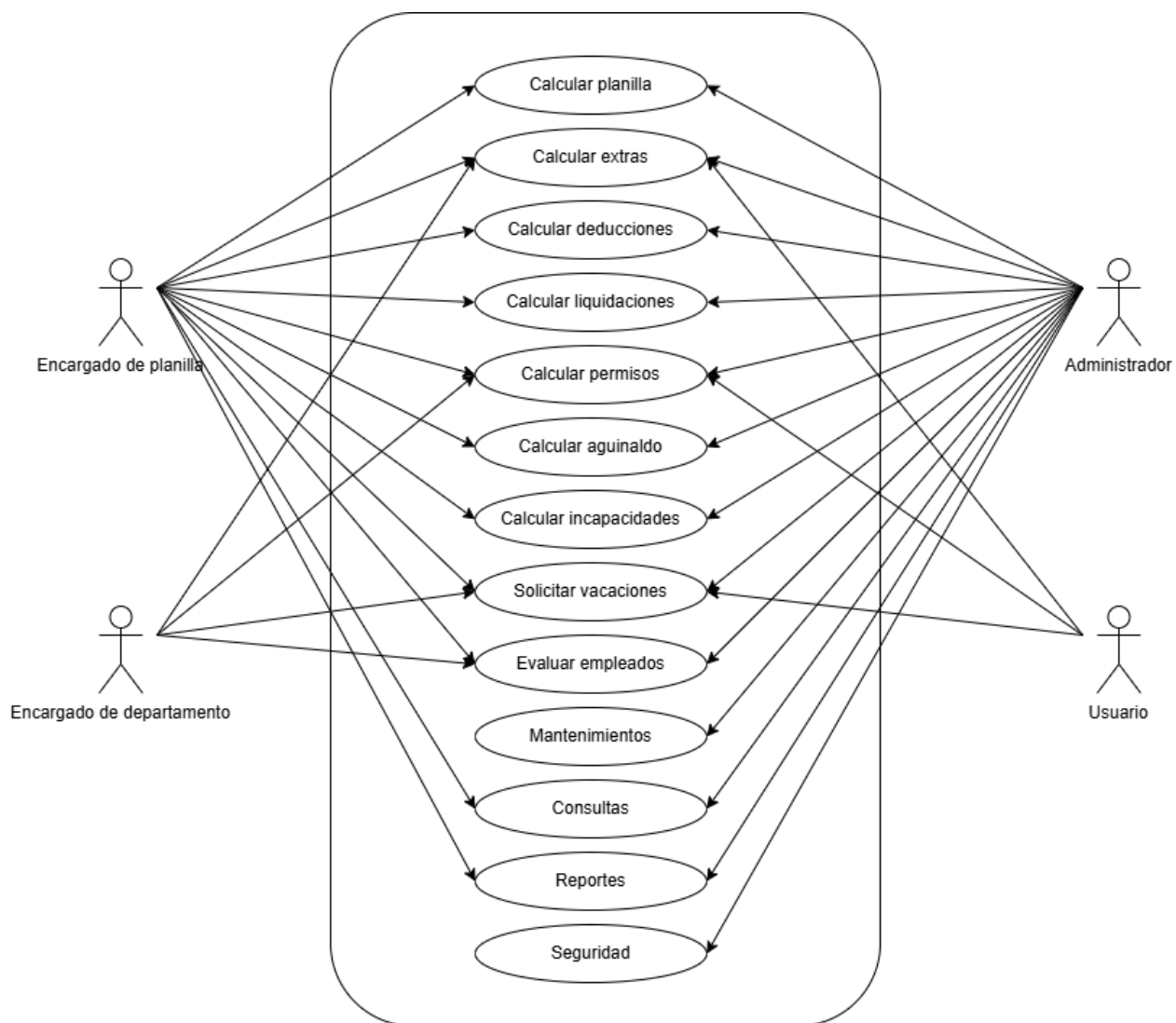
El sistema de recursos humanos fue diseñado considerando el perfil actual del personal de la empresa y sus capacidades tecnológicas, de forma que su implementación no simbolice una curva de aprendizaje compleja. Dado que la organización cuenta con personal familiarizados en el uso de computadoras y aplicaciones web, el nivel de conocimiento requerido para utilizar el sistema es básico-intermedio en el manejo de herramientas informáticas.

El sistema presenta una interfaz intuitiva y guiada, con mensajes y validaciones claras, lo que permite que los usuarios puedan adaptarse rápidamente tras una breve capacitación. No se requiere personal adicional ni cambios en la estructura organizativa, ya que cada usuario desempeñará sus funciones habituales desde la aplicación, con accesos restringidos según su rol (Empleado, Jefatura, Recursos Humanos o Administrador). Con ello, se garantiza la continuidad operativa y la adopción eficiente del sistema sin impacto negativo en la productividad del personal.

Casos de uso

A continuación, se proceden a mostrar el diagrama de caso de uso y la descripción de cada uno de ellos.

Figura 12. Diagrama de casos de uso



Fuente: Elaboración propia.

Descripción de casos de uso

En esta sección, se detallan los casos de uso que representan las principales funcionalidades del sistema propuesto, desarrolladas en respuesta a los requerimientos planteados por la empresa. Cada caso de uso describe una interacción específica entre los actores del sistema y el prototipo funcional, lo cual permite visualizar el comportamiento esperado ante distintas situaciones operativas.

Tabla 9: Caso de uso “Crear empleado”

Prototipo funcional para la gestión del recurso humano en la empresa Nutrimax, ubicada en Curridabat	
Número de caso de uso: CU-CE-01	Nombre del caso de uso: Crear empleado
Fecha elaboración:	25/09/2025
Descripción del caso de uso:	Recursos Humanos registra un nuevo empleado en el sistema con sus datos personales, laborales y de seguridad (rol/usuario), quedando disponible para procesos de planilla y control de asistencia.
Autor del caso de uso:	Marvin Rodrigo Núñez Sandoval
Actores relacionados:	Recursos Humanos (actor primario) Administrador del sistema (actor secundario, para creación de usuario/rol si aplica)
Precondiciones:	Usuario de Recursos Humanos autenticado con permisos para “Empleados”. Catálogos disponibles: Departamentos, Horarios. Reglas de validación definidas (formato de cédula, correos, fechas).
Flujo básico del caso de uso	
<ol style="list-style-type: none"> 1. El usuario de Recursos Humanos ingresa al módulo Empleados y presiona Agregar nuevo Empleado. 2. El sistema muestra el formulario de alta con campos obligatorios (cédula, nombre, apellidos, fecha ingreso, salario base, departamento, horario, correo, teléfono, estado). 3. El usuario de Recursos Humanos completa los campos requeridos. 4. El usuario de Recursos Humanos presiona Guardar. 5. El sistema valida los datos (obligatorios, formatos, duplicidad por cédula). 	

6. El sistema crea el registro del empleado.	
7. El sistema muestra confirmación y redirige al Listado con el nuevo empleado visible.	
Subflujos	
SF-1	El sistema mostrará por medio de una pantalla un resumen de los empleados del sistema.
Flujos alternos	
FA-01 Datos inválidos/faltantes	En el paso 6, el sistema resalta campos con error; el usuario de Recursos Humanos corrige y vuelve al paso 5.
FA-02 Cédula duplicada	En el paso 6, el sistema informa duplicidad y bloquea el guardado; el usuario de Recursos Humanos corrige y vuelve al paso 3.
Requerimientos especiales	
Auditoría: guardar en AuditLog los campos Antes/Después (alta).	
Protección de datos personales (acceso restringido por rol).	
Validaciones de país (CR): formato de cédula y teléfono.	
Poscondiciones	
Empleado creado en estado Activo y visible en listados y búsquedas.	

Fuente: Elaboración propia.

Tabla 10: Caso de uso “Editar empleado”

Prototipo funcional para la gestión del recurso humano en la empresa Nutrimax, ubicada en Curridabat	
Número de caso de uso: CU-EMP-02	Nombre del caso de uso: Editar empleado
Fecha elaboración:	25/09/2025
Descripción del caso de uso:	Recursos Humanos actualiza la información de un empleado existente.
Autor del caso de uso:	Marvin Rodrigo Núñez Sandoval
Actores relacionados:	Recursos Humanos (actor primario)
Precondiciones:	Empleado existente en el sistema.
Flujo básico del caso de uso	
<ol style="list-style-type: none"> 1. El usuario de Recursos Humanos abre listado de empleados y selecciona Editar. 2. El sistema muestra datos actuales. 3. Recursos Humanos modifica campos y guarda. 4. El sistema valida y actualiza. 5. El sistema confirma cambios. 	
Subflujos	
SF-1	N/A
Flujos alternos	
FA-01 Datos inválidos	Sistema resalta errores.
FA-02 Cédula duplicada	Sistema bloquea guardado.
Requerimientos especiales	
Auditoría en edición.	
Poscondiciones	

Empleado actualizado.

Fuente: Elaboración propia.

Tabla 11: Caso de uso “Eliminar empleado”

Prototipo funcional para la gestión del recurso humano en la empresa Nutrimax, ubicada en Curridabat	
Número de caso de uso: CU-EMP-03	Nombre del caso de uso: Eliminar empleado
Fecha elaboración:	25/09/2025
Descripción del caso de uso:	Recursos Humanos elimina o inactiva un empleado.
Autor del caso de uso:	Marvin Rodrigo Núñez Sandoval
Actores relacionados:	Recursos Humanos (actor primario)
Precondiciones:	Empleado registrado en el sistema.
Flujo básico del caso de uso	
<ol style="list-style-type: none"> 1. El usuario de Recursos Humanos selecciona un empleado y presiona Eliminar. 2. El sistema pide confirmación. 3. El usuario de Recursos Humanos confirma. 4. El sistema valida referencias. 5. Si procede, elimina/inactiva y confirma. 	
Subflujos	
SF-1	Ninguno
Flujos alternos	
FA-01 Asignaciones	El empleado tiene planillas o registros: sistema bloquea y sugiere inactivar.
Requerimientos especiales	
Auditoría en eliminación.	
Poscondiciones	
Empleado eliminado/inactivo.	

Fuente: Elaboración propia.

Tabla 12: Caso de uso “Leer empleado”

Prototipo funcional para la gestión del recurso humano en la empresa Nutrimax, ubicada en Curridabat	
Número de caso de uso: CU-EMP-04	Nombre del caso de uso: Leer empleado
Fecha elaboración:	25/09/2025
Descripción del caso de uso:	Recursos Humanos consulta el listado de empleados.
Autor del caso de uso:	Marvin Rodrigo Núñez Sandoval
Actores relacionados:	Recursos Humanos (actor primario)
Precondiciones:	Usuario de Recursos Humanos autenticado con permisos para “Empleados”.
Flujo básico del caso de uso	
<ol style="list-style-type: none"> 1. El usuario de Recursos Humanos abre módulo Empleados. 2. El sistema muestra listado. 3. El usuario de Recursos Humanos navega. 	
Subflujos	
SF-1	N/A
Flujos alternos	
FA-01	N/A
Requerimientos especiales	
Acceso restringido a el usuario de Recursos Humanos.	
Poscondiciones	
Listado visible para gestión.	

Fuente: Elaboración propia.

Tabla 13: Caso de uso “Crear departamento”

Prototipo funcional para la gestión del recurso humano en la empresa Nutrimax, ubicada en Curridabat	
Número de caso de uso: CU-DEP-01	Nombre del caso de uso: Crear departamento
Fecha elaboración:	25/09/2025
Descripción del caso de uso:	Recursos Humanos registra un nuevo departamento para asignarlo a empleados y puestos.
Autor del caso de uso:	Marvin Rodrigo Núñez Sandoval
Actores relacionados:	Recursos Humanos (actor primario)
Precondiciones:	Usuario Recursos Humanos autenticado con permisos para “Departamentos”. No debe existir un departamento activo con el mismo nombre.
Flujo básico del caso de uso	
<ol style="list-style-type: none"> 1. Recursos Humanos ingresa al módulo Departamentos y presiona Agregar departamento. 2. El sistema muestra formulario con campos obligatorios (Nombre, Descripción, Fecha de creación). 3. Recursos Humanos completa los campos y presiona Guardar. 4. El sistema valida obligatorios y duplicidad por Nombre. 5. El sistema crea el registro y redirige al Listado mostrando confirmación. 	
Subflujos	
SF-1	Cancelar creación: Recursos Humanos puede cancelar antes de guardar y volver al listado.
Flujos alternos	
FA-01 Datos inválidos/faltantes	En el paso 4, el sistema resalta errores; Recursos Humanos corrige y vuelve al paso 3.
FA-02 Nombre duplicado	En el paso 4, el sistema informa duplicidad y bloquea el guardado; Recursos Humanos corrige y vuelve al paso 3.
Requerimientos especiales	

Registrar alta en AuditLog (Antes/Después). Control de acceso por rol.
Poscondiciones
Departamento creado y disponible en listas desplegables y filtros.

Fuente: Elaboración propia.

Tabla 14: Caso de uso “Editar departamento”

Prototipo funcional para la gestión del recurso humano en la empresa Nutrimax, ubicada en Curridabat	
Número de caso de uso: CU-DEP-Editar	Nombre del caso de uso: Editar departamento
Fecha elaboración:	25/09/2025
Descripción del caso de uso:	Recursos Humanos actualiza datos de un departamento existente.
Autor del caso de uso:	Marvin Rodrigo Núñez Sandoval
Actores relacionados:	Recursos Humanos (actor primario) Administrador del sistema (actor secundario, para creación de usuario/rol si aplica)
Precondiciones:	Departamento existente. Usuario Recursos Humanos autenticado.
Flujo básico del caso de uso	
<ol style="list-style-type: none"> 1. Recursos Humanos abre Departamentos y elige Editar en el registro deseado. 2. El sistema muestra formulario con datos actuales. 3. Recursos Humanos modifica campos y presiona Guardar. 4. El sistema valida y actualiza el registro. 5. El sistema muestra confirmación y vuelve al Listado. 	
Subflujos	
SF-1	Cancelar edición: Recursos Humanos descarta cambios y vuelve al listado.
Flujos alternos	
FA-01 Datos inválidos/faltantes	En el paso 4, el sistema señala errores; Recursos Humanos corrige y vuelve al paso 3.
FA-02 Duplicidad por nombre	En el paso 4, el sistema advierte duplicidad y bloquea actualización.
Requerimientos especiales	

Registrar edición en AuditLog. Control de acceso por rol.
Poscondiciones
Departamento actualizado y visible en listados/combo.

Fuente: Elaboración propia.

Tabla 15: Caso de uso “Eliminar departamento”

Prototipo funcional para la gestión del recurso humano en la empresa Nutrimax, ubicada en Curridabat	
Número de caso de uso: CU-DEP-Eliminar	Nombre del caso de uso: Eliminar departamento
Fecha elaboración:	25/09/2025
Descripción del caso de uso:	Recursos Humanos elimina o marca Inactivo un departamento, según políticas de referencia.
Autor del caso de uso:	Marvin Rodrigo Núñez Sandoval
Actores relacionados:	Recursos Humanos (actor primario) Administrador del sistema (actor secundario, para creación de usuario/rol si aplica)
Precondiciones:	Departamento existente. Usuario Recursos Humanos autenticado.
Flujo básico del caso de uso	
<ol style="list-style-type: none"> 1. Recursos Humanos abre Departamentos y presiona Eliminar en el registro. 2. El sistema solicita confirmación. 3. Recursos Humanos confirma. 4. El sistema valida referencias (empleados/puestos). 5. Si no hay referencias, elimina; si existen, marca Inactivo (según política). 6. El sistema muestra confirmación. 	
Subflujos	
SF-1 Revisión de referencias	El sistema lista entidades relacionadas, si las hubiera.
Flujos alternos	
FA-01 Referencias críticas	El sistema bloquea eliminación y sugiere inactivar.
FA-02 Cancelación del usuario	Recursos Humanos cancela en el paso 2; no hay cambios.
Requerimientos especiales	

Registrar eliminación/inactivación en AuditLog. Control de acceso por rol.
Poscondiciones
Departamento eliminado o inactivo; no disponible para nuevas asignaciones si está inactivo.

Fuente: Elaboración propia.

Tabla 16: Caso de uso “Leer departamento”

Prototipo funcional para la gestión del recurso humano en la empresa Nutrimax, ubicada en Curridabat	
Número de caso de uso: CU-DEP-Listar	Nombre del caso de uso: Listar departamentos
Fecha elaboración:	25/09/2025
Descripción del caso de uso:	Recursos Humanos visualiza el listado de departamentos con filtros y paginación.
Autor del caso de uso:	Marvin Rodrigo Núñez Sandoval
Actores relacionados:	Recursos Humanos (actor primario) Administrador del sistema (actor secundario, para creación de usuario/rol si aplica)
Precondiciones:	Usuario Recursos Humanos autenticado.
Flujo básico del caso de uso	
<ol style="list-style-type: none"> 1. Recursos Humanos accede a Departamentos. 2. El sistema muestra listado de los Departamentos. 3. Recursos Humanos navega entre páginas. 	
Subflujos	
SF-1	Cancelar accion.
Flujos alternos	
FA-01 Sin resultados	El sistema informa “sin registros para filtros aplicados”.
Requerimientos especiales	
<p>Auditoría de consultas. Control de acceso por rol.</p>	

Poscondiciones
Información visible para gestión y navegación a Crear/Editar/Eliminar.

Fuente: Elaboración propia.

Tabla 17: Caso de uso “Crear puesto”

Prototipo funcional para la gestión del recurso humano en la empresa Nutrimax, ubicada en Curridabat	
Número de caso de uso: CU-PUE-Crear	Nombre del caso de uso: Crear Puesto
Fecha elaboración:	25/09/2025
Descripción del caso de uso:	Recursos Humanos registra un nuevo puesto asociado a un departamento.
Autor del caso de uso:	Marvin Rodrigo Núñez Sandoval
Actores relacionados:	Recursos Humanos (actor primario) Administrador del sistema (actor secundario, para creación de usuario/rol si aplica)
Precondiciones:	Usuario Recursos Humanos autenticado. Debe existir al menos un Departamento activo.
Flujo básico del caso de uso	
<ol style="list-style-type: none"> 1. Recursos Humanos ingresa a Puestos y presiona Agregar Puesto. 2. Formulario: Nombre, Departamento, Descripción, Estado. 3. Recursos Humanos completa y Guarda. 4. Validación de obligatorios y duplicidad por Nombre dentro del mismo Departamento. 5. El sistema crea el puesto y muestra confirmación. 	
Subflujos	
SF-1	Cancelar creación antes de guardar.
Flujos alternos	
FA-01 Datos inválidos/faltantes	Corregir y volver a guardar.
FA-02 Departamento inexistente/inactivo:	Sistema impide guardado y solicita selección válida.

Requerimientos especiales
Auditoría de alta. Control de acceso.
Poscondiciones
Puesto disponible para asignación a empleados.

Fuente: Elaboración propia.

Tabla 18: Caso de uso “Editar puesto”

Prototipo funcional para la gestión del recurso humano en la empresa Nutrimax, ubicada en Curridabat	
Número de caso de uso: CU-PUE-Editar	Nombre del caso de uso: Editar puesto
Fecha elaboración:	25/09/2025
Descripción del caso de uso:	Recursos Humanos actualiza información de un puesto.
Autor del caso de uso:	Marvin Rodrigo Núñez Sandoval
Actores relacionados:	Recursos Humanos (actor primario) Administrador del sistema (actor secundario, para creación de usuario/rol si aplica)
Precondiciones:	Puesto existente. Usuario Recursos Humanos autenticado.
Flujo básico del caso de uso	
<ol style="list-style-type: none"> 1. Recursos Humanos selecciona Editar en el listado de Puestos. 2. El sistema muestra datos actuales. 3. Recursos Humanos modifica y Guarda. 4. Sistema valida y actualiza. 5. Muestra confirmación. 	
Subflujos	
SF-1	Cancelar edición.
Flujos alternos	
FA-01 Datos inválidos/duplicados.	Sistema muestra errores.
Requerimientos especiales	
<p>Auditoría de edición.</p> <p>Control de acceso.</p>	

Poscondiciones
Puesto actualizado.

Fuente: Elaboración propia.

Tabla 19: Caso de uso “Eliminar puesto”

Prototipo funcional para la gestión del recurso humano en la empresa Nutrimax, ubicada en Curridabat	
Número de caso de uso: CU-PUE-Elimniar	Nombre del caso de uso: Eliminar/Inactivar puesto
Fecha elaboración:	25/09/2025
Descripción del caso de uso:	El usuario de Recursos Humanos elimina o marca Inactivo un puesto.
Autor del caso de uso:	Marvin Rodrigo Núñez Sandoval
Actores relacionados:	Recursos Humanos (actor primario) Administrador del sistema (actor secundario, para creación de usuario/rol si aplica)
Precondiciones:	Puesto existente. Sin empleados activos asociados (o política de inactivación).
Flujo básico del caso de uso	
<ol style="list-style-type: none"> 1. El usuario de Recursos Humanos presiona Eliminar en el puesto. 2. El sistema pide confirmación. 3. El usuario de Recursos Humanos confirma. 4. El sistema valida referencias (empleados). 5. El sistema elimina o inactiva según política y confirma. 	
Subflujos	
SF-1	Listar referencias (si las hay) para decisión del usuario.
Flujos alternos	
FA-01 Puesto en uso	El sistema bloquea eliminación y sugiere inactivar.
FA-02 Cancelación del usuario.	El usuario cancela la opción.

Requerimientos especiales
Auditoría. Control de acceso.
Poscondiciones
Puesto eliminado o inactivo; ya no seleccionable si está inactivo.

Fuente: Elaboración propia.

Tabla 20: Caso de uso “Leer puesto”

Prototipo funcional para la gestión del recurso humano en la empresa Nutrimax, ubicada en Curridabat	
Número de caso de uso: CU-PUE-Listar	Nombre del caso de uso: Listar puestos
Fecha elaboración:	25/09/2025
Descripción del caso de uso:	El usuario de Recursos Humanos consulta y filtra los puestos.
Autor del caso de uso:	Marvin Rodrigo Núñez Sandoval
Actores relacionados:	Recursos Humanos (actor primario) Administrador del sistema (actor secundario, para creación de usuario/rol si aplica)
Precondiciones:	Usuario de Recursos Humanos autenticado.
Flujo básico del caso de uso	
<ol style="list-style-type: none"> 1. Abrir Puestos. 2. Ver listado de los puestos con Nombre, filtro por Departamento y Estado. 	
Subflujos	
SF-1	Exportar listado (opcional) a Excel/PDF.
Flujos alternos	
FA-01 Sin resultados	Para filtros aplicados.
Requerimientos especiales	
Control de acceso.	
Poscondiciones	

Listado disponible para gestión.

Fuente: Elaboración propia.

Tabla 21: Caso de uso “Crear Horario”

Prototipo funcional para la gestión del recurso humano en la empresa Nutrimax, ubicada en Curridabat	
Número de caso de uso: CU-HOR-CREAR	Nombre del caso de uso: Crear Horario
Fecha elaboración:	25/09/2025
Descripción del caso de uso:	El usuario de Recursos Humanos define un nuevo horario (entrada/salida, días hábiles, tolerancias).
Autor del caso de uso:	Marvin Rodrigo Núñez Sandoval
Actores relacionados:	Recursos Humanos (actor primario) Administrador del sistema (actor secundario, para creación de usuario/rol si aplica)
Precondiciones:	Usuario de Recursos Humanos autenticado.
Flujo básico del caso de uso	
<ol style="list-style-type: none"> 1. El usuario de Recursos Humanos abre Horarios, Agregar Horario. 2. Formulario: Nombre, Descripción, Horas por día, Horas por semana, Hora entrada, Hora salida. 3. El usuario de Recursos Humanos completa y Guarda. 4. Validación de consistencia: entrada < salida, días válidos, tolerancias. 5. El sistema crea el horario y confirma. 	
Subflujos	
SF-1	Cancelar edición.
Flujos alternos	
FA-01 Datos inconsistentes	Inconsistencia de datos en horas/días.
Requerimientos especiales	
Auditoría de alta. Control de acceso.	

Poscondiciones
Horario disponible para asignación a empleados.

Fuente: Elaboración propia.

Tabla 22: Caso de uso “Editar Horario”

Prototipo funcional para la gestión del recurso humano en la empresa Nutrimax, ubicada en Curridabat	
Número de caso de uso: CU-HOR-Editar	Nombre del caso de uso: Editar Horario
Fecha elaboración:	25/09/2025
Descripción del caso de uso:	El usuario de Recursos Humanos ajusta parámetros de un horario.
Autor del caso de uso:	Marvin Rodrigo Núñez Sandoval
Actores relacionados:	Recursos Humanos (actor primario) Administrador del sistema (actor secundario, para creación de usuario/rol si aplica)
Precondiciones:	Horario existente.
Flujo básico del caso de uso	
<ol style="list-style-type: none"> 1. El usuario de Recursos Humanos elige Editar en Horarios. 2. Modifica campos y Guarda. 3. Validaciones de consistencia. 4. El sistema actualiza y confirma. 	
Subflujos	
SF-1	Cancelar edición.
Flujos alternos	
FA-01 Datos inválidos/inconsistentes.	Sistema muestra errores.
Requerimientos especiales	
Auditoría de edición.	
Poscondiciones	

Horario actualizado.

Fuente: Elaboración propia.

Tabla 23: Caso de uso “Eliminar Horario”

Prototipo funcional para la gestión del recurso humano en la empresa Nutrimax, ubicada en Curridabat	
Número de caso de uso: CU-HOR-Eliminar	Nombre del caso de uso: Eliminar/Inactivar horario
Fecha elaboración:	25/09/2025
Descripción del caso de uso:	El usuario de Recursos Humanos elimina o marca Inactivo un horario.
Autor del caso de uso:	Marvin Rodrigo Núñez Sandoval
Actores relacionados:	Recursos Humanos (actor primario) Administrador del sistema (actor secundario, para creación de usuario/rol si aplica)
Precondiciones:	Horario existente. Sin empleados activos con ese horario (o política de inactivar).
Flujo básico del caso de uso	
<ol style="list-style-type: none"> 1. Eliminar en el registro de horario. 2. Confirmación del usuario. 3. Validación de referencias (empleados). 4. Eliminar o inactivar y confirmar. 	
Subflujos	
SF-1	Cancelar acción.
Flujos alternos	
FA-01 Horario en uso:	Bloquear eliminación y sugerir inactivar.
Requerimientos especiales	
Auditoría	

Poscondiciones
Horario eliminado/inactivo.

Fuente: Elaboración propia.

Tabla 24: Caso de uso “Leer Horario”

Prototipo funcional para la gestión del recurso humano en la empresa Nutrimax, ubicada en Curridabat	
Número de caso de uso: CU-HOR-Listar	Nombre del caso de uso: Leer horarios
Fecha elaboración:	25/09/2025
Descripción del caso de uso:	Recursos Humanos visualiza horarios.
Autor del caso de uso:	Marvin Rodrigo Núñez Sandoval
Actores relacionados:	Recursos Humanos (actor primario) Administrador del sistema (actor secundario, para creación de usuario/rol si aplica)
Precondiciones:	Usuario autenticado.
Flujo básico del caso de uso	
<ol style="list-style-type: none"> 1. Abrir Horarios. 2. Ver listado de horarios por Nombre y Estado. 3. Navegar. 	
Subflujos	
SF-1	Exportar (opcional).
Flujos alternos	
FA-01	N/A
Requerimientos especiales	
(Opcional) Auditoría de vistas.	
Poscondiciones	

Listado disponible para gestión.

