

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

**PROYECTO DE GRADUACIÓN
PARA OPTAR POR EL GRADO DE BACHILLERATO EN INGENIERIA EN
SISTEMAS DE INFORMACION**

**PROTOTIPO FUNCIONAL PARA LA GESTIÓN DEL RECURSO HUMANO PARA LA
TIENDA GIRASOL, UBICADA EN DESAMPARADOS**

**JOSELYN RAQUEL ALVARADO REYES
AUTOR**

**LIC. FERNANDO RIOS VARGAS
TUTOR**

Sede Central

JULIO, 2025

DEDICATORIA

Dedico este logro, con profundidad gratitud y amor, a Dios por ser mi guía durante todo el proceso, por darme la fuerza necesaria para superar cada reto y sabiduría para tomar decisiones correctas. Gracias por iluminar mi camino incluso en los momentos más oscuros y llenos de incertidumbre, recordándome siempre que la fe y la perseverancia abren puertas.

A mi familia, mi mayor motivación y refugio en cada etapa de este recorrido. A mis padres por su amor incondicional, sus sacrificios silenciosos y su ejemplo de lucha y entrega. Sus palabras de aliento, que me han sostenido aun en los días más difíciles y sus enseñanzas han sido la base sobre la cual he construido este logro.

A mis amigos, por su apoyo sincero, por sus palabras de ánimo y por compartir conmigo tanto las risas como las preocupaciones. Gracias por estar presentes en cada momento importante, por celebrar cada paso adelante y alentarme a seguir aun cuando el camino parecía incierto.

A mis profesores, por compartir su conocimiento con sinceridad, por su guía constante y por motivarme a dar siempre lo mejor de mí. Gracias por sembrar en mi la pasión por aprender, crecer y superarme día a día.

A mi directora, por su paciencia, dedicación y orientación a lo largo de este proceso. Gracias por confiar en mi trabajo, por escucharme y por guiarme con compromiso hacia la meta final.

Y a mí mismo, por no rendirme cuando las dificultades parecían superar mis fuerzas, por resistir las dudas y por creer que todo esfuerzo tiene su recompensa.

Este trabajo representa más que un requisito académico: es el cierre de un ciclo importante de mi vida y el inicio de nuevos sueños y metas que me impulsa a seguir adelante.

AGRADECIMIENTOS

Quiero expresar mi más sincero agradecimiento a todas las personas que, de una u otra manera, hicieron posible la realización de este trabajo, cada una aportando un grano de arena que ha sido fundamental en este camino.

A Dios, por darme salud, fuerza y sabiduría necesarias en cada paso del camino. Gracias por guiarme incluso en los momentos de claridad y, sobre todo, en los más difíciles, cuando la fe y la perseverancia fueron mis únicas herramientas para seguir adelante.

A mi familia, por ser mi pilar, por estar siempre ahí, por su amor incondicional. Por su amor sin límites, por los sacrificios que muchas veces hicieron en silencio y por creer en mi incluso en esos momentos en los que yo misma dudaba. Ustedes me han enseñado que el verdadero apoyo es el que se da sin condiciones y por eso este logro también les pertenece.

A mis profesores, por compartir de su conocimiento con paciencia y dedicación, por sus enseñanzas y por su acompañamiento constante durante este proceso formativo. Su guía ha sido clave no solo para el desarrollo de este trabajo, sino también para mi crecimiento personal y profesional.

A mi tutor, por su orientación, paciencia y su compromiso para guiarme en cada etapa de este proyecto. Brindándome siempre valiosos consejos para mejorar y avanzar con confianza.

A mi directora, por su apoyo, compromiso con mi proceso académico y por la confianza en mi capacidad para llevar este proyecto a buen término. Su guía y su visión fueron un faro en los momentos de duda

A mis amigos, por estar presentes en cada paso, por acompañarme en los momentos de alegría y también en los de agotamiento, por escucharme cuando lo necesitaba y por recordarme que nunca estoy sola. Su compañía y sus palabras de aliento fueron un impulso invaluable.

Y a todas aquellas personas que aportaron con una palabra de aliento, un consejo o un gesto de apoyo cuando más lo necesitaba. Gracias por ser parte de este capítulo tan importante en mi vida.

Contenido

Tablas.....	7
Figuras.....	9
Resumen Ejecutivo	11
CAPÍTULO I: INTRODUCCIÓN.....	12
Objetivo General	13
Objetivos específicos	13
Justificación.....	13
Viabilidad técnica	13
Viabilidad operativa.....	15
Viabilidad económica	16
Viabilidad legal.....	17
Proyecciones.....	18
Alcance funcional	18
Alcance metodológico	19
Alcance tecnológico.....	20
CAPÍTULO II: MARCO REFERENCIAL	22
CAPÍTULO III: MARCO METODOLÓGICO.....	38
Enfoques de la Investigación.....	38
Enfoque Cualitativo	39
Enfoque de Investigación Seleccionado.	39
Tipos de Investigación	40
Investigación Descriptiva.....	40

Tipo de Investigación Seleccionado	41
Fuentes de Información.....	41
Fuentes de Información Primarias	42
Fuentes de Información Secundarias	42
Fuentes de Información Terciarias.....	43
Variables.....	43
Variable conceptual	44
Variable operacional	44
Variable instrumental.....	44
Instrumentos de recolección de datos.....	46
Proceso de recolección y análisis de datos.....	48
CAPÍTULO IV: ANÁLISIS DE RESULTADOS	49
Entrevista.....	49
Observación Directa.....	51
CAPÍTULO V: CONCLUSIONES Y RECOMENDACIONES.....	53
Conclusiones	53
Recomendaciones.....	54
CAPITULO VI: PROPUESTA.....	56
Análisis Detallado del Software por Desarrollar.....	56
Análisis Detallado del Hardware.....	59
Hardware para el desarrollo del prototipo	59
Hardware para la producción del prototipo	60
Análisis Detallado de las Telecomunicaciones	60
Descripción detallada de las herramientas técnicas	62
Descripción Detallada del Conocimiento Básico para Operar el Sistema	63

Elaboración de los Casos de Uso	66
Diseño.....	77
Arquitectura del sistema	77
Arquitectura del software.....	78
Diseño de entradas	80
Diagrama Entidad-Relación	84
Diccionario de Datos	85
Diseño de Procesos.....	90
Diseño de Salidas	98
Diagramas UML.....	102
PROGRAMACIÓN	109
PRUEBAS.....	120
Referencias.....	125
Apéndice A	128
GUÍA DE ENTREVISTA.....	128
Apéndice B.....	129
GUÍA DE OBSERVACIÓN	129

Tablas

Tabla 1	14
Tabla 2	15
Tabla 3	16
Tabla 4	45
Tabla 5	66
Tabla 6	67
Tabla 7	68
Tabla 8	69
Tabla 9	70
Tabla 10	71
Tabla 11	72
Tabla 12	73
Tabla 13	74
Tabla 14	75
Tabla 15	76
Tabla 16	85
Tabla 17	85
Tabla 18	86
Tabla 19	86
Tabla 20	86
Tabla 21	86
Tabla 22	87
Tabla 23	87
Tabla 24	87

Tabla 25	87
Tabla 26	88
Tabla 27	88
Tabla 28	88
Tabla 29	88
Tabla 30	89
Tabla 31	89
Tabla 32	89
Tabla 33	89
Tabla 34	90
Tabla 35	90
Tabla 36	90
Tabla 35	120
Tabla 36	121
Tabla 37	122
Tabla 38	123
Tabla 39	124

Figuras

Figura 1	61
Figura 2	¡Error! Marcador no definido.
Figura 3	77
Figura 4	79
Figura 5	80
Figura 6	81
Figura 7	82
Figura 8	83
Figura 9	83
Figura 10	84
Figura 11	91
Figura 12	92
Figura 13	93
Figura 14	94
Figura 15	95
Figura 16	96
Figura 17	97
Figura 18	98
Figura 19	99
Figura 20	99
Figura 21	100
Figura 22	101
Figura 23	102
Figura 24	103

Figura 25	103
Figura 26	104
Figura 27	105
Figura 28	106
Figura 29	107
Figura 30	108
Figura 31	109
Figura 32	110
Figura 33	111
Figura 34	112
Figura 35	113
Figura 36	114
Figura 37	115
Figura 38	115
Figura 39	116
Figura 40	116
Figura 41	117
Figura 42	118
Figura 43	119

Resumen Ejecutivo

Este documento tiene como objetivo resolver la problemática que enfrenta el Departamento de Recursos Humanos de la Tienda Girasol, ubicada en Desamparados, mediante la automatización de los procesos relacionados con la gestión del talento humano. Actualmente, existe un descontrol administrativo, ya que la mayoría de las gestiones se realizan manualmente, lo que provoca demoras y errores.

El objetivo general de esta investigación es desarrollar un prototipo funcional para la gestión de Recursos Humanos en la tienda Girasol, que permita cubrir las necesidades de la entidad. En caso de que el prototipo cumpla con las expectativas, la tienda será la encargada de realizar el procedimiento correspondiente para su implementación.

Este prototipo se centrará en optimizar procesos clave como la elaboración de planillas, control de asistencia, liquidaciones, registro de incapacidades, gestionar las vacaciones y control de permisos.

Finalmente, el desarrollo del prototipo se realizará en función de los requerimientos expuestos por el personal de la entidad, quienes han señalado, de forma detallada, la problemática que actualmente enfrentan.

CAPÍTULO I: INTRODUCCIÓN

La tienda “Girasol” ubicada en Desamparados, tiene más de 20 años de estar en el mercado. Actualmente, cuenta con 10 empleados y uno de ellos se encarga de distribuir el trabajo del día a día, para la atención a clientes. En el local se venden una gran variedad de productos: desde artículos de cocina y útiles escolares, hasta electrodomésticos, juguetes, flores, verduras y frutas.

A pesar del tiempo que lleva en el mercado, la tienda no cuenta con el sistema de planilla. Lo cual ha provocado consecuencias y que el pago, la gestión de la planilla y del personal sean poco eficientes. El responsable de realizar el pago de salarios, cálculo de horas extra, vacaciones, aguinaldos y la gestión de permisos lo realiza manualmente, lo que requiere mucho tiempo y esfuerzo de parte de la empresa.

Esta situación genera insatisfacción entre los trabajadores, ya que en ocasiones hay atrasos en los pagos o se hacen con montos equivocados o fechas incorrectas y surgen a raíz de errores de naturaleza humana.

Algunos problemas encontrados, según la investigación preliminar:

- Se genera atrasos y errores en los cálculos de planilla. Actualmente los cálculos para realizar el pago de cada trabajador se generan de forma manual, con cálculos en Excel y esto puede generar errores al realizar el pago, disconformidad y retraso hacia los trabajadores.
- Insuficiencia en registro de vacaciones disponibles para cada colaborador. Actualmente, las vacaciones se calculan o se asignan de forma manual, no se lleva ningún registro de los días vacacionados de cada trabajador y esto puede generar errores a la hora de pagar sus respectivas vacaciones y disconformidad hacia los trabajadores.
- Dificultad en el registro y pagos de horas extras. Actualmente, el proceso de control, registro y pagos de horas extras se lleva de una manera manual y se han presentado múltiples errores en los cálculos al pagar horas extras, por lo que ocasiona gran molestia y reclamos de trabajadores afectados.
- Insuficiencia en registro de cada una de las incapacidades de cada colaborador. Se autoriza que el colaborador tenga permisos, pero no se lleva un registro de cuáles permisos ha solicitado cada colaborador.
- Se genera errores al realizar la sumatoria de salarios. Actualmente, para calcular el aguinaldo de cada trabajador se genera de forma manual, con cálculos en Excel y esto

puede generar errores al realizar la suma de los salarios, disconformidad y retrasos hacia los trabajadores.

Objetivo General

Desarrollar un prototipo funcional para la Gestión del Recurso Humano para la tienda Girasol, ubicada en San José, Desamparados

Objetivos específicos

- Analizar los requerimientos que se van a incorporar en el prototipo funcional.
- Elaborar el diseño del prototipo de acuerdo con los resultados del análisis.
- Desarrollar la programación del prototipo, tomando como base, el diseño, previamente creado.
- Ejecutar las pruebas funcionales necesarias en el prototipo final.

Justificación

El objetivo principal del proyecto es mejorar la forma de trabajar de la planilla y registros de la empresa, ya que, actualmente, no cuentan un sistema de gestión de recursos humanos.

El propósito del prototipo es facilitar tareas como el cálculo de salarios, aguinaldos y horas extras, reduciendo al mismo tiempo los errores humanos que se suelen presentar en estos procesos.

Esto ayudará al responsable, a realizar los cálculos de manera eficiente porque tendrá a su disposición una herramienta que le facilitará su labor y optimizará tu tiempo.

Viabilidad técnica

El proyecto de desarrollo del prototipo de sistema de gestión de Recursos Humanos para la tienda “Girasol” es técnicamente viable, por los siguientes motivos:

Se utilizará ASP.NET en un entorno Modelo Vista Controlador (MVC), utilizando el lenguaje de programación C# para desarrollar un sistema robusto con alta flexibilidad y eficiente. La elección de esta tecnología se basa en que el sistema puede ejecutarse, sin necesidad de una computadora de alta tecnología, lo que lo hace más accesible y adaptable a distintos entornos.

El Entorno de Desarrollo Integrado (IDE), seleccionado será Visual Studio 2022, ya que proporciona una amplia gama de herramientas que se adaptan al desarrollo del prototipo funcional. Con este entorno de desarrollo avanzado se busca mejorar la eficiencia del proyecto, permitiendo incorporar de características avanzadas y un mejor rendimiento del sistema.

Para la gestión de base de datos se utilizará Microsoft SQL Server Management Studio 2022, es una herramienta eficaz que ofrece una interfaz intuitiva y amigable, lo que facilita las tareas relacionadas con el manejo de datos.

Es importante destacar que el sistema será programado en C#, un lenguaje altamente recomendado, debido a sus características y capacidades, que permiten el desarrollo eficiente, estructurado y sostenible del sistema.

El área de licencias se enfocará en desarrollar prototipos que utilizarán licencias gratuitas y complementándose con herramientas código abierto, disponible públicamente. Este plan se adopta con el propósito de evitar costos adicionales, minimizar costos asociados para la empresa y garantizar el cumplimiento.

Tomar este enfoque no solo favorece la eficiencia financiera, sino que también proporciona una base sólida para la entrega exitosa del proyecto y garantiza la confiabilidad de las herramientas utilizadas. Además, esta elección se alinea con el principio de construir sobre tecnologías establecidas y respaldadas por la comunidad, esto aumenta la robustez y la estabilidad del proyecto.

Tabla 1

Herramientas de Hardware

Recursos	Especificaciones
Laptop	Computadora Dell
Pantalla	Pantalla de 15.6" FHD
Procesador	Procesador Core i5
Memoria RAM	16 GB DDR4
Almacenamiento	Unidad en estado sólido de 1 TB
Sistema Operativo	Windows 10 Home 64

Fuente: Elaboración propia, 2025

En la Tabla 1 se observan los componentes de hardware que se van a utilizar para el desarrollo del prototipo.

Tabla 2

Herramientas para Software

Herramientas	Especificaciones
IDE de desarrollo	Visual Studio 2022
Motor Base de Datos	Microsoft SQL Server Management Studio 2022
Lenguaje de Programación	C# version 13

Fuente: elaboración propia, 2025

En la Tabla 2 se observan los componentes de software que se van a utilizar para el desarrollo del prototipo.

Viabilidad operativa

Hace referencia a la efectividad del prototipo una vez sea utilizado por los responsables de determinado departamento de la empresa y la capacidad de utilizarlo adecuadamente. Se mencionan ciertos puntos para confirmar la viabilidad.

- El personal cuenta con el conocimiento básico y habilidades necesarias para utilizar y manejar, de forma efectiva, el prototipo funcional. Además, cuentan con la capacidad de interactuar con las herramientas específicas que trae el sistema.
- El personal posee la capacidad necesaria para llevar a cabo las tareas asignadas, sin embargo, se proporcionará un manual básico que explicará el uso del prototipo funcional, con el fin de asegurar una correcta implementación y aprovechamiento del sistema.
- El prototipo funcional será utilizado, por parte de todos los colaboradores de la empresa, incluyendo tanto a los administradores, como a los empleados.
- El prototipo funcional no provocará reducción de personal, dado que nadie desempeñaba esta función.
- Este sistema provocará beneficios significativos para la empresa, permitiendo un mejor control del personal, mayor eficiencia en las tareas diarias y una optimización en el control administrativo.

Viabilidad económica

La empresa donde se desarrollará el prototipo, no tendrá que usar costos habituales relacionados con software o hardware. El software se construirá con licencias gratuitas y tecnologías de código abierto.

Con relación al hardware se utilizará el que posee la empresa actualmente, el cual, según sus especificaciones actuales, es suficiente para ejecutar el sistema de una manera óptima.

Es importante recalcar que la mano de obra asociada a este proyecto, realizado con fines académicos para obtener el grado de bachillerato en Ingeniería en Sistemas de Información, no implica ningún costo.

Según la información recopilada, el costo de trabajar en el proyecto será de ₡15 983,96 colones diarios, siendo este el costo de un programador informático, según el Ministerio de Trabajo en 2025.

A continuación, se muestra una tabla que detalla los costos que incurrirá en el proyecto.

Tabla 3

Costo del desarrollo del sistema

Proceso	Tiempo en días	Total
Análisis de requerimientos	15	₡ 234 208,65
Creación del diseño	20	₡ 312 278,2
Programación del sistema	35	₡ 546 486,85
Pruebas de funcionamiento	5	₡ 78 069,55
Total	70	₡ 1 171 043,25

Fuente: elaboración propia, 2025

En la tabla 3 se puede observar que el proyecto tendrá una duración aproximada de 70 días, distribuidos en cuatro fases para el desarrollo de la aplicación, cada fase cuenta con una duración y precio estimado de acuerdo a los costos de mano de obra del desarrollador, estableciendo el precio de la mano de obra para el proyecto es de ₡1 171 043,25, es importante destacar que este monto es aproximado y está sujeto a cambios, sin embargo, en el caso de este proyecto, este monto no será cobrado, ya que se trata de una investigación con fines académicos para obtener el grado de bachillerato en ingeniería en sistemas de información.

Viabilidad legal

A continuación, se detallan una serie de normas legales que son importantes para el desarrollo de este proyecto

- Según la Ley 8148, en los artículos 196 bis., 217 bis., 229 bis, la obtención de datos personales de un tercero sin su consentimiento, así como la alteración de dichos datos, su utilización para fines no especificados, su aprovechamiento para beneficio personal individual o la revelación de dicha información se consideran actos ilícitos y conlleva consecuencias legales. Esta infracción se encuentra sujeta a la legislación vigente, la cual establece sanciones penales para prevenir que no sean alterados y que no comprometan la privacidad y seguridad de los individuos.

Esta ley, para proteger la integridad de la información personal, establece medidas disciplinarias para asegurar que los datos no sean objeto de alteraciones, divulgaciones indebidas o manipulación por parte de terceros. Estas sanciones cumplen una función crucial en el mantenimiento de la confidencialidad y la protección de la privacidad de los individuos, contribuyendo así, a un entorno digital seguro y ético.

- La Ley 6683 de 1982, de Derechos de Autor, la Asamblea Legislativa de la República de Costa Rica.

Establece un compromiso firme, con respecto a la protección de las obras y creaciones intelectuales de sus creadores originales. La finalidad primordial es prevenir el robo y la apropiación no autorizada de la propiedad intelectual del autor que dio origen a la obra.

Esta ley, se fundamenta en el resguardo de la propiedad intelectual, busca proteger los derechos y reconocer el mérito de los creadores, promoviendo así un entorno propicio para la innovación y la expresión creativa. Esta normativa está diseñada para preservar la integridad de las obras y fomentar un marco legal que respete y proteja la labor de los artistas y creadores.

- La Ley 8968, sobre la protección de las personas físicas contra el tratamiento de sus datos personales.

Con el objetivo primordial de proteger la información de todos los residentes en Costa Rica, esta ley establece ciertas normas y procesos que todas las organizaciones dentro del país deben cumplir, con respecto a la información de terceros.

Sobre los principios fundamentales de protección e integridad de los datos, la ley reconoce y garantiza a los usuarios el derecho de modificar la información personal que les ha sido

proporcionada, Además, concede a los individuos la facultad de solicitar la eliminación de dicha información. Esta disposición, respaldada por la legislación, refleja el compromiso con la autonomía y el control que los individuos deben tener sobre sus datos personales en un entorno cada vez más digitalizado.

Proyecciones

El sistema tiene como objetivo optimizar los procesos relacionados con el pago de salarios a los empleados de la empresa. En el pasado se identificaron problemas relacionados a este ámbito, entre ellos el pago incorrecto de los montos en la planilla, así como deficiencia en módulos para la gestión de permisos, vacaciones,

Además, se incorporará un módulo de asistencia que permitirá llevar un mejor control del registro de cada colaborador.

Esto conlleva una serie de beneficios tales como la agilización de procesos, la reducción de errores humanos en los cálculos, la disminución de la carga del personal encargado y la facilitación del trabajo para quien administre estas labores.

Alcance funcional

El sistema podrá realizar diversos procesos que facilitarán, significativamente, el cálculo y gestión del recurso humano. Los módulos se describirán a continuación.

- **Generar planilla:** Este módulo se encargará de calcular el pago correspondiente para cada uno de los colaboradores, tomando en cuenta sus respectivas deducciones (C.C.S.S), vacaciones, horas extras.
- **Calcular vacaciones:** Este módulo se encargará de verificar las vacaciones de cada uno de los colaboradores, a partir de su primer día de ingreso y los días que tiene disponibles, el colaborador deberá enviar una solicitud a la jefatura inmediata, este se encargará de enviar la solicitud a recursos humanos y este se encargará de aceptar o rechazar dicha solicitud, también se le notificará al colaborador.
- **Calcular horas extras:** Este módulo se encargará de gestionar el cálculo del monto a pagar por cada hora extra realizada por el colaborador, este deberá enviar una solicitud sobre las horas extras trabajadas a la jefatura inmediata, la cual se encargará de enviar la solicitud a Recursos Humanos para aceptar o rechazar dicha solicitud. También se le notificará al

colaborador, según la decisión de la jefatura, si se ejecuta el proceso de calcular horas extras.

- Control permisos: Este módulo se encargará de llevar un control de los permisos solicitados por cada colaborador, quien envía una solicitud a la jefatura inmediata y esta se encarga de aprobar o rechazar dicha solicitud y se le notificará al colaborador.
- Calcular deducciones: Este módulo se encargará de calcular las deducciones de cada colaborador, tomando en cuenta C.C.S.S y I.V.M y así se podrá calcular el monto correspondiente, sin que nadie salga perjudicado.
- Calcular aguinaldo: Este módulo se encargará de calcular la sumatoria de los salarios para cada uno de los empleados, con su respectivo puesto y desglose de este, que va desde el 1° diciembre del año anterior al 30 de noviembre del año siguiente y así, realizar el cálculo de manera correcta.
- Calcular liquidación: Este módulo se encargará de calcular la liquidación correspondiente a cada uno de empleados activos en la compañía, a través de su periodo laborado.
- Tramitar incapacidad: Este módulo se encargará de manejar las incapacidades de los empleados, incluyendo la fecha de inicio y final de la incapacidad, el tipo de la incapacidad y el historial de las incapacidades.
- Marcar asistencia: Este módulo se encargará de llevar un registro de asistencia de cada colaborador, especificando hora de entrada y salida de la jornada laboral.

Alcance metodológico

Kirvan (2023), indica que “el modelo en cascada es un enfoque lineal y secuencial del ciclo de vida del desarrollo de software (SDLC) y es popular en la ingeniería de software y el desarrollo de productos.

El modelo de cascada utiliza una progresión lógica de los pasos del ciclo de vida del desarrollo de software (SDLC) para un proyecto, similar a la dirección en la que fluye el agua por el borde de un acantilado. Establece objetivos o puntos finales específicos para cada fase del proceso de desarrollo. Estos objetivos o puntos finales no pueden revisarse una vez finalizados.

El Dr. Winston W. Royce, del Centro de Tecnología de Software de Lockheed, introdujo el concepto en un artículo publicado en 1970, sobre su experiencia en el desarrollo de software

para satélites. Sin embargo, Royce no utilizó el término «cascada»; en su lugar, se refirió al valor de la documentación en la fase posterior” ([traducción propia], párrs. 1-3).

Esta metodología se divide en las siguientes fases

- **Requisitos:** Los posibles requisitos, plazos y directrices del proyecto se analizan y se incorporan a un documento formal de requisitos, también denominado especificación funcional. Esta primera fase de desarrollo define y planifica el proyecto, sin mencionar procesos específicos.
- **Análisis:** Se analizan las especificaciones del sistema para generar modelos de producto y lógica de negocio que guíen la producción. En este proceso también se auditan los recursos financieros y técnicos para determinar su viabilidad.
- **Diseño:** Se crea un documento de especificaciones de diseño para definir los requisitos técnicos durante la fase de diseño. Estos incluyen el lenguaje de programación, el hardware, las fuentes de datos, la arquitectura y los servicios.
- **Pruebas:** En este proceso, el control de calidad, las pruebas unitarias, de sistema y beta identifican problemas que deben resolverse. Esto obliga a repetir la fase de codificación para la depuración. Si el sistema supera las pruebas de integración y pruebas, el proceso en cascada continúa.
- **Instalación:** Esta es la etapa en la que se aprueba el uso del producto de acuerdo con todos los requisitos. En esta etapa se pueden realizar varios procedimientos de prueba
- **Mantenimiento:** En esta fase de desarrollo, se proporcionan instrucciones del sistema, así como de operación y mantenimiento. Es importante mencionar que la instalación y mantenimiento de este sistema no están contemplados como parte del proyecto de graduación.

Alcance tecnológico

Los puntos siguientes cubren, brevemente, las soluciones de software que se utilizarán para crear el sistema, tanto durante su creación, como durante su gestión.

Para gestión de la base de datos se utilizará Microsoft SQL Server Management Studio 2022 (SSMS), incluida dentro de la licencia estándar de Microsoft SQL Server, permite administrar datos, generar informes y analizar los datos para el sistema.

SSMS es compatible con una amplia variedad de herramientas y tecnologías, lo que la convierte en una excelente opción gracias a las bibliotecas disponibles en C# y .NET, esto permite aprovechar al máximo sus capacidades, incluso con la versión estándar. Además, funciona de una manera eficiente en entornos donde se requiere la ejecución de consultas complejas y la creación de procedimientos almacenados.

El lenguaje de programación utilizado para este proyecto es C#, trabajando sobre la plataforma .NET, versión 4.8. Esta combinación fue elegida porque es flexible, confiable y cuenta con el respaldo y soporte de Microsoft. Gracias a estas características se puede garantizar la viabilidad y eficiencia en la creación de proyectos que se alinean, perfectamente, con las necesidades de la empresa.

Las herramientas de desarrollo que se utilizarán incluyen Microsoft Visual Studio, un entorno de desarrollo integrado (IDE) que ofrece una variedad de herramientas, bibliotecas y soporte, lo que lo hace ideal para crear proyectos pequeños, medianos y grandes.

El sistema se desarrollará en un equipo con Windows 10 Home con licencia, garantizando la compatibilidad y estabilidad necesarias para que el proyecto se desarrolle sin contratiempos.

El proyecto utilizará ASP.NET para el hospedaje de la aplicación web, mientras que la base de datos será creada en un entorno de desarrollo Visual Studio 2022. Esta herramienta hará más fácil la administración del sistema permitiendo realizar ajustes de manera sencilla conforme van cambiando las necesidades del usuario final. Esta elección busca asegurar un buen rendimiento del sistema y una experiencia de usuario fluida y consistente, a lo largo del desarrollo del proyecto.

CAPÍTULO II: MARCO REFERENCIAL

En el capítulo 1 se presentan las herramientas a utilizar para el desarrollo del prototipo funcional, así como una propuesta para solucionar el tema de gestión de planillas en la empresa “Girasol”. A continuación, se explicarán los conceptos necesarios para comprender mejor la solución planteada, brindando mejor claridad sobre el enfoque del proyecto.

Las Aplicaciones Web

Las aplicaciones web han cambiado, significativamente, la forma en que se utilizan herramientas y servicios digitales. En vez de depender de un sistema instalado en el dispositivo, ahora es posible acceder a múltiples de funciones y servicios directamente de un navegador web.

Esto significa que cualquier persona puede utilizar las herramientas y servicios web desde cualquier dispositivo que tenga una conexión a internet. Además, las aplicaciones web suelen ofrecer una experiencia familiar, ya que se ejecutan en navegadores conocidos y de uso común.

Según lo mencionado por Salminen (2025): “Una aplicación web, es un programa que se ejecuta en un navegador web. A diferencia de las aplicaciones de escritorio tradicionales, las aplicaciones web no requieren descarga ni instalación: los usuarios acceden a ellas directamente a través de internet desde cualquier dispositivo.

En segundo plano, las aplicaciones web dependen de varios componentes clave para funcionar correctamente. Estos incluyen un servidor backend, una interfaz de usuario frontend, bases de datos, API y un navegador que integra todo.

Las aplicaciones web vienen en diversos formatos, adaptándose a diferentes necesidades e industrias” ([traducción propia], párrs. 1-3).

Referente a la cita anterior, las aplicaciones web son programas informáticos donde las personas pueden acceder a través de cualquier navegador web, sin necesidad de instalar ningún sistema adicional. Estas aplicaciones pueden ser páginas simples que solo muestran información básica o más complejas con múltiples funciones como las redes sociales. Una de sus principales ventajas es que se puede acceder y utilizar desde cualquier dispositivo, con conexión a internet.