Fuente: Elaboración propia.

Tabla 25: Caso de uso “Crear Rol”

Prototipo funcional para la gestión del recurso humano en la empresa Nutrimax, ubicada en Curridabat	
Número de caso de uso: CU-ROL-Crear	Nombre del caso de uso: Crear Rol
Fecha elaboración:	25/09/2025
Descripción del caso de uso:	Administrador define un rol con permisos del sistema.
Autor del caso de uso:	Marvin Rodrigo Núñez Sandoval
Actores relacionados:	Administrador (actor primario)
Precondiciones:	Administrador autenticado.
Flujo básico del caso de uso	
<ol style="list-style-type: none"> 1. Administrador abre Roles, Agregar Rol. 2. Formulario: Nombre del Rol. 3. Guardar. 4. Validación de Nombre (obligatorio/duplicidad). 5. El sistema crea el rol y confirma. 	
Subflujos	
SF-1	Cancelar acción.
Flujos alternos	
FA-01 Nombre duplicado	Sistema muestra errores.
Requerimientos especiales	
<p>Auditoría.</p> <p>Control de acceso.</p>	
Poscondiciones	

Rol disponible para asignación a usuarios.

Fuente: Elaboración propia.

Tabla 26: Caso de uso “Editar Rol”

Prototipo funcional para la gestión del recurso humano en la empresa Nutrimax, ubicada en Curridabat	
Número de caso de uso: CU-ROL-Editar	Nombre del caso de uso: Editar Rol
Fecha elaboración:	25/09/2025
Descripción del caso de uso:	Administrador ajusta permisos y metadatos de un rol.
Autor del caso de uso:	Marvin Rodrigo Núñez Sandoval
Actores relacionados:	Administrador (actor primario)
Precondiciones:	Rol existente.
Flujo básico del caso de uso	
<ol style="list-style-type: none"> 1. Elegir Editar en Roles. 2. Modificar. 3. Guardar. 4. Validaciones y actualización. 5. Confirmación. 	
Subflujos	
SF-1	Revertir cambios (cancelar).
Flujos alternos	
FA-01 Nombre duplicado	Sistema muestra errores.
Requerimientos especiales	
Auditoría	
Poscondiciones	

Rol actualizado.

Fuente: Elaboración propia.

Tabla 27: Caso de uso “Eliminar Rol”

Prototipo funcional para la gestión del recurso humano en la empresa Nutrimax, ubicada en Curridabat	
Número de caso de uso: CU-ROL-Eliminar	Nombre del caso de uso: Eliminar Rol
Fecha elaboración:	25/09/2025
Descripción del caso de uso:	Administrador elimina o inactiva un rol.
Autor del caso de uso:	Marvin Rodrigo Núñez Sandoval
Actores relacionados:	Administrador (actor primario)
Precondiciones:	Rol existente. Sin usuarios asociados (o política de inactivar).
Flujo básico del caso de uso	
<ol style="list-style-type: none"> 1. Eliminar en el rol. 2. Confirmación. 3. Validación de usuarios asociados. 4. Eliminar o inactivar y confirmar. 	
Subflujos	
SF-1	Cancelar acción.
Flujos alternos	
FA-01 Rol en uso	Bloquear eliminación y sugerir inactivar.
FA-02 Cancelación	Del administrador.
Requerimientos especiales	
Auditoría	

Poscondiciones
Rol eliminado/inactivo.

Fuente: Elaboración propia.

Tabla 28: Caso de uso “Leer Rol”

Prototipo funcional para la gestión del recurso humano en la empresa Nutrimax, ubicada en Curridabat	
Número de caso de uso: CU-ROL-Listar	Nombre del caso de uso: Leer/filtrar roles
Fecha elaboración:	25/09/2025
Descripción del caso de uso:	Administrador consulta y filtra roles.
Autor del caso de uso:	Marvin Rodrigo Núñez Sandoval
Actores relacionados:	Administrador (actor primario)
Precondiciones:	Administrador autenticado.
Flujo básico del caso de uso	
<ol style="list-style-type: none"> 1. Abrir Roles. 2. Ver listado de Roles con Nombre. 3. Navegar. 	
Subflujos	
SF-1	Exportar (opcional).
Flujos alternos	
FA-01	N/A
Requerimientos especiales	
Auditoría de vistas.	
Poscondiciones	
Listado visible para gestión.	

Fuente: Elaboración propia.

Tabla 29: Caso de uso “Crear Usuario”

Prototipo funcional para la gestión del recurso humano en la empresa Nutrimax, ubicada en Curridabat	
Número de caso de uso: CU-USR-Crear	Nombre del caso de uso: Crear usuario
Fecha elaboración:	25/09/2025
Descripción del caso de uso:	Administrador o Recursos Humanos crea un usuario del sistema para un empleado existente, asignándole rol y credenciales iniciales.
Autor del caso de uso:	Marvin Rodrigo Núñez Sandoval
Actores relacionados:	Administrador (actor primario) Recursos Humanos (actor secundario, si la política lo permite)
Precondiciones:	Empleado existente y activo. Políticas de complejidad de contraseña y roles configuradas.
Flujo básico del caso de uso	
<ol style="list-style-type: none"> 1. Administrador abre Usuarios, Agregar Usuario. 2. Selecciona Empleado. 3. Define Nombre de Usuario, Contraseña, Empleado y Rol. 4. Presiona Guardar. 5. El sistema valida duplicidad de Usuario/Correo y políticas de contraseña. 6. El sistema crea el usuario. 	
Subflujos	
SF-1	Cancelar acción.
Flujos alternos	
FA-01 Empleado ya tiene usuario	Sistema bloquea creación.

FA-02 Usuario/Correo duplicado:	Sistema bloquea guardado.
Requerimientos especiales	
Auditoría en alta. Controles de seguridad (hash de contraseña, expiración de contraseña temporal).	
Poscondiciones	
Usuario creado y activo; empleado puede iniciar sesión según rol.	

Fuente: Elaboración propia.

Tabla 30: Caso de uso “Editar Usuario”

Prototipo funcional para la gestión del recurso humano en la empresa Nutrimax, ubicada en Curridabat	
Número de caso de uso: CU-USR-Editar	Nombre del caso de uso: Editar usuario
Fecha elaboración:	25/09/2025
Descripción del caso de uso:	Administrador/ Recursos Humanos actualiza rol, estado o correo de un usuario.
Autor del caso de uso:	Marvin Rodrigo Núñez Sandoval
Actores relacionados:	Administrador (actor primario) Recursos Humanos (actor secundario, si aplica)
Precondiciones:	Usuario existente. Políticas de seguridad vigentes.
Flujo básico del caso de uso	
<ol style="list-style-type: none"> 1. Abrir Usuarios, Editar. 2. Modificar Rol, Contraseña, Empleado y Rol. 3. Guardar. 4. Validaciones y actualización. 5. Confirmación. 	
Subflujos	
SF-1	Restablecer contraseña (opcional): genera clave temporal.
Flujos alternos	
FA-01	N/A
Requerimientos especiales	
Auditoría de edición. Notificación al usuario si cambió su rol/estado (opcional).	

Poscondiciones
Usuario actualizado.

Fuente: Elaboración propia.

Tabla 31: Caso de uso “Eliminar Usuario”

Prototipo funcional para la gestión del recurso humano en la empresa Nutrimax, ubicada en Curridabat	
Número de caso de uso: CU-USR-Eliminar	Nombre del caso de uso: Eliminar/Desactivar usuario
Fecha elaboración:	25/09/2025
Descripción del caso de uso:	Administrador desactiva o elimina un usuario del sistema según política.
Autor del caso de uso:	Marvin Rodrigo Núñez Sandoval
Actores relacionados:	Administrador (actor primario)
Precondiciones:	Usuario existente. Política: preferir desactivar para conservar trazabilidad.
Flujo básico del caso de uso	
<ol style="list-style-type: none"> 1. En Usuarios, seleccionar Eliminar. 2. Confirmar acción. 3. El sistema desactiva (o elimina si la política lo permite) y confirma. 	
Subflujos	
SF-1	Cancelar acción.
Flujos alternos	
FA-01 Bloqueo por políticas	Usuarios críticos del sistema no se pueden eliminar.
FA-02 Cancelación del administrador	Se cancela la solicitud.
Requerimientos especiales	

Auditoría de baja.
Poscondiciones
Usuario desactivado/eliminado; no puede iniciar sesión.

Fuente: Elaboración propia.

Tabla 32: Caso de uso “Leer Usuario”

Prototipo funcional para la gestión del recurso humano en la empresa Nutrimax, ubicada en Curridabat	
Número de caso de uso: CU-USR-Eliminar	Nombre del caso de uso: Leer Usuario
Fecha elaboración:	25/09/2025
Descripción del caso de uso:	Administrador desactiva o elimina un usuario del sistema según política.
Autor del caso de uso:	Marvin Rodrigo Núñez Sandoval
Actores relacionados:	Administrador (actor primario)
Precondiciones:	Usuario existente. Política: preferir desactivar para conservar trazabilidad.
Flujo básico del caso de uso	
<ol style="list-style-type: none"> 1. Abrir Usuarios. 2. Ver listado de Usuarios con Nombre. 3. Navegar. 	
Subflujos	
SF-1	N/A
Flujos alternos	
FA-01	N/A
Requerimientos especiales	
Permisos de Rol.	
Poscondiciones	

Listado visible para gestión.

Fuente: Elaboración propia.

Tabla 33: Caso de uso “Crear Vacaciones”

Prototipo funcional para la gestión del recurso humano en la empresa Nutrimax, ubicada en Curridabat	
Número de caso de uso: CU-VAC-Solicitar	Nombre del caso de uso: Crear Vacaciones
Fecha elaboración:	25/09/2025
Descripción del caso de uso:	El empleado registra una solicitud de vacaciones; el sistema valida elegibilidad y saldo disponible y la deja en estado “Pendiente de jefatura”.
Autor del caso de uso:	Marvin Rodrigo Núñez Sandoval
Actores relacionados:	Empleado (actor primario) Jefatura (actor secundario) Recursos Humanos (actor secundario)
Precondiciones:	Empleado activo con cuenta válida. Reglas CR vigentes (elegibilidad ≥ 1 año o proporcional, 12 días hábiles/año).
Flujo básico del caso de uso	
<ol style="list-style-type: none"> 1. Empleado abre Acciones, Solicitar vacaciones. 2. Ingresa rango de fechas. 3. El sistema calcula días hábiles del rango. 4. El sistema verifica saldo disponible. 5. Muestra resumen (fechas, días, saldo antes/después). 6. Empleado confirma. 7. El sistema registra Pendiente de jefatura y notifica a jefatura. 	
Subflujos	
SF-1	Cancelar solicitud antes de confirmar.
Flujos alternos	

FA-01 Saldo insuficiente	Sistema indica falta de saldo y sugiere acortar rango.
FA-02 Traslape con otra ausencia	El sistema notifica.
Requerimientos especiales	
Auditoría de creación (Antes/Después). Cómputo de días hábiles (excluir fines de semana/feriados).	
Poscondiciones	
Solicitud registrada en estado Pendiente de jefatura.	

Fuente: Elaboración propia.

Tabla 34: Caso de uso “Editar Vacaciones”

Prototipo funcional para la gestión del recurso humano en la empresa Nutrimax, ubicada en Curridabat	
Número de caso de uso: CU-VAC-Editar	Nombre del caso de uso: Editar Vacaciones
Fecha elaboración:	25/09/2025
Descripción del caso de uso:	El empleado modifica una solicitud de vacaciones mientras esté pendiente.
Autor del caso de uso:	Marvin Rodrigo Núñez Sandoval
Actores relacionados:	Empleado (actor primario)
Precondiciones:	Solicitud en estado Pendiente.
Flujo básico del caso de uso	
<ol style="list-style-type: none"> 1. Empleado abre su solicitud y presiona Editar. 2. Ajusta fechas y guarda. 3. El sistema recalcula días y saldo. 4. El sistema actualiza la solicitud. 	
Subflujos	
SF-1	Cancelar edición (descartar cambios).
Flujos alternos	
FA-01 Cambios inválidos/sin saldo	Sistema muestra errores; empleado corrige.
Requerimientos especiales	
Auditoría de edición.	
Poscondiciones	

Solicitud actualizada y aún Pendiente.

Fuente: Elaboración propia.

Tabla 35: Caso de uso “Cancelar Vacaciones”

Prototipo funcional para la gestión del recurso humano en la empresa Nutrimax, ubicada en Curridabat	
Número de caso de uso: CU-VAC-Cancelar	Nombre del caso de uso: Cancelar solicitud de vacaciones
Fecha elaboración:	25/09/2025
Descripción del caso de uso:	El empleado cancela una solicitud antes de su aprobación.
Autor del caso de uso:	Marvin Rodrigo Núñez Sandoval
Actores relacionados:	Empleado (actor primario)
Precondiciones:	Solicitud Pendiente o Aprobada con política que permita cancelación anticipada.
Flujo básico del caso de uso	
<ol style="list-style-type: none"> 1. Empleado selecciona Cancelar. 2. Sistema solicita confirmación. 3. Empleado confirma. 4. Sistema cambia estado a Cancelada y libera saldo si corresponde. 	
Subflujos	
SF-1	Motivo de cancelación (opcional).
Flujos alternos	
FA-01 Cancelación no permitida	Por política o fecha ya iniciada.
Requerimientos especiales	
Auditoría de cancelación.	
Poscondiciones	

Solicitud Cancelada.

Fuente: Elaboración propia.

Tabla 36: Caso de uso “Aprobar Vacaciones”

Prototipo funcional para la gestión del recurso humano en la empresa Nutrimax, ubicada en Curridabat	
Número de caso de uso: CU-VAC-Aprobar	Nombre del caso de uso: Aprobar solicitud de vacaciones
Fecha elaboración:	25/09/2025
Descripción del caso de uso:	La jefatura aprueba una solicitud; el sistema descuenta saldo y notifica.
Autor del caso de uso:	Marvin Rodrigo Núñez Sandoval
Actores relacionados:	Jefatura (actor primario) Empleado (secundario) Recursos Humanos (secundario)
Precondiciones:	Solicitud Pendiente de jefatura. Jefatura autenticada.
Flujo básico del caso de uso	
<ol style="list-style-type: none"> 1. Jefatura abre solicitudes Pendientes. 2. Revisa detalle y presiona Aprobar. 3. El sistema valida vigencia y saldo. 4. El sistema cambia a Aprobada y bloquea el rango. 5. Sistema notifica al empleado. 	
Subflujos	
SF-1	Comentario de aprobación (opcional).
Flujos alternos	
FA-01 Falta de saldo	El sistema bloquea aprobación. Sistema solicita intervención de Recursos Humanos.

FA-02 Rango ya iniciado	Sistema solicita intervención de Recursos Humanos.
Requerimientos especiales	
Auditoría de aprobación.	
Poscondiciones	
Solicitud Aprobada.	

Fuente: Elaboración propia.

Tabla 37: Caso de uso “Rechazar Vacaciones”

Prototipo funcional para la gestión del recurso humano en la empresa Nutrimax, ubicada en Curridabat	
Número de caso de uso: CU-VAC-Rechazar	Nombre del caso de uso: Rechazar solicitud de vacaciones
Fecha elaboración:	25/09/2025
Descripción del caso de uso:	La jefatura rechaza una solicitud indicando motivo.
Autor del caso de uso:	Marvin Rodrigo Núñez Sandoval
Actores relacionados:	Jefatura (actor primario) Empleado (secundario)
Precondiciones:	Solicitud Pendiente de jefatura.
Flujo básico del caso de uso	
<ol style="list-style-type: none"> 1. Jefatura elige Rechazar. 2. Ingresar motivo y confirma. 3. Sistema cambia a Rechazada y notifica. 	
Subflujos	
SF-1	N/A
Flujos alternos	
FA-01 Falta motivo	Sistema impide finalizar.
Requerimientos especiales	
Auditoría de rechazo.	
Poscondiciones	

Solicitud Rechazada.

Fuente: Elaboración propia.

Tabla 38: Caso de uso “Crear Permiso”

Prototipo funcional para la gestión del recurso humano en la empresa Nutrimax, ubicada en Curridabat	
Número de caso de uso: CU-PER-Solicitar	Nombre del caso de uso: Crear Permiso
Fecha elaboración:	25/09/2025
Descripción del caso de uso:	El empleado solicita un permiso (con goce / sin goce) indicando fechas/horas.
Autor del caso de uso:	Marvin Rodrigo Núñez Sandoval
Actores relacionados:	Empleado (actor primario) Jefatura (secundario) Recursos Humanos (secundario)
Precondiciones:	Empleado activo. Tipos de permiso parametrizados.
Flujo básico del caso de uso	
<ol style="list-style-type: none"> 1. Empleado abre Acciones, Solicitar Permisos. 2. Define tipo, rango y motivo. 3. Guardar. 4. Sistema registra Pendiente y notifica a jefatura. 	
Subflujos	
SF-1	Cancelar antes de guardar.
Flujos alternos	
FA-01 Días no disponible	Solicitud por rango, tipo.
FA-02 Traslape con vacaciones/marcas	Se cruzan con días no disponibles.

Requerimientos especiales	
Auditoría de creación.	
Poscondiciones	
Permiso Pendiente.	

Fuente: Elaboración propia.

Tabla 39: Caso de uso “Aprobar Permiso”

Prototipo funcional para la gestión del recurso humano en la empresa Nutrimax, ubicada en Curridabat	
Número de caso de uso: CU-PER-Aprobar	Nombre del caso de uso: Aprobar Permiso
Fecha elaboración:	25/09/2025
Descripción del caso de uso:	La jefatura aprueba el permiso y lo deja listo para impactar planilla si aplica.
Autor del caso de uso:	Marvin Rodrigo Núñez Sandoval
Actores relacionados:	Jefatura (actor primario) Empleado (secundario)
Precondiciones:	Permiso Pendiente.
Flujo básico del caso de uso	
<ol style="list-style-type: none"> 1. Jefatura revisa permiso. 2. Presiona Aprobar. 3. Sistema valida rango y políticas. 4. Cambia a Aprobado y notifica. 	
Subflujos	
SF-1	Comentario (opcional).
Flujos alternos	
FA-01 Información insuficiente	Solicitar corrección.
Requerimientos especiales	
Auditoría.	
Poscondiciones	

Permiso Aprobado.

Fuente: Elaboración propia.

Tabla 40: Caso de uso “Rechazar Permiso”

Prototipo funcional para la gestión del recurso humano en la empresa Nutrimax, ubicada en Curridabat	
Número de caso de uso: CU-PER-Rechazar	Nombre del caso de uso: Rechazar permiso
Fecha elaboración:	25/09/2025
Descripción del caso de uso:	La jefatura rechaza el permiso con un motivo.
Autor del caso de uso:	Marvin Rodrigo Núñez Sandoval
Actores relacionados:	Jefatura (actor primario) Empleado (secundario)
Precondiciones:	Permiso Pendiente.
Flujo básico del caso de uso	
1. Jefatura elige Rechazar. 2. Sistema cambia a Rechazado y notifica.	
Subflujos	
SF-1	N/A
Flujos alternos	
FA-01	N/A
Requerimientos especiales	
Auditoría.	
Poscondiciones	

Permiso Rechazado.

Fuente: Elaboración propia.

Tabla 41: Caso de uso “Leer Permiso”

Prototipo funcional para la gestión del recurso humano en la empresa Nutrimax, ubicada en Curridabat	
Número de caso de uso: CU-PER-Listar	Nombre del caso de uso: Leer/consultar permisos
Fecha elaboración:	25/09/2025
Descripción del caso de uso:	Recursos Humanos lista y filtra permisos por estado, tipo, empleado y rango.
Autor del caso de uso:	Marvin Rodrigo Núñez Sandoval
Actores relacionados:	Recursos Humanos (actor primario)
Precondiciones:	Usuario de Recursos Humanos autenticado.
Flujo básico del caso de uso	
<ol style="list-style-type: none"> 1. El usuario de Recursos Humanos abre Permisos. 2. Aplica filtros y consulta. 3. Revisa estados y detalles. 	
Subflujos	
SF-1	Exportar (opcional).
Flujos alternos	
FA-01	N/A
Requerimientos especiales	
Auditoría de vistas.	
Poscondiciones	

Información visible para gestión.

Fuente: Elaboración propia.

Tabla 42: Caso de uso “Crear Horas Extras”

Prototipo funcional para la gestión del recurso humano en la empresa Nutrimax, ubicada en Curridabat	
Número de caso de uso: CU-HEX-Solicitar	Nombre del caso de uso: Crear Horas Extra
Fecha elaboración:	25/09/2025
Descripción del caso de uso:	El empleado solicita horas extra indicando fecha, horas y motivo; queda pendiente de aprobación
Autor del caso de uso:	Marvin Rodrigo Núñez Sandoval
Actores relacionados:	Empleado (actor primario) Jefatura (secundario) Recursos Humanos (secundario)
Precondiciones:	Empleado activo. Políticas de tope y factor configuradas.
Flujo básico del caso de uso	
<ol style="list-style-type: none"> 1. Empleado abre Acciones, Solicitar Horas Extras. 2. Indica fecha, cantidad y motivo. 3. Guardar. 4. Sistema registra Pendiente y notifica a jefatura. 	
Subflujos	
SF-1	Cancelar antes de guardar.
Flujos alternos	
FA-01 Horas solicitadas durante horario laboras	Sistema bloquea solicitud.
FA-02 Fecha inválida	Sistema bloquea solicitud.

Requerimientos especiales
Auditoría.
Poscondiciones
Solicitud Pendiente.

Fuente: Elaboración propia.

Tabla 43: Caso de uso “Aprobar Horas Extras”

Prototipo funcional para la gestión del recurso humano en la empresa Nutrimax, ubicada en Curridabat	
Número de caso de uso: CU-HEX-Aprobar	Nombre del caso de uso: Aprobar Horas Extras
Fecha elaboración:	25/09/2025
Descripción del caso de uso:	La jefatura aprueba la solicitud; el sistema calculará el monto al integrarse con planilla.
Autor del caso de uso:	Marvin Rodrigo Núñez Sandoval
Actores relacionados:	Jefatura (actor primario) Empleado (secundario)
Precondiciones:	Solicitud Pendiente.
Flujo básico del caso de uso	
<ol style="list-style-type: none"> 1. Jefatura revisa detalle. 2. Presiona Aprobar. 3. Sistema cambia a Aprobada y notifica. 	
Subflujos	
SF-1	Comentario (opcional).
Flujos alternos	
FA-01 Inconsistencias	Fecha/horas.
Requerimientos especiales	
Auditoría.	

Poscondiciones
Solicitud Aprobada (lista para planilla).

Fuente: Elaboración propia.

Tabla 44: Caso de uso “Rechazar Horas Extras”

Prototipo funcional para la gestión del recurso humano en la empresa Nutrimax, ubicada en Curridabat	
Número de caso de uso: CU-HEX-Rechazar	Nombre del caso de uso: Rechazar Horas Extras
Fecha elaboración:	25/09/2025
Descripción del caso de uso:	La jefatura rechaza la solicitud con motivo.
Autor del caso de uso:	Marvin Rodrigo Núñez Sandoval
Actores relacionados:	Jefatura (actor primario) Empleado (secundario)
Precondiciones:	Solicitud Pendiente.
Flujo básico del caso de uso	
1. Jefatura presiona Rechazar. 2. Sistema cambia a Rechazada y notifica.	
Subflujos	
SF-01	N/A
Flujos alternos	
FA-01	N/A
Requerimientos especiales	
Auditoría.	
Poscondiciones	

Solicitud Rechazada.

Fuente: Elaboración propia.

Tabla 45: Caso de uso “Leer Horas Extras”

Prototipo funcional para la gestión del recurso humano en la empresa Nutrimax, ubicada en Curridabat	
Número de caso de uso: CU-HEX-Listar	Nombre del caso de uso: Leer/consultar horas extra
Fecha elaboración:	25/09/2025
Descripción del caso de uso:	Recursos Humanos lista solicitudes y estados.
Autor del caso de uso:	Marvin Rodrigo Núñez Sandoval
Actores relacionados:	Recursos Humanos (actor primario)
Precondiciones:	Recursos Humanos autenticado.
Flujo básico del caso de uso	
<ol style="list-style-type: none"> 1. El usuario de Recursos Humanos abre Horas Extra. 2. Visualiza monto estimado según factor (si aplica). 	
Subflujos	
SF-1	Exportar (opcional).
Flujos alternos	
FA-01	N/A
Requerimientos especiales	
(Opcional) Auditoría de vistas.	
Poscondiciones	
Información visible para gestión.	

Fuente: Elaboración propia.

Tabla 46: Caso de uso “Crear Planilla”

Prototipo funcional para la gestión del recurso humano en la empresa Nutrimax, ubicada en Curridabat	
Número de caso de uso: CU-PLA-Calcular	Nombre del caso de uso: Calcular/Crear planilla quincenal
Fecha elaboración:	25/09/2025
Descripción del caso de uso:	El usuario de Recursos Humanos genera una nueva planilla en estado “Calculada”, consolidando horas, horas extra aprobadas, ausencias/vacaciones y deducciones automáticas (incluida CCSS).
Autor del caso de uso:	Marvin Rodrigo Núñez Sandoval
Actores relacionados:	Recursos Humanos (actor primario) Sistema de marcas (secundario)
Precondiciones:	Parámetros y tasas vigentes. Marcas y solicitudes aprobadas en el rango.
Flujo básico del caso de uso	
<ol style="list-style-type: none"> 1. El usuario de Recursos Humanos abre Generar, Planilla. 2. Define Fecha inicio/fin. 3. Sistema valida no traslape con planillas cerradas. 4. Sistema consolida horas, integra horas extra aprobadas y ausencias. 5. Calcula Bruto, Deducciones automáticas y Neto. 6. Muestra resumen (por empleado y totales). 7. Guarda. 	
Subflujos	
SF-1	Cancelar cálculo antes de guardar.
Flujos alternos	
FA-01 Datos errneos	Traslape de rangos, parámetros faltantes, inconsistencias de marcas.

Requerimientos especiales
Auditoría (Antes/Después).
Poscondiciones
Planilla en estado Calculada.

Fuente: Elaboración propia.

Tabla 47: Caso de uso “Generar Deducciones”

Prototipo funcional para la gestión del recurso humano en la empresa Nutrimax, ubicada en Curridabat	
Número de caso de uso: CU-DED-Automáticas	Nombre del caso de uso: Generar deducciones automáticas
Fecha elaboración:	25/09/2025
Descripción del caso de uso:	Recursos Humanos ejecuta la generación de deducciones automáticas (CCSS)
Autor del caso de uso:	Marvin Rodrigo Núñez Sandoval
Actores relacionados:	Recursos Humanos (actor primario)
Precondiciones:	Parámetros/porcentajes vigentes.
Flujo básico del caso de uso	
<ol style="list-style-type: none"> 1. Recursos Humanos ejecuta Generar deducciones. 2. Sistema calcula por empleado según reglas. 3. Muestra resumen y aplica a la planilla activa (o precálculo). 	
Subflujos	
SF-1	Vista previa antes de aplicar.
Flujos alternos	
FA-01	N/A
Requerimientos especiales	
Auditoría de proceso. Trazabilidad por empleado.	
Poscondiciones	

Deducciones registradas para el rango

Fuente: Elaboración propia.

Tabla 48: Caso de uso “Previsualizar Aguinaldo”

Prototipo funcional para la gestión del recurso humano en la empresa Nutrimax, ubicada en Curridabat	
Número de caso de uso: CU-AGU-Previsualizar	Nombre del caso de uso: Previsualizar Aguinaldo
Fecha elaboración:	25/09/2025
Descripción del caso de uso:	Recursos Humanos obtiene una vista previa del cálculo de aguinaldo por empleado (sin afectar datos definitivos) para revisar montos antes de la generación masiva.
Autor del caso de uso:	Marvin Rodrigo Núñez Sandoval
Actores relacionados:	Recursos Humanos (actor primario)
Precondiciones:	Parámetros de cálculo vigentes (período, reglas CR). Información de planillas del período de aguinaldo disponible.
Flujo básico del caso de uso	
<ol style="list-style-type: none"> 1. El usuario de Recursos Humanos abre Generar, Aguinaldo → Previsualizar. 2. El sistema calcula montos estimados por empleado. 3. El sistema muestra lista previsualizada (empleado, base de cálculo, deducciones si aplica, total). 	
Subflujos	
SF-1	Exportar previsualización (Excel/PDF) para revisión externa.
Flujos alternos	
FA-01 Falta de datos del período	El sistema informa y sugiere revisar planillas del año.
FA-02 Parámetros faltantes:	El sistema indica qué parámetro configurar.

Requerimientos especiales
Auditoría de consulta (opcional).
Poscondiciones
Previsualización generada.

Fuente: Elaboración propia.

Tabla 49: Caso de uso “Generar Aguinaldo”

Prototipo funcional para la gestión del recurso humano en la empresa Nutrimax, ubicada en Curridabat	
Número de caso de uso: CU-AGU-GenerarMasivo	Nombre del caso de uso: Generar aguinaldo (Automático – Generar masivo)
Fecha elaboración:	25/09/2025
Descripción del caso de uso:	Recursos Humanos ejecuta el botón automático de generación masiva de aguinaldo que calcula y persiste los montos por empleado para el período seleccionado.
Autor del caso de uso:	Marvin Rodrigo Núñez Sandoval
Actores relacionados:	Recursos Humanos (actor primario)
Precondiciones:	Previsualización revisada (recomendado). Parámetros vigentes y datos de planillas consolidados.
Flujo básico del caso de uso	
<ol style="list-style-type: none"> 1. El usuario de Recursos Humanos abre Generar, Aguinaldo → Generar masivo. 2. Selecciona período y confirma Generar. 3. El sistema calcula montos por empleado (base anual/12, reglas CR). 4. El sistema persiste resultados y muestra resumen (totales, excepciones). 	
Subflujos	
SF-1	Reintentar casos con error (empleados con datos incompletos).
Flujos alternos	
FA-01 Empleado con datos faltantes	Listado de excepciones (no se genera para ellos).
FA-02 Parámetros faltantes	Bloqueo con mensaje.
Requerimientos especiales	
AuditLog del proceso (antes/después, período, usuario).	

Poscondiciones
Aguinaldo generado y almacenado para el período.

Fuente: Elaboración propia.

Tabla 50: Caso de uso “Crear Liquidación”

Prototipo funcional para la gestión del recurso humano en la empresa Nutrimax, ubicada en Curridabat	
Número de caso de uso: CU-LIQ-NuevaCalcular	Nombre del caso de uso: Crear Liquidación (seleccionar empleado y calcular)
Fecha elaboración:	25/09/2025
Descripción del caso de uso:	Recursos Humanos crea una nueva liquidación: selecciona empleado, fecha de salida, tipo de terminación y si hubo preaviso “en tiempo”, y calcula los rubros.
Autor del caso de uso:	Marvin Rodrigo Núñez Sandoval
Actores relacionados:	Recursos Humanos (actor primario)
Precondiciones:	Empleado activo. Reglas CR vigentes (preaviso, cesantía, vacaciones pendientes, aguinaldo proporcional).
Flujo básico del caso de uso	
<ol style="list-style-type: none"> 1. El usuario de Recursos Humanos abre Generar, Liquidaciones → Nueva liquidación. 2. Selecciona Empleado. 3. Define Fecha de salida, Tipo de terminación y marca “Dado en tiempo” si aplica. 4. Presiona Calcular. 5. El sistema calcula rubros (aguinaldo proporcional, vacaciones pendientes, preaviso/cesantía según tipo) y muestra detalle. 	
Subflujos	
SF-1	Editar parámetros específicos antes de confirmar.
Flujos alternos	
FA-01 Datos inválidos/incompletos	El sistema destaca errores y detiene el cálculo.
Requerimientos especiales	
Auditoría del cálculo (snapshot).	

Poscondiciones
Liquidación calculada visible en la lista.

Fuente: Elaboración propia.

Tabla 51: Caso de uso “Leer Liquidaciones”

Prototipo funcional para la gestión del recurso humano en la empresa Nutrimax, ubicada en Curridabat	
Número de caso de uso: CU-LIQ-Listar	Nombre del caso de uso: Listar liquidaciones
Fecha elaboración:	25/09/2025
Descripción del caso de uso:	Recursos Humanos visualiza la lista de liquidaciones por empleado, fecha y estado, incluyendo acceso a detalles y descarga de soporte.
Autor del caso de uso:	Marvin Rodrigo Núñez Sandoval
Actores relacionados:	Recursos Humanos (actor primario)
Precondiciones:	Liquidaciones calculadas o cerradas en el sistema.
Flujo básico del caso de uso	
<ol style="list-style-type: none"> 1. El usuario de Recursos Humanos abre Liquidaciones. 2. El sistema muestra lista con totales y enlaces a detalle. 	
Subflujos	
SF-1	N/A
Flujos alternos	
FA-01	N/A
Requerimientos especiales	
(Opcional) Auditoría de vistas.	
Poscondiciones	

Listado visible para gestión.

Fuente: Elaboración propia.

Tabla 52: Caso de uso “Consultar Empleados”

Prototipo funcional para la gestión del recurso humano en la empresa Nutrimax, ubicada en Curridabat	
Número de caso de uso: CU-CON-Empleados	Nombre del caso de uso: Consultar Empleados (búsqueda avanzada)
Fecha elaboración:	25/09/2025
Descripción del caso de uso:	Recursos Humanos consulta empleados con filtros avanzados (estado, departamento, puesto, fecha ingreso).
Autor del caso de uso:	Marvin Rodrigo Núñez Sandoval
Actores relacionados:	Recursos Humanos (actor primario)
Precondiciones:	Recursos Humanos autenticado.
Flujo básico del caso de uso	
<ol style="list-style-type: none"> 1. Abrir Consultas, Empleados. 2. Aplicar filtros y Buscar. 3. Ver resultado (tabla) y acceder a detalle. 	
Subflujos	
SF-1	Exportar (Excel/PDF).
Flujos alternos	
FA-01	N/A
Requerimientos especiales	
(Opcional) Auditoría de vistas.	
Poscondiciones	
Datos mostrados para análisis rápido.	

Fuente: Elaboración propia.

Tabla 53: Caso de uso “Consultar Vacaciones”

Prototipo funcional para la gestión del recurso humano en la empresa Nutrimax, ubicada en Curridabat	
Número de caso de uso: CU-CON-Vacaciones	Nombre del caso de uso: Consultar Vacaciones (saldos y solicitudes)
Fecha elaboración:	25/09/2025
Descripción del caso de uso:	Recursos Humanos consulta saldos, solicitudes por estado y rangos de fechas.
Autor del caso de uso:	Marvin Rodrigo Núñez Sandoval
Actores relacionados:	Recursos Humanos (actor primario)
Precondiciones:	Datos de vacaciones disponibles.
Flujo básico del caso de uso	
<ol style="list-style-type: none"> 1. Abrir Consultas, Vacaciones. 2. Filtros: empleado, estado, rango. 3. Ver resultados con saldo, días tomados y pendientes. 	
Subflujos	
SF-1	Exportar (Excel/PDF).
Flujos alternos	
FA-01	N/A
Requerimientos especiales	
Auditoría de consultas (opcional).	
Poscondiciones	
Información visible para seguimiento.	

Fuente: Elaboración propia.

Tabla 54: Caso de uso “Consultar Permisos”

Prototipo funcional para la gestión del recurso humano en la empresa Nutrimax, ubicada en Curridabat	
Número de caso de uso: CU-CON-Permisos	Nombre del caso de uso: Consultar permisos
Fecha elaboración:	25/09/2025
Descripción del caso de uso:	Recursos Humanos consulta permisos por tipo, estado y fechas.
Autor del caso de uso:	Marvin Rodrigo Núñez Sandoval
Actores relacionados:	Recursos Humanos (actor primario)
Precondiciones:	Datos de permisos registrados.
Flujo básico del caso de uso	
<ol style="list-style-type: none"> 1. Abrir Consultas, Permisos. 2. Aplicar filtros y Buscar. 3. Ver resultados y detalles. 	
Subflujos	
SF-1	Exportar (opcional).
Flujos alternos	
FA-01	N/A
Requerimientos especiales	
Poscondiciones	
Información disponible para gestión.	

Fuente: Elaboración propia.

Tabla 55: Caso de uso “Consultar Deducciones”

Prototipo funcional para la gestión del recurso humano en la empresa Nutrimax, ubicada en Curridabat	
Número de caso de uso: CU-CON-Deducciones	Nombre del caso de uso: Consultar deducciones
Fecha elaboración:	25/09/2025
Descripción del caso de uso:	Recursos Humanos consulta deducciones por empleado, tipo (automática/manual) y período.
Autor del caso de uso:	Marvin Rodrigo Núñez Sandoval
Actores relacionados:	Recursos Humanos (actor primario)
Precondiciones:	Deducciones generadas/registradas.
Flujo básico del caso de uso	
<ol style="list-style-type: none"> 1. Abrir Consultas, Deducciones. 2. Filtros: empleado, tipo, rango. 3. Ver resultados y totales. 	
Subflujos	
SF-1	Exportar lista (Excel/PDF) — opcional.
Flujos alternos	
FA-01	N/A
Requerimientos especiales	
Poscondiciones	
Información visible para revisión	

Fuente: Elaboración propia.

Tabla 56: Caso de uso “Reporte de Planilla”

Prototipo funcional para la gestión del recurso humano en la empresa Nutrimax, ubicada en Curridabat	
Número de caso de uso: CU-REP-ResumenPlanilla	Nombre del caso de uso: Reporte – Resumen de planilla
Fecha elaboración:	25/09/2025
Descripción del caso de uso:	Recursos Humanos genera el reporte de totales de planilla por quincena (bruto, deducciones, neto).
Autor del caso de uso:	Marvin Rodrigo Núñez Sandoval
Actores relacionados:	Recursos Humanos (actor primario)
Precondiciones:	Planillas calculadas/cerradas.
Flujo básico del caso de uso	
<ol style="list-style-type: none"> 1. Abrir Reportes, Planilla Mensual 2. Seleccionar rango/quincena. 3. Generar vista previa. 4. Exportar a PDF/Excel (opcional). 	
Subflujos	
SF-1	Filtrar por departamento/empleador (opcional).
Flujos alternos	
FA-01	N/A
Requerimientos especiales	
Consistencia con totales de planilla.	
Poscondiciones	
Reporte generado/exportado.	

Fuente: Elaboración propia.

Tabla 57: Caso de uso “Reporte de Deducciones”

Prototipo funcional para la gestión del recurso humano en la empresa Nutrimax, ubicada en Curridabat	
Número de caso de uso: CU-REP-Deducciones	Nombre del caso de uso: Reporte – Deducciones por período
Fecha elaboración:	25/09/2025
Descripción del caso de uso:	Recursos Humanos obtiene detalle de deducciones por tipo y empleado en un rango.
Autor del caso de uso:	Marvin Rodrigo Núñez Sandoval
Actores relacionados:	Recursos Humanos (actor primario)
Precondiciones:	Deducciones registradas.
Flujo básico del caso de uso	
<ol style="list-style-type: none"> 1. Abrir Reportes, Deducciones/Tipo 2. Filtros: tipo, empleado, rango. 3. Generar. 4. Exportar (opcional). 	
Subflujos	
SF-1	N/A
Flujos alternos	
FA-01	N/A
Requerimientos especiales	
Poscondiciones	
Reporte disponible.	

Fuente: Elaboración propia.

Tabla 58: Caso de uso “Reporte de Vacaciones”

Prototipo funcional para la gestión del recurso humano en la empresa Nutrimax, ubicada en Curridabat	
Número de caso de uso: CU-REP-Vacaciones	Nombre del caso de uso: Reporte – Vacaciones (tomadas y saldo)
Fecha elaboración:	25/09/2025
Descripción del caso de uso:	Recursos Humanos genera reporte de días tomados y saldo por empleado.
Autor del caso de uso:	Marvin Rodrigo Núñez Sandoval
Actores relacionados:	Recursos Humanos (actor primario)
Precondiciones:	Datos de vacaciones.
Flujo básico del caso de uso	
<ol style="list-style-type: none"> 1. Abrir Reportes, Vacaciones / Saldo. 2. Filtro por empleado/departamento y rango. 3. Generar y Exportar (opcional). 	
Subflujos	
SF-1	N/A
Flujos alternos	
FA-01	N/A
Requerimientos especiales	
Poscondiciones	
Reporte disponible.	

Fuente: Elaboración propia.

Tabla 59: Caso de uso “Reporte de Horas Extras”

Prototipo funcional para la gestión del recurso humano en la empresa Nutrimax, ubicada en Curridabat	
Número de caso de uso: CU-REP-HorasExtra	Nombre del caso de uso: Reporte – Horas extra aprobadas
Fecha elaboración:	25/09/2025
Descripción del caso de uso:	Recursos Humanos genera reporte de horas extra aprobadas y montos asociados por período.
Autor del caso de uso:	Marvin Rodrigo Núñez Sandoval
Actores relacionados:	Recursos Humanos (actor primario)
Precondiciones:	Solicitudes aprobadas.
Flujo básico del caso de uso	
<ol style="list-style-type: none"> 1. Abrir Reportes, Horas extra / Depto. 2. Filtros: rango, empleado, estado (aprobadas). 3. Generar y Exportar. 	
Subflujos	
SF-1	N/A
Flujos alternos	
FA-01	N/A
Requerimientos especiales	
Poscondiciones	
Reporte disponible.	

Fuente: Elaboración propia.

Tabla 60: Caso de uso “Crear Plantilla de Evaluación”

Prototipo funcional para la gestión del recurso humano en la empresa Nutrimax, ubicada en Curridabat	
Número de caso de uso: CU-EVA-CrearPlantilla	Nombre del caso de uso: Crear Plantilla de Evaluación (criterios y ponderaciones)
Fecha elaboración:	25/09/2025
Descripción del caso de uso:	Recursos Humanos define una plantilla con criterios y ponderaciones para evaluar empleados.
Autor del caso de uso:	Marvin Rodrigo Núñez Sandoval
Actores relacionados:	Recursos Humanos (actor primario)
Precondiciones:	Catálogo de criterios disponible (o se crean en esta acción).
Flujo básico del caso de uso	
<ol style="list-style-type: none"> 1. El usuario de Recursos Humanos abre Evaluaciones → Plantillas → Agregar Plantilla. 2. Define nombre, periodicidad, criterios y ponderaciones. 3. Guardar. 4. El sistema valida sumatoria de ponderaciones (=100 %). 5. El sistema crea la plantilla. 	
Subflujos	
SF-1	N/A
Flujos alternos	
FA-01	N/A
Requerimientos especiales	
Auditoría de alta.	
Poscondiciones	
Plantilla lista para programar evaluaciones.	

Fuente: Elaboración propia.

Tabla 61: Caso de uso “Iniciar Sesión”

Prototipo funcional para la gestión del recurso humano en la empresa Nutrimax, ubicada en Curridabat	
Número de caso de uso: CU-IniciarSesión	Nombre del caso de uso: Iniciar Sesión
Fecha elaboración:	25/09/2025
Descripción del caso de uso:	El usuario ingresa sus credenciales para autenticarse en el sistema; el sistema valida identidad, estado y políticas de seguridad, crea la sesión y registra la bitácora de acceso para permitir el uso de los módulos según su rol.
Autor del caso de uso:	Marvin Rodrigo Núñez Sandoval
Actores relacionados:	Actor primario: Administrador, Recursos Humanos, Jefatura y Usuario
Precondiciones:	El usuario debe estar registrado en la base de datos, con cuenta activa y con un rol asignado.
Flujo básico del caso de uso	
<ol style="list-style-type: none"> 1. El usuario accede a la pantalla “Iniciar Sesión”. 2. El usuario ingresa sus credenciales (nombre de usuario) y contraseña. 3. El sistema valida formato de los campos. 4. El sistema compara las credenciales con los registros almacenados mediante verificación de hash de contraseña. 5. El sistema crea la sesión del usuario, emite el cookie de autenticación, establece el tiempo de expiración por inactividad y carga el menú según los permisos del rol. 6. El sistema redirige al usuario a la página de inicio del sistema. 	
Subflujos	
SF-1	N/A
Flujos alternos	
FA-01 Credenciales inválidas	Si el usuario o la contraseña no coinciden, el sistema muestra mensaje genérico “Credenciales incorrectas” y no crea sesión.

FA-02 Usuario inactivo o sin rol	Si la cuenta está desactivada o sin rol asignado, el sistema deniega el acceso e informa que contacte a Recursos Humanos/Administrador.
Requerimientos especiales	
Las contraseñas se almacenan con un hash seguro y nunca se guardan en texto claro.	
Poscondiciones	
Sesión creada y vigente con permisos cargados según rol.	

Fuente: Elaboración propia.

Diseño

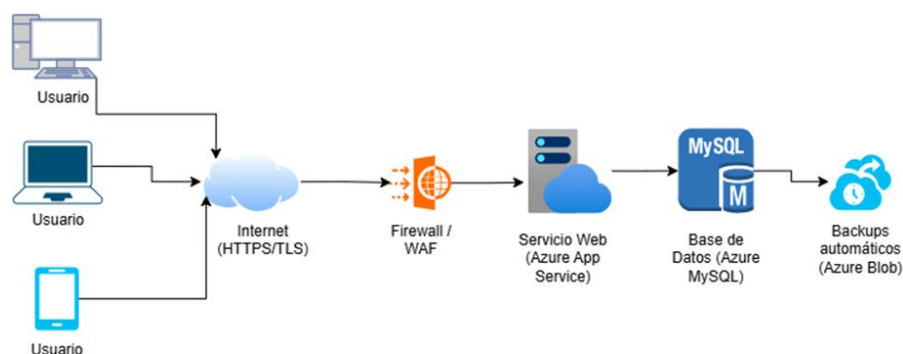
En esta sección se describe la forma en que el sistema fue desarrollado desde el punto de vista arquitectónico, lógico y visual, comprendiendo aspectos como la arquitectura del sistema, arquitectura del *software*, diseño de entradas y salidas, diseño físico de la base de datos, diagramas de procesos y los diagramas UML correspondientes.

El propósito de esta etapa es garantizar que el sistema propuesto posea una base sólida, escalable y mantenible, además de una interfaz clara y funcional para los usuarios. A través de los distintos diagramas y representaciones gráficas, se busca facilitar la comprensión del flujo interno del sistema, las relaciones entre sus módulos y la manera en que los datos son procesados, almacenados y presentados al usuario final.

Arquitectura del sistema

La solución se implementa sobre la plataforma Microsoft Azure y sigue un modelo cliente-servidor con despliegue en la nube. El cliente (navegador web) consume una aplicación web desarrollada en ASP.NET Core que ejecuta la lógica de negocio y expone interfaces para la gestión de recursos humanos. La capa de persistencia maneja Azure Database for MySQL (Flexible Server) para almacenar la información y Azure Blob Storage para los archivos y documentos (adjuntos, imágenes, respaldos). La arquitectura considera mecanismos para seguridad, protección de la entrada (WAF), escalado automático y operación.

Figura 13: Arquitectura del sistema

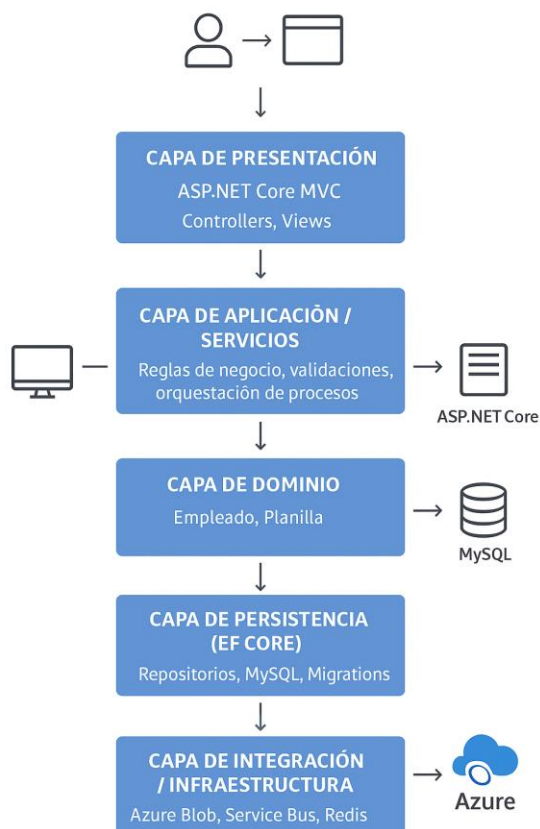


Fuente: Elaboración propia.

Arquitectura de software

La arquitectura del *software* del prototipo se basa en un enfoque en capas combinado con el patrón Modelo–Vista–Controlador (MVC) propio de ASP.NET Core. Esta arquitectura busca separar responsabilidades, facilitar las pruebas unitarias, mejorar la mantenibilidad y permitir la evolución del sistema. La solución se organiza en distintas capas lógicas que colaboran entre sí. En la capa de presentación se gestiona la interacción con el usuario mediante controladores, vistas y modelos de presentación (Clases), que son las encargadas de validar entradas básicas y devolver respuestas HTML o JSON. Los controladores como EmpleadosController, PlanillaController o VacacionesController interactúan directamente con los servicios de aplicación. La capa de servicios encapsula la lógica de negocio del sistema, utilizando reglas específicas, coordinando el acceso a datos y gestionando transacciones. Interfaces como IPlanillaService o IVacacionesService representan esta capa, que actúa como puente entre la presentación y el dominio. El modelo de dominio define las entidades centrales del negocio, como Empleado, Planilla, Vacaciones o Usuario, manteniendo y evitando la mezcla con lógica de presentación o persistencia. Para el acceso a datos, se utiliza Entity Framework Core con proveedor MySQL, permitiendo mapear entidades a tablas, realizar consultas optimizadas y gestionar migraciones. Finalmente, la infraestructura externa se compone mediante servicios como almacenamiento de archivos (Blob Storage), envío de correos (SMTP), con soporte para entornos en la nube como Azure. Esta arquitectura permite escalar el sistema, mantenerlo limpio y facilitar la colaboración entre desarrolladores.

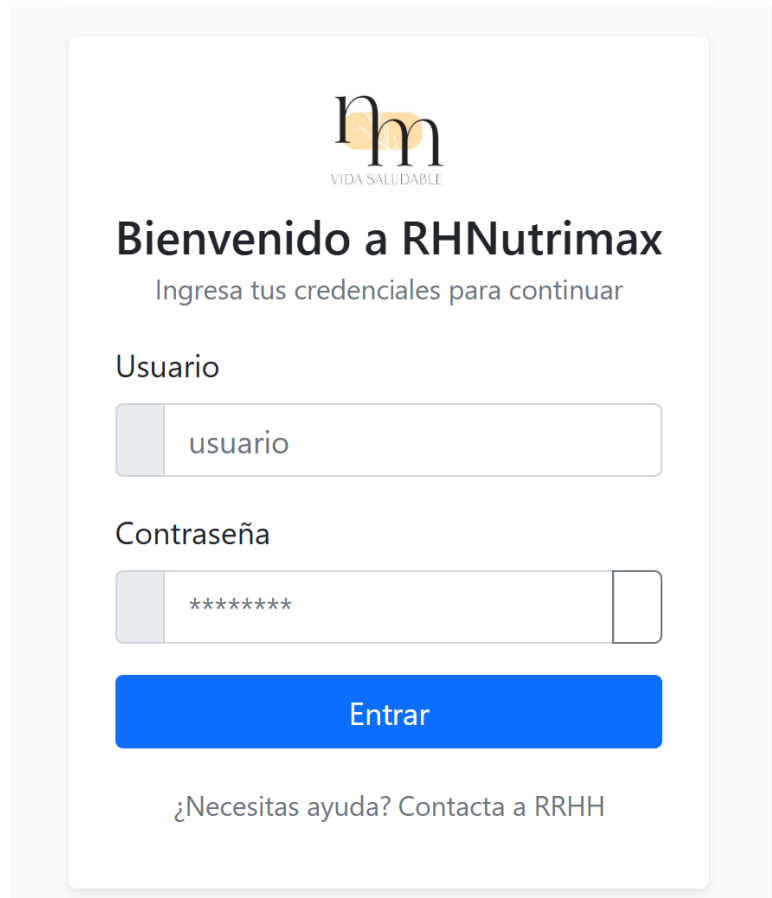
Figura 14: Arquitecta de software



Fuente: Elaboración propia.

Diseño de entradas

El diseño de entradas corresponde a las interfaces mediante las cuales los usuarios introducen datos al sistema. Estas pantallas deben estar orientadas a facilitar la captura correcta de datos, minimizar errores y validar la información antes de su almacenamiento. Se prioriza la usabilidad, accesibilidad, claridad en las etiquetas, y validaciones tanto del lado cliente como servidor.