Los Tipos de Aplicaciones Web

Los tipos de aplicaciones web hacen referencia a las categorías que agrupan distintas funciones y características que estas pueden ofrecer. Estas pueden variar dependiendo de su propósito y su necesidad, sirven para comprender las diferentes formas en que las aplicaciones pueden ser utilizadas. Ayudan a organizar y analizar las distintas soluciones y servicios disponibles dentro de las aplicaciones web. Además, es útil, tanto para desarrolladores como para empresas, ya que les permite identificar qué tipo de aplicación se adapta mejor a sus objetivos. Como lo indica Franzolini (2023):

Algunos tipos de aplicaciones son: “Aplicaciones web estáticas, aplicaciones web dinámicas, aplicaciones web de comercio electrónico, aplicaciones web de gestión de contenido, aplicaciones web de página única, aplicaciones web de portal y aplicaciones web progresivas” (párr. 4).

En relación con la cita anterior, cada tipo de aplicación web posee sus respectivas características, propósitos y requerimientos técnicos. Estas características permiten distinguir entre diferentes clases de aplicaciones y entender mejor lo que cada una puede ofrecer. El objetivo de cada tipo de aplicación puede ir cambiando, desde mejorar la productividad, hasta facilitar transacciones comerciales.

Los Beneficios de las Aplicaciones Web

Los beneficios de las aplicaciones web son las ventajas que las aplicaciones brindan a los usuarios que las utilizan. Son claves para comprender la vida de los usuarios, también abarcan aspectos como conveniencia y seguridad, además de proporcionar un acceso instantáneo a una amplia variedad de herramientas y servicios, también reduce la carga de los recursos de almacenamiento. Tal como lo indica Silva (2023):

Algunas de las ventajas de las aplicaciones web son: “Puedes crear tus propias herramientas, no se requieren equipos especializados, funciona en una gran diversidad de dispositivos, no se requiere instalación y se actualizan automáticamente” (párr. 7-10).

Los beneficios pueden variar según el entorno y de las necesidades de cada usuario o empresa. En general, están estrechamente relacionados con factores como la accesibilidad, la compatibilidad y escalabilidad del sistema. Las ventajas se reflejan en como las aplicaciones

contribuyen a mejorar la eficiencia, productividad y experiencia del usuario, pero más allá de sus ventajas prácticas, las aplicaciones web se han convertido en herramientas indispensables en el día a día.

Los Entornos de Desarrollo Integrado

Conocido también como IDE, por sus siglas en inglés (por sus siglas en inglés: Integrated Development Environment), son la parte fundamental en el proceso de desarrollo de un sistema. Los entornos combinan diversas funciones y herramientas eficaces que permite a los desarrolladores poder escribir, probar, depurar e implementar sus aplicaciones. A lo largo del tiempo, los IDE han evolucionado significativamente, adaptándose a las necesidades cambiantes de los desarrolladores. De acuerdo con Lima (2022):

El entorno de desarrollo integrado es una herramienta de desarrollo para editar código, acceder a una terminal, ejecutar un script, depurar y compilar utilizando un único entorno. Un IDE es un programa con todas las funciones que se puede usar para muchos aspectos del desarrollo de software, que incluye herramientas de finalización de código, complementos y muchas otras funciones para facilitar el proceso de desarrollo de software. Sin embargo, actualmente existen varios IDE disponibles en el mercado, algunos desarrollados para lenguajes de programación específicos, otros editores de código que se comportan como IDE. (párrs. 1-2).

El Entorno de Desarrollo Integrado es un sistema que ofrece a los programadores diversas herramientas necesarias para poder desarrollar y mantener una aplicación, también sirve para poder mejorar la productividad y la eficiencia de los desarrolladores durante todo el desarrollo del sistema. Aparte de un editor de código, también ofrece un intérprete para poder convertir el código en un programa funcional, un detector para poder identificar y corregir errores, así como también ayuda a mantener los proyectos organizados.

Las Características del IDE

Las características de un IDE son diferentes y puede variar según sea la necesidad y preferencias de los desarrolladores. Tienen como objetivo principal mejorar la productividad del programador durante el desarrollo de un sistema, esto significa que proporciona herramientas que

facilitan las tareas diarias, automatizando procesos y eliminando la necesidad de cambiar constantemente entre diferentes aplicaciones. Según como lo menciona Canle (2023):

Entre las características más destacadas de los entornos de desarrollo integrado se encuentran las siguientes:

- editor de texto: El IDE cuenta con un editor de código que proporciona funciones de edición avanzadas, como el resaltado de sintaxis y la corrección ortográfica.
- Depurador de código: El detector es esencial para detectar errores en el código y solucionarlos, además de proporcionar la capacidad de realizar seguimientos en tiempo real.
- Herramientas de automatización: El IDE integra herramientas externas para realizar control de pruebas y analizar códigos.
- Soporte para múltiples lenguajes: Un buen entorno de desarrollo integrado admite múltiples lenguajes de programación en los proyectos.
- Facilidad de uso: Por lo general, son fáciles de usar gracias a que cuenta con una interfaz bastante intuitiva. Todo ello mejora la experiencia de usuario. (párr.7-13).

Las características son atributos que ofrece un sistema para facilitar el desarrollo de aplicaciones, son adaptables y flexibles. Además, están diseñadas para mejorar el flujo de trabajo de los desarrolladores durante el desarrollo de un sistema. Proporcionan un entorno completo para escribir, compilar y ejecutar código de manera eficaz. Les ayuda a los desarrolladores a mejorar el flujo de trabajo durante el proceso de desarrollo del sistema.

Los Principales Tipos de IDE

Los IDE son herramientas diseñadas para facilitar y agilizar el proceso de escritura, edición y mantenimiento del código, también tienen la capacidad de brindar un ambiente integrado, que permite a los desarrolladores trabajar de manera más eficiente.

Como lo expresa Nafiul (2025): “Existen diferentes tipos de Entornos de Desarrollo Integrado (IDE) según su propósito, pero en términos generales se pueden clasificar en tres tipos principales:

El primer tipo es un IDE completo, diseñado para trabajar con un lenguaje específico. Un ejemplo de esto es PyCharm, un IDE desarrollado especialmente para el lenguaje de programación Python, aunque también brinda cierto soporte limitado para otros lenguajes como JavaScript.

El segundo tipo es un IDE de propósito general. El mejor ejemplo es Visual Studio Code (VSCode), que es compatible con una gran variedad de lenguajes de programación gracias

a su arquitectura basada en extensiones o plugins. Esto le permite ser flexible y adaptarse a múltiples tecnologías, aunque en algunos casos no ofrece el soporte más completo para todos los lenguajes.

El tercer tipo es un IDE diseñado para una plataforma propietaria, que trabaja con su propio lenguaje de programación. Un ejemplo de este tipo es SalesforceCode, un IDE en línea desarrollado para la plataforma Salesforce. Aunque utiliza internamente la estructura de VSCode, incorpora muchas más funciones específicas, permitiendo trabajar, directamente, con directorios dentro de la propia instancia de Salesforce” ([traducción propia], párrs. 7-10).

Estos son algunos ejemplos de los Entornos de Desarrollo Integrado (IDE), utilizados en la industria del desarrollo. La elección de uno u otro depende de diferentes factores como el lenguaje de programación que se utilizará y las necesidades específicas del proyecto. Les permite a los desarrolladores trabajar de manera más productiva y organizada en sus proyectos.

Los Patrones de Arquitectura

El patrón de arquitectura es un aspecto fundamental en el diseño del sistema, ya que permite establecer una estructura organizada. También es utilizado para el desarrollo de aplicaciones robustas y escalables, lo que proporciona una base sólida sobre la cual construir. Al seguir el patrón establecido, los desarrolladores pueden garantizar la capacidad de crecer y adaptarse con el tiempo, respondiendo a nuevas necesidades sin comprometer su rendimiento ni su estabilidad. Tal como lo indica Gupta (2025):

“La arquitectura de software explica las ideas y características fundamentales de un sistema en relación con sus relaciones, entorno y otros principios de diseño. La arquitectura de software incluye la estructura organizativa de un sistema de software, sus componentes de comportamiento y la composición de dichos componentes en subsistemas más complejos. La base para gestionar el rendimiento, la tolerancia a fallos, la escalabilidad y la fiabilidad en el futuro reside en una arquitectura óptima. A medida que escala, elegir la arquitectura adecuada para su software se traducirá en un rendimiento más fiable en circunstancias difíciles” ([traducción propia], párrs. 1-2).

Complementando con lo anterior, un patrón de arquitectura es una solución comprobada para resolver problemas comunes en el diseño y desarrollo de sistemas. Los patrones ofrecen un

conjunto de buenas prácticas para organizar y estructurar el código de manera efectiva. Es como una plantilla para guiar a los desarrolladores a resolver problemas recurrentes que se presentan en proyectos similares. Este patrón es esencial para el rendimiento, la escalabilidad y el mantenimiento de las aplicaciones web.

El Patrón de Arquitectura MVC

El Patrón de Arquitectura MVC, es una estructura organizada y modular que se divide en tres componentes y esta división hace mejorar la mantenibilidad del código al separar las responsabilidades de cada elemento y apoya la escalabilidad al permitir que cada componente pueda ser modificado, esto facilita la adaptación del sistema, además, de que facilita la colaboración entre equipos de desarrollo. De acuerdo con Nwonah (2024):

“El patrón Modelo-Vista-Controlador (MVC) es uno de los patrones de diseño más utilizados en el desarrollo de software. Proporciona una clara separación de intereses al dividir una aplicación en tres componentes interconectados: el Modelo, la Vista y el Controlador. Esta separación facilita la gestión, las pruebas y el escalado de las aplicaciones” ([traducción propia], párr. 1).

En relación con lo anterior, los patrones de arquitectura facilitan el trabajo, al dividir las responsabilidades de una aplicación en tres partes, esto permite que cada componente sea desarrollado de manera independiente y pueda ser probado de manera eficiente, también permite la evolución continua del sistema a lo largo del tiempo y facilita la colaboración entre equipos de desarrollo.

Los Componentes del Patrón MVC

Los componentes trabajan en conjunto para brindar una arquitectura flexible que facilita el desarrollo, las pruebas y el mantenimiento de las aplicaciones. Estos desempeñan un rol específico dentro del sistema para trabajar en conjunto y proporcionar una estructura organizada y eficiente. Dentro de los componentes existe un sistema interrelacionado que permite una mejor coordinación entre las distintas partes del sistema, lo que a su vez facilita el desarrollo y optimiza las pruebas de las aplicaciones. Según Nwonah (2024):

“Modelo: El Modelo representa los datos y la lógica de negocio de la aplicación. Es responsable de gestionar los datos de la aplicación, responder a las solicitudes de información y actualizar los datos.

Vista: La Vista es la interfaz de usuario de la aplicación. Muestra datos del Modelo al usuario y envía comandos al Controlador.

Controlador: El controlador actúa como intermediario entre el modelo y la vista. Procesa la entrada del usuario, interactúa con el modelo y selecciona la vista adecuada para mostrar” ([traducción propia], párrs. 3-7).

En resumen, el diseño de arquitectura MVC se divide en tres partes, el modelo que representa los datos y las operaciones comerciales, la vista que muestra la información al usuario y el controlador que maneja de las interacciones con el usuario y actualiza el modelo y la vista cuando es necesario. La flexibilidad permite adaptarse, fácilmente, a los cambios en los requisitos, sin perjudicar el funcionamiento ni la estabilidad del sistema.

Las Bases de Datos

Las bases de datos son una parte fundamental, prácticamente para cualquier sistema informático. Almacenan, organizan y gestionan grandes cantidades de información, de manera eficiente, se suelen utilizar datos específicos que permitan estructurar la información de manera lógica. Se utilizan diferentes herramientas y estructuras para organizar la información, lo que hace más fácil su almacenamiento y recuperación. Según lo menciona Kosinski (2024):

“Una base de datos es un repositorio digital para almacenar, gestionar y proteger colecciones organizadas de datos.

Los distintos tipos de bases de datos almacenan datos de distintas maneras. Por ejemplo, las bases de datos relacionales los almacenan en tablas definidas con filas y columnas, mientras que las bases de datos no relacionales pueden almacenarlos en diversas estructuras de datos, como pares clave-valor o grafos” ([traducción propia], párrs. 1-2)

La base de datos es un sistema organizado que guarda y administra información de manera estructurada, permitiendo su fácil acceso, manipulación y actualización de datos de manera eficiente y se pueden almacenar en un medio digital o físico. Suelen ofrecer funciones y herramientas para asegurar la integridad y consistencia de los datos, como reglas de integridad y transacciones completas.

Tipos de Bases de Datos

Los tipos de bases de datos son diferentes categorías en las que se pueden unir, según sea su estructura, modelo, almacenamiento y uso específico. Sirven de guía para entender y dividir las diferentes tecnologías disponibles en el mercado. La variedad de tipos se debe a las diferentes necesidades en que se utilizan. Se clasifican en función de diversos criterios como la estructura el modelo y el entorno. Los tipos de base de datos son, de acuerdo con Larque (2023):

Base de datos relacionales: “Las bases de datos relacionales son las más utilizadas como tecnología para la industria. Están diseñadas para almacenar datos estructurados en tablas relacionadas entre sí”

Bases de datos NoSQL: “Las bases de datos NoSQL son una alternativa a las bases de datos relacionales. No utilizan tablas y claves como las bases de datos relacionales, sino que utilizan una variedad de estructuras de datos, como documentos, gráficos”

Bases de datos de objetos: “Las bases de datos de objetos son un tipo de base de datos NoSQL que almacenan datos como objetos. Están diseñadas para trabajar con lenguajes de programación orientados a objetos, como Java o Python” (párrs. 2-4).

Según lo anterior, estos son algunos ejemplos de tipos de bases de datos, los cuales permiten agrupar diferentes sistemas de gestión de bases de datos, según sus características y usos específicos. La comprensión de los tipos es importante para elegir la tecnología más adecuada según las necesidades del proyecto. La elección de estos también depende de factores como los requisitos funcionales, el volumen de datos, la estructura de la información y la forma en que será consultada o actualizada.

La Base de Datos Microsoft SQL Server

Microsoft SQL Server ha evolucionado para ser una plataforma potente, robusta y escalable para almacenar, administrar y analizar datos. Se destaca por su capacidad de manejar transacciones de gran volumen y tener una amplia gama de características adaptadas a las necesidades de los entornos empresariales. Su constante evolución la convierte en una de las preferidas, tanto por grandes corporaciones, como por pequeñas empresas. Tal como menciona con Parada (2019):

Microsoft SQL Server es la alternativa de Microsoft a otros potentes sistemas gestores de bases de datos. Es un sistema de gestión de base de datos relacional desarrollado como un servidor que da servicio a otras aplicaciones de software que pueden funcionar ya sea en el

mismo ordenador o en otro ordenador a través de una red (incluyendo Internet). Los servidores SQL Server suelen presentar como principal característica una alta disponibilidad al permitir un gran tiempo de actividad y una conmutación más rápida. Todo esto sin sacrificar los recursos de memoria del sistema. (párrs. 3-4).

SQL Server creado por Microsoft es un sistema de gestión de base de datos, utilizado para almacenar, recuperar y administrar información de forma eficiente, proporciona diversas características como funciones definidas por el usuario, replicación de datos y alta disponibilidad, utiliza el lenguaje de consulta estructurada para interactuar con la base de datos y también proporciona herramientas de administración gráfica y ofrece diversas ediciones diseñadas para cumplir necesidades específicas de los usuarios.