Figura 15: Inicio de sesión

nm
VIDA SALUDABLE

Bienvenido a RHNutrimax

Ingresa tus credenciales para continuar

Usuario

Contraseña

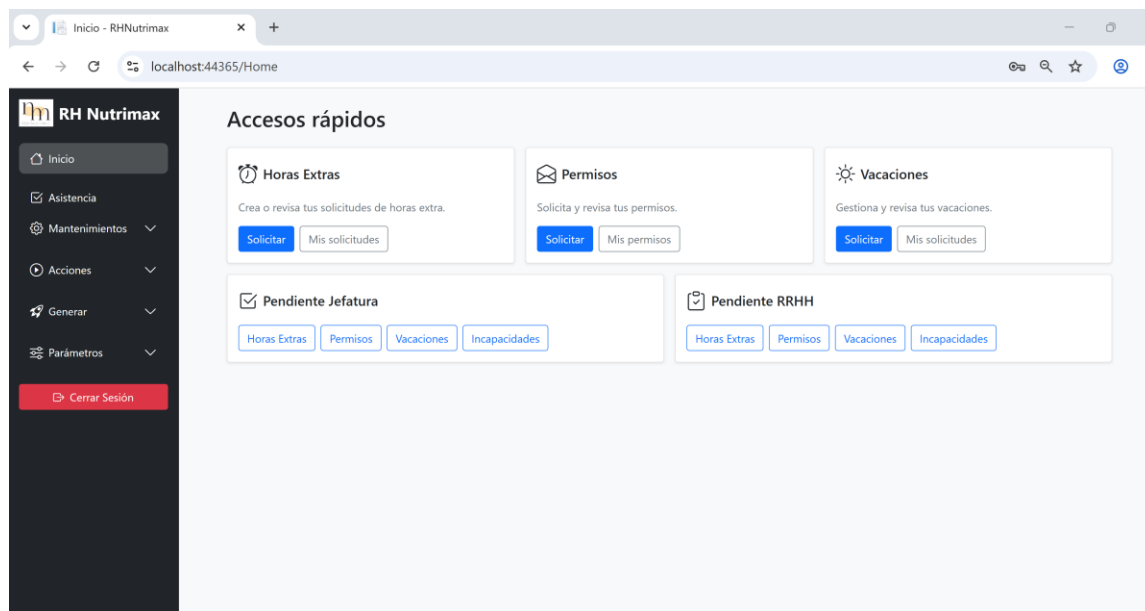
Entrar

[¿Necesitas ayuda? Contacta a RRHH](#)

Fuente: Elaboración propia.

Esta pantalla permite el acceso autenticado al sistema mediante las credenciales del usuario (usuario y contraseña). Es el punto de entrada para todos los módulos del sistema de Recursos Humanos.

Figura 16: Menú principal



Fuente: Elaboración propia.

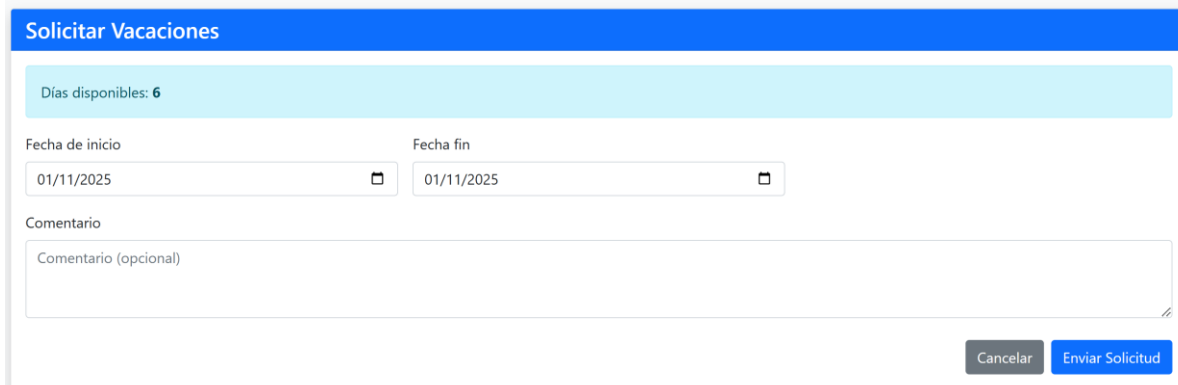
El *dashboard* o menú actúa como la página principal del sistema, brindando una vista general y acceso rápido a los módulos más utilizados (Empleados, Vacaciones, Planilla, Deducciones, etc.) Permite también ingresar pequeños filtros o parámetros para buscar información de forma ágil.

Figura 17: Agregar empleado

Fuente: Elaboración propia.

Esta pantalla permite registrar un nuevo empleado en el sistema mediante un formulario estructurado por secciones. Representa una de las principales entradas de datos maestros del sistema.

Figura 18: Solicitar vacaciones



Fuente: Elaboración propia.

Permite al empleado registrar una solicitud de vacaciones, aplicando automáticamente las reglas establecidas (mínimo un año de trabajo y máximo de 12 días hábiles por año). Los supervisores podrán, posteriormente, aprobar o rechazar las solicitudes.

Figura 19: Planilla



Empleado	Fecha inicio	Fecha fin	Monto total	Acciones
Maria Brenes sanchez	2025-11-16	2025-11-30	€210 000,00	Detalles
Rolando Fonseca Gutierrez	2025-11-16	2025-11-30	€400 000,00	Detalles
Fabian Hernandez	2025-11-16	2025-11-30	€257 000,00	Detalles
Roberto Navarro Salas	2025-11-16	2025-11-30	€400 000,00	Detalles
Fernanda Mora García	2025-11-16	2025-11-30	€260 000,00	Detalles
Ericka Villarreal Morales	2025-11-16	2025-11-30	€750 000,00	Detalles
Fanny Sandoval Sánchez	2025-11-16	2025-11-30	€300 000,00	Detalles

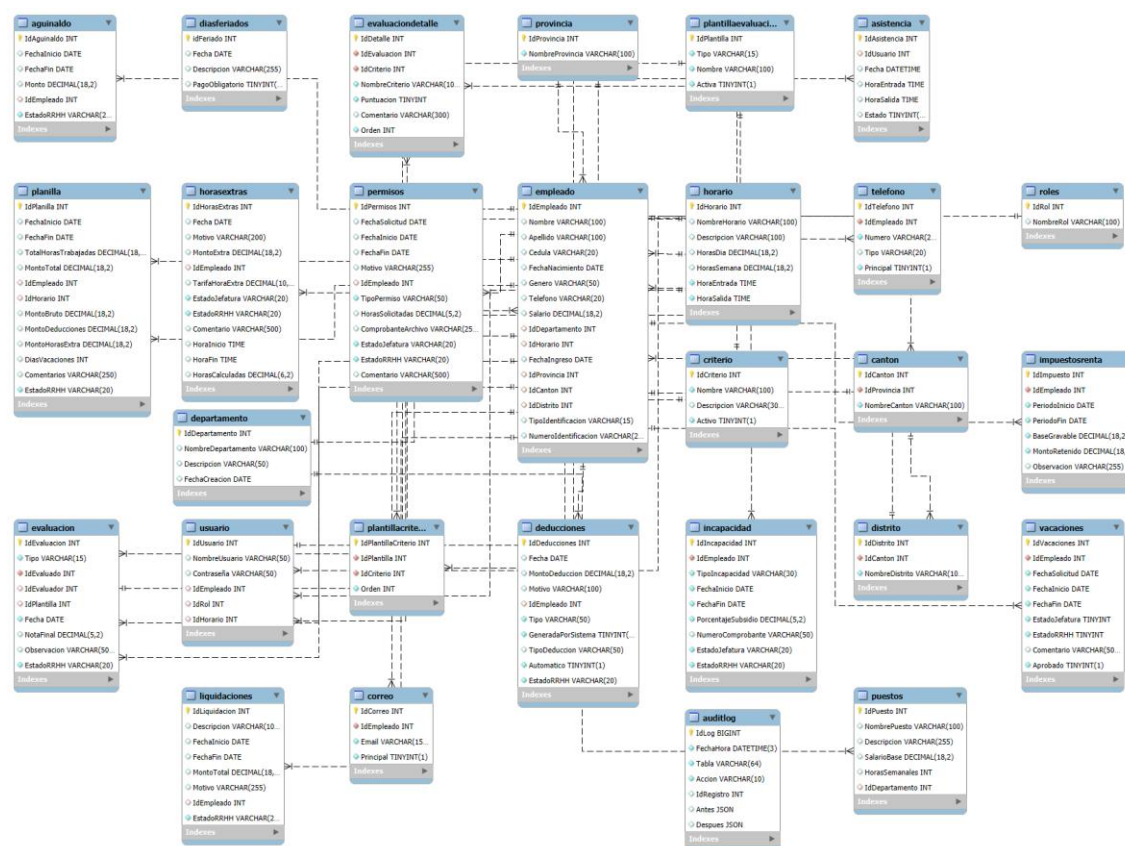
Fuente: Elaboración propia.

Esta pantalla permite al usuario administrativo generar y visualizar los detalles de las planillas de pago. Además, se ofrecen funciones para simular asistencias, eliminar registros existentes y generar nuevas planillas, lo que convierte esta vista en una de las principales entradas de parámetros del sistema.

Diseño de base de datos

El diseño de base de datos compone una de las etapas más importantes dentro del desarrollo del sistema, ya que define la estructura y organización de la información que será utilizada por la aplicación. En esta fase se identifican las entidades principales, sus atributos y las relaciones existentes entre ellas, garantizando la integridad, consistencia y disponibilidad de los datos.

Figura 20. Diseño de la base de datos



Fuente: Elaboración propia.

Diccionario de datos

Este apartado documenta la estructura de datos del sistema de Recursos Humanos, detallando todas las tablas que conforman la base de datos, incluyendo aquellas relacionadas con empleados, departamentos, horarios, planillas, vacaciones, deducciones y otros componentes clave. Cada tabla se presenta con sus respectivos campos, descripción funcional, tipo de dato y

restricciones aplicables, lo que permite comprender la lógica estructural que respalda las operaciones del sistema.

Tabla 62: Estructura tabla Aguinaldo

Campo	Tipo de dato	PK	FK	Not Null	Descripción
IdAguinaldo	INT(11)	X		X	Clave única de aguinaldo
FechaInicio	DATE				Fecha de inicio del cálculo del aguinaldo
FechaFin	DATE				Fecha final del cálculo del aguinaldo
Monto	DECIMAL(18,2)				Monto final del aguinaldo
IdEmpleado	INT(11)		X		Llave foránea del empleado
EstadoRRHH	VARCHAR(20)			X	Estado solicitud en recursos humanos

Fuente: Elaboración propia.

Tabla 63: Estructura tabla Asistencia

Campo	Tipo de dato	PK	FK	Not Null	Descripción
IdAsistencia	INT(11)	X		X	Clave única de registro de asistencia
IdUsuario	INT(11)		X		Relación con el usuario que marca asistencia
Fecha	DATETIME				Fecha y hora del registro de asistencia
HoraEntrada	TIME				Hora en que el usuario ingresó
HoraSalida	TIME				Hora en que el usuario salió
Estado	TINYINT(1) (Generada)				Estado del registro (campo calculado)

Fuente: Elaboración propia.

Tabla 64: Estructura tabla AuditLog

Campo	Tipo de dato	PK	FK	Not Null	Descripción
-------	--------------	----	----	----------	-------------

IdLog	BIGINT	X		X	Clave única del registro en la bitácora
FechaHora	DATETIME(3)			X	Fecha y hora del evento (con milisegundos)
Tabla	VARCHAR(64)			X	Nombre de la tabla donde ocurrió la acción
Accion	VARCHAR(10)			X	Acción realizada (INSERT, UPDATE, DELETE)
IdRegistro	INT				Identificador del registro afectado
Antes	JSON				Valores del registro antes de la acción
Despues	JSON				Valores del registro después de la acción

Fuente: Elaboración propia.

Tabla 65: Estructura tabla Canton

Campo	Tipo de dato	PK	FK	Not Null	Descripción
IdCanton	INT	X		X	Identificador único del cantón
IdProvincia	INT		X	X	Relación con la provincia
NombreCanton	VARCHAR(100)			X	Nombre del cantón

Fuente: Elaboración propia.

Tabla 66: Estructura tabla Correo

Campo	Tipo de dato	PK	FK	Not Null	Descripción
IdCorreo	INT	X		X	Identificador único del correo
IdEmpleado	INT		X	X	Relación con el empleado

Email	VARCHAR(150)			X	Dirección de correo electrónico
Principal	TINYINT(1)			X	Indica si es el correo principal (1 = sí, 0 = no)

Fuente: Elaboración propia.

Tabla 67: Estructura tabla Criterio

Campo	Tipo de dato	PK	FK	Not Null	Descripción
IdCriterio	INT	X		X	Identificador único del criterio
Nombre	VARCHAR(100)			X	Nombre del criterio (único)
Descripcion	VARCHAR(300)				Descripción del criterio
Activo	TINYINT(1)			X	Estado del criterio (1 = activo)

Fuente: Elaboración propia.

Tabla 68: Estructura tabla Deducciones

Campo	Tipo de dato	PK	FK	Not Null	Descripción
IdDeducciones	INT(11)	X		X	Clave única de deducciones
Fecha	DATE				Fecha de la deducción
MontoDeducccion	DECIMAL(18,2)				Monto de la deducción
Motivo	VARCHAR(100)				Motivo de la deducción
IdEmpleado	INT(11)		X		Llave foránea del empleado
IdEstado	INT(11)		X		Llave foránea del estado
Tipo	VARCHAR(50)			X	Tipo de deducción automática o generada por usuario
GeneradaPorSistema	TINYINT(1)			X	Validación de la deducción
TipoDeducccion	VARCHAR(50)				Validación de deducción legal o de la CCSS o voluntaria

Automatico	TINYINT(1)			X	Deducción generada automática
EstadoRRHH	VARCHAR(20)			X	Estado de la deducción en recursos humanos

Fuente: Elaboración propia.

Tabla 69: Estructura tabla Departamento

Campo	Tipo de dato	PK	FK	Not Null	Descripción
IdDepartamento	INT(11)	X			Clave única de registro departamentos
NombreDepartamento	VARCHAR(100)				Nombre del departamento
Descripcion	VARCHAR(50)				Descripción para el departamento
FechaCreacion	DATE				Fecha de la creación del departamento

Fuente: Elaboración propia.

Tabla 70: Estructura tabla DiasFeriados

Campo	Tipo de dato	PK	FK	Not Null	Descripción
IdFeriado	INT	X		X	Identificador único del feriado
Fecha	DATE				Fecha del feriado
Descripcion	VARCHAR(255)				Descripción del feriado
PagoObligatorio	TINYINT(1)				Indica si el feriado es de pago obligatorio

Fuente: Elaboración propia.

Tabla 71: Estructura tabla Distrito

Campo	Tipo de dato		PK	FK	Not Null	Descripción
IdDistrito	INT		X		X	Identificador único del distrito

IdCanton	INT			X	X	Relación con el cantón
NombreDistrito	VARCHAR(100)				X	Nombre del distrito

Fuente: Elaboración propia.

Tabla 72: Estructura tabla Empleado

Campo	Tipo de dato	PK	FK	Not Null	Descripción
IdEmpleado	INT(11)	X			Clave única de registro empleado
Nombre	VARCHAR(100)				Nombre del empleado
Apellido	VARCHAR(100)				Apellido del empleado
Cedula	VARCHAR(20)				Cédula del empleado
FechaNacimiento	DATE				Fecha de nacimiento del empleado
Genero	VARCHAR(50)				Género del empleado
Telefono	VARCHAR(20)				Teléfono del empleado
Salario	DECIMAL(18,2)				Salario bruto del empleado
IdDepartamento	INT		X		Llave foránea a departamento
IdHorario	INT		X		Llave foránea a horario
FechaIngreso	Date				Fecha de ingreso del empleado
IdProvincia	INT				Llave foránea a provincia
IdCanton	INT				Llave foránea a cantón
IdDistrito	INT				Llave foránea a distrito
TipoIdentificacion	VARCHAR(15)				Tipo de identificación
NumeroIdentificacion	VARCHAR(20)				Número de identificación

Fuente: Elaboración propia.

Tabla 73: Estructura tabla Evaluación

Campo	Tipo de dato	PK	FK	Not Null	Descripción
IdEvaluacion	INT(11)	X		X	Clave única de evaluación

Tipo	VARCHAR(15)			X	Tipo de la evaluación
IdEvaluado	INT		X	X	Llave foránea al usuario
IdEvaluador	INT		X		Llave foránea a Empleado
IdPlantilla	INT		X		Llave foránea de la plantilla que se va a usar
Fecha	DATE			X	Fecha de la creación de la evaluación
NotaFinal	DECIMAL(5,2)			X	Valor numérico del 1 al 10 para nota final
Observacion	VARCHAR(500)			X	Observación de la evaluación
EstadoRRHH	VARCHAR(20)				Estado por <i>default</i> en Pendiente de la solicitud

Fuente: Elaboración propia.

Tabla 74: Estructura tabla EvaluacionDetalle

Campo	Tipo de dato	PK	FK	Not Null	Descripción
IdDetalle	INT	X		X	Identificador del detalle de evaluación
IdEvaluacion	INT		X	X	Relación con la evaluación
IdCriterio	INT		X	X	Relación con el criterio
NombreCriterio	VARCHAR(100)			X	Nombre del criterio evaluado
Puntuacion	TINYINT			X	Puntuación asignada
Comentario	VARCHAR(300)				Comentario opcional
Orden	INT			X	Orden del criterio dentro de la evaluación

Fuente: Elaboración propia.

Tabla 75: Estructura tabla Horario

Campo	Tipo de dato	PK	FK	Not Null	Descripción
IdHorario	INT(11)	X			Clave única de registro horario

NombreHorario	VARCHAR(100)				Nombre del horario
Descripcion	VARCHAR(100)				Descripción para los horarios
HorasDia	Decimal(18,2)				Horas laboradas por día
HorasSemana	Decimal(18,2)				Horas laboradas por semana
HoraEntrada	Time(7)			X	Hora de entrada por día
HoraSalida	Time(7)			X	Hora de salida por día

Fuente: Elaboración propia.

Tabla 76: Estructura tabla HorasExtras

Campo	Tipo de dato	PK	FK	Not Null	Descripción
IdHorasExtras	INT(11)	X			Clave única de horas extra
Fecha	DATE				Fecha de solicitud de horas extra
Motivo	VARCHAR(200)				Motivo de la solicitud
MontoExtra	DECIMAL(18,2)				Monto extra total
TarifaHoraExtra	DECIMAL(10,2)				Tarifa de hora extra
EstadoJefatura	VARCHAR(20)			X	Estado de solicitud en jefatura
EstadoRRHH	VARCHAR(20)			X	Estado de solicitud en recursos humanos
Comentario	VARCHAR(500)			X	Comentarios de la solicitud
HoraInicio	TIME				Hora de inicio hora extra
HoraFin	TIME				Hora de fin hora extra
HorasCalculadas	DECIMAL(6,2)				Cantidad horas calculadas

Fuente: Elaboración propia.

Tabla 77: Estructura tabla ImpuestosRenta

Campo	Tipo de dato	PK	FK	Not Null	Descripción
IdImpuesto	INT	X		X	Identificador único del registro de impuesto

IdEmpleado	INT		X	X	Relación con el empleado
PeriodoInicio	DATE			X	Inicio del periodo
PeriodoFin	DATE			X	Fin del periodo
BaseGravable	DECIMAL(18,2)			X	Monto base para cálculo
MontoRetenido	DECIMAL(18,2)			X	Monto retenido por impuesto
Observacion	VARCHAR(255)				Observaciones adicionales

Fuente: Elaboración propia.

Tabla 78: Estructura tabla Incapacidad

Campo	Tipo de dato	PK	FK	Not Null	Descripción
IdIncapacidad	INT	X		X	Identificador de la incapacidad
IdEmpleado	INT		X	X	Relación con el empleado
TipoIncapacidad	VARCHAR(30)			X	Tipo de incapacidad
FechaInicio	DATE			X	Fecha de inicio
FechaFin	DATE			X	Fecha de fin
PorcentajeSubsidio	DECIMAL(5,2)			X	Porcentaje de subsidio aplicado
NumeroComprobante	VARCHAR(50)				Número de comprobante (opcional)
EstadoJefatura	VARCHAR(20)			X	Estado de aprobación en jefatura
EstadoRRHH	VARCHAR(20)			X	Estado de aprobación en Recursos Humanos

Fuente: Elaboración propia.

Tabla 79: Estructura tabla Liquidaciones

Campo	Tipo de dato	PK	FK	Not Null	Descripción
IdLiquidacion	INT(11)	X		X	Clave única de la liquidación

Descripcion	VARCHAR(100)				Descripción de la liquidación
FechaInicio	DATE				Fecha de inicio de la liquidación
FechaFin	DATE				Fecha final de la liquidación
MontoTotal	DECIMAL(18,2)				Monto total de la liquidación
Motivo	VARCHAR(255)				Motivo de la liquidación
IdEmpleado	INT(11)		X		Llave foránea del empleado
EstadoRRHH	VARCHAR(20)			X	Estado por <i>default</i> en Pendiente de la solicitud

Fuente: Elaboración propia.

Tabla 80: Estructura tabla Permisos

Campo	Tipo de dato	PK	FK	Not Null	Descripción
IdPermisos	INT(11)	X		X	Clave única de permisos
FechaSolicitud	DATE				Fecha de solicitud del permiso
FechaInicio	DATE				Fecha de inicio del permiso
FechaFin	DATE				Fecha final del permiso
Motivo	VARCHAR(255)				Motivo del permiso
IdEmpleado	INT(11)		X		Llave foránea del empleado
TipoPermiso	VARCHAR(50)			X	
HorasSolicitadas	DECIMAL(5,2)				Horas solicitadas para el permiso
ComprobanteArchivo	VARCHAR(255)				Comprobante del permiso
EstadoJefatura	VARCHAR(20)			X	Estado de la solicitud en jefatura
EstadoRRHH	VARCHAR(20)			X	Estado de la solicitud en recursos humanos
Comentario	VARCHAR(500)				Comentarios

Fuente: Elaboración propia.

Tabla 81: Estructura tabla Planilla

Campo	Tipo de dato	PK	FK	Not Null	Descripción
IdPlanilla	INT(11)	X		X	Clave única de la planilla
FechaInicio	DATE				Fecha de inicio de la planilla
FechaFin	DATE				Fecha final de la planilla
TotalHorasTrabajadas	DECIMAL(18,2)				Total de horas trabajadas calculadas
MontoTotal	DECIMAL(18,2)				Monto total designado a la planilla
IdEmpleado	INT(11)		X		Llave foránea del empleado
IdHorario	INT(11)		X		Llave foránea del horario
MontoBruto	DECIMAL(18,2)				Monto bruto de salario por empleado
MontoDeducciones	DECIMAL(18,2)				Monto deducciones del salario por empleado
MontoHorasExtra	DECIMAL(18,2)				Monto de horas extra hechas por el empleado
DiasVacaciones	INT				Días disponibles de vacaciones por empleado
Comentarios	VARCHAR(250)				Comentarios
EstadoRRHH	VARCHAR(20)				Estado por <i>default</i> en Pendiente de la solicitud

Fuente: Elaboración propia.

Tabla 82: Estructura tabla PlantillaCriterio

Campo	Tipo de dato	PK	FK	Not Null	Descripción
IdPlantillaCriterio	INT	X		X	Identificador único
IdPlantilla	INT		X	X	Relación con la plantilla

IdCriterio	INT		X	X	Relación con el criterio
Orden	INT			X	Orden del criterio en la plantilla

Fuente: Elaboración propia.

Tabla 83: Estructura tabla PlantillaEvaluacion

Campo	Tipo de dato	PK	FK	Not Null	Descripción
IdPlantilla	INT	X		X	Identificador único de la plantilla
Tipo	VARCHAR(15)			X	Tipo de evaluación
Nombre	VARCHAR(100)			X	Nombre de la plantilla
Activa	TINYINT(1)			X	Indica si la plantilla está activa

Fuente: Elaboración propia.

Tabla 84: Estructura tabla Provincia

Campo	Tipo de dato	PK	FK	Not Null	Descripción
IdProvincia	INT	X		X	Identificador único de la provincia
NombreProvincia	VARCHAR(100)			X	Nombre de la provincia (único)

Fuente: Elaboración propia.

Tabla 85: Estructura tabla Puestos

Campo	Tipo de dato	PK	FK	Not Null	Descripción
IdPuesto	INT	X		X	Identificador del puesto
NombrePuesto	VARCHAR(100)				Nombre del puesto

Descripcion	VARCHAR(255)				Descripción del puesto
SalarioBase	DECIMAL(18,2)				Salario base asignado
HorasSemanales	INT				Cantidad de horas semanales
IdDepartamento	INT		X		Relación con el departamento

Fuente: Elaboración propia.

Tabla 86: Estructura tabla Roles

Campo	Tipo de dato	PK	FK	Not Null	Descripción
IdRol	INT	X		X	Identificador único del rol
NombreRol	VARCHAR(100)				Nombre del rol

Fuente: Elaboración propia.

Tabla 87: Estructura tabla Teléfono

Campo	Tipo de dato	PK	FK	Not Null	Descripción
IdTelefono	INT	X		X	Identificador único del teléfono
IdEmpleado	INT		X	X	Relación con el empleado
Numero	VARCHAR(20)			X	Número de teléfono
Tipo	VARCHAR(20)				Tipo de teléfono (ej. celular, oficina)
Principal	TINYINT(1)			X	Indica si es el teléfono principal

Fuente: Elaboración propia.

Tabla 88: Estructura tabla Usuario

Campo	Tipo de dato	PK	FK	Not Null	Descripción
IdUsuario	INT(11)	X			Clave única del usuario
IdEmpleado	INT(11)		X		Llave foránea a empleado

NombreUsuario	VARCHAR(50)				Nombre del usuario
Contraseña	VARCHAR(50)				Contraseña del usuario
IdRol	INT(11)		X		Llave foránea a rol
IdHorario	INT(11)		X		Llave foránea a Horario

Fuente: Elaboración propia.

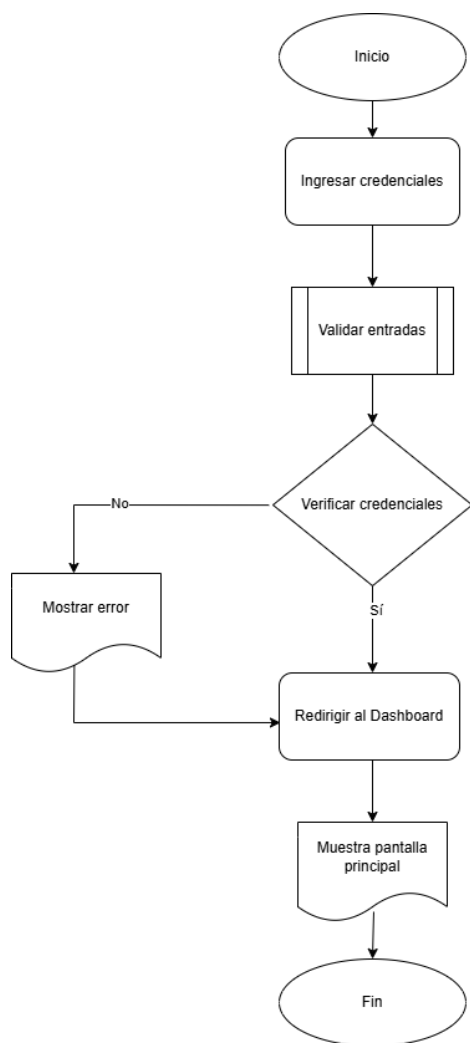
Tabla 89: Estructura tabla Vacaciones

Campo	Tipo de dato	PK	FK	Not Null	Descripción
IdVacaciones	INT(11)	X		X	Clave única de vacaciones
IdEmpleado	INT(11)		X	X	Llave foránea del empleado
FechaSolicitud	DATE			X	Fecha de solicitud de las vacaciones
FechaInicio	DATE			X	Fecha inicio de las vacaciones
FechaFin	DATE			X	Fecha final de las vacaciones
EstadoJefatura	TINYINT			X	Estado de jefatura
EstadoRRHH	TINYINT			X	Estado de recursos humanos
Comentario	VARCHAR(500)				Comentarios de las vacaciones
Aprobado	BIT			X	Estado de la solicitud

Fuente: Elaboración propia.