SQL Server Management Studio

SQL Server Management Studio es una herramienta fundamental que facilita la administración y manejar entornos SQL Server. Ofrece una variedad de funciones que facilita el trabajo de desarrollo y mantenimiento de bases de datos de SQL Server, suele combinar una variedad de herramientas gráficas con editores de texto para brindar acceso a SQL Server a programadores y administradores de varios niveles. Como señala Microsoft (2025):

“SQL Server Management Studio (SSMS) es un entorno integrado para administrar cualquier infraestructura SQL. Use SSMS para acceder, configurar, administrar y desarrollar todos los componentes de SQL Server, Azure SQL Database, Azure SQL Managed Instance, SQL Server en Azure VM y Azure Synapse Analytics. SSMS proporciona una utilidad integral que combina un amplio conjunto de herramientas gráficas con numerosos editores de scripts enriquecidos para brindar acceso a SQL Server a desarrolladores y administradores de bases de datos de todos los niveles” ([traducción propia], párr. 1).

SQL Server Management Studio diseñada por Microsoft, es una aplicación para gestionar y acceder a las bases de datos relacionales de SQL Server, lo que facilita al usuario realizar diversas tareas, como diseñar bases de datos, hacer consultas, modificar datos, configurar servidores de forma sencilla y organizada. Es fundamental tanto para administradores de bases de datos, como para desarrolladores, ya que las múltiples funciones que tiene ayudan a simplificar y gestionar los datos.

Los Lenguajes de Programación

Los lenguajes de programación son la parte más importante de la evolución digital, siguen en constante evolución para adaptarse a las demandas y desafíos de la tecnología, proporcionan herramientas necesarias para que los desarrolladores puedan crear sistemas y aplicaciones, los lenguajes facilitan la comunicación entre los programadores y las computadoras y poder expresar algoritmos de manera efectiva. Tal como indica Cimas (2024):

Un lenguaje de programación es el conjunto de instrucciones a través del cual los humanos interactúan con las computadoras. Nos permite comunicarnos con las computadoras a través de algoritmos e instrucciones escritas en una sintaxis que la computadora entiende e interpreta en lenguaje de máquina. También permiten a las computadoras procesar de forma rápida y eficientemente grandes y complejas cantidades de información. (párrs. 4-6).

Un lenguaje de programación implica reglas y símbolos utilizados para escribir instrucciones que una computadora pueda entender. Estos lenguajes permiten a los desarrolladores crear programas informáticos y automatizar procesos. Todos los lenguajes tienen características diferentes, varían su sintaxis y propósito, cada uno está diseñado para cumplir diferentes necesidades.

Tipos de Lenguajes de Programación

Los tipos de lenguajes de programación hacen referencia a las clasificaciones para agrupar diferentes lenguajes, según sea su estructura y propósito. La clasificación de estos permite a los desarrolladores identificar características y funciones que mejor se adapten a las necesidades específicas y facilitando la elección del más adecuado. El aprendizaje de nuevos lenguajes se hace más fácil, ya que, con los conocimientos adquiridos al trabajar con un lenguaje, pueden aplicarse a otros tipos. Los tipos de lenguajes son, según lo expresado por Saavedra (2023):

Lenguaje de bajo nivel (Lenguaje de máquina): Es el único lenguaje que puede ser leído por el hardware de la computadora a través de la CPU (unidad central de procesamiento). El lenguaje de máquina usa código binario. Todas las instrucciones se codifican por los bits 0 y 1 en forma de cadenas de información. Estos dígitos son datos de la tensión de corriente eléctrica; las señales de bajo voltaje y alto voltaje permiten que la máquina interprete y ejecute cualquier indicación.

Lenguajes de medio nivel (Lenguajes ensambladores): Pueden ser entendidos, tanto por los programadores, como por las máquinas. Se aplican para escribir instrucciones precisas para la computadora. Usan mnemónicos que incluyen letras, dígitos, caracteres especiales y palabras reservadas. Un programa utilitario traduce el código escrito en este lenguaje al lenguaje de máquina que puede procesar la CPU.

Lenguajes de alto nivel: Son los lenguajes de programación más comunes, ya que son fáciles de entender para el ser humano. El uso de declaraciones similares en inglés permite crear un código sencillo y comprensible. Cada lenguaje de alto nivel tiene palabras claves predeterminadas y un conjunto de reglas llamado sintaxis. Se usan compiladores o intérpretes para convertir un lenguaje de alto nivel al lenguaje de máquina para que la computadora pueda cumplir las indicaciones. (párrs. 4-6).

Los tipos de lenguajes son una forma de organizar los diferentes lenguajes que se utilizan para el desarrollo de sistemas. Al comprender los diversos tipos, los desarrolladores pueden tomar una decisión más informada sobre cual lenguaje utilizar y cuál se ajusta más a las necesidades y objetivos específicos, tomando en cuenta los requisitos del proyecto, preferencias de los usuarios y si existen limitaciones de compatibilidad o recursos.

El Lenguaje de Programación C#

El lenguaje de Programación C# fue creado en el año 2000 y a lo largo del tiempo, se ha convertido en un pilar fundamental en el mundo del desarrollo de software y desde entonces ha ganado popularidad, gracias a su capacidad de ofrecer equilibrio entre notable potencia, flexibilidad y facilidad de uso y ha ganado reconocimiento como uno de los lenguajes más utilizados en la industria. Según lo indica Staff (2024):

“C#, que se pronuncia "C sharp", es un lenguaje de programación de propósito general y orientado a objetos que se utiliza para desarrollar una amplia gama de programas, incluyendo software empresarial, videojuegos y aplicaciones móviles. Introducido por Microsoft en 2002 con el lanzamiento de Visual Studio .NET, C# forma parte de la familia de lenguajes C, que también incluye C y C++. C# es el más moderno de los tres y el más fácil de aprender, ya que es un lenguaje de programación de alto nivel. En este artículo, aprenderá más sobre C#: qué es, sus aplicaciones y los trabajos que lo utilizan habitualmente” ([traducción propia], párr. 1).

El lenguaje de programación C#, desarrollado por Microsoft que se utiliza para crear diversos sistemas, tanto aplicaciones de escritorio como aplicaciones web, es reconocido por su sintaxis clara y su capacidad para trabajar con la plataforma .Net, que ha contribuido al desarrollo de su éxito, se acopla bien con el IDE Visual Studio, lo que facilita la creación y mantenimiento de proyectos. Además, este lenguaje sigue creciendo de manera continua, debido a las necesidades cambiantes de la industria.

Las Características del Lenguaje de Programación C#

Las características del lenguaje de programación C# son atributos que definen el comportamiento y la función, proporcionando a los desarrolladores características y herramientas específicas para crear sistemas eficientes. Están creadas para cubrir necesidades específicas en el desarrollo del sistema. Además, ofrecen características orientadas a objetos que hacen más fácil la creación de software modular y fácil de mantener. Al entender las características del lenguaje C#, los programadores pueden aprovechar, al máximo, su potencial, de acuerdo con Ramírez (2023):

Entre las características de C# se incluyen: gestión automática de memoria, interoperabilidad con otros lenguajes de programación, compatibilidad con múltiples plataformas, tiene capacidad de crear aplicaciones con una gran cantidad de componentes, capacidad de crear aplicaciones de escritorio, aplicaciones web y juegos, cuenta con una amplia comunidad de desarrolladores y una gran cantidad de recursos en línea. (párr. 3).

Las características del lenguaje C# se distinguen como elementos fundamentales que han cooperado a su popularidad por el mundo del desarrollo, han sido importantes para saber el funcionamiento del lenguaje C# y las capacidades que tiene y también hacen que el lenguaje C# sea una herramienta potente para la creación de diversas aplicaciones. Este lenguaje brinda a los programadores diversas herramientas necesarias para desarrollar un sistema de alta calidad.

Este proyecto está orientado en el área de Recursos Humanos de la empresa “Girasol”. La elección de este enfoque se fundamenta debido a los datos obtenidos en la investigación inicial, que identificó problemas relacionados con la gestión de planilla, un proceso directamente vinculado con dicho departamento. Es importante destacar que el personal abarca todos los individuos que integran la organización y también contribuyen a diversas áreas. Su papel es esencial para el funcionamiento, ya que aportan sus conocimientos y habilidades para alcanzar los objetivos comerciales.

Recursos Humanos

El Departamento de Recursos Humanos representa una parte fundamental para el éxito y crecimiento de una empresa, no solo para reclutar, sino también para potenciar las habilidades de los empleados, además, de emplear un ambiente que promueve el crecimiento personal y profesional de cada empleador. Como menciona O'Reilly (2024):

“Recursos humanos (RR. HH.) se refiere al personal de su organización que trabaja en conjunto para alcanzar sus objetivos a corto y largo plazo. La gestión de Recursos Humanos es el proceso de encontrar, contratar, capacitar y gestionar la fuerza laboral de su empresa. El departamento de RR. HH. de su empresa actúa como el nexo que une sus diversas operaciones en una sola entidad coherente y eficiente. Actúa como un recurso para empleados y gerentes. Sin embargo, el verdadero valor de RR. HH. para una organización reside en su capacidad para optimizar las contribuciones de las personas y los equipos, así como la eficiencia y la productividad de sus diversos departamentos” ([traducción propia], párr. 1).

Complementando con lo anterior, los Recursos Humanos es una función clave dentro de una empresa encargada de gestionar el capital humano, entre sus funciones principales están reclutar, seleccionar y capacitar a los empleados de la empresa, también establecer política interna, administrar los beneficios, manejar conflictos y promover un ambiente agradable dentro de la empresa.

Funciones del departamento de Recursos Humanos

Las funciones son responsabilidades que debe llevar a cabo el Departamento de Recursos Humanos dentro de una empresa. Están diseñadas para gestionar el capital humano de una manera eficaz, son esenciales para crear un ambiente laboral saludable y para garantizar el cumplimiento de políticas laborales. Las principales funciones de Recursos Humanos, tal como lo indica Chimphondah (2025), son:

“Planificación y gestión de recursos humanos: La planificación y gestión de recursos humanos es un proceso estratégico de toma de decisiones que visualiza la planificación futura de una empresa. Implica integrar la plantilla actual de la organización con sus objetivos y visión. Es la principal función de RR. HH. y se centra en anticipar las necesidades futuras de la organización. ¿Cuántas y qué tipo de personas requiere la

organización? Comprender esto influirá en la configuración de las funciones de recursos humanos.

Reclutamiento y selección: Las organizaciones deben atraer y retener al mejor talento para mantenerse competitivas en el sector. El Departamento de Recursos Humanos crea una estrategia para incorporar talento nuevo a la organización. Es responsable de buscar empleados y determinar cómo serán estos y sus descripciones de puestos. Las organizaciones deben crear descripciones de puestos que definan, claramente, las funciones y responsabilidades. Al reclutar, el Departamento de Recursos Humanos debe comprender cómo llevar a cabo correctamente el proceso de reclutamiento y selección para retener al personal de alto nivel y desarrollar un equipo sólido y estable. Si bien la formación, las habilidades y la experiencia son importantes, Recursos Humanos también debe considerar las cualidades personales y el estilo de comunicación para aumentar la probabilidad de una compatibilidad a largo plazo.

Capacitación y desarrollo: Se espera que los nuevos empleados de una empresa reciban la orientación y la capacitación adecuadas para prosperar en sus puestos, desde su incorporación. Como parte de este proceso, deberá comunicar los valores fundamentales, la visión, la misión, las normas y los procedimientos internos de la organización.

Sistema de Gestión del Desempeño: El desempeño de los empleados tiene un efecto significativo en el éxito de una empresa. Por lo tanto, las organizaciones deben monitorear y evaluar el desempeño. Los sistemas de gestión del desempeño buscan aumentar la productividad, mediante el establecimiento de indicadores clave de desempeño (KPI) eficaces. Tradicionalmente, el Departamento de Recursos Humanos revisa el desempeño mediante evaluaciones semestrales o anuales” ([traducción propia], párrs. 1-7).

“Recompensas y reconocimientos: Otra función de Recursos Humanos que el Departamento de RR. HH. puede utilizar en una organización. El departamento de RR. HH. puede crear diferentes formas de recompensas y reconocimiento. Las recompensas pueden ser vacaciones patrocinadas, aumentos de sueldo, bonificaciones o pagos por desempeño. Los reconocimientos pueden ser un puesto más alto, estabilidad laboral, oportunidades de

crecimiento, un buen ambiente laboral, reconocimientos y credibilidad en toda la empresa” ([traducción propia], (párr. 11).

Componentes del Sistema de Planilla

El sistema de planilla es una parte fundamental en la estructura operativa de una empresa, está compuesto por diferentes componentes para garantizar una correcta gestión de planilla y pago de los empleados. El sistema se compone de una serie de procedimientos asociados al personal de la empresa, para calcular y administrar los salarios y beneficios.

Planilla. Este módulo se encargará de calcular el pago correspondiente para cada uno de los colaboradores, tomando en cuenta sus respectivas deducciones (CCSS), el registro de vacaciones y las horas extras laboradas.

Vacaciones. El sistema verifica las vacaciones de cada colaborador, a partir de su primer día de ingreso y los días que tiene disponibles. El colaborador deberá enviar una solicitud a su jefatura inmediata, quien se encargará de remitir la solicitud a Recursos Humanos. Este departamento es responsable de aceptar o rechazar la solicitud y se notifica la decisión tomada al colaborador.

Horas extras. Se encarga de gestionar el cálculo del monto correspondiente a pagar por cada hora extra realizada por el colaborador, este deberá enviar una solicitud sobre las horas extras trabajadas a la jefatura inmediata, quien remite la solicitud a Recursos Humanos y este departamento es el encargado de aceptar o rechazar dicha solicitud, también notifica al colaborador el estado de dicha solicitud.

Control permisos. El objetivo de este proceso es llevar un control de los permisos solicitados por cada trabajador, este envía una solicitud a la jefatura inmediata, quien se encarga de aprobar o rechazar las solicitudes y de notificarlo al colaborador.

Deducciones. El sistema calcula las deducciones de cada colaborador tomando en cuenta las deducciones como C.C.S.S y I.V.M y así se podrá calcular el monto correspondiente,

garantizando que las rebajas se realicen de manera justa y precisa, evitando que algún colaborador se vea perjudicado.

Aguinaldo. Este componente tiene la función de calcular el aguinaldo de cada empleado, de acuerdo con su puesto, desde el 1° de diciembre del año anterior hasta el 30 de noviembre del presente año y realizar ese cálculo de manera correcta.

Liquidación. Su función es calcular la liquidación correspondiente a cada empleado activo en la compañía, tomando en cuenta el periodo laborado y los beneficios que por ley corresponden.

Tramitar incapacidad. Su objetivo es gestionar las incapacidades de los empleados, incluyendo la fecha de inicio y la fecha fin, el tipo de la incapacidad y el historial completo de las incapacidades de cada trabajador.

CAPÍTULO III: MARCO METODOLÓGICO

Dentro del marco metodológico, se detallarán los procedimientos y las herramientas para la recolección y análisis de datos, para garantizar una investigación exitosa, asegurando la validez y fiabilidad de los resultados. Además, se explicará, de manera general, los instrumentos utilizados durante el desarrollo de la investigación y así poder interpretar, de manera eficiente, los resultados obtenidos.