Diseño de procesos

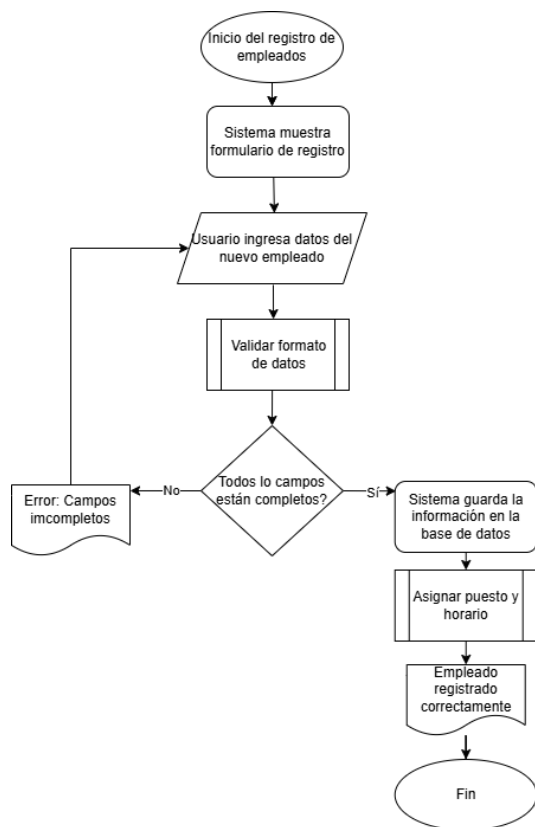
En esta sección se muestran los diagramas de flujo que representan los principales procesos del sistema de Recursos Humanos. Estos diagramas representan la secuencia de pasos, decisiones y acciones realizadas por el usuario y el sistema durante la ejecución de las funcionalidades principales. Los procesos modelados incluyen: inicio de sesión, registro de empleados, solicitud de vacaciones y cálculo de planilla.

Figura 21: Proceso de inicio de sesión

Fuente: Elaboración propia.

El proceso inicia cuando el usuario accede a la aplicación e ingresa sus credenciales en el formulario principal. El sistema valida que los campos no estén vacíos y verifica las credenciales contra los registros de la base de datos. Si son correctas, se identifica el rol asignado (como Administrador, Recursos Humanos, Jefatura o Usuario) y se redirige al panel principal con las opciones correspondientes. En caso contrario, se muestra un mensaje de error y se permite reintentar. Este flujo asegura que solo usuarios autorizados accedan al sistema, respetando sus permisos y garantizando el control de acceso.

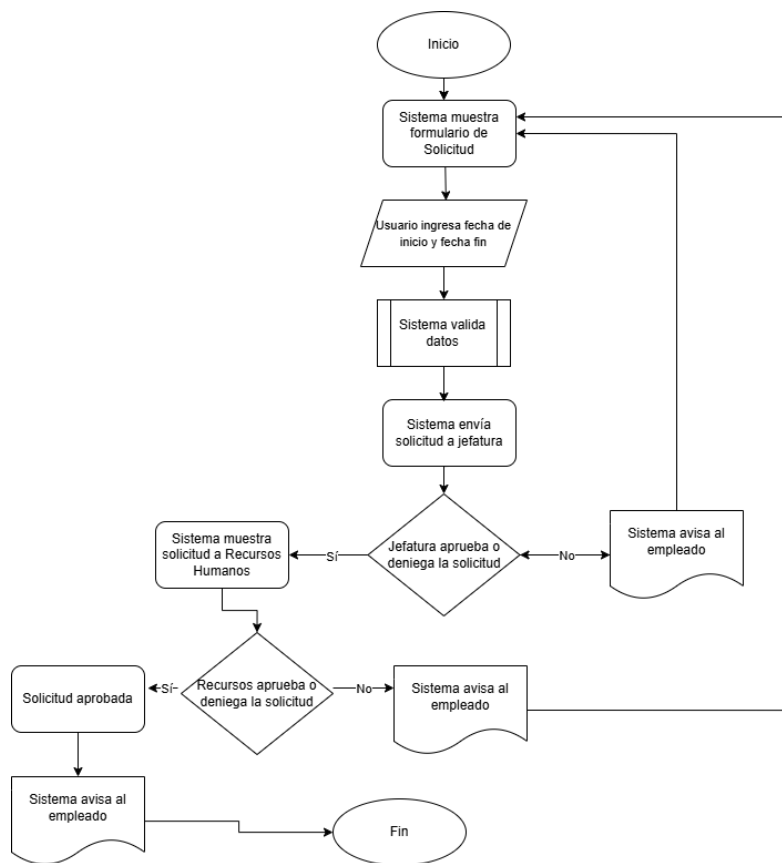
Figura 22: Agregar empleado



Fuente: Elaboración propia.

El proceso de registro de un nuevo empleado inicia cuando el usuario accede al formulario correspondiente dentro del sistema. Una vez en pantalla, se ingresan los datos requeridos como nombre, identificación, puesto y horario asignado. El sistema valida el formato de los datos y comprueba que todos los campos obligatorios estén completos. Si la información es válida, se guarda en la base de datos y se asignan automáticamente el puesto y el horario correspondiente. En caso de que falte algún dato, se muestra un mensaje de error indicando los campos incompletos, permitiendo al usuario corregirlos antes de continuar. Este flujo garantiza que el registro se realice de forma estructurada, evitando inconsistencias y asegurando que cada empleado quede correctamente vinculado al sistema.

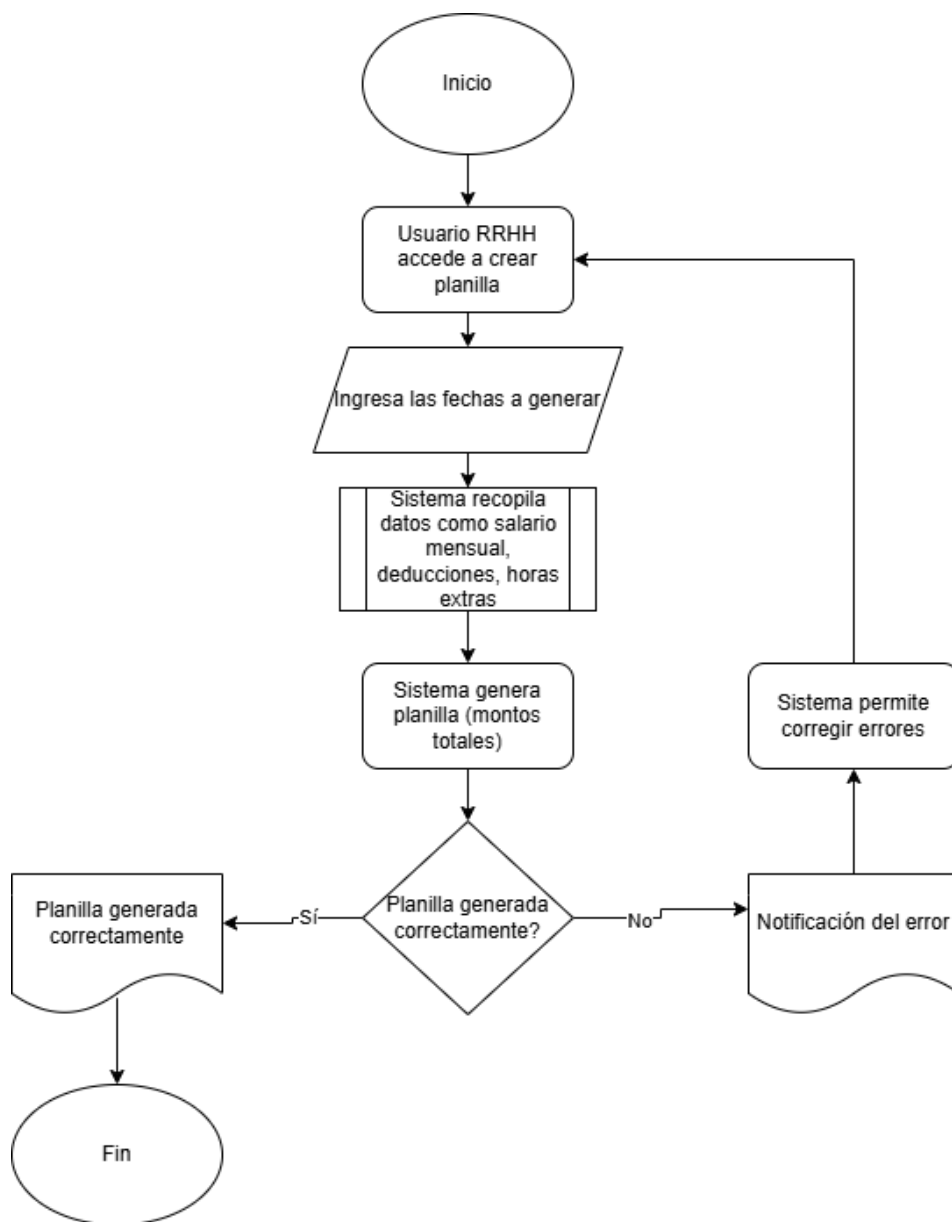
Figura 23: Solicitud de vacaciones



Fuente: Elaboración propia.

El proceso inicia cuando el usuario accede al formulario de solicitud y registra las fechas de inicio y fin del período deseado. El sistema valida los datos ingresados y envía la solicitud a la jefatura correspondiente. Si la jefatura aprueba, la solicitud se envía al departamento de Recursos Humanos para su validación final. En caso de ser aprobada por ambas partes, el sistema notifica al empleado que la solicitud ha sido aceptada. Si alguna de las partes la rechaza, se informa al empleado del resultado. Este flujo asegura que las solicitudes de vacaciones sean evaluadas de forma estructurada, respetando los niveles de autorización establecidos y garantizando trazabilidad en el proceso.

Figura 24: Generar planilla

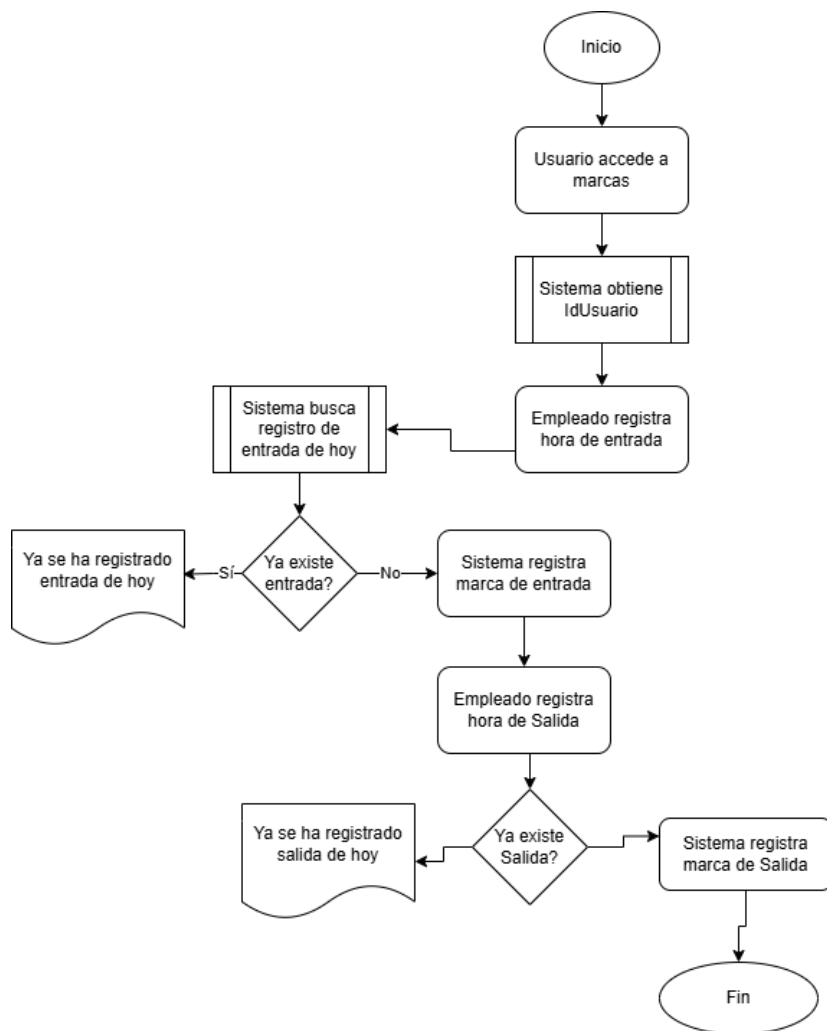


Fuente: Elaboración propia.

El proceso inicia cuando el usuario del área de Recursos Humanos accede a la opción de crear una nueva planilla. Seguidamente, se ingresan las fechas correspondientes al período que se desea calcular. El sistema recopila automáticamente los datos necesarios, como el salario mensual, las deducciones aplicables, las horas extra aprobadas y otros factores relevantes. Con esta información, se realiza el cálculo de los montos totales y se genera la planilla. Si el proceso se completa adecuadamente, se confirma la generación y se finaliza el flujo. En caso de que se

detecten errores durante el cálculo, el sistema permite corregirlos y notifica al usuario para que pueda ajustar los datos antes de continuar. Este procedimiento garantiza que las planillas se generen de forma precisa.

Figura 25: Asistencia



Fuente: Elaboración propia.

Este proceso permite registrar las marcas de entrada y salida de los empleados de forma controlada. Inicia cuando el usuario accede al módulo de marcas, momento en el cual el sistema identifica al usuario y verifica si ya existe un registro de entrada para el día actual. Si no se ha registrado la entrada, el empleado puede marcarla y el sistema la guarda. Si la entrada ya existe, se evalúa si también se ha registrado la salida. En caso de que falte, el empleado puede registrar su hora de salida y el sistema la almacena. Si ambas marcas ya están registradas, el proceso finaliza

sin duplicaciones. Este flujo garantiza que cada jornada laboral quede correctamente registrada, evitando errores y duplicidades en el control de asistencia.

Diseño de salidas

El sistema de recursos humanos implementado en el prototipo genera múltiples salidas que permiten a los usuarios consultar, verificar y gestionar la información en los distintos módulos. Estas salidas se presentan en forma de pantallas, reportes y vistas detalladas. A continuación, se describen cinco de las principales interfaces de salida incluidas en el sistema, cada una diseñada para ofrecer claridad, accesibilidad y trazabilidad sobre los datos registrados.

Figura 26: Detalle de planillas

Detalles de Planilla	
Empleado	Maria Brenes sanchez
Fecha Inicio	2025-11-16
Fecha Fin	2025-11-30
Monto Bruto	€210 000,00
Monto Deducciones	€21 714,00
Rebajo por incapacidad	€0,00
Pago patronal por incapacidad	€0,00
Monto Horas Extra	€0,00
Días Vacaciones	0
Monto Total	€188 286,00
No se encontraron deducciones aplicadas para este periodo.	
No se encontraron permisos para este periodo.	
No se encontraron vacaciones aprobadas para este periodo.	
Impuesto al salario	
Base gravable (mensual)	€420 000,00
Retenido (quincena)	€0,00
Observación	Impuesto al salario (quincenal) automático

Fuente: Elaboración propia.

Figura 27: Lista de empleados

Lista de Empleados

[Agregar Nuevo Empleado](#)

Nombre	Apellidos	Identificación	Fecha de nacimiento	Género	Salario	Departamento	Horario	Fecha de Ingreso	Estado	Acciones
Maria	Brenes sanchez	Cédula 304009891	03/06/1990	Femenino	€420 000,00	Sistemas	Diurno	01/01/2025	Activo	Editar Dar de baja
Rolando	Fonseca Gutierrez	Cédula 409000678	06/05/1988	Masculino	€800 000,00	Recursos Humanos	Diurno	01/01/2024	Activo	Editar Dar de baja
Fabian	Hernandez	Cédula 304008988	06/01/1999	Masculino	€514 000,00	Mercadeo	Diurno	01/01/2025	Activo	Editar Dar de baja
Roberto	Navarro Salas	Cédula 205000616	15/02/1997	Masculino	€800 000,00	Sistemas	Diurno	15/02/2024	Activo	Editar Dar de baja
Fernanda	Mora García	DIMEX 657553695	20/06/2005	Femenino	€520 000,00	Sistemas	Diurno	01/01/2024	Activo	Editar Dar de baja
Ericka	Villarreal Morales	Cédula 102780110	21/07/2000	Femenino	€1 500 000,00	Medicina de Empresa	Diurno	21/04/2025	Activo	Editar Dar de baja
Fanny	Sandoval Sánchez	Cédula 304009890	14/02/1887	Femenino	€600 000,00	Ventas	Diurno	05/08/2025	Activo	Editar Dar de baja

Fuente: Elaboración propia.

Figura 28: Vacaciones solicitadas

Mis Solicitudes de Vacaciones

[Solicitar Vacaciones](#)

Fecha Solicitud	Inicio	Fin	Dias	Estado Jefatura	Estado RRHH	Aprobado
22/10/2025	04/11/2025	06/11/2025	3	Rechazado	Rechazado	No
22/10/2025	05/11/2025	06/11/2025	2	Aprobado	Aprobado	Si
10/10/2025	14/10/2025	14/10/2025	1	Aprobado	Aprobado	Si
02/09/2025	10/09/2025	12/09/2025	3	Aprobado	Aprobado	Si
29/08/2025	04/09/2025	05/09/2025	2	Aprobado	Rechazado	No

Fuente: Elaboración propia.

Figura 29: Asistencias

Consultas

Asistencias | Deducciones | Empleados | Horas Extra | Permisos | Vacaciones | Planillas | Liquidaciones | Aguinaldo

Evaluaciones

Empleado: Departamento: Desde: Hasta: Tardías:

[Buscar](#) [Exportar](#)

Fecha	Empleado	Entrada	Salida
2025-10-20	Maria Brenes sanchez	08:00:00	18:00:00
2025-10-16	Maria Brenes sanchez	08:00:00	18:00:00
2025-10-16	Roberto Navarro Salas	08:00:00	18:00:00
2025-10-16	Fernanda Mora García	08:00:00	18:00:00
2025-10-16	Rolando Fonseca Gutierrez	08:00:00	18:00:00
2025-10-16	Fanny Sandoval Sánchez	08:00:00	18:00:00
2025-10-16	Marianela Sánchez Pereira	08:00:00	18:00:00
2025-10-16	Tiffany Brenes Ulloa	08:00:00	18:00:00
2025-10-16	Fabian Hernandez	08:00:00	18:00:00

Fuente: Elaboración propia.

Figura 30: Permisos

Mis Solicitudes de Permiso

[Solicitar Permiso](#)

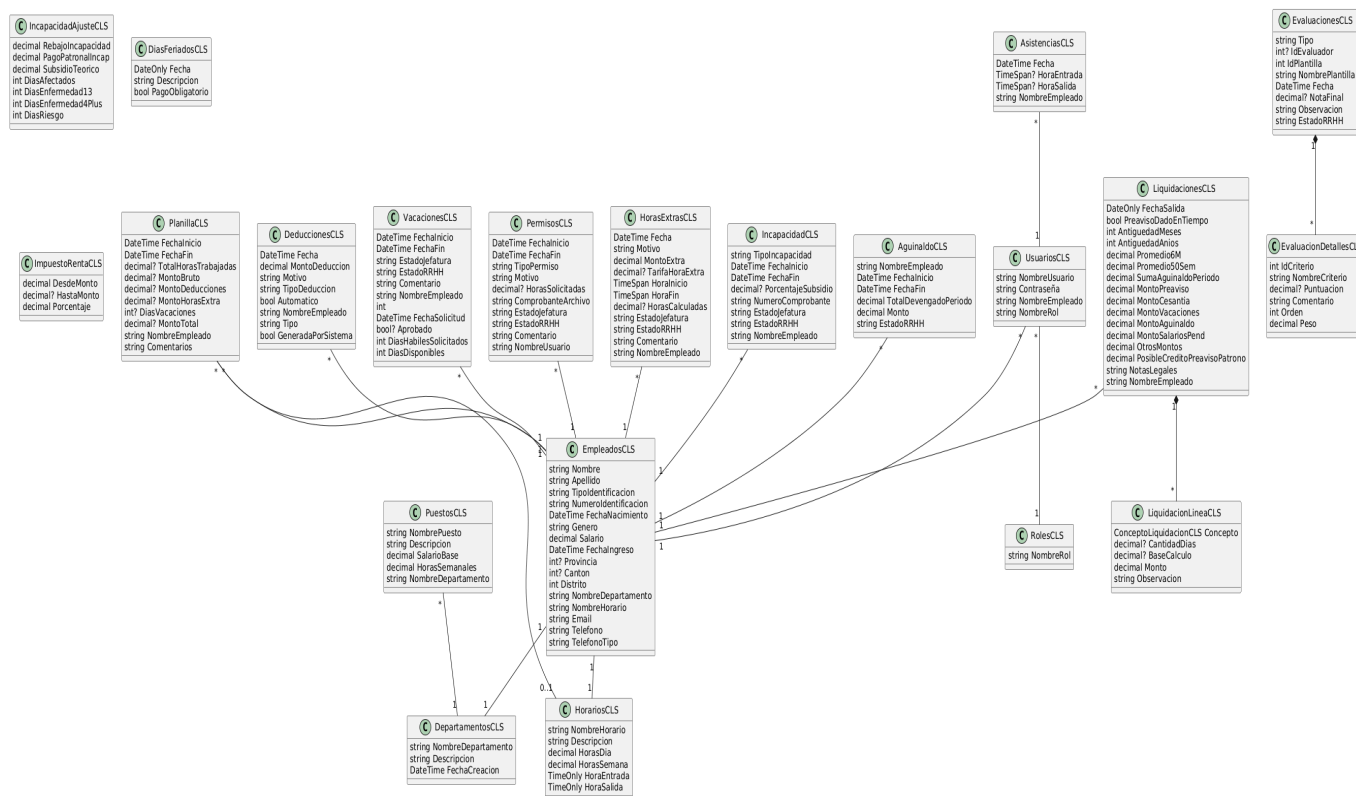
Tipo	Inicio	Fin	Jefatura	RRHH	Comentario
Personal	13/10/2025	13/10/2025	Aprobado	Aprobado	
Sin goce	15/10/2025	15/10/2025	Aprobado	Aprobado	
Sin goce	10/10/2025	10/10/2025	Rechazado	Rechazado	
Sin goce	29/10/2025	29/10/2025	Aprobado	Aprobado	

Fuente: Elaboración propia.

Diagramas UML

El diagrama de clases permite representar de forma ordenada todas las clases desarrolladas durante la implementación del prototipo. Este tipo de diagrama muestra los atributos, métodos y relaciones existentes entre las clases, ofreciendo una visión general clara de la arquitectura interna del sistema.

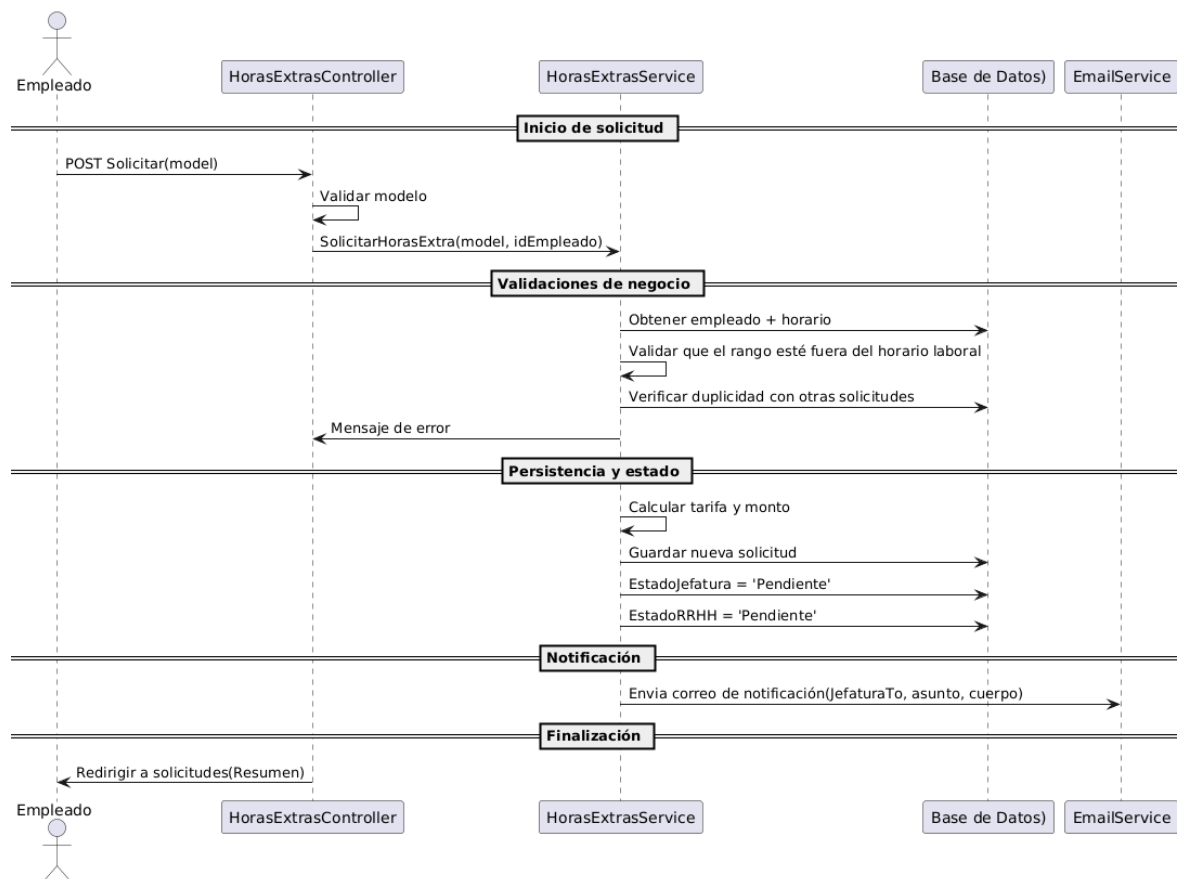
Figura 31: Diagrama de clases



Fuente: Elaboración propia.