Enfoques de la Investigación

En una investigación, el enfoque es fundamental, porque determina la manera en que se abordará y resolverá el problema en estudio. El enfoque puede variar dependiendo de diferentes factores, como la naturaleza del problema, el contexto de la investigación y las herramientas y recursos disponibles. Cada uno cuenta con sus propias características, que los distinguen de los demás. Cada uno de los enfoques ofrecen un conjunto de herramientas necesarias para poder avanzar en la comprensión y el objetivo la investigación. Como menciona Hassan (2024),

“Un enfoque de investigación se refiere a la estrategia o dirección general adoptada para responder a las preguntas de investigación o comprobar hipótesis. Proporciona una vía estructurada para la recopilación y el análisis de datos, garantizando que la metodología del estudio se ajuste a sus objetivos.

Por ejemplo, un investigador que investiga el comportamiento humano podría elegir un enfoque cualitativo para explorar experiencias subjetivas, mientras que podría utilizarse un enfoque cuantitativo para medir y analizar datos numéricos” ([traducción propia], párrs. 3-4).

Según lo mencionado anteriormente, el enfoque de la investigación se refiere a diferentes formas que se utilizan para resolver una problemática. El investigador cuenta con diferentes métodos para estudiar un problema y determinar cuál se adecúa mejor para la solución. Además, es importante seleccionar el enfoque que mejor se adapte a la naturaleza del problema, para obtener los resultados deseados. Los enfoques se dividen en 3 tipos: el enfoque cuantitativo, el enfoque cualitativo y el enfoque mixto.

Enfoque Cualitativo

El enfoque cualitativo es una herramienta que permite comprender fenómenos complejos, desde una perspectiva amplia y profunda. A diferencia del enfoque cuantitativo, se basa en la recolección y medición de datos no numéricos, se enfoca en la calidad y emplea técnicas como entrevistas en profundidad y observación participante, que permiten capturar opiniones y experiencias de los participantes de manera detallada. Tal como indica Bhandari (2023):

“La investigación cualitativa implica la recopilación y el análisis de datos no numéricos (p. ej., texto, vídeo o audio) para comprender conceptos, opiniones o experiencias. Puede utilizarse para profundizar en un problema o generar nuevas ideas para la investigación.

La investigación cualitativa es lo opuesto a la investigación cuantitativa, que implica la recopilación y el análisis de datos numéricos para el análisis estadístico.

La investigación cualitativa se utiliza habitualmente en las humanidades y las ciencias sociales, en disciplinas como la antropología, la sociología, la educación, las ciencias de la salud, la historia, etc.” ([traducción propia], párrs. 1-3).

El enfoque cualitativo se basa en una metodología para recopilar información, con el objetivo de abordar y comprender, a profundidad, un problema de investigación, se aplica a estudios que involucran experiencias humanas, historias de la vida o resolución de un determinado problema. Las investigaciones cualitativas que utilizan esta metodología tienen diversas herramientas y técnicas para la recopilación y análisis de datos, las cuales se ajustan a las necesidades que surjan durante el proceso de investigación.

Enfoque de Investigación Seleccionado.

Se eligió el enfoque cualitativo porque permite investigar, en profundidad y de forma detallada, el problema que se va a abordar, lo que facilita una comprensión más completa de la situación. Es flexible y adaptable en analizar datos, lo que permite ajustarse a las necesidades específicas de la investigación. Es ideal para la recolección de datos pequeños y eficaz para una mejor comprensión de ellos. También tiene diversidad de herramientas que se pueden usar para la recolección de datos como entrevistas, observación directa, cuestionarios, etc.

Tipos de Investigación

Los tipos de investigación se refieren a las diferentes formas para llevar a cabo un estudio, este puede variar dependiendo del factor como el propósito de la investigación, los objetivos que se desee alcanzar y características de la investigación. Seleccionar el tipo de investigación adecuado es esencial para poder crear una investigación sólida y eficaz, ya que orienta y facilita a obtener los resultados deseados. De acuerdo con Suarez (2023):

Los tipos de investigación son las maneras en las que un estudio se puede abordar en concordancia con los aspectos que lo definen, tales como el título, la profundidad, objetivos, tratamiento de la muestra y los datos, tipo de inferencia, manipulación de las variables, temporalidad, entre otros. En este sentido, la adecuada escogencia de un tipo de investigación permitirá mejorar la comprensión del tema que estás estudiando, siendo crucial para aplicar los procesos de recolección de datos que sean más efectivos ([traducción propia], párrs. 1-3).

Haciendo referencia a lo anterior, cada tipo tiene sus propias características y se debe elegir en función a las necesidades y objetivo del estudio. Se clasifican en: Exploratorios: consisten en investigar un tema poco estudiado, los descriptivos: se refieren a especificar las características y perfiles de personas o fenómenos que se sometan a un análisis, los correlacionales: estos responden a preguntas de investigación, todas dentro de un mismo contexto y los explicativos: estos responden a la causa de eventos o fenómenos físicos o sociales.

Investigación Descriptiva

La investigación descriptiva es un método que permite describir, explorar, comprender fenómenos y situaciones detalladamente, pretende seleccionar o medir información de manera independiente. Su objetivo principal es brindar información valiosa y para lograrlo recurre a diferentes técnicas y herramientas de recolección de datos. Al proporcionar una visión más detallada de los fenómenos en estudio, sirve como un punto de partida para el análisis y la interpretación de los resultados en etapas posteriores. Según menciona Mejía (2020):

La investigación descriptiva es un tipo de investigación que se encarga de describir la población, situación o fenómeno alrededor del cual se centra su estudio. Procura brindar información acerca del qué, cómo, cuándo y dónde, relativo al problema de investigación,

sin darle prioridad a responder al “por qué” ocurre dicho problema. Como dice su propio nombre, esta forma de investigar “describe”, no explica. (párr. 1).

Su objetivo principal es especificar las características y perfiles de personas, grupos, procesos u objetos que se puedan someter a un análisis, es decir se orientan al análisis de datos que conllevan diversos componentes o aspectos del tema investigado. En este tipo de investigación se seleccionan una serie de preguntas apropiadas y se procede a la recopilación de información de cada una de ellas, esto con el fin de presentar de manera precisa y completa el propósito de la investigación.

Tipo de Investigación Seleccionado

Se eligió la investigación descriptiva, porque puede servir como punto de partida para explorar un tema del cual se tiene poco conocimiento, es útil para mostrar, con claridad, las dimensiones del problema, así como para definir conceptos claves relacionados con el tema de investigación, sirve para validar cuáles instrumentos o técnicas son más adecuados para la recolección de datos y también cuenta con diferentes métodos tales como la observación, método de estudio de caso y encuestas.

Fuentes de Información

Desde tiempos antiguos la búsqueda de fuentes confiables ha sido una tarea fundamental para los investigadores. Las fuentes de información forman el soporte para construir el conocimiento, para poder obtener información de calidad y confiable para llevar a cabo una investigación. Una fuente de información fiable es crucial para garantizar validez en los resultados obtenidos en una investigación. Tal como indica Ashikuzzaman (2023):

“Las fuentes de información se refieren a los orígenes o canales de los cuales las personas obtienen datos, hechos, conocimientos o perspectivas. Estas fuentes pueden ser diversas y abarcar diversos medios y formatos. Las fuentes de información son cruciales para moldear nuestra comprensión del mundo, contribuyendo a la base del conocimiento y a los procesos de toma de decisiones. Tradicionalmente, las fuentes de información incluyen libros, revistas, periódicos y otros medios impresos. Sin embargo, con la llegada de la era digital, las fuentes de información se han expandido para incluir plataformas en línea, bases de datos, redes sociales, contenido audiovisual y más” ([traducción propia], párr. 2).

Las fuentes de información son recursos que proporcionan datos, conocimientos o ideas sobre un tema específico, esta puede ser tanto tangible, como intangible, desde páginas como revistas, libros, periódicos, etc., las cuales sirven como referencia para realizar una investigación o estudio y también contribuyen al aprendizaje y así poder garantizar la calidad y confiabilidad de los resultados de una investigación.

Fuentes de Información Primarias

Este tipo de información es creada, directamente, por personas o instituciones, son aquellas que contienen datos en su estado original. Su principal característica es la inmediatez y autenticidad, brindando un acceso directo a los datos originales sin intermediarios. Las fuentes primarias permiten a los investigadores explorar la multiplicidad de voces y perspectivas que conforman una sociedad. Como lo menciona Kowalski (2025):

“Las fuentes primarias son materiales originales. Son registros de evidencia sobre un evento o descubrimiento, creados en el momento en que ocurrió dicho evento o descubrimiento. A menudo, comparten información nueva que no se había publicado antes. Imagina que eres un chef preparando un plato. Tus ingredientes son como fuentes primarias: son los alimentos crudos y sin procesar que vas a utilizar. En el mundo académico, las fuentes primarias son la materia prima de la investigación. Son los relatos originales y de primera mano, o la evidencia directa, de un tema investigado” ([traducción propia], párrs. 3-4).

Fuentes de Información Secundarias

Brinda información no original, es decir, obtenida desde la fuente principal que pueden ser de revistas, libros, enciclopedias, sitios web, etc. A diferencia de las fuentes primarias, las secundarias procesan, interpretan la información para brindar una visión más clara y comprensible. Se utilizan, especialmente, cuando se busca obtener una perspectiva general de un tema específico. Según lo expresa Kowalski (2025):

“Las fuentes secundarias están un paso más allá del evento o la información original y se crearon en un momento posterior.

Piense en las fuentes secundarias como si fueran un plato preparado. Al igual que un chef combina ingredientes para crear una comida, las fuentes secundarias sintetizan, interpretan y analizan la

información de las fuentes primarias. En el ámbito académico, las fuentes secundarias describen y analizan las fuentes primarias proporcionando contexto y diferentes perspectivas sobre un tema” ([traducción propia], párrs. 7-8).

Fuentes de Información Terciarias

Se refieren a resúmenes que facilitan el acceso a información de fuentes primarias y secundarias, es una herramienta de referencia para localizar datos de un tema en específico de manera rápida y eficiente. A diferencia de las fuentes secundarias, las fuentes terciarias actúan como una guía rápida y eficiente para encontrar información relevante sobre un tema específico. Tal como lo indica Kowalski (2025):

“Las fuentes terciarias compilan, resumen o indexan información de fuentes primarias y secundarias.

Si las fuentes primarias son los ingredientes crudos y las secundarias son la comida cuidadosamente preparada, las fuentes terciarias son como un menú: ofrecen una visión general rápida de lo disponible. En el ámbito académico, las fuentes terciarias son puntos de partida útiles para la investigación y pueden ayudarte a obtener información de fondo sobre tu tema de investigación” ([traducción propia], párrs. 11-12).

Variables

Las variables permiten medir, describir y analizar los fenómenos que están en investigación. Hace referencia a cualquier atributo o característica que sea capaz de asumir distintos valores, brinda la capacidad de poder dar forma a los fenómenos estudiados. Se puede clasificar en diferentes tipos, según sea su naturaleza y función en la investigación, variable conceptual, variable operacional y variable instrumental. Son esenciales en una investigación para garantizar la fiabilidad de los resultados obtenidos. Según menciona González (2020):

Las variables de investigación son las distintas características o propiedades de los seres vivos, objetos o fenómenos que tienen la particularidad de sufrir cambios y que pueden observarse, medirse, ser objeto de análisis y controlarse durante el proceso de una investigación. Una variable es una propiedad del objeto de estudio que puede asumir dos o más valores (es decir, puede cambiar). De forma que, si esto no ocurre, la característica observada no es una variable sino una constante. (párrs. 1-3).

Una variable es una característica o condición que puede cambiar en diferentes momentos o situaciones, este puede asumir distintos valores y es el objeto de estudio de una investigación, se usan para medir comportamientos o conceptos dentro de una investigación, son fundamentales para una interpretación adecuada en el análisis del estudio. Las variables garantizan la validez de los resultados obtenidos.

Variable conceptual

Es un concepto abstracto que representa una característica del fenómeno en estudio. Proporciona la base teórica sobre la cual se desarrolla la investigación y se formulan las hipótesis. Sin embargo, este tipo de variable necesita ser tangible y medible para que pueda ser estudiada de manera efectiva.

Variable operacional

Es el proceso específico utilizado para observar o manipular una variable conceptual y con el fin de convertir una idea abstracta en una tangible y medible.

Variable instrumental

Es la variable utilizada como instrumento para medir, controlar o manipular otra variable dentro de una investigación. Su función principal es facilitar la recolección de datos y garantizar que la medición sea precisa y confiable.

Tabla 4*Cuadro de variables*

Objetivo Especifico	Variable	Variable Conceptual	Variable Operacional	Variable Instrumental
Analizar los requerimientos que se van a incorporar en el prototipo funcional	Requerimientos	Gomez (2022) “Un requerimiento define las funciones, capacidades o atributos intrínsecos de un sistema de software, en otras palabras, describe como un sistema debe comportarse” (párr.	Entrevista, Observación	Guia de entrevista, guía de observación
Elaborar el diseño del prototipo de acuerdo con los resultados del análisis.	Diseño	De acuerdo con Mordecki (2004) es elegir cuál es la solución que construiremos entre las infinitas soluciones válidas para un problema. Cuando un cliente nos plantea un problema y nos pide un diseño, nos plantea el desafío de decidir cuál es la mejor de todas las construcciones posibles para intentar dar solución a ese	Diagrama de base de datos, diagrama de arquitectura	Draw.io, SQL Management Studio
Desarrollar la programación del prototipo tomando como base el diseño previamente creado	Programación	Según Cassingena (2023) La programación es lo que le permite a tu computadora ejecutar los programas que usas cada día y es lo que le permite a tu smartphone ejecutar las aplicaciones que te encantan. Es una parte esencial del mundo que	Casos de Uso, Entidad-Relación, Arquitectura MVC	Visual Studio 2022, SQL Management Studio, C#
Ejecutar las pruebas funcionales necesarias en el prototipo final	Pruebas	Tal como indica Santos (2024) “Son los procesos en los que se verifica la funcionalidad y rendimiento del software de un determinado producto o aplicación” (párr. 1).	Pruebas	Visual Studio 2022, Guia de pruebas

Fuente: elaboración propia, 2025

Instrumentos de recolección de datos

La entrevista es una herramienta de investigación importante para obtener información directa de las personas, es la más utilizada en la investigación cualitativa, debido a su capacidad de obtener información de manera profunda. Mediante diversas preguntas, los investigadores pueden explorar, con mayor detalle, los temas en estudio. Ofrece una relación de confianza entre el participante y el investigador, lo que impulsa a una mayor sinceridad en las respuestas. De acuerdo con Angarita (2024):

La entrevista es un intercambio verbal entre dos o más personas con roles de interrogador(es) e interrogado(s) para un propósito expreso. Existe un objetivo previamente establecido por los interlocutores a través de acuerdo mutuo. De este modo, la entrevista permite reunir datos durante un encuentro donde una persona se dirige a otra contando su historia, dando su versión de los hechos y respondiendo preguntas en torno a un tema específico.