Diagrama de secuencia

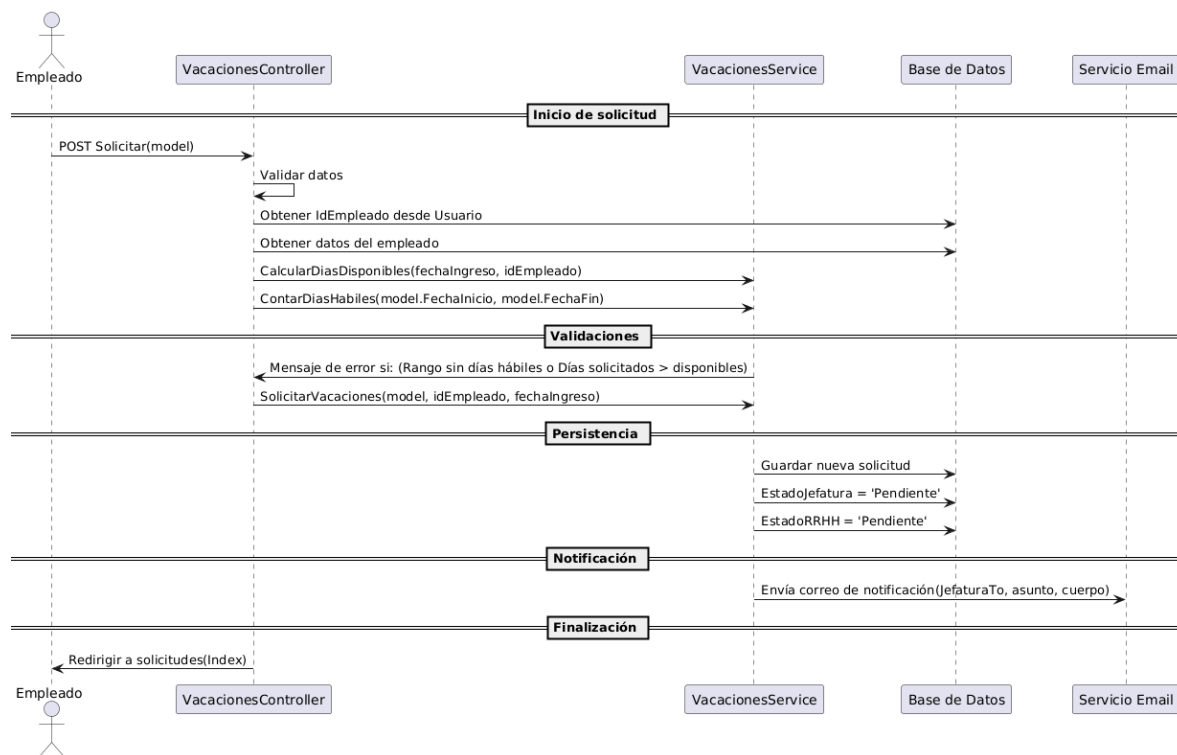
Figura 32: Solicitud de horas extra



Fuente: Elaboración propia.

Este diagrama representa el flujo completo que se ejecuta cuando un empleado realiza una solicitud de horas extra dentro del sistema. El proceso inicia con el envío del formulario por parte del empleado, el cual es recibido por el controlador correspondiente. Luego de validar el modelo, se invoca el servicio de horas extra, que realiza múltiples validaciones de negocio: verifica que el horario solicitado esté fuera de la jornada laboral, que no existan solicitudes duplicadas, y que los datos sean coherentes. Si alguna validación falla, se devuelve un mensaje de error. En caso contrario, el sistema calcula la tarifa y el monto, guarda la solicitud en la base de datos y la marca con estado “Pendiente” tanto para Recursos Humanos como para el flujo general. Finalmente, se envía una notificación por correo electrónico al destinatario correspondiente, y el controlador redirige al empleado a la vista resumen de sus solicitudes. Este flujo garantiza que las solicitudes sean procesadas de forma estructurada y trazable.

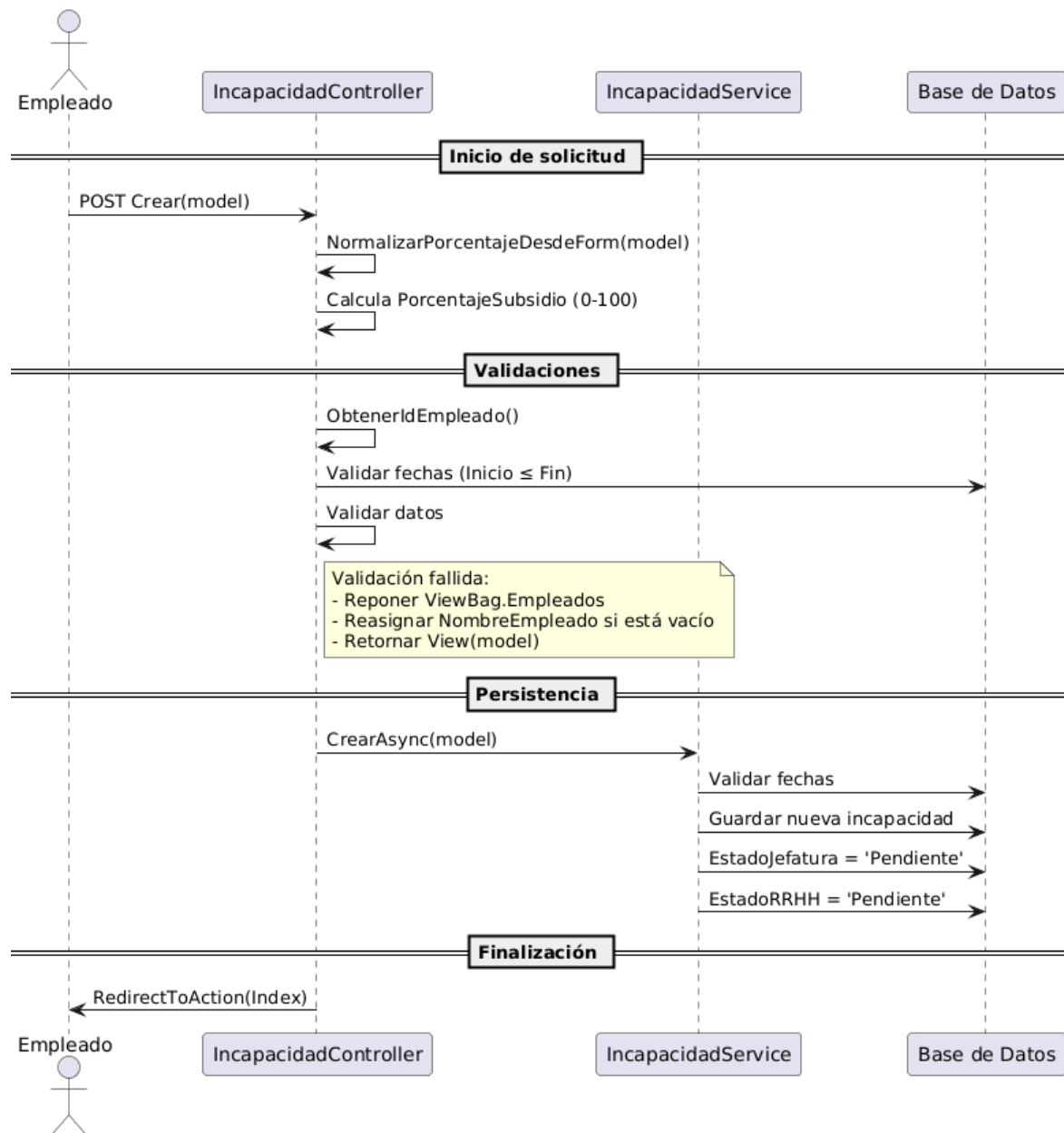
Figura 33: Solicitud de vacaciones



Fuente: Elaboración propia.

Este diagrama representa el flujo completo que se ejecuta cuando un empleado realiza una solicitud de vacaciones dentro del sistema. El proceso inicia con el envío del formulario por parte del empleado, el cual es recibido por el controlador correspondiente. Luego de validar los datos, el sistema identifica al usuario y obtiene la información del empleado. A partir de las fechas ingresadas, se calcula la cantidad de días solicitados y se verifica si están dentro del rango disponible. Si la solicitud es válida, se guarda en la base de datos y se asigna el estado “Pendiente”. Posteriormente, se envía una notificación por correo electrónico al destinatario correspondiente con los detalles de la solicitud. Finalmente, el controlador redirige al empleado a la vista de solicitudes, completando el proceso.

Figura 34: Solicitud de incapacidades

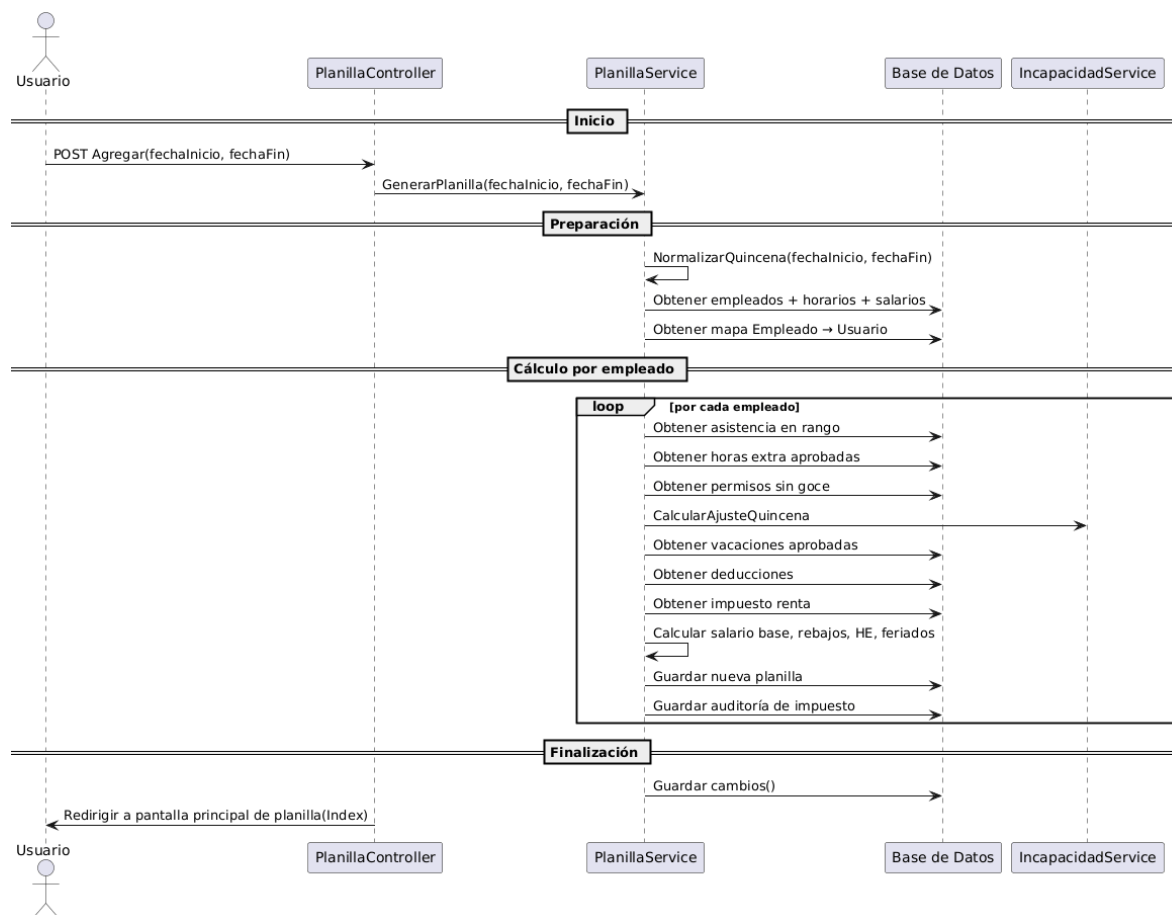


Fuente: Elaboración propia.

Este diagrama representa el flujo que se ejecuta cuando un empleado de Recursos Humanos crea una solicitud de incapacidad dentro del sistema. El proceso inicia con el envío del formulario por parte del usuario, el cual es recibido por el controlador. El sistema normaliza el porcentaje ingresado y calcula el subsidio correspondiente. Luego, se obtienen los datos del empleado y se validan las fechas y condiciones de la solicitud. Si la validación falla, se devuelve el formulario con los datos ingresados para su corrección. Si la solicitud es válida, se guarda en la base de datos

y se asignan los estados “Pendiente” tanto para el flujo general como para Recursos Humanos. Finalmente, el controlador redirige al usuario a la vista principal, completando el proceso.

Figura 35: Generación de planilla



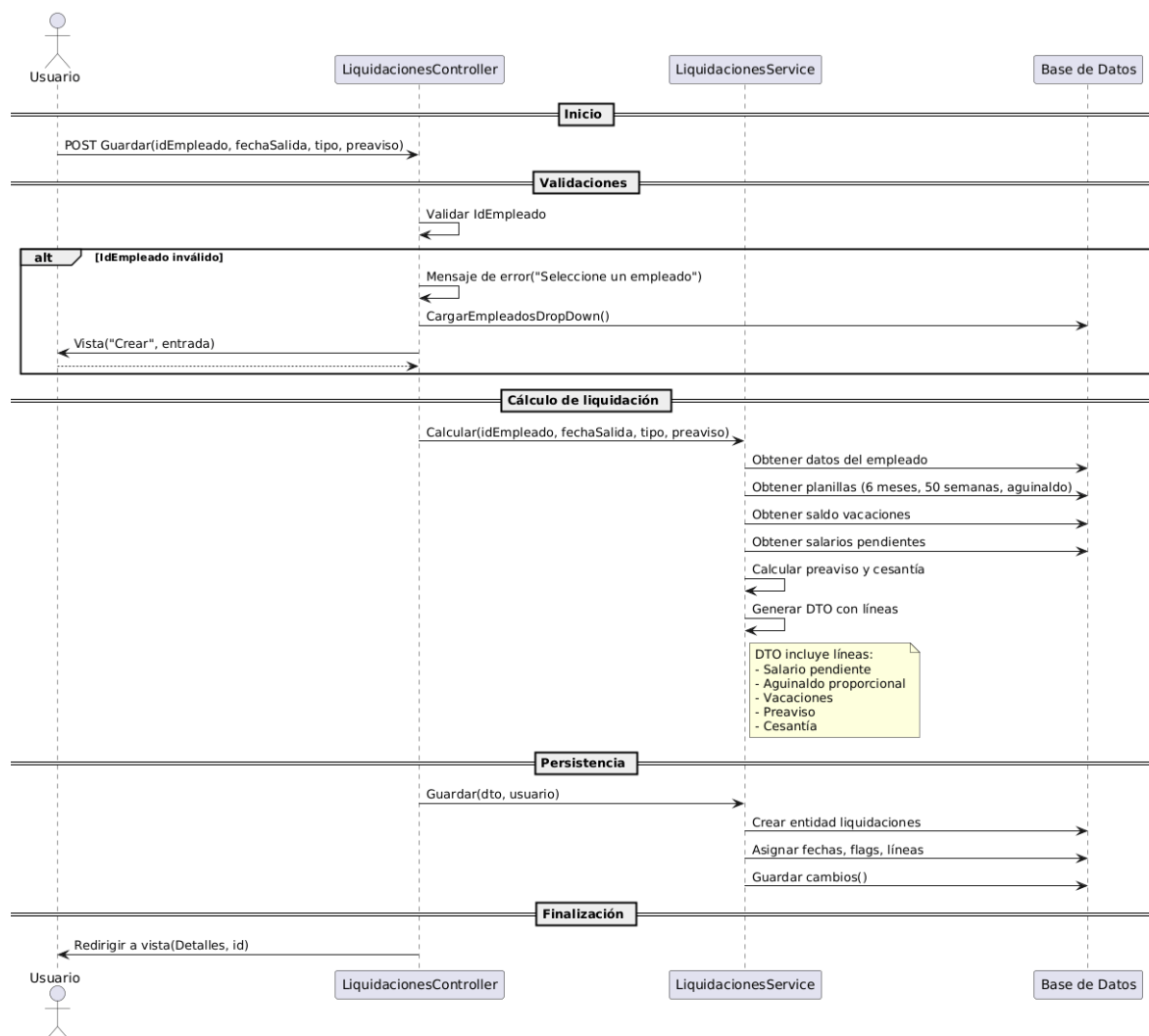
Fuente: Elaboración propia.

Este diagrama representa el flujo completo que se ejecuta cuando el usuario del área de Recursos Humanos genera una nueva planilla dentro del sistema. El proceso inicia con el envío de una solicitud POST que incluye el rango de fechas a procesar. El controlador invoca el servicio de planillas, el cual normaliza el período quincenal y recopila los datos necesarios: empleados activos, horarios asignados, salarios base. Luego, se realiza un ciclo por cada empleado, en el que se obtienen las asistencias registradas, permisos sin goce, vacaciones aprobadas, deducciones, horas extra y feriados, así como el impuesto sobre la renta. Asimismo, se consulta el servicio de incapacidades para calcular ajustes salariales. Con esta información, el sistema calcula el salario

bruto y neto, y guarda los registros proporcionados en la base de datos, incluyendo la auditoría del impuesto aplicado.

Finalmente, se consolidan los cambios y se redirige al usuario hacia la pantalla principal de planillas. Este flujo garantiza que el cálculo salarial se realice de forma precisa, considerando todas las variables laborales y asegurando trazabilidad en cada etapa del proceso.

Figura 36: Generación de liquidaciones



Fuente: Elaboración propia.

Este diagrama representa el flujo que se ejecuta cuando el usuario del sistema registra la liquidación de un empleado. El proceso inicia con el envío de una solicitud POST que incluye parámetros como la fecha de salida y el tipo de liquidación. El controlador valida que se haya

seleccionado un empleado; si no, se muestra un mensaje de error y se repuebla el menú desplegable con los empleados disponibles.

Una vez validado, el sistema obtiene los datos del empleado y recopila información relevante como planillas anteriores, semanas trabajadas, aguinaldo proporcional, vacaciones acumuladas, preaviso y cesantía. Posteriormente, se guarda la información en la base de datos, asignando fechas, etiquetas y detalles correspondientes. Finalmente, el sistema redirige al usuario a la vista de detalles de la liquidación, completando el proceso. Este flujo garantiza que las liquidaciones se generen de forma precisa, considerando todos los componentes legales y contractuales asociados con el cierre laboral.

Programación

En este apartado se presentan fragmentos específicos del código del prototipo para evidenciar reglas de negocio, lógica aplicada y funcionalidades clave. Los ejemplos muestran cómo se implementaron controladores, servicios, modelos y vistas, manteniendo buenas prácticas como inyección de dependencias, validaciones y control de acceso por roles.

Entradas y salidas

El manejo de datos en el sistema se ejemplifica mediante el formulario de registro de empleados. Las figuras incluidas muestran el código responsable de capturar las entradas del usuario, tales como identificación, nombre y apellidos, todos definidos como campos obligatorios. Este componente resulta esencial para el procesamiento de la información, ya que permite generar las salidas correspondientes, como la visualización de datos, la generación de planillas y la trazabilidad de registros en el sistema.

Figura 37: Agregar empleado

```

<div class="container mt-4">
  <div class="card shadow">
    <div class="card-header bg-primary text-white">
      <h4 class="mb-0">Agregar Empleado</h4>
    </div>
    <div class="card-body">
      <form asp-action="Agregar" method="post">
        <div class="row g-3">
          <div class="col-md-6">
            @Html.LabelFor(m => m.Nombre)
            @Html.TextBoxFor(m => m.Nombre, new { @class = "form-control" })
            @Html.ValidationMessageFor(m => m.Nombre, "", new { @class = "text-danger" })
          </div>
          <div class="col-md-6">
            @Html.LabelFor(m => m.Apellido)
            @Html.TextBoxFor(m => m.Apellido, new { @class = "form-control" })
            @Html.ValidationMessageFor(m => m.Apellido, "", new { @class = "text-danger" })
          </div>
          <div class="col-md-4">
            @Html.LabelFor(m => m.TipoIdentificacion)
            @Html.DropDownListFor(m => m.TipoIdentificacion, tiposId, "-- Seleccione --", new { @class = "form-select" })
            @Html.ValidationMessageFor(m => m.TipoIdentificacion, "", new { @class = "text-danger" })
          </div>
          <div class="col-md-4">
            @Html.LabelFor(m => m.NumeroIdentificacion)
            @Html.TextBoxFor(m => m.NumeroIdentificacion, new { @class = "form-control" })
            @Html.ValidationMessageFor(m => m.NumeroIdentificacion, "", new { @class = "text-danger" })
          </div>
          <div class="col-md-4">
            @Html.LabelFor(m => m.Genero)
            @Html.DropDownListFor(m => m.Genero, generos, "-- Seleccione --", new { @class = "form-select" })
            @Html.ValidationMessageFor(m => m.Genero, "", new { @class = "text-danger" })
          </div>
          <div class="col-md-4">
            @Html.LabelFor(m => m.FechaNacimiento)
            @Html.TextBoxFor(m => m.FechaNacimiento, "{0:yyyy-MM-dd}", new { @class = "form-control", type = "date" })
            @Html.ValidationMessageFor(m => m.FechaNacimiento, "", new { @class = "text-danger" })
          </div>
        </div>
      </form>
    </div>
  </div>
</div>

```

Fuente: Elaboración propia.

Figura 38: Agregar empleado 2

```

</div>
<div class="col-md-4">
  @Html.LabelFor(m => m.FechaNacimiento)
  @Html.TextBoxFor(m => m.FechaNacimiento, "{0:yyyy-MM-dd}", new { @class = "form-control", type = "date" })
  @Html.ValidationMessageFor(m => m.FechaNacimiento, "", new { @class = "text-danger" })
</div>
<div class="col-md-4">
  @Html.LabelFor(m => m.Salario)
  @Html.TextBoxFor(m => m.Salario, new { @class = "form-control", type = "number", step = "0.01" })
  @Html.ValidationMessageFor(m => m.Salario, "", new { @class = "text-danger" })
</div>
<div class="col-md-4">
  @Html.LabelFor(m => m.FechaIngreso)
  @Html.TextBoxFor(m => m.FechaIngreso, "{0:yyyy-MM-dd}", new { @class = "form-control", type = "date" })
  @Html.ValidationMessageFor(m => m.FechaIngreso, "", new { @class = "text-danger" })
</div>
<div class="col-md-6">
  @Html.LabelFor(m => m.IdDepartamento)
  @Html.DropDownListFor(m => m.IdDepartamento, listaDepto, new { @class = "form-select" })
  @Html.ValidationMessageFor(m => m.IdDepartamento, "", new { @class = "text-danger" })
</div>
<div class="col-md-6">
  @Html.LabelFor(m => m.IdHorario)
  @Html.DropDownListFor(m => m.IdHorario, listaHora, new { @class = "form-select" })
  @Html.ValidationMessageFor(m => m.IdHorario, "", new { @class = "text-danger" })
</div>
<!-- Geografia -->
<div class="col-md-4">
  @Html.LabelFor(m => m.IdProvincia, "Provincia")
  @Html.DropDownListFor(m => m.IdProvincia, listaProv, new { @class = "form-select", id = "ddlProvincia" })
  @Html.ValidationMessageFor(m => m.IdProvincia, "", new { @class = "text-danger" })
</div>
<div class="col-md-4">
  @Html.LabelFor(m => m.IdCanton, "Cantón")
  @Html.DropDownListFor(m => m.IdCanton, listaCant, new { @class = "form-select", id = "ddlCanton" })
  @Html.ValidationMessageFor(m => m.IdCanton, "", new { @class = "text-danger" })
</div>

```

Fuente: Elaboración propia.

Figura 39: Agregar empleado 3

```

  @Html.ValidationMessageFor(m => m.IdCanton, "", new { @class = "text-danger" })
</div>
<div class="col-md-4">
  @Html.LabelFor(m => m.IdDistrito, "Distrito")
  @Html.DropDownListFor(m => m.IdDistrito, listaDist, new { @class = "form-select", id = "ddlDistrito" })
  @Html.ValidationMessageFor(m => m.IdDistrito, "", new { @class = "text-danger" })
</div>
<!-- Contacto principal (opcional) -->
<div class="col-md-6">
  @Html.LabelFor(m => m.Email)
  @Html.TextBoxFor(m => m.Email, new { @class = "form-control", type = "email" })
  @Html.ValidationMessageFor(m => m.Email, "", new { @class = "text-danger" })
</div>
<div class="col-md-4">
  @Html.LabelFor(m => m.Telefono)
  @Html.TextBoxFor(m => m.Telefono, new { @class = "form-control" })
  @Html.ValidationMessageFor(m => m.Telefono, "", new { @class = "text-danger" })
</div>
<div class="col-md-2">
  @Html.LabelFor(m => m.TelefonoTipo, "Tipo Teléfono")
  @Html.DropDownListFor(m => m.TelefonoTipo, tiposTel, "-- Seleccione --", new { @class = "form-select" })
  @Html.ValidationMessageFor(m => m.TelefonoTipo, "", new { @class = "text-danger" })
</div>
</div>
<div class="mt-4 d-flex justify-content-between">
  <button type="submit" class="btn btn-primary">Guardar</button>
  <a asp-action="Index" class="btn btn-secondary">Cancelar</a>
</div>
</form>
</div>
</div>

```

Fuente: Elaboración propia.

Figura 40: salida de empleados

```

<table class="table table-bordered table-hover table-striped align-middle shadow-sm">
  <thead class="table-dark text-center">
    <tr>
      <th>@Html.DisplayNameFor(p => p[0].Nombre)</th>
      <th>@Html.DisplayNameFor(p => p[0].Apellido)</th>
      <th>Identificación</th>
      <th>@Html.DisplayNameFor(p => p[0].FechaNacimiento)</th>
      <th>@Html.DisplayNameFor(p => p[0].Genero)</th>
      <th>Salario</th>
      <th>Departamento</th>
      <th>Horario</th>
      <th>@Html.DisplayNameFor(p => p[0].FechaIngreso)</th>
      <th>Acciones</th>
    </tr>
  </thead>

```

Fuente: Elaboración propia.

Figura 41: salida de empleados 2

```

<tbody>
  @foreach (var item in Model)
  {
    <tr>
      <td>@item.Nombre</td>
      <td>@item.Apellido</td>
      <td>@($"{NiceTipo(item.TipoIdentificacion)} {item.NumeroIdentificacion}")</td>
      <td>@(item.FechaNacimiento.HasValue ? item.FechaNacimiento.Value.ToString("dd/MM/yyyy") : "-")</td>
      <td>@NiceGenero(item.Genero)</td>
      <td>@item.Salario.ToString("C", new System.Globalization.CultureInfo("es-CR"))</td>
      <td>@item.NombreDepartamento</td>
      <td>@item.NombreHorario</td>
      <td>@($"{item.FechaIngreso:dd/MM/yyyy}")</td>
      <td class="text-nowrap">
        <a class="btn btn-warning btn-sm" asp-action="Editar" asp-route-id="@item.IdEmpleado">
          <i class="bi bi-pencil-square"></i> Editar
        </a>
        <a class="btn btn-danger btn-sm" asp-action="Eliminar" asp-route-id="@item.IdEmpleado">
          <i class="bi bi-trash"></i> Eliminar
        </a>
      </td>
    </tr>
  }
</tbody>
</table>

```

Fuente: Elaboración propia.

Esta pantalla permite visualizar el listado completo de empleados registrados en el sistema. La información se presenta en una tabla estructurada que incluye campos clave del registro del empleado. Esta salida es fundamental para la administración del recurso humano, ya que permite consultar, modificar y mantener actualizada la información laboral de cada colaborador.

Procesos

La lógica de programación vinculada al manejo de colaboradores se implementa a través de los controladores específicos del sistema, los cuales dirigen las operaciones de registro, edición y consulta de empleados en la base de datos.

Figura 42: Controlador GET agregar

```

0 referencias
public IActionResult Agregar()
{
    var model = new EmpleadosCLS
    {
        FechaIngreso = DateTime.Today,
        TipoIdentificacion = "Cédula",
        Genero = "Masculino",
        TelefonoTipo = "Móvil"
    };

    CargarCombosBasicosEmpleado(model.TipoIdentificacion, model.Genero, model.TelefonoTipo);
    CargarCombosGeografiaYOtros(null);
    return View(model);
}

```

Fuente: Elaboración propia.

En el controlador correspondiente al registro de empleados, el método GET llamado Agregar tiene como propósito preparar el formulario inicializando el modelo con valores por defecto. Esto incluye asignar la fecha de ingreso como la del día actual, definir el tipo de identificación como “Cédula”, el género como “Masculino” y el tipo de teléfono como “Móvil”. Todo este preparativo garantiza que al seleccionar la vista “Agregar”, el formulario esté listo para recibir datos del usuario de forma organizada, con listas desplegables correctamente pobladas y valores iniciales. Este fragmento de código es útil para ilustrar cómo se manejan las entradas en el sistema, combinando lógica de inicialización con la carga dinámica de datos de referencia.

Figura 43: Controlador POST agregar

```

[HttpPost]
0 referencias
public IActionResult Agregar(EmpleadosCLS model)
{
    model.TipoIdentificacion = NormalizarTipoId(model.TipoIdentificacion);
    model.Genero = NormalizarGenero(model.Genero);
    model.TelefonoTipo = NormalizarTipoTelefono(model.TelefonoTipo);

    if (!ModelState.IsValid)
    {
        CargarCombosBasicosEmpleado(model.TipoIdentificacion, model.Genero, model.TelefonoTipo);
        CargarCombosGeografiaYOtros(model);
        return View(model);
    }

    _empleadoService.AgregarEmpleado(model);
    TempData["mensaje"] = "Empleado agregado correctamente.";
    return RedirectToAction("Index");
}

```

Fuente: Elaboración propia.

El controlador encargado del registro de empleados, el método POST Agregar se encarga de manejar el proceso de creación de un nuevo colaborador. Recibe el modelo desde el formulario, y lo primero que se hace es normalizar ciertos campos clave como el tipo de identificación, el género y el tipo de teléfono, con el fin de evitar inconsistencias en la base de datos (por ejemplo, variantes como “Cédula” y “cedula”). Luego, se verifica si el modelo es válido. En caso de que existan errores de validación, se retorna la misma vista con el modelo.

Si el modelo pasa todas las validaciones, se delega la lógica de persistencia al servicio correspondiente mediante `_empleadoService.AgregarEmpleado(model)`, lo que permite mantener una separación clara entre el controlador y la lógica de negocio. Finalmente, se establece un mensaje temporal para informar al usuario que el registro fue exitoso y se redirige a la acción Index.

Figura 44: Servicio AgregarEmpleado

```

2 referencias
public void AgregarEmpleado(EmpleadosCLS m)
{
    if (!m.FechaIngreso.HasValue)
        throw new ArgumentException("Fecha de ingreso es obligatoria.");

    var e = new empleado
    {
        Nombre = m.Nombre!,
        Apellido = m.Apellido!,
        TipoIdentificacion = m.TipoIdentificacion!,
        NumeroIdentificacion = m.NumeroIdentificacion!,
        FechaNacimiento = m.FechaNacimiento.HasValue ? DateOnly.FromDateTime(m.FechaNacimiento.Value) : (DateOnly?)null,
        Genero = m.Genero,
        Salario = m.Salario,
        IdDepartamento = m.IdDepartamento,
        IdHorario = m.IdHorario,
        FechaIngreso = DateOnly.FromDateTime(m.FechaIngreso.Value),
        IdProvincia = m.IdProvincia,
        IdCanton = m.IdCanton,
        IdDistrito = m.IdDistrito
    };

    _context.Set<empleado>().Add(e);
    _context.SaveChanges();

    if (!string.IsNullOrWhiteSpace(m.Email))
    {
        var actuales = _context.Set<correo>().Where(x => x.IdEmpleado == e.IdEmpleado && x.Principal == true).ToList();
        foreach (var a in actuales) a.Principal = false;

        _context.Set<correo>().Add(new correo
        {
            IdEmpleado = e.IdEmpleado,
            Email = m.Email!,
            Principal = true
        });
    }
}

```

Fuente: Elaboración propia.

Figura 45: Servicio AgregarEmpleado 2

```

if (!string.IsNullOrEmpty(m.Telefono))
{
    var actualesT = _context.Set<telefono>().Where(x => x.IdEmpleado == e.IdEmpleado && x.Principal == true).ToList();
    foreach (var a in actualesT) a.Principal = false;

    _context.Set<telefono>().Add(new telefono
    {
        IdEmpleado = e.IdEmpleado,
        Numero = m.Telefono!,
        Tipo = m.TelefonoTipo,
        Principal = true
    });
}

_context.SaveChanges();
}

```

Fuente: Elaboración propia.

En el servicio responsable de la gestión de empleados, el método `AgregarEmpleado` encapsula la lógica de negocio y la constancia asociada al registro de un nuevo colaborador. El proceso inicia con una validación fundamental: se exige que la fecha de ingreso esté presente, ya que representa un requisito mínimo para considerar válida la incorporación. A continuación, se realiza el mapeo entre clases (`EmpleadosCLS`) y la entidad del modelo de datos.

Una vez construido el objeto empleado, se agrega al contexto y se ejecuta un primer guardar cambios, necesario para obtener el `IdEmpleado` generado por la base de datos.

Finalmente, se confirma la operación con un segundo guardar cambios, consolidando todos los cambios en la base de datos. Este método es representativo de buenas prácticas en la capa de servicios, ya que demuestra cómo aplicar validaciones de negocio y realizar mapeos precisos entre capas.

Figura 46: Controlador Index

```

0 referencias
public IActionResult Index()
{
    var empleados = _empleadoService.ObtenerEmpleados();
    return View(empleados);
}

```

Fuente: Elaboración propia.

El método `Index` del controlador obtiene la lista de empleados desde el servicio y la envía a la vista principal. Este proceso permite visualizar los registros existentes y facilita la gestión desde la interfaz.

Figura 47: Servicio ObtenerEmpleados

```

2 referencias
public List<EmpleadosCLS> ObtenerEmpleados()
{
    return _context.Set<empleado>()
        .Include(e => e.IdDepartamentoNavigation)
        .Include(e => e.IdHorarioNavigation)
        .AsNoTracking()
        .Select(e => new EmpleadosCLS
            {
                IdEmpleado = e.IdEmpleado,
                Nombre = e.Nombre,
                Apellido = e.Apellido,
                TipoIdentificacion = e.TipoIdentificacion,
                NumeroIdentificacion = e.NumeroIdentificacion,
                FechaNacimiento = e.FechaNacimiento.HasValue
                    ? e.FechaNacimiento.Value.ToDateTime(TimeOnly.MinValue)
                    : (DateTime?)null,
                Genero = e.Genero,
                Salario = e.Salario,
                IdDepartamento = e.IdDepartamento,
                IdHorario = e.IdHorario,
                FechaIngreso = e.FechaIngreso.ToDateTime(TimeOnly.MinValue),
                IdProvincia = e.IdProvincia,
                IdCanton = e.IdCanton,
                IdDistrito = e.IdDistrito,
                NombreDepartamento = e.IdDepartamentoNavigation != null ? e.IdDepartamentoNavigation.NombreDepartamento : "",
                NombreHorario = e.IdHorarioNavigation != null ? e.IdHorarioNavigation.NombreHorario : ""
            })
        .ToList();
}

```

Fuente: Elaboración propia.

El método ObtenerEmpleados consulta la base de datos y proyecta los registros en objetos EmpleadosCLS, incluyendo datos personales, fechas, salario y nombres de departamento y horario. Usa Include y AsNoTracking para optimizar la consulta y manejar relaciones.

Validaciones

El sistema realiza validaciones en múltiples capas para garantizar la integridad de los datos, claridad y el cumplimiento de las reglas de negocio. En primer lugar, se aplican DataAnnotations directamente en los modelos, lo que permite definir requisitos como campos obligatorios, formatos específicos y restricciones de longitud. En segundo lugar, el controlador se encarga de normalizar las entradas y determinar la respuesta adecuada, ya sea reenviar la vista con mensajes de error o continuar con el proceso. Esta arquitectura distribuida de validación permite mantener datos consistentes, minimizar errores y asegurar que el sistema funcione acorde a las reglas internas definidas por la organización.

Figura 48: Validaciones

```

namespace RHNutrimax2.Models
{
    27 referencias
    public class EmpleadosCLS
    {
        [Display(Name = "Id Empleado")]
        11 referencias
        public int IdEmpleado { get; set; }

        [Required(ErrorMessage = "El campo Nombre es obligatorio.")]
        [StringLength(50, ErrorMessage = "El nombre no debe exceder los 50 caracteres.")]
        [Display(Name = "Nombre")]
        18 referencias
        public string? Nombre { get; set; }

        [Required(ErrorMessage = "El campo Apellido es obligatorio.")]
        [StringLength(100, ErrorMessage = "El apellido no debe exceder los 100 caracteres.")]
        [Display(Name = "Apellidos")]
        18 referencias
        public string? Apellido { get; set; }

        [Required(ErrorMessage = "Seleccione el tipo de identificación.")]
        [StringLength(15, ErrorMessage = "Tipo de identificación inválido.")]
        [Display(Name = "Tipo Identificación")]
        26 referencias
        public string? TipoIdentificacion { get; set; }

        [Required(ErrorMessage = "El número de identificación es obligatorio.")]
        [StringLength(20, ErrorMessage = "El número de identificación no debe exceder los 20 caracteres.")]
        [Display(Name = "Número Identificación")]
        16 referencias
        public string? NumeroIdentificacion { get; set; }

        [DataType(DataType.Date)]
        [DisplayFormat(DataFormatString = "{0:yyyy-MM-dd}", ApplyFormatInEditMode = true)]
        [Display(Name = "Fecha de nacimiento")]
        16 referencias
        public DateTime? FechaNacimiento { get; set; }
    }
}

```

Fuente: Elaboración propia.

Figura 49: Validaciones 2

```

[StringLength(50, ErrorMessage = "El género no debe exceder los 50 caracteres.")]
[Display(Name = "Género")]
24 referencias
public string? Genero { get; set; }

[Required(ErrorMessage = "El salario es obligatorio.")]
[Range(0, double.MaxValue, ErrorMessage = "El salario debe ser un valor positivo.")]
[Display(Name = "Salario")]
12 referencias
public decimal Salario { get; set; }

[Required(ErrorMessage = "Debe seleccionar un departamento.")]
[Display(Name = "Departamento")]
12 referencias
public int IdDepartamento { get; set; }

[Required(ErrorMessage = "Debe seleccionar un horario.")]
[Display(Name = "Horario")]
11 referencias
public int IdHorario { get; set; }

[Required(ErrorMessage = "La fecha de ingreso es obligatoria.")]
[DataType(DataType.Date)]
[DisplayFormat(DataFormatString = "{0:yyyy-MM-dd}", ApplyFormatInEditMode = true)]
[Display(Name = "Fecha de Ingreso")]
19 referencias
public DateTime? FechaIngreso { get; set; }

[Display(Name = "Provincia")]
13 referencias
public int? IdProvincia { get; set; }

[Display(Name = "Cantón")]
12 referencias
public int? IdCanton { get; set; }

[Required(ErrorMessage = "Debe seleccionar un distrito.")]
[Display(Name = "Distrito")]
11 referencias
public int IdDistrito { get; set; }

```

Fuente: Elaboración propia.

Figura 50: Validaciones 3

```

7 referencias
public string? NombreDepartamento { get; set; }
7 referencias
public string? NombreHorario { get; set; }

[EmailAddress(ErrorMessage = "Email no es válido.")]
[StringLength(150, ErrorMessage = "El email no debe exceder los 150 caracteres.")]
[Display(Name = "Email")]
12 referencias
public string? Email { get; set; }

[Phone(ErrorMessage = "Teléfono no válido.")]
[StringLength(20, ErrorMessage = "El teléfono no debe exceder los 20 caracteres.")]
[Display(Name = "Teléfono")]
12 referencias
public string? Telefono { get; set; }

[StringLength(20, ErrorMessage = "El tipo de teléfono no debe exceder los 20 caracteres.")]
[Display(Name = "Tipo Teléfono")]
21 referencias
public string? TelefonoTipo { get; set; }
}

```

Fuente: Elaboración propia.

Figura 51: Validaciones 4

```

[HttpPost]
0 referencias
public IActionResult Agregar(EmpleadosCLS model)
{
    model.TipoIdentificacion = NormalizarTipoId(model.TipoIdentificacion);
    model.Genero = NormalizarGenero(model.Genero);
    model.TelefonoTipo = NormalizarTipoTelefono(model.TelefonoTipo);

    if (!ModelState.IsValid)
    {
        CargarCombosBasicosEmpleado(model.TipoIdentificacion, model.Genero, model.TelefonoTipo);
        CargarCombosGeografiaVotros(model);
        return View(model);
    }

    _empleadoService.AgregarEmpleado(model);
    TempData["mensaje"] = "Empleado agregado correctamente.";
    return RedirectToAction("Index");
}

```

Fuente: Elaboración propia.

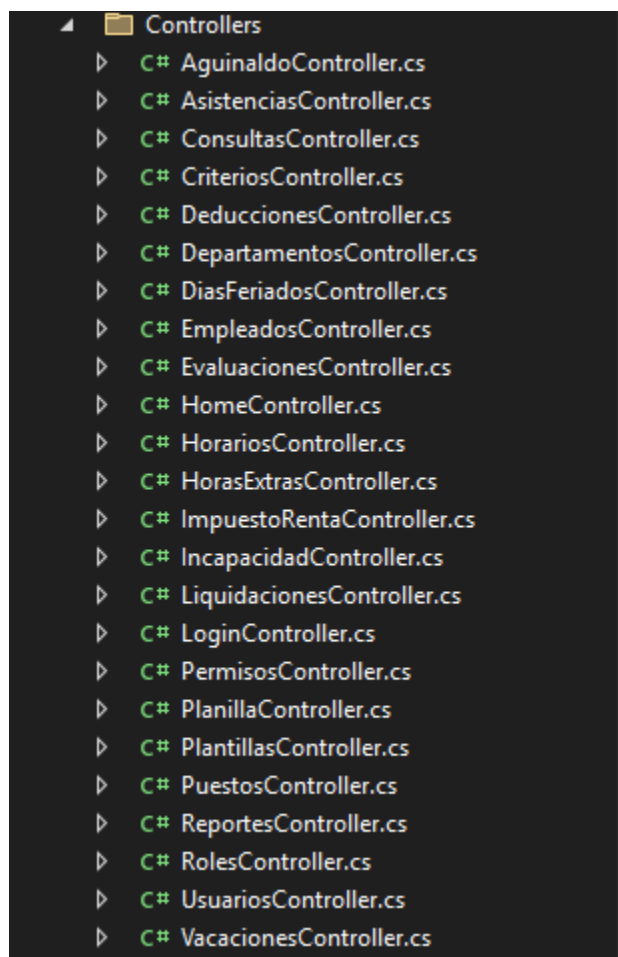
Figura 52: Validaciones 5

```

<div class="card-body">
<form asp-action="Agregar" method="post">
<div class="row g-3">
<div class="col-md-6">
@Html.LabelFor(m => m.Nombre)
@Html.TextBoxFor(m => m.Nombre, new { @class = "form-control" })
@Html.ValidationMessageFor(m => m.Nombre, "", new { @class = "text-danger" })
</div>
<div class="col-md-6">
@Html.LabelFor(m => m.Apellido)
@Html.TextBoxFor(m => m.Apellido, new { @class = "form-control" })
@Html.ValidationMessageFor(m => m.Apellido, "", new { @class = "text-danger" })
</div>
<div class="col-md-6">
@Html.LabelFor(m => m.TipoIdentificacion)
@Html.DropDownListFor(m => m.TipoIdentificacion, tiposId, "Seleccione --", new { @class = "form-select" })
@Html.ValidationMessageFor(m => m.TipoIdentificacion, "", new { @class = "text-danger" })
</div>
<div class="col-md-6">
@Html.LabelFor(m => m.NumeroIdentificacion)
@Html.TextBoxFor(m => m.NumeroIdentificacion, new { @class = "form-control" })
@Html.ValidationMessageFor(m => m.NumeroIdentificacion, "", new { @class = "text-danger" })
</div>
<div class="col-md-6">
@Html.LabelFor(m => m.Genero)
@Html.DropDownListFor(m => m.Genero, generos, "Seleccione --", new { @class = "form-select" })
@Html.ValidationMessageFor(m => m.Genero, "", new { @class = "text-danger" })
</div>
<div class="col-md-6">
@Html.LabelFor(m => m.FechaNacimiento)
@Html.TextBoxFor(m => m.FechaNacimiento, "{0:yyyy-MM-dd}", new { @class = "form-control", type = "date" })
@Html.ValidationMessageFor(m => m.FechaNacimiento, "", new { @class = "text-danger" })
</div>

```

Fuente: Elaboración propia.

Figura 53: Módulos alcanzados

Fuente: Elaboración propia.

Pruebas

A continuación, se presentan ejemplos específicos de las pruebas realizadas sobre el prototipo de gestión de recursos humanos. Estas pruebas fueron realizadas en distintos escenarios funcionales de los módulos desarrollados, con el objetivo de verificar el comportamiento esperado, validar la lógica y anticipar posibles errores que podrían afectar la experiencia del usuario. El proceso de prueba contribuye a fortalecer la confiabilidad del sistema, con lo cual se asegura que las funcionalidades respondan de manera correcta ante diversas condiciones de uso.

Tabla 90: Caso de prueba de inicio de sesión

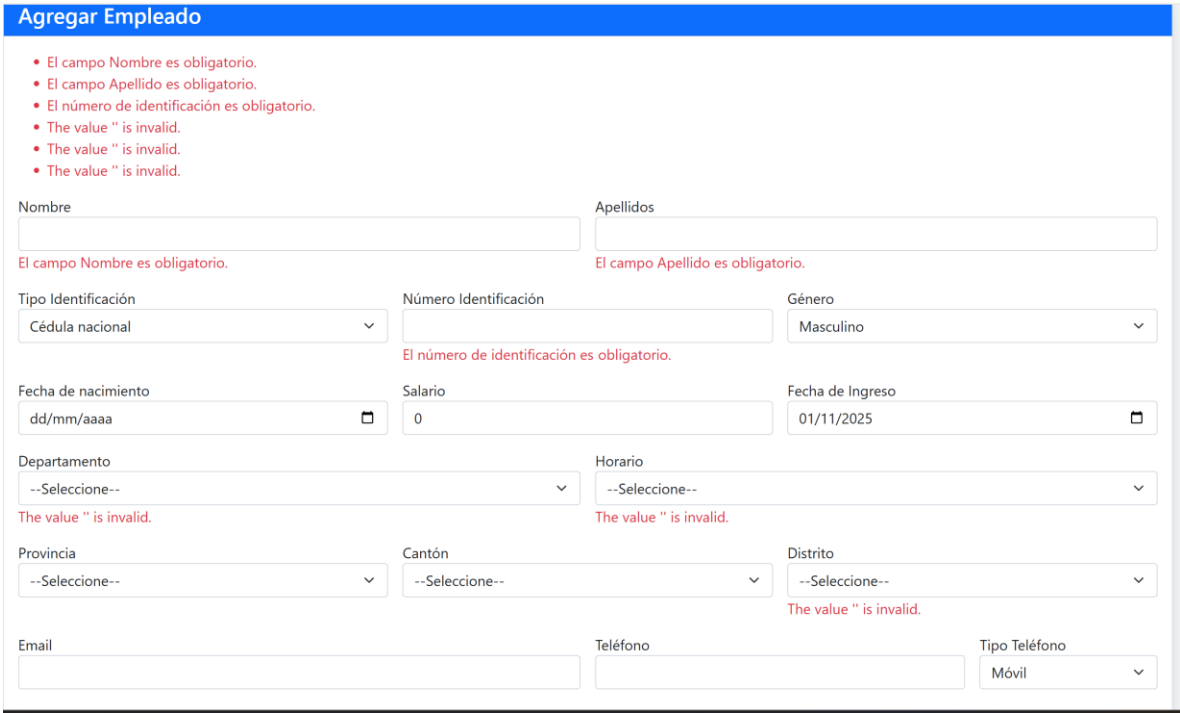
Verificación de inicio de sesión	
Numero de caso de prueba: 01	Módulo de seguridad
Descripción	
Se ejecuta la prueba de ingreso al prototipo mediante el inicio de sesión con credenciales anteriormente ingresados por el usuario, verificando el acceso.	
Resultado deseado	
El ingreso al prototipo es denegado por el ingreso de credenciales incorrectas.	
Prueba realizada:	Ingreso de contraseña incorrecta.
Resultado obtenido	
El prototipo no permite el acceso.	



The image shows a login interface for RHNutrimax. At the top, there is a logo with the letters 'nm' and the tagline 'VIDA SALUDABLE'. Below the logo, the text reads 'Bienvenido a RHNutrimax' and 'Ingresa tus credenciales para continuar'. A red error message box states 'Usuario o contraseña incorrectos'. There are two input fields: 'Usuario' containing 'rnunez' and 'Contraseña' containing '*****'. A blue 'Entrar' button is positioned below the password field. At the bottom, there is a link that says '¿Necesitas ayuda? Contacta a RRHH'.

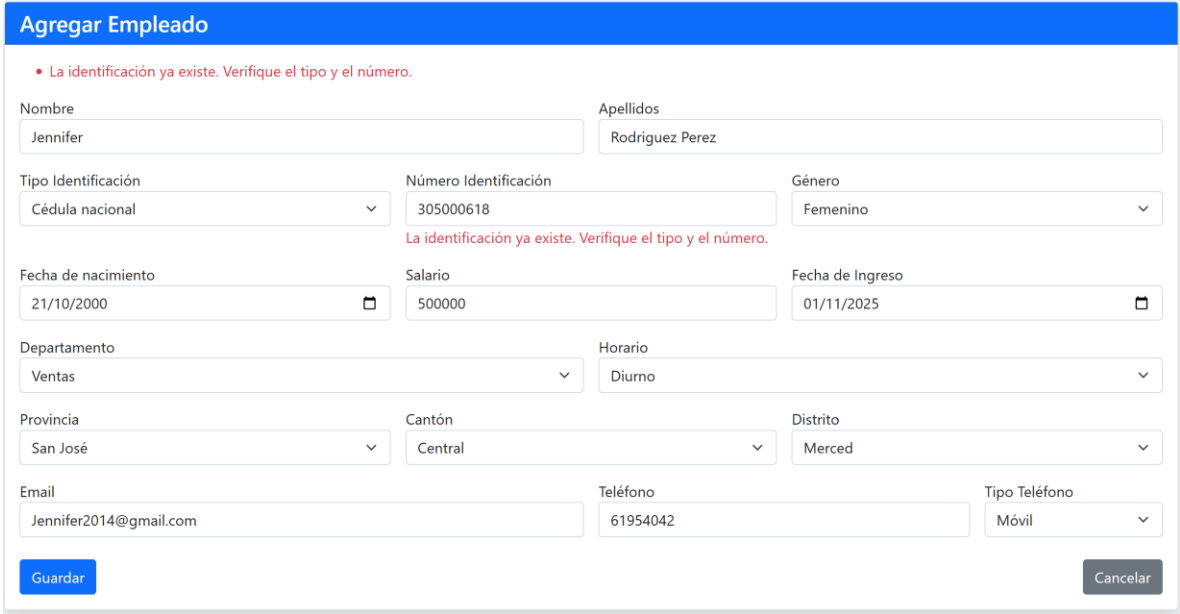
Fuente: Elaboración propia.

Tabla 91: Caso de prueba de verificación de campos del colaborador

Verificación de agregar empleado	
Numero de caso de prueba: 02	Módulo de empleados
Descripción	
Se realiza la prueba de ingreso de datos del nuevo empleado y el sistema verifican los campos.	
Resultado deseado	
El sistema muestra un mensaje de alerta de que hacen falta campos por llenar.	
Prueba realizada:	Agregar empleado con campos faltantes.
Resultado obtenido	
El prototipo muestra un mensaje de error con los datos que faltan por llenar.	
 <p>Agregar Empleado</p> <ul style="list-style-type: none"> • El campo Nombre es obligatorio. • El campo Apellido es obligatorio. • El número de identificación es obligatorio. • The value "" is invalid. • The value "" is invalid. • The value "" is invalid. <p>Nombre: <input type="text"/> Apellidos: <input type="text"/> El campo Nombre es obligatorio. El campo Apellido es obligatorio.</p> <p>Tipo Identificación: <input type="text" value="Cédula nacional"/> Número Identificación: <input type="text" value=""/> Género: <input type="text" value="Masculino"/> El número de identificación es obligatorio.</p> <p>Fecha de nacimiento: <input type="text" value="dd/mm/aaaa"/> Salario: <input type="text" value="0"/> Fecha de Ingreso: <input type="text" value="01/11/2025"/></p> <p>Departamento: <input type="text" value="--Seleccione--"/> Horario: <input type="text" value="--Seleccione--"/> The value "" is invalid. The value "" is invalid.</p> <p>Provincia: <input type="text" value="--Seleccione--"/> Cantón: <input type="text" value="--Seleccione--"/> Distrito: <input type="text" value="--Seleccione--"/> The value "" is invalid.</p> <p>Email: <input type="text"/> Teléfono: <input type="text"/> Tipo Teléfono: <input type="text" value="Móvil"/></p>	

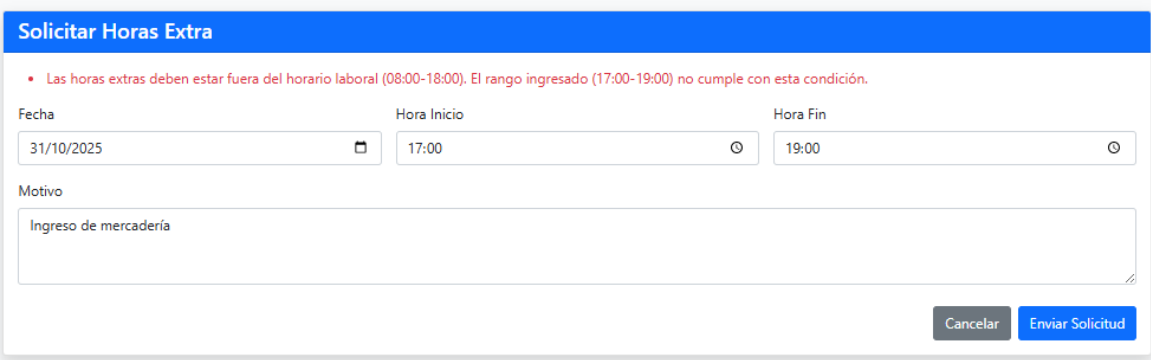
Fuente: Elaboración propia.