Debe tenerse en cuenta que la entrevista en la investigación cualitativa es flexible, por lo que el principio y el final de la misma no se predeterminan ni se definen con claridad. Asimismo, las preguntas, su orden, ritmo, dirección y las normas y el lenguaje empleados se adecúan a los participantes (Hernández-Sampieri y Mendoza, 2018). Las preguntas son abiertas y neutrales, ya que pretenden obtener perspectivas detalladas de los participantes. También es importante considerar que el contexto social resulta fundamental para la interpretación de significados, dado que toda entrevista es un proceso dinámico y multifuncional atravesado por las transformaciones del contexto complejo de vida del entrevistado. (párrs. 1-2).

La entrevista es un método que se utiliza para recopilar datos cualitativos, su objetivo es poder profundizar en el tema de investigación. Durante el proceso de la entrevista, es recomendable que el entrevistador establezca confianza con el entrevistado, muestre respeto en todo momento y un comportamiento ético. Este instrumento puede aplicarse de diferentes formas, virtual, presencial o vía telefónica, esto va a depender de las preferencias y disponibilidad de los participantes.

La observación es una herramienta de investigación que permite analizar el comportamiento humano y conductas durante el desarrollo de la investigación. Puede ser participativa o no participativa, dependiendo del observador y la situación que está siendo

observada. Una de las ventajas de la observación es poder capturar comportamientos y eventos, tal como ocurren en la vida real. Esto sirve a los investigadores para poder obtener una comprensión más auténtica de los fenómenos en estudio. Tal como indica Rodríguez (2024):

“La observación es una técnica fundamental en la investigación que permite recopilar datos de manera directa sobre el comportamiento, fenómenos o eventos que se estudian. Este artículo explora qué es la observación, su importancia en la investigación y cómo se utiliza eficazmente para obtener resultados valiosos y precisos.

La observación es un método de recolección de datos que implica la observación directa y sistemática de un fenómeno, grupo o comportamiento en su entorno natural. Su objetivo principal es registrar información tal como ocurre, sin intervención del investigador, lo que permite obtener datos más auténticos y menos influenciados por la interpretación subjetiva. Es ampliamente utilizada en disciplinas como la sociología, la psicología, la antropología y la educación” ([traducción propia], párrs. 1-2).

La observación es el proceso de recopilación de datos mediante la visualización de eventos, fenómenos o comportamientos estando en su entorno natural, sin interrumpir. Está puede realizarse de dos maneras: en donde el observador interactúa con los sujetos que se encuentran bajo estudio y o donde el observador solamente examina el sujeto bajo investigación. En la investigación cualitativa sirve para recolectar datos para comprender, de manera más profunda, el fenómeno en estudio.

Proceso de recolección y análisis de datos

Para la recolección y análisis de datos, se establece una serie de pasos organizados que garanticen la obtención de información precisa y relevante. En primer lugar, es necesario coordinar una cita con la persona encargada de la gestión del sistema de Recursos Humanos dentro de la empresa. Esta coordinación implicará acordar de manera anticipada la fecha, la hora y el lugar más convenientes para ambas partes. Una vez establecidos estos aspectos logísticos, se procede a la realización de la entrevista, siguiendo un guion, previamente elaborado, que permita obtener los datos necesarios para el desarrollo de la investigación.

Después de completar la entrevista, se procede con el siguiente método establecido: la observación. Para ello, se contactará, nuevamente, al encargado del área, con el objetivo de coordinar una visita a la empresa, en una fecha y hora acordadas, que permitan presenciar el desarrollo habitual de las actividades. Durante esta visita, se observarán, de manera sistemática, los procedimientos y flujos de trabajo que se llevan a cabo en el departamento, con especial atención a aquellos directamente relacionados con la problemática detectada. El propósito de esta etapa es recopilar la mayor cantidad de información posible, tanto explícita como implícita, que permita comprender a fondo la dinámica interna y así diseñar un sistema que se adapte, adecuadamente, a las necesidades de la organización.

CAPÍTULO IV: ANÁLISIS DE RESULTADOS

Dentro del análisis de resultados, se presentarán, de forma detallada, los resultados obtenidos a partir del proceso de recolección y análisis de datos. Para ello, se mostrarán los resultados con cada uno de los instrumentos aplicados, de manera que se evidencie como cada técnica contribuyó a comprender mejor la situación estudiada.

Entrevista

Se llevó a cabo el lunes 15 de julio del 2025, dentro de las instalaciones de la empresa, lo que permite una interacción directa con la persona encargada, la cual fue el principal participante de la conservación, durante la entrevista se comentaron varios temas importantes relacionados al tema principal de Recursos Humanos.

A continuación, se mencionarán los puntos más importantes discutidos, durante la entrevista.

La persona encargada informó al entrevistador que la empresa no cuenta con un sistema especializado para la gestión del personal, en su lugar utilizan hojas de Excel, para realizar los diferentes cálculos. Actualmente, los registros se hacen de manera manual todos los meses, por ejemplo, los datos de los empleados, las horas extras y el monto a pagar a cada empleado. Además, se debe verificar que haya concordancia en los datos con las hojas de cálculo en Excel. Es importante mencionar que en la empresa hay un único encargado de realizar esta tarea (la misma persona que participó en la entrevista), la cual se encarga de comunicar a los trabajadores cuando el pago se efectuó.

Después, se realizaron preguntas sobre los procesos que están relacionados con el Departamento de Recursos Humanos. Entre los procesos que se pudo abordar están el de las vacaciones, donde los empleados deben hablar con la persona encargada y luego él se encarga de aceptar o rechazar dicha solicitud. Además, para poder determinar cuántos días disponibles tiene cada empleado, se realiza el cálculo basado en los días laborados y así se gestiona el trámite realizado por el empleado.

Por otro lado, el procedimiento relacionado con las horas extras, que es similar al proceso de las vacaciones, porque en ambos casos se debe comunicar con el encargado para poder realizar una solicitud adecuada. El trabajador solicita la autorización para laborar horas adicionales y el

encargado es quien debe aceptar o rechazar la solicitud, si fue aceptada la solicitud el encargado debe, manualmente ingresar las horas extras realizadas por cada empleado en las hojas de cálculo de Excel y seguidamente se realiza el cálculo para saber el monto de las horas extras laboradas, el cual es dividir el salario entre 30 días y luego entre las horas que dura la jornada normalmente y después se multiplica por 1.5.

Por otra parte, se tiene el aguinaldo, un derecho de todo empleado que esté laborando, según lo conversado con la persona encargada, por medio de una fórmula en hojas de Excel se hace el debido cálculo, sumando el salario de 12 meses de cada trabajador, que va desde el 1 de diciembre del año anterior al 30 de noviembre del presente año, incluyendo horas extras y bonos. Además, es importante mencionar que no debe pasar la fecha límite que va del 1 de diciembre al 20 de diciembre para realizar el debido pago.

Dentro de los cálculos que se realizan en el área de Recursos Humanos, están las liquidaciones, requieren una atención especial, ya que es fundamental incluir todos los aspectos necesarios para efectuar el pago de manera correcta. Se debe calcular los días disponibles de vacaciones pendientes que tiene el empleado, calcular un aguinaldo proporcional correspondiente al tiempo laborado, se debe incluir el salario pendiente, las horas extras y bonos del empleado, se deben añadir como parte de la liquidación y también se deben calcular las deducciones legales correspondientes. Además, es importante incluir la fecha y forma de pago de la liquidación, un desglose del pago realizado y un comprobante de recibido. Es importante destacar que los cálculos mencionados anteriormente, se realizan mediante fórmulas de Excel.

Después de terminada la entrevista, se conversó con la persona a cargo acerca de los múltiples beneficios de implementar un prototipo funcional de software en el área de Recursos Humanos, al presentar la idea de implementar un sistema para la gestión de cálculos, el encargado expuso que traería diversos beneficios para la empresa. Además, el sistema a implementar podría reducir, significativamente, errores humanos y mejorar la eficiencia de los procesos, ayudando a optimizar los tiempos de trabajo.

Contar con un prototipo funcional de software, creado, especialmente para el Departamento de Recursos Humanos es una herramienta indispensable para las empresas. Se reducen los errores humanos, se agilizan los tiempos de los procesos, lo cual tendría un impacto positivo en la gestión del recurso humano y demás actividades dentro de la empresa. La entrevista aclara la necesidad

de optimizar los procesos de gestión de Recursos Humanos en la empresa, mediante herramientas apropiadas y calificadas.

En resumen, la entrevista realizada el lunes 15 de julio del 2025, con el encargado de gestionar el sistema de Recursos Humanos en la empresa, presentó las deficiencias existentes en los procesos actuales. La empresa se ayuda, en gran medida, con hojas de cálculo Excel para mantener los registros de los empleados y gestionar los procesos de horas extras, vacaciones, aguinaldos, liquidaciones, entre otros. La falta de un sistema especializado y el depender de una sola persona para realizar las tareas hacen que los riesgos sean considerables en términos de precisión y efectividad. Al implementar un sistema de Recursos Humanos, la empresa obtiene ventajas como:

Mejora la precisión: Si los datos de los empleados no se actualizan o se encuentra algún error en la información de la planilla, el sistema de Recursos Humanos garantiza la precisión de los datos, mediante la automatización de los procesos.

Reducción de errores: Al usar hojas de Excel y procesos manuales para la gestión del personal puede llevar a errores y mayor carga administrativa. El sistema de Recursos Humanos podrá reducir errores y podrá agilizar los procesos administrativos, lo que permitirá más eficiencia del tiempo.

Fomenta el compromiso: Al tener un sistema de Recursos Humanos que pueda automatizar los procesos como la gestión de vacaciones y horas extras, mejora la experiencia del trabajador, al facilitar el acceso a beneficios, lo que lleva a aumentar su compromiso dentro de la organización.

Observación Directa

Se llevó a cabo el jueves 17 de julio del 2025, dentro de las instalaciones de la empresa, aquí se espera obtener información valiosa que pueda ser de ayuda para la investigación en proceso y también tener el conocimiento del entorno actual laboral, así como también posibles oportunidades de mejora.

A continuación, se detallan los acontecimientos que se pudieron observar.

En la actualidad, la empresa no cuenta con un sistema especializado para la gestión de recursos humanos, sino que utiliza la herramienta de Excel para este propósito. En las hojas de Excel el encargado almacena todo tipo de información necesaria y se realizan los cálculos para los pagos de planilla, vacaciones, horas extras y liquidaciones. Cada mes del año tiene su respectiva hoja de cálculo donde se lleva un registro del salario de cada empleado. En el caso de las vacaciones y horas extras el administrador y su equipo de trabajo, después de evaluar la solicitud de los empleados, se encargan de aceptar o denegar la solicitud.

Si un trabajador realiza horas extras, solicita un permiso o una incapacidad, la información debe ser añadida a la hoja correspondiente para ajustar el pago. En cuanto a las liquidaciones, se manejan en un Excel aparte, en donde se calculan únicamente la liquidación de cada empleado. Además, se debe añadir la fecha del contrato y el salario del empleado. El administrador, junto con su equipo de trabajo, toma una decisión en conjunto, respecto a la liquidación correspondiente para cada empleado.

Por otro lado, el cálculo del aguinaldo en la empresa se realiza dentro del documento de Excel que contiene los registros de los pagos mensuales de la planilla, específicamente, en el mes de diciembre. De acuerdo con los datos obtenidos, se puede determinar que el tiempo estimado para llevar a cabo el cálculo de la planilla en la empresa, se estima de 40 minutos a 1 hora. El encargado de realizar el proceso una vez terminado y de haber verificado que se realizó correctamente, anuncia a los trabajadores que el pago de la planilla se efectuó satisfactoriamente.

En resumen, durante la visita a la empresa se pudo observar que la gestión de Recursos Humanos se realiza manualmente por una única persona a cargo, dependiendo de hojas de Excel. Cada mes del año, con su respectiva hoja para registrar y calcular pagos de planilla, lo que implica una carga de trabajo, retrasos y aumenta la posibilidad de errores. El aguinaldo se efectúa únicamente en el mes de diciembre, además, se observó también que el procedimiento para realizar el cálculo de planilla es bastante tedioso, requiriendo entre 40 minutos a 1 hora para completar el proceso. Estos descubrimientos resaltan la necesidad de automatizar los procesos para la gestión de Recursos Humanos en la empresa y la implementación del sistema puede corregir hallazgos como: mejorar la eficiencia, disminuir el riesgo de errores y optimizar el uso del tiempo.

CAPÍTULO V: CONCLUSIONES Y RECOMENDACIONES

Conclusiones

- El desarrollo del prototipo funcional para la gestión de Recursos Humanos en la empresa *Girasol* permitió cumplir de manera satisfactoria con los objetivos propuestos en la investigación. A partir del análisis de requerimientos, se lograron identificar las principales necesidades del área y diseñar una solución tecnológica que se ajusta a la realidad y a los procesos internos de la empresa. Este proceso permitió comprender de forma más profunda las problemáticas existentes y proponer una herramienta que aporte valor tanto operativo como organizacional.
- El prototipo demostró ser una herramienta eficaz para automatizar tareas clave como la gestión de planillas, vacaciones, incapacidades y deducciones. Con ello, se logró reducir significativamente los errores humanos que ocurrían durante los cálculos manuales y se mejoró la eficiencia del trabajo administrativo. Además, la implementación del sistema contribuye a optimizar el tiempo de trabajo del personal y facilita la generación de reportes precisos, lo que fortalece la toma de decisiones dentro del departamento de Recursos Humanos.
- Las pruebas realizadas al sistema evidenciaron que la aplicación es estable, funcional y de fácil uso, adaptándose correctamente a las necesidades de los usuarios. Su diseño amigable e intuitivo facilita la navegación, incluso para personas con conocimientos técnicos limitados, lo que garantiza una adopción efectiva dentro de la empresa.
- Este proyecto representa una contribución significativa para la empresa *Girasol*, al modernizar su forma de trabajo e impulsar la digitalización de sus procesos administrativos. El sistema no solo mejora la productividad y la organización interna, sino que también sienta las bases para una futura expansión tecnológica dentro del área de Recursos Humanos, orientada hacia la mejora continua, la transparencia y la eficiencia en la gestión del talento humano.

Recomendaciones

- Realizar un seguimiento constante durante los primeros meses de uso del sistema: Se recomienda que el departamento de Recursos Humanos, en coordinación con el encargado de soporte técnico, sea el responsable de realizar un seguimiento continuo durante los tres primeros meses posteriores a la implementación del sistema. Este seguimiento debe incluir la observación del uso real del sistema, la recopilación de comentarios de los empleados y la identificación de posibles mejoras o ajustes en las funcionalidades. La participación conjunta de estos perfiles permitirá detectar fallos de manera oportuna, optimizar los procesos y adaptar el sistema a las necesidades reales de la empresa, asegurando una adopción exitosa y un funcionamiento estable desde el inicio.

- Implementar capacitaciones periódicas a los usuarios del sistema: Se recomienda que el departamento de Recursos Humanos, en coordinación con el encargado de tecnología o soporte técnico, sea el responsable de planificar y coordinar capacitaciones periódicas dirigidas a los usuarios del sistema, incluyendo al personal administrativo y a los nuevos colaboradores que se integren a la empresa. Estas capacitaciones deben abarcar el uso correcto de las funcionalidades principales, las buenas prácticas en el manejo de la información y el aprovechamiento óptimo del sistema. La capacitación continua contribuirá a reducir errores operativos, mejorar la eficiencia de los procesos y asegurar un uso adecuado del sistema conforme el sistema evoluciona o se incorporan nuevas funcionalidades.

- Establecer políticas de respaldo y seguridad de la información: Se recomienda que el encargado de soporte técnico sea el responsable de definir, implementar y supervisar un sistema de copias de seguridad automáticas de la información, con una periodicidad diaria, asegurando que los respaldos se almacenen en una ubicación segura, como un servidor externo o servicios de almacenamiento en la nube. Estas políticas de respaldo son fundamentales para prevenir la pérdida de datos ante fallos del sistema, errores humanos o incidentes técnicos, así como para garantizar la disponibilidad y continuidad operativa del sistema.

- Actualizar la documentación técnica y de usuario: Se recomienda que el encargado del proyecto sea el encargado de actualizar la documentación técnica y el manual de usuario cada vez que se realicen cambios, mejoras o ajustes al sistema. Esta actualización debe efectuarse de manera inmediata tras cada modificación relevante, con el fin de evitar inconsistencias entre el funcionamiento real del sistema y la información documentada. Mantener la documentación actualizada facilita la capacitación del personal, el mantenimiento futuro del sistema y la correcta comprensión de la estructura y funcionamiento del sistema por parte de nuevos colaboradores.
- Ampliar gradualmente las funcionalidades del sistema: Se recomienda que el equipo desarrollador, en coordinación con el departamento de Recursos Humanos, sea el responsable de planificar y ejecutar la incorporación gradual de nuevas funcionalidades o módulos al sistema, de acuerdo con las necesidades que vayan surgiendo en la empresa. Una de las principales ampliaciones sugeridas es la inclusión de un módulo de evaluación del desempeño, el cual permitirá medir el rendimiento de los colaboradores de forma objetiva, identificar fortalezas y áreas de mejora, y apoyar la toma de decisiones relacionadas con capacitaciones, incentivos o ascensos. Se sugiere desarrollar este módulo en una segunda fase del proyecto, una vez que el sistema se encuentre completamente operativo. Para su implementación, se estima un tiempo aproximado de cuatro a seis semanas, considerando las etapas de análisis, diseño, desarrollo y pruebas, así como un costo estimado que oscila entre ₡800.000 y ₡1.200.000 colones, dependiendo del nivel de complejidad y los requerimientos específicos definidos por la empresa. Esta ampliación permitirá fortalecer la gestión del talento humano y promover la mejora continua en la organización.

CAPITULO VI: PROPUESTA

Análisis Detallado del Software por Desarrollar

En esta etapa se describen, de forma detallada, los módulos que conforman el sistema, junto con las funciones específicas de cada uno. El propósito de esta descripción es que el sistema quede claramente definido y que cada módulo cumpla con los objetivos planteados y los requisitos establecidos.

Módulo de Seguridad: Se encarga de la seguridad del sistema. Para poder ingresar al sistema se debe colocar el usuario, en este caso el número de cédula y la contraseña correspondiente. Según el rol asignado anteriormente, la persona que inicie sesión tendrá acceso a diferentes funciones, según lo que corresponda a su perfil, por ejemplo: si ingresa un usuario con el rol de administrador tendrá acceso a la mayoría de los módulos, pero si ingresa un usuario con el rol de empleado, este tendrá limitaciones con los módulos.

Módulo de Mantenimiento: Dentro de este módulo, se debe iniciar sesión con el rol únicamente de administrador para poder tener el acceso correspondiente, el cual podrá modificar, editar y eliminar información, según sea su necesidad, en cada una de las tablas correspondientes a las disponibles en la base de datos. Por ejemplo, si el administrador desea agregar un nuevo empleado o modificar la información del algún trabajador, se podrá realizar dentro de este módulo. Además, va a contener una interfaz de usuario administrador donde podrá ver y gestionar todas las funciones disponibles, también podrá realizar mantenimientos como limpiar registros antiguos de ser necesario, corregir datos incorrectos y también mantener actualizadas cada una de las tablas.

Módulo de Consultas: En este módulo se podrá consultar información más específica sobre cada colaborador que esté laborando, actualmente, dentro de la empresa y se debe iniciar sesión con el rol de administrador para poder tener el acceso correspondiente. Dentro de las consultas que se podrán hacer están: información personal del colaborador, vacaciones disfrutadas, permisos con su respectiva información, horas extras realizadas y trámites de incapacidades.

Módulo de Reportes: El objetivo de este módulo es generar reportes con la información disponible de cada empleado que labora en la empresa, con el fin de mantener los datos actualizados y organizados. Para acceder a esta funcionalidad, se debe ingresar al sistema con el rol de administrador, lo que permite un acceso completo a la información y a las opciones de generación de reportes. Sin embargo, también se contempla que los empleados, desde su propio rol, puedan generar reportes básicos relacionados con su información personal. En este caso, el acceso es más limitado, garantizando que cada usuario solo consulte lo que le corresponde, según su perfil dentro del sistema.

Módulo Calcular Planilla: El departamento de Recursos Humanos debe iniciar sesión en el sistema para poder acceder a este módulo. Una vez dentro, tendrá la posibilidad de realizar el cálculo correspondiente al pago de cada empleado que labora en la empresa. Este cálculo incluye no solo el salario base, sino también las horas extras trabajadas, los bonos percibidos y las deducciones establecidas por la ley. Además, el sistema mostrará, de manera organizada y clara, la información relacionada con los pagos de los colaboradores, facilitando el control y la transparencia en la gestión de planillas.

Módulo Calcular Vacaciones: Dentro de este módulo se debe iniciar sesión para ingresar al sistema, en el caso de los empleados, para poder solicitar vacaciones, deberán completar los campos correspondientes fecha de inicio, fecha fin, con base en los días disponibles. En el caso del administrador, este es el responsable de aprobar o rechazar las solicitudes de vacaciones en el campo denominado “Estado_Jefatura”. De manera similar, el Departamento de Recursos Humanos también revisa dichas solicitudes y toma la decisión correspondiente en el campo llamado “Estado_RH”. Además de esta función, el área de Recursos Humanos tiene la tarea de realizar el cálculo del pago de las vacaciones.

Módulo Calcular Horas: En este módulo se debe iniciar sesión para poder ingresar al sistema, en el caso de los empleados, para poder solicitar el pago de las horas extra, deberán completar los campos correspondientes fecha y cantidad horas, en el caso del administrador este se encarga de aceptar o rechazar la solicitud creada en el campo denominado “Estado_Jefatura”. De manera similar, el departamento de Recursos Humanos también revisa dichas solicitudes y toma la

decisión correspondiente en el campo llamado “Estado_RH”. Además de esta función, el área de Recursos Humanos tiene la tarea de realizar el cálculo del pago de las horas extras de cada empleado laborando dentro de la empresa.

Módulo Control Permisos: Dentro de este módulo se debe iniciar sesión para poder ingresar al sistema, en el caso de los empleados, para poder solicitar un permiso, deberán completar los campos correspondientes descripción, fecha de inicio y fecha fin, en el caso del administrador este se encarga de aceptar o rechazar la solicitud creada en el campo denominado “Estado_Jefatura”. De manera similar, el departamento de Recursos Humanos también revisa dichas solicitudes y toma la decisión correspondiente en el campo llamado “Estado_RH” y de verificar si el tipo de permiso corresponde a un permiso con goce salarial o sin goce salarial. Esto tiene como propósito llevar un control claro y preciso de cada permiso solicitado por los empleados, asegurando que la información se mantenga organizada y actualizada.

Módulo Calcular Deducciones: Dentro de este módulo, el Departamento de Recursos Humanos debe iniciar sesión para poder acceder al sistema y realizar el cálculo correspondiente a las deducciones de cada empleado de la empresa. Dichas deducciones se efectúan tomando en cuenta las normativas legales vigentes, como las establecidas por la Caja Costarricense del Seguro Social (CCSS). Además, en los casos en que corresponda y dependiendo del salario del colaborador, también se debe considerar la aplicación del impuesto sobre la renta. Este proceso garantiza que los cálculos se realicen de manera justa, transparente y conforme a las leyes establecidas.

Módulo Calcular Aguinaldo: En este módulo, el Departamento de Recursos Humanos debe iniciar sesión para poder ingresar al sistema y realizar el cálculo correspondiente al pago del aguinaldo de cada empleado de la empresa. El cálculo se debe realizar con base en la información generada desde el 1° de diciembre del año anterior, hasta el 30 de noviembre del presente año, siguiendo lo estipulado en el Código de Trabajo de Costa Rica. Es importante destacar que, conforme a la legislación, el aguinaldo debe ser cancelado a más tardar el 20 de diciembre de cada año, garantizando que los colaboradores reciban este derecho de manera puntual y conforme a la normativa vigente.

Módulo Calcular Liquidación: El Departamento de Recursos Humanos debe iniciar sesión en el sistema para acceder a este módulo, el cual está diseñado para gestionar el pago de la liquidación de cada empleado de la empresa. Este proceso se realiza siguiendo lo establecido en la legislación laboral vigente de Costa Rica, garantizando que el trabajador reciba un pago justo y completo en el momento de finalizar su relación laboral. Además, el sistema permite mantener un registro ordenado y transparente de cada liquidación realizada, lo que facilita el control administrativo y la trazabilidad de la información.

Módulo Tramitar Incapacidad: Dentro de este módulo, el Departamento de Recursos Humanos debe iniciar sesión en el sistema para poder registrar y gestionar las incapacidades de los empleados. Esta funcionalidad permite llevar un control actualizado de los casos que se presentan, garantizando un seguimiento adecuado. Además, el sistema brinda la opción de calcular el pago correspondiente a cada incapacidad, tomando en cuenta el tipo de incapacidad reportada, de manera que se respete lo establecido en la normativa laboral y se realice un proceso justo y transparente para cada colaborador.

Análisis Detallado del Hardware

Para el desarrollo e implementación del prototipo, ninguna de las partes, ni el estudiante ni la empresa, deberá incurrir en gastos adicionales, ya que ambos cuentan con el hardware necesario para llevar a cabo el proceso de desarrollo y ejecución del sistema. A continuación, se describen los componentes del hardware que se utilizarán, con el propósito de garantizar que el equipo tenga la capacidad suficiente para ejecutar el sistema sin comprometer el rendimiento ni del prototipo ni de la computadora. De esta manera, se asegura que los recursos disponibles sean adecuados para cumplir con los objetivos planteados y que el desarrollo pueda realizarse de forma eficiente y estable.

Hardware para el desarrollo del prototipo

A continuación, se describen las características de la computadora portátil que se utilizará para el desarrollo del prototipo. Este equipo constituye la herramienta principal para llevar a cabo las tareas de programación, pruebas y documentación necesarias dentro del proyecto:

Marca: Computadora portátil marca Dell.

Pantalla: Contiene una pantalla de 15.6” con resolución FHD (Full High Definition).

Procesador: Incluye un procesador Core i5.

Sistema Operativo: Windows 10 Home de 64 bits.

Memoria RAM: Dispone de 8GB de memoria RAM DDR4.

Almacenamiento: Cuenta con 1 unidad de estado sólido de 1TB.

Costo: Se menciona que el equipo no tendrá ningún costo, lo cual es un beneficio significativo.

Hardware para la producción del prototipo

A continuación, se describen las características de la computadora portátil que se utilizará para la producción del prototipo. Este equipo fue seleccionado considerando su capacidad de respuesta y compatibilidad con las herramientas de desarrollo que requiere el proyecto, asegurando un rendimiento adecuado durante las fases de programación, pruebas y documentación:

Marca: Computadora portátil marca Dell.

Pantalla: Contiene una pantalla de 15.6” con resolución FHD (Full High Definition).

Procesador: Incluye un procesador Core i5.

Sistema Operativo: Windows 10 Home de 64 bits.

Memoria RAM: Dispone de 8GB de memoria RAM DDR4.

Almacenamiento: Cuenta con 1 unidad de estado sólido de 1TB.

Costo: Se menciona que el equipo no tendrá ningún costo, lo cual es un beneficio significativo.

Análisis Detallado de las Telecomunicaciones

A continuación, se detallarán los elementos más relevantes relacionados con las telecomunicaciones. En este apartado se describen aspectos fundamentales como el tipo de red utilizada, el acceso y uso del internet, la configuración de puertos y la implementación de equipos especializados. En caso de que resulte necesario, también se incluirán los costos asociados a estos recursos, con el fin de ofrecer una comprensión más completa de los requisitos técnicos que garantizan la eficiencia y estabilidad del sistema.

Router: Sirve como el centro de comunicación de una red, permitiendo que los dispositivos se puedan comunicar entre sí y puedan acceder a internet de manera segura. En español se conoce como enrutador.

Internet: Es una red global que conectar a personas, comunidades y empresas de todo el mundo. Para el proyecto se utilizarán 600megas de internet.

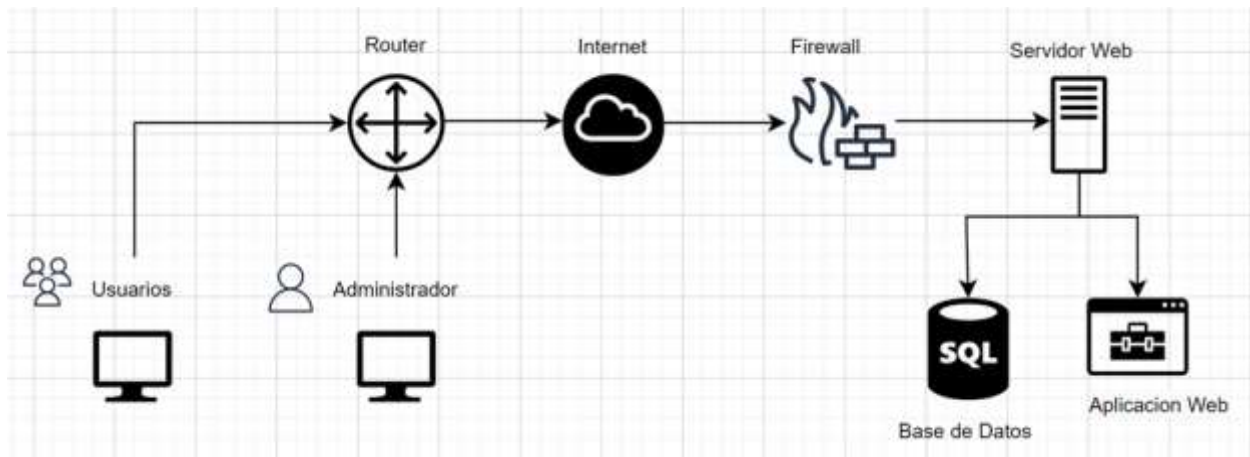
Firewall: En español como Cortafuegos, esencial para proteger una red al controlar y filtrar datos. Ayuda a prevenir ataques cibernéticos y la filtración de información confidencial.

Servidor Web: Es un dispositivo que responde a las solicitudes de los clientes, entregando el recurso solicitado como páginas web, archivos. También se encarga de procesar, almacenar y enviar información, a través de internet, cuando un cliente lo solicita.