Tabla 92: Caso de prueba de validación de la identificación del empleado

Verificación de identificación del empleado	
Numero de caso de prueba: 03	Módulo de empleados
Descripción	
Se realiza la prueba de ingreso de datos del nuevo empleado y el sistema verifica si la identificación ya existe.	
Resultado deseado	
El sistema muestra un mensaje de alerta de que la identificación ya existe.	
Prueba realizada:	Ingreso de identificación duplicada.
Resultado obtenido	
El prototipo muestra un mensaje de que la identificación ya existe.	
	

Fuente: Elaboración propia.

Tabla 93: Caso de prueba de verificación de solicitud de horas extra

Verificación de solicitud de horas extra	
Numero de caso de prueba: 04	Módulo de horas extra.
Descripción	
Comprobar que el prototipo no permita la solicitud de horas extra en horario laboral.	
Resultado deseado	
El prototipo debe mostrar una leyenda de que no se pueden solicitar horas extra dentro del horario laboral.	
Prueba realizada:	Validación de solicitud de horas extra
Resultado obtenido	
El prototipo muestra un mensaje de que no puede solicitar horas extra en horario laboral.	
	

Fuente: Elaboración propia.

Tabla 94: Caso de prueba de verificación de solicitud de vacaciones

Verificación de disfrute de vacaciones	
Numero de caso de prueba: 05	Módulo de vacaciones
Descripción	
Comprobar que el prototipo no permita solicitar vacaciones si no posee los días suficientes para realizar la solicitud.	
Resultado deseado	
El prototipo debe mostrar una leyenda de que no posee días suficientes.	
Prueba realizada:	Validación de fechas.
Resultado obtenido	
El prototipo muestra una leyenda de que no posee días suficientes para la solicitud.	
	

Fuente: Elaboración propia.

CAPÍTULO VI: CONCLUSIONES Y RECOMENDACIONES

Este capítulo presenta las reflexiones finales derivadas del desarrollo del prototipo funcional para la gestión de recursos humanos en la empresa Nutrimax. A partir del análisis de los resultados obtenidos, se presentan las conclusiones que demuestran el cumplimiento de los objetivos planteados, así como el impacto del sistema en la optimización de procesos administrativos. Además, se proponen recomendaciones orientadas a la implementación, mejora continua y evolución futura del sistema, con base en los hallazgos técnicos, funcionales y operativos identificados durante el proyecto.

Conclusiones

El desarrollo del prototipo funcional para la gestión de recursos humanos en la empresa Nutrimax representa un avance significativo en la modernización de sus procesos administrativos. A partir del análisis de los requerimientos y la situación actual de la organización, se comprobó que la gestión manual de planillas, permisos, vacaciones y liquidaciones no solo era propensa a errores humanos, sino que también generaba ineficiencias operativas y riesgos de manipulación. El sistema propuesto responde directamente a estas necesidades, brindando una solución automatizada, segura y escalable que mejora la trazabilidad, la precisión en los cálculos y la experiencia tanto del personal administrativo como de los colaboradores.

Entre los principales logros alcanzados se destaca la implementación de módulos clave como el cálculo automático de planillas, deducciones, liquidaciones, incapacidades y aguinaldos, así como la gestión electrónica de solicitudes de vacaciones, permisos y horas extra. Estos módulos fueron diseñados respetando las reglas de negocio propias de la empresa, como la coherencia entre fechas laborales y la validación de datos críticos. Además, se incorporaron mecanismos de seguridad como autenticación por perfiles y control de acceso, lo que garantiza la protección de los datos sensibles.

Desde el punto de vista técnico, el uso de tecnologías modernas como ASP.NET Core, MySQL Workbench, Bootstrap y Microsoft Azure permitió construir una arquitectura robusta, con disponibilidad inmediata y capacidad de crecimiento. La interfaz fue diseñada para ser intuitiva y accesible, considerando el nivel digital del personal actual. Asimismo, se logró una separación clara entre capas (controlador, servicio, modelo y vista), lo que facilita el mantenimiento y la evolución futura del sistema.

En cuanto a la validación del prototipo, se realizaron pruebas funcionales en diferentes escenarios, confirmando que el sistema responde adecuadamente ante entradas válidas e inválidas de datos, y que los cálculos se ejecutan conforme a los parámetros establecidos. La retroalimentación obtenida durante las entrevistas y encuestas con los usuarios permitió ajustar detalles de usabilidad y reforzar la lógica de negocio en puntos críticos.

Finalmente, el proyecto demuestra que es posible implementar soluciones tecnológicas efectivas en pequeñas empresas como Nutrimax, sin incurrir en altos costos ni alterar la estructura organizativa. El prototipo desarrollado no solo cumple con los objetivos planteados, sino que también sienta las bases para una transformación digital progresiva en la gestión del talento humano.

Recomendaciones

A partir del desarrollo y evaluación del prototipo funcional para la gestión de recursos humanos en Nutrimax, se plantean una serie de recomendaciones orientadas a su implementación efectiva, mejora continua y evolución futura. En primer lugar, se sugiere implementar el sistema de forma gradual, iniciando con los módulos de mayor impacto operativo como el cálculo de planillas, cálculo de liquidaciones y la gestión de permisos. Esta implementación puede realizarse por medio de una fase piloto que incluya acompañamiento técnico y sesiones de capacitación dirigidas al personal administrativo, idealmente durante el primer mes posterior a la validación del prototipo.

Asimismo, se recomienda establecer un plan de capacitación breve y dirigido, enfocado en el uso del sistema, la interpretación de reportes y la gestión de solicitudes. Para ello, se pueden utilizar manuales interactivos y sesiones prácticas que preparen al personal antes de la puesta en marcha oficial. En una segunda fase, se sugiere integrar el sistema con herramientas contables o de facturación mediante API o servicios web que permitan sincronizar datos financieros, una vez que el sistema haya sido adoptado por todos los usuarios internos y se haya estabilizado su uso.

Es importante también monitorear el uso del sistema y recopilar retroalimentación periódica de los usuarios, lo cual puede lograrse mediante encuestas internas y análisis de logs de actividad, aplicados trimestralmente durante el primer año de operación. En paralelo, se recomienda fortalecer las medidas de seguridad incorporando cifrado de datos sensibles, autenticación multifactor y auditoría de accesos, utilizando servicios disponibles en Microsoft

Azure y buenas prácticas de desarrollo seguro, especialmente antes de migrar el sistema a un entorno productivo en la nube.

Otro aspecto clave es la documentación técnica del sistema, que debe incluir diagramas de arquitectura, flujos de proceso, reglas de negocio y manuales de usuario. Esta documentación puede elaborarse utilizando plataformas de documentación en línea, y debe completarse durante la fase final del desarrollo, previo a la entrega oficial del prototipo.

Finalmente, se recomienda establecer indicadores de desempeño como precisión en cálculos, reducción de errores administrativos y tiempos de respuesta, los cuales pueden medirse mediante métricas automatizadas y reportes mensuales desde el momento en que el sistema entre en operación formal.

Estas recomendaciones buscan asegurar que el sistema no solo cumpla con los objetivos planteados, sino que también se mantenga sostenible, seguro y adaptable a las necesidades futuras de la empresa.

REFERENCIAS

- Álvaro, A. (4 de agosto de 2023). *Bizneo blog*. Obtenido de Cómo elaborar una planilla de pago con éxito. <https://www.bizneo.com/blog/planilla-de-pago/#:~:text=La%20planilla%20de%20pago%20es,seg%C3%BAAn%20su%20tipo%20de%20contrato.>
- Amazon. (01 de 10 de 2023). *Amazon*. <http://www.amazon.com>
- Anandmeg. (8 de mayo de 2023). *¿Qué es Visual Studio?* Microsoft Learn. <https://learn.microsoft.com/es-es/visualstudio/get-started/visual-studio-ide?view=vs-2022>
- Arias, Á. (2 de marzo de 2014). *Computación en la Nube: 2ª Edición*. Obtenido de https://books.google.co.cr/books?id=-0_mCgAAQBAJ&printsec=frontcover&hl=es#v=onepage&q&f=false
- Arias, Á. (s.f.). *Aprende a Programar ASP .NET y C#: 2ª Edición*. <https://books.google.es/books?hl=es&lr=&id=9WnjCgAAQBAJ&oi=fnd&pg=PA1&dq=que+es+asp.net&ots=kiB9-gShd5&sig=cQdve7gCZDva-2W2azJwJkTVYNc#v=onepage&q=que%20es%20asp.net&f=false>
- Assembler Institute. (14 de julio de 2022). *¿Qué es la programación y para qué sirve?* <https://assemblerinstitute.com/blog/que-es-la-programacion/>
- Barrantes Echevarría, R. (2002). *Investigación: un camino al conocimiento, un enfoque cuantitativo y cualitativo*. EUNED.
- BG&A Abogados Corporativos. (26 de Julio de 2023). *BG&A Abogados Corporativos*. <https://bgacorp.com/porcentaje-aporteccss/#:~:text=%C2%BFCu%C3%A1nto%20se%20rebaja%20del%20salario,trabajador%20un%2010%2C67%25.>
- Bigelow, S. (17 de septiembre de 2021). *ComputerWeekly.es*. Sistema operativo. <https://www.computerweekly.com/es/definicion/Sistema-operativo>
- Brismark, A. (s.f.). *Definición de IaaS, PaaS y SaaS ¿En qué se diferencian?* Definición de IaaS, PaaS y SaaS ¿En qué se diferencian? <https://www.ambit-bst.com/blog/definici%C3%B3n-de-iaas-paas-y-saas-en-qu%C3%A9-se-diferencian>
- buhoagenciadigital.com. (18 de julio de 2022). *Software de aplicación: ¿Qué es?* Software de aplicación: ¿Qué es? <https://buhoagenciadigital.com/software-de-aplicacion-que-es/>

- Clickittech, D. (26 de marzo de 2020). *AWS Precios: Como funcionan los costos en la nube de Amazon*. AWS Precios: Como funcionan los costos en la nube de Amazon. <https://www.scribbr.com/citation/generator/folders/4Ie5Uuy9U1Bh9D29mIB1oM/lists/5eDTof9zD101GIomQgL4mI/sources/2BNwz1TvAqKbRXNKyIDkgw/edit/>
- Condor, E. y Soria, I. (10 de octubre de 2014). *Programación Web con CSS, JavaScript, PHP y AJAX*. <https://books.google.es/books?hl=es&lr=&id=QRG-CQAAQBAJ&oi=fnd&pg=PA1&dq=que+es+CSS&ots=Vq0WhTsmYC&sig=22EsqzvxJrRgwK4NG2Mfsl8vZEE#v=onepage&q=que%20es%20CSS&f=false>
- Consejo Nacional de Rectores. (8 de diciembre de 2021). *CONARE. ¿Qué es el salario bruto?*. <https://www.conare.ac.cr/ufaq/que-es-el-salario-bruto/>
- Consejo Nacional de Rectores. (8 de diciembre de 2021). *CONARE. ¿Qué es el salario neto? ~ Consejo Nacional de Rectores*. <https://www.conare.ac.cr/ufaq/que-es-el-salario-neto/>
- Conzultek. (s.f.). *Microsoft Azure: qué es, cómo funciona y cómo ayuda en su empresa*. <https://blog.conzultek.com/microsoft-azure-que-es-como-funciona-como-ayuda-a-las-empresas>
- Coppola, M. (9 de octubre de 2023). *HubSpot*. Qué es HTML y cómo utilizarlo (guía para principiantes). <https://blog.hubspot.es/website/html>
- Digital, I. (2025). *¿Qué es MySQL Workbench y para qué sirve?* <https://informatecdigital.com/que-es-mysql-workbench/>
- Figueiras, S. (27 de julio de 2021). *Diferencias entre hardware y software*. Diferencias entre hardware y software. <https://www.ceupe.mx/blog/diferencias-entre-hardware-y-software.html>
- Galindo, J. y Camps, J. (14 de enero de 2008). *Diseño e implementación de un marco de*. <https://openaccess.uoc.edu/bitstream/10609/876/1/00765tfc.pdf>
- Gallay, R. (21 de septiembre de 2023). *TUTFG*. Recuperado el 10 de octubre de 2023, de Fuentes de información: primarias y secundarias. <https://tutfg.es/fuentes-primarias-y-secundarias/>
- García, I. (2023 de diciembre de 2021). *Qué es Modelo-Vista-Controlador*. Qué es Modelo-Vista-Controlador. <https://carontestudio.com/blog/que-es-modelo-vista-controlador/>
- Gómez, Á. P. (diciembre de 2017). *Fundamentos sobre la Gestión de Base de Datos*. <http://repositorio.unesum.edu.ec/bitstream/53000/2077/1/Fundamento%20sobre%20la%20Gestio%20de%20bases%20de%20datos.pdf>

- Gómez, M. (11 de agosto de 2023). *HubSpot*. Qué es una encuesta, para qué sirve y qué tipos existen. <https://blog.hubspot.es/service/que-es-una-encuesta>
- Gonzalez, R. (26 de octubre de 2023). *SignificadosWeb.com*. Definición de JavaScript: que es, 5 ejemplos, tipos y para que sirve + sinónimo y significado. https://significadosweb.com/definicion-de-javascript-que-es-ejemplos-tipos-y-para-que-sirve-sinonimo-y-significado/#google_vignette
- Grapsas, T. (12 de febrero de 2021). *Rockcontent*. ¿Qué es cloud computing o computación en la nube? Conoce sobre el término a continuación. <https://rockcontent.com/es/blog/computacion-en-la-nube/>
- Gutierrez J., J. (2006). *¿Qué es un framework web?* http://www.lsi.us.es/~javierj/investigacion_ficheros/Framework.pdf
- Herebero, C. (2004). *Informática y comunicaciones en la empresa*. <https://books.google.com.co/books?id=U0MXWtqjxtsC&printsec=frontcover#v=onepage&q&f=false>
- Hernández Sampieri, R., Fernández Collado, C. y Baptista Lucio, P. (2014). *Metodología de la Investigación Sexta edición*. McGRAW-HILL.
- Hernandez, R. (28 de junio de 2021). *freeCodeCamp.org*. El patrón modelo-vista-controlador: Arquitectura y frameworks explicados. <https://www.freecodecamp.org/espanol/news/el-modelo-de-arquitectura-view-controller-pattern/>
- Hernández, S. (11 de diciembre de 2020). *zademy*. ¿Qué es el patrón de diseño MVC?. <https://zademy.com/publicacion/6/Qu%C3%A9-es-el-patr%C3%B3n-de-dise%C3%B1o-MVC->
- Hosting Plus. (6 de julio de 2021). *Modelo de prototipos: ¿qué es y cuáles son sus etapas?* Modelo de prototipos: ¿qué es y cuáles son sus etapas?
- Hostinger tutoriales. (11 de enero de 2023). *¿Qué es Bootstrap? – Una guía para principiantes*. <https://www.hostinger.es/tutoriales/que-es-bootstrap>
- IBM. (s.f.). *¿Qué es la prueba de software?* <https://www.ibm.com/es-es/topics/software-testing>
- Informatec Digital. (s.f.). *¿Qué es MySQL Workbench y para qué sirve?* <https://informatecdigital.com/que-es-mysql-workbench/>

- Innovación Digital 360. (s.f.). *Guía completa de MySQL Workbench: descarga, instalación y uso*.
<https://www.innovaciondigital360.com/big-data/guia-completa-de-mysql-workbench-descarga-instalacion-y-uso/>
- Joyanes, L. (2006). *Programación en C++. Algoritmos, estructuras*.
<http://biblioteca.univalle.edu.ni/files/original/ec89128132c7434f1765ceb9cbf1160828ae85f5.pdf>
- jQuery.com. (s.f.). *¿Qué es jQuery?* e <http://jquery.com/>
- Leandro, A. (1 de agosto de 2023). *Definición de aplicación web*. Alegsa.com.ar.
https://www.alegsa.com.ar/Dic/aplicacion_web.php#gsc.tab=0
- Londoño, P. (21 de enero de 2023). *HubSpot*. Obtenido de Qué son las aplicaciones web y 8 ejemplos. <https://blog.hubspot.es/website/que-es-aplicacion-web>
- Luján Mora, S. (31 de octubre de 2002). *Programación de aplicaciones web: historia, principios básicos y clientes web*. https://rua.ua.es/dspace/bitstream/10045/16995/1/sergio_lujan-programacion_de_aplicaciones_web.pdf
- Maida, Pacienza, E. y Julián. (Diciembre de 2015). *Metodologías de desarrollo de software*.
<https://repositorio.uca.edu.ar/bitstream/123456789/522/1/metodologias-desarrollo-software.pdf>
- Marker, G. (12 de noviembre de 2022). *Tecnología + Informática*. ¿Qué es un programa de computadora?.
https://www.tecnologia-informatica.com/que-es-programa-computadora/#%C2%BFQu%C3%A9_es_un_programa_de_computadora?
- Marker, G. (9 de mayo de 2023). *Tecnología + Informática*. Qué es una computadora?.
<https://www.tecnologia-informatica.com/que-es-una-computadora/>
- Martín, D. (31 de julio de 2023). *OpenHR*. Recursos Humanos: definición, concepto y funciones.
<https://www.openhr.cloud/blog/recursos-humanos-funciones-dentro-de-la-empresa#que>
- Martins, J. (19 de junio de 2023). *Scrum: conceptos clave y cómo se aplica en la gestión de proyectos*. <https://asana.com/es/resources/what-is-scrum>
- Microsoft. (s.f.). *¿Qué es .NET?* Obtenido de <https://dotnet.microsoft.com/es-es/learn/dotnet/what-is-dotnet>
- Microsoft. (15 de febrero de 2023). *Paseo por el lenguaje C#*. <https://learn.microsoft.com/es-es/dotnet/csharp/tour-of-csharp/>

- Microsoft Learn. (2023 de agosto de 2023). *Descarga de SQL Server Management Studio (SSMS)*.
<https://learn.microsoft.com/es-es/sql/ssms/download-sql-server-management-studio-ssms?view=sql-server-ver16>
- Microsoft.com. (s.f.). *C#. El lenguaje de programación moderno, innovador y de código abierto para crear todas sus aplicaciones*. <https://dotnet.microsoft.com/es-es/languages/csharp>
- Ministerio de Trabajo y Seguridad Social [MTSS]. (s.f.). Asuntos laborales.
<https://www.mtss.go.cr/temas-laborales/index.html>
- Ministerio de Trabajo y Seguridad Social [MTSS]. (s.f.). *Preaviso y Cesantía*.
<https://escuelajudicialpj.poder-judicial.go.cr/Archivos/bibliotecaVirtual/otrasPublicaciones/brochures/Preaviso%20y%20Cesantia.pdf>
- Morales, F. (1 de marzo de 2021). *Economipedia*. Fuente primaria.
<https://economipedia.com/definiciones/fuente-primaria.html>
- Moreno Galindo, E. (9 de marzo de 2018). *Metodología de investigación, pautas para hacer Tesis*.
<https://tesis-investigacion-cientifica.blogspot.com>
- Mugira, A. (31 de agosto de 2023). *QuestionPro*. ¿Qué es una entrevista? Todo lo que debes saber al respecto. <https://www.questionpro.com/blog/es/tecnicas-de-recoleccion-de-datos-entrevista/>
- Northware. (26 de MAYO de 2022). *Requerimientos en el desarrollo de software y aplicaciones*.
<https://www.northware.mx/blog/requerimientos-en-el-desarrollo-de-software-y-aplicaciones/>
- Norton, P. (2006). *Introducción a la computación*.
<https://alkedua.files.wordpress.com/2015/07/218982559-introduccion-a-la-computacion-peter-norton.pdf>
- Organizadores Gráficos. (14 de octubre de 2022). *Población y muestra de investigación (Definición, Proceso, técnicas y Fórmulas)*.
<https://www.organizadoresgraficos.org/poblacion-y-muestra-de-investigacion/#respond>
- Ortega, C. (23 de febrero de 2023). *QuestionPro*. 5 instrumentos para recopilar información.
<https://www.questionpro.com/blog/es/instrumentos-para-recopilar-informacion/>
- Ortega, C. (16 de junio de 2023). *QuestionPro*. ¿Qué es un cuestionario?.
<https://www.questionpro.com/blog/es/que-es-un-cuestionario/>

- Parada, M. (14 de abril de 2023). *OpenWebinars*. Qué es SQL Server. <https://openwebinars.net/blog/que-es-sql-server/>
- Pérez, M. (2011). *Microsoft SQL Server 2008 R2. Motor de base de datos y administración*. <https://books.google.co.cr/books?id=ExK0AQRjPk4C&printsec=frontcover&hl=es#v=onepage&q=2011&f=false>
- Pezua, S. (26 de febrero de 2023). *Steven Lizarzaburu Pezúa*. ¿Qué es el Patrón Modelo-Vista-Controlador (MVC)? <https://stevenlizarzaburupezua.com/que-es-el-patron-modelo-vista-controlador-mvc/>
- Proyectos Ágiles. (20 de septiembre de 2021). *Atlassian*. Proyectos Ágiles. <https://proyectosagiles.org/que-es-scrum/>
- Revista Escuela de Negocios y Dirección. (18 de octubre de 2022). *¿Qué es scrum? Conoce el framework que agiliza el trabajo en equipo*. <https://www.escueladenegociosydireccion.com/revista/business/scrum-framework-agiliza-trabajo-equipo/>
- Rodríguez, G. (28 de agosto de 2023). *Azul School*. ASP.NET: ¿Qué es y para qué sirve? - Azul School. <https://www.azulschool.net/asp-net-que-es-y-para-que-sirve/>
- Ruiz Jiménez, K. (9 de septiembre de 2022). *Aselecom Abogados*. ASPECTOS ESENCIALES SOBRE EL PREAVISO EN MATERIA LABORAL. <https://aselecom.com/aspectos-esenciales-sobre-el-preaviso-en-materia-laboral/>
- Saavedra, J. (01 de Junio de 2023). *¿Qué es AJAX y cómo se utiliza?* <https://ebac.mx/blog/que-es-ajax>
- Sánchez, M. (2012). *JavaScript*. <https://books.google.co.cr/books?id=3x09sewjaHIC&printsec=frontcover#v=onepage&q&f=false>
- Santos, D. (28 de Julio de 2023). *Hubspot*. Introducción al CSS: qué es, para qué sirve y otras 10 preguntas frecuentes. <https://blog.hubspot.es/website/que-es-css>
- Sommerville, I. (2011). *Ingeniería de Software*. https://gc.scalahed.com/recursos/files/r161r/w25469w/ingdelsoftwarelibro9_compressed.pdf
- Tesis y Másters. (9 de octubre de 2023). *Tesis y Másters*. ¿Qué son los instrumentos de recolección de datos?

- <https://www.scribbr.es/citar/generador/folders/1DVkhDh5ImoOYFoLdoGv16/lists/1Phn3AGi3eCLWtmrCVKL5N/fuentes/6oLassjhtKNWpbuwl6VJbf/editar/>
- Tocci, R. y Widmer, N. (2003). *Sistemas Digitales*.
https://books.google.co.cr/books?id=bmLuH0CsIh0C&printsec=frontcover&hl=es&source=gbs_ge_summary_r&cad=0#v=onepage&q&f=false
- Toptrabajos. (5 de diciembre de 2022). *Toptrabajos*. Horas extras en Costa Rica. calcula tu pago fácilmente. <https://www.toptrabajos.com/blog/cr/calcular-pago-de-horas-extras/>
- Toptrabajos. (7 de diciembre de 2022). *Toptrabajos*. ¿Cómo calcular la liquidación de contrato en Costa Rica? [Paso a paso]. <https://www.toptrabajos.com/blog/cr/liquidacion-costa-rica/>
- Ungoti, S. (24 de octubre de 2021). *Ciclo de vida del desarrollo de software (SDLC) explicado - Ungoti*. Ciclo de vida del desarrollo de software (SDLC) explicado -Ungoti. <https://ungoti.com/es/soluciones/desarrollo-de-software/sdlc/>
- Universidad de Guadalajara. (s.f.). *Clasificación general de las fuentes de información*. <http://biblioteca.udgvirtual.udg.mx/portal/clasificacion-general-de-las-fuentes-de-informacion#:~:text=Fuentes%20terciarias%3A%20son%20gu%C3%ADas%20f%C3%ADas,de%20referencia%20de%20una%20biblioteca.>

APÉNDICES

Apéndice A – Encuesta aplicada a colaboradores

Instrumento: Cuestionario de percepción sobre procesos de Recursos Humanos
Objetivo: Recopilar información sobre la percepción de los colaboradores respecto a la eficiencia de los procesos actuales en el área de recursos humanos.

1. ¿A qué departamento pertenece dentro de la empresa?

- Contabilidad
- Ventas
- Recursos Humanos
- Tecnología / Sistemas
- Otro

2. ¿Cuánto tiempo lleva laborando en la empresa?

- Menos de 1 año
- De 1 a 3 años
- De 3 a 6 años
- Más de 6 años

3. ¿Cómo califica la eficiencia de los procesos actuales del área de recursos humanos?

- Muy eficiente
- Eficiente
- Normal
- Ineficiente
- Muy ineficiente

4. ¿Cómo califica el proceso actual de cálculo de salarios?

- Muy eficiente
- Eficiente
- Normal
- Ineficiente
- Muy ineficiente

5. ¿De qué forma se gestionan actualmente las solicitudes de permisos?

- Hojas de Excel
- Formularios en papel
- Otro

6. ¿Cómo califica el proceso actual de manejo de permisos?

- Muy eficiente
- Eficiente
- Normal
- Ineficiente
- Muy ineficiente

7. ¿Cuál área considera que requiere mayor mejora dentro de los procesos de recursos humanos?

- Gestión de planilla
- Gestión de solicitudes
- Gestión de vacaciones
- Gestión de liquidaciones

() Otro

Apéndice B – Entrevista aplicada a la encargada de planillas

Instrumento: Entrevista

Objetivo: Obtener información detallada sobre los procesos actuales de gestión de planillas y recursos humanos desde la perspectiva de la encargada del área.

Preguntas:

1. ¿Podría detallar el rol y las responsabilidades como encargada de planillas de la Macrobiótica?
2. ¿Posee experiencia en la evaluación y generación de planillas?
3. ¿Podría detallar el proceso actual para el cálculo de planilla de los trabajadores de la macrobiótica?
4. ¿Podría detallar cómo se realiza el cálculo de horas extra de los empleados?
5. ¿Puede hablarme acerca del proceso para verificar la cantidad de días de vacaciones que tiene cada empleado?
6. ¿Cómo calcula el monto que tiene que pagar de aguinaldo a finalizar el año?
7. ¿Ha sufrido manipulación o extravío de los documentos formales donde se lleva el control de estas horas extra?
8. ¿Ha sufrido atrasos al momento de generar los pagos correspondientes de los empleados?
9. ¿Cuáles son los desafíos más importantes que afronta el departamento de recursos humanos?