Base de Datos: Es como un almacén gigante donde se guarda todo tipo de información de manera estructurada y organizada, permitiendo que se puedan actualizar o eliminar, según sea necesario.

Figura 1

Diagrama Telecomunicaciones



Fuente: Elaboración propia, 2025

Descripción detallada de las herramientas técnicas

En esta etapa se detallan las herramientas técnicas empleadas para el desarrollo del prototipo, incluyendo el motor de base de datos y otros recursos esenciales. Asimismo, se abordan aspectos relacionados con el licenciamiento y los costos del sistema, lo cual permite comprender mejor la viabilidad técnica y económica del proyecto.

A continuación, se presenta una descripción de cada elemento, con el fin de ofrecer una visión más clara y completa de los recursos utilizados y su importancia dentro del proceso de desarrollo.

Motor Base de Datos: Microsoft SQL Server Management Studio 2022, incluido como parte de la licencia estándar, este recurso ofrece una plataforma completa para la gestión de datos. Permite organizar la información, generar informes y realizar análisis de manera práctica, lo que facilita el control y la toma de decisiones dentro del sistema.

Entorno de Desarrollo Integrado: Visual Studio 2022, se utiliza como el principal Entorno de Desarrollo Integrado (IDE), una plataforma que reúne, en un solo lugar, diferentes herramientas diseñadas para facilitar la creación del prototipo funcional. Este entorno permite programar, probar y depurar el sistema de manera más organizada y rápida. Además, ofrece funciones avanzadas que ayudan a mejorar la eficiencia del proyecto, permiten incorporar nuevas características con facilidad y optimizan el rendimiento general del sistema.

Lenguaje de Programación: Se opta por el uso del lenguaje de programación C#, ampliamente recomendado por sus características y capacidades, las cuales permiten desarrollar un sistema eficiente y confiable. Para garantizar mayor robustez y flexibilidad, este lenguaje se implementará en conjunto con ASP.NET, dentro de un entorno Modelo–Vista–Controlador (MVC) y bajo el Framework .NET. Esta combinación facilita la organización del proyecto, mejora la escalabilidad del sistema y asegura un desarrollo más ordenado y sostenible en el tiempo.

Licencias: Se optará por el uso de licencias gratuitas y recursos de código abierto en el desarrollo de los prototipos, con el objetivo de reducir costos adicionales y minimizar los gastos asociados para la empresa. Además, este enfoque está alineado con la utilización de tecnologías que cuentan

con el respaldo de la comunidad, lo cual contribuye a fortalecer la robustez, estabilidad y confiabilidad del proyecto.

Descripción Detallada del Conocimiento Básico para Operar el Sistema

El equipo cuenta con los conocimientos básicos y las habilidades necesarias para utilizar y manejar, de manera efectiva, el prototipo funcional, ya que el sistema ha sido diseñado para ser amigable e intuitivo. Los colaboradores tienen la capacidad de interactuar con las herramientas específicas integradas en el sistema y de completar las tareas asignadas. No obstante, con el fin de garantizar una comprensión uniforme entre todos los miembros del equipo, se les proporcionará un manual básico que explique, detalladamente, el uso del prototipo funcional. De esta manera, se asegura que cada usuario pueda aprovechar plenamente las funcionalidades del sistema y trabajar de manera eficiente.

Competencia en Tecnologías de Información: Conocimiento en aplicaciones de oficina como Microsoft Word, Excel y PowerPoint para la gestión de documentos y tener familiaridad con el uso de computadoras y sistemas con Windows 10 u 11.

Operación del Sistema: Habilidad para iniciar y cerrar sesión en el sistema de manera segura. Conocimiento básico de la interfaz de usuario, incluyendo navegación por los diferentes menús, ventanas y paneles de control y capacidad para realizar operaciones básicas como búsqueda y recuperación de información.

Gestión de Datos: Tener comprensión de como poder ingresar, editar y eliminar datos de manera precisa y eficiente. Conocimiento básico de gestión de datos, incluyendo la organización, categorización y almacenamiento de información.

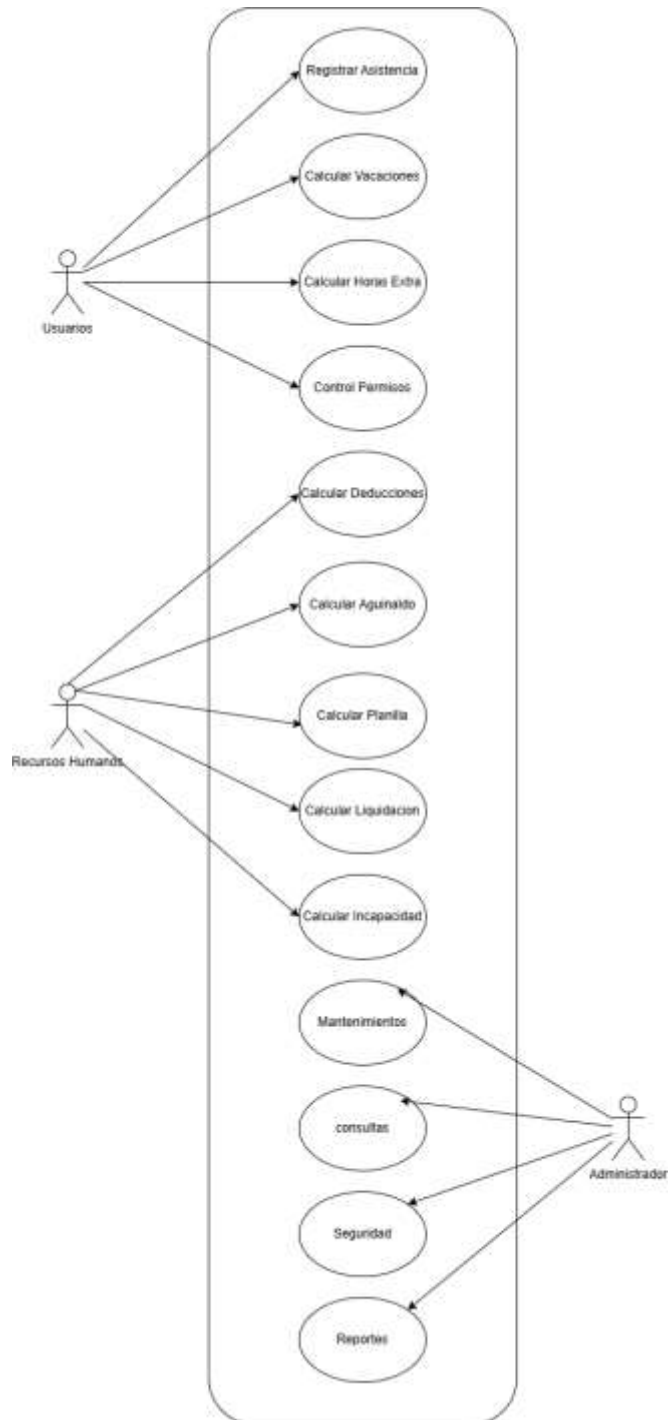
Procedimientos de Seguridad: Conocimiento básico de las mejores prácticas de seguridad informática, incluyendo la gestión de contraseñas y el reconocimiento de amenazas. Además, habilidad para seguir protocolos de seguridad establecidos para la integridad y confidencialidad de los datos.

Solución de Problemas: Capacidad para identificar y resolver problemas comunes que puedan surgir durante la operación del sistema. Conocimiento de los procesos para reportar incidencias más complejas al soporte técnico.

Manuales: Habilidad para leer e interpretar manuales y guías de usuario y capacidad para seguir instrucciones detalladas y aplicar procedimientos operativos descritos en los manuales. Instrucciones detalladas sobre cómo utilizar el sistema, realizar tareas comunes y solucionar problemas básicos.

Figura 2

Diagrama Casos de uso



Fuente: elaboración propia, 2025

Elaboración de los Casos de Uso

Los casos de uso son una técnica utilizada en el análisis y diseño de sistemas que permiten describir, de manera clara y estructurada, las interacciones entre los usuarios y otros sistemas.

Su objetivo principal es representar, de manera sencilla y comprensible, el comportamiento esperado del sistema frente a diferentes situaciones. De esta manera, ayudan a garantizar que el sistema cumpla con las necesidades del negocio.

Tabla 5

Caso de uso Calcular Planilla

Prototipo: Sistema RH	
Número Caso de Uso: 1	Nombre del Caso de Uso: Calcular Planilla
Fecha elaboración:	10/09/2025
Descripción Caso de Uso:	Realiza la planilla de empleados.
Autor caso de uso:	Joselyn Raquel Alvarado Reyes.
Actores relacionados:	Usuario encargado del sistema de recursos humanos.
Precondiciones:	Se tuvo que iniciar sesión en el sistema con un usuario activo con el rol de Recursos Humanos.
Flujo Básico	
1. El de recursos humanos ingresa al módulo "Calcular Planilla".	
2. El sistema muestra la pantalla principal.	
3. El de recursos humanos selecciona la pestaña de "Calcular Planilla".	
4. El sistema muestra la página para realizar el cálculo de planilla.	
5. El de recursos humanos debe presionar el botón de "Calcular Planilla" (FA-01) y se extrae la información de la planilla (SF-01).	
6. El sistema muestra una tabla con la información generada en la planilla mensual.	
7. Fin del caso de uso.	
Subflujos	
SF-01	El sistema consulta y extrae la información de la planilla mediante un procedimiento almacenado.
Flujos Alternos	
FA-01	Se verifica el número de planillas generadas en el mes no sea mayor a 2, de lo contrario se mostrará el mensaje "Ya existe una planilla calculada para el mes actual."
Requerimientos especiales	
No hay.	
Postcondiciones	
PC-01	La planilla del período se calcula y se guarda en el sistema de manera correcta y el resultado se muestra en pantalla para que pueda revisarse.

Fuente: elaboración propia, 2025

Tabla 6*Caso de uso Calcular Vacaciones*

Prototipo: Sistema RH	
Número Caso de Uso: 2	Nombre del Caso de Uso: Calcular Vacaciones
Fecha elaboración:	10/09/2025
Descripción Caso de Uso:	Mostrará los días disponibles de cada empleado y hara el cálculo de pago de estos días.
Autor caso de uso:	Joselyn Raquel Alvarado Reyes.
Actores relacionados:	Recursos Humanos.
Precondiciones:	Se debe iniciar sesión con un usuario activo.
Flujo Básico	
1.	El de recursos humanos ingresa al módulo "Calcular Vacaciones".
2.	El sistema muestra la pantalla principal del módulo.
3.	El de recursos humanos debe seleccionar la pestaña de "Calcular Vacaciones".
4.	El sistema muestra la pantalla para el cálculo de vacaciones y habilita una caja de texto.
5.	El de recursos humanos coloca el ID del empleado correspondiente.
6.	El de recursos humanos presiona el botón de "seleccionar empleado" para verificar que exista en el sistema (SF-01).
7.	El sistema muestra la informacion actual de las vacaciones.
8.	Fin del proceso.
Subflujos	
SF-01	El sistema valida que el ID ingresado se encuentre en los registros del sistema
Flujos Alternos	
FA-01	En caso de que no se encuentre el ID, el sistema mostrará un mensaje que indique "El ID ingresado no existe", también mostrará un botón que redirija al de recursos humanos al formulario nuevamente.
Requerimientos especiales	
	No hay.
Postcondiciones	
PC-01	Los días disponibles de vacaciones del empleado quedan calculados y la información se muestra en pantalla.

Fuente: elaboración propia, 2025

Tabla 7*Caso de uso Calcular Horas Extras*

Prototipo: Sistema RH	
Número Caso de Uso: 3	Nombre del Caso de Uso: Calcular Horas Extras
Fecha elaboración:	10/09/2025
Descripción Caso de Uso:	Calcular las horas extras de cada empleado.
Autor caso de uso:	Joselyn Raquel Alvarado Reyes.
Actores relacionados:	Recursos Humanos.
Precondiciones:	Se debe iniciar sesión con un usuario activo.
Flujo Básico	
1.	El de recursos humanos ingresa al módulo "Calcular Horas Extras".
2.	El sistema muestra la pantalla principal del módulo.
3.	El de recursos humanos debe seleccionar la pestaña de "Calcular Horas Extras".
4.	El sistema muestra la pantalla para el cálculo de las horas extras y habilita una caja de texto.
5.	El de recursos humanos coloca el ID del empleado correspondiente.
6.	El de recursos humanos presiona el botón de "seleccionar empleado" para verificar que exista en el sistema (SF-01).
7.	El sistema muestra el monto total de las horas extras.
8.	Fin del proceso.
Subflujos	
SF-01	El sistema consulta y extrae la información de la planilla mediante un procedimiento almacenado.
Flujos Alternos	
FA-01	Se verifica el número de planillas generadas en el mes no sea mayor a 2, de lo contrario se mostrará el mensaje "No se pueden crear más planillas en el mes establecido".
Requerimientos especiales	
	No hay.
Postcondiciones	
PC-01	El pago de las horas extra se calcula y se guarda en el sistema y el resultado se muestra en pantalla para su revisión.

Fuente: elaboración propia, 2025

Tabla 8*Caso de uso Control Permisos*

Prototipo: Sistema RH	
Número Caso de Uso: 4	Nombre del Caso de Uso: Control Permisos
Fecha elaboración:	10/09/2025
Descripción Caso de Uso:	Muestra un control de cuales permisos solicita cada empleado.
Autor caso de uso:	Joselyn Raquel Alvarado Reyes.
Actores relacionados:	Usuario.
Precondiciones:	Se debe iniciar sesión con un usuario activo.
Flujo Básico	
1. El usuario ingresa al módulo "Control Permisos".	
2. El sistema muestra la pantalla principal del módulo.	
3. El usuario debe seleccionar la pestaña de "Control Permisos".	
4. El sistema muestra una tabla con la información de los permisos y las siguientes opciones: Agregar (SF-01), Editar (SF-02), Eliminar (SF-03) y Detalle (SF-04).	
5. Fin del proceso.	
Subflujos	
SF-01	1. El usuario debe seleccionar la opción de "Crear".
	2. El sistema muestra varios campos vacios donde se debe agregar la información que se solicita.
	3. El usuario completa los campos correspondientes y presiona el boton "agregar".
	4. El sistema valida los datos (FA-01) y se registra en el sistema.
SF-02	1. El usuario debe seleccionar la opcion de "Editar".
	2. El sistema muestra varios campos donde se puede editar la información.
	3. El usuario realiza los cambios necesarios y presiona el botón de "guardar cambios".
	4. El sistema valida los datos (FA-02) y se guarda en el sistema.
SF-03	1. El usuario debe seleccionar la opcion de "Editar".
	2. El sistema muestra la información correspondiente a eliminar.
	3. El usuario presiona el botón "Eliminar" y se realiza el borrado en el sistema.
SF-04	1. El usuario debe seleccionar la opcion de "Detalle".
	2. El sistema muestra la información seleccionada por el administrador
Flujos Alternos	
FA-01	El sistema valida que los campos enviados sean los correctos para poder insertar los permisos en la Base de Datos, de lo contrario mostrará un mensaje indicando que hay un campo incompleto o el permiso ya existe
FA-02	El sistema valida que los campos enviados sean los correctos para poder insertar los cambios en la Base de Datos, de lo contrario mostrará un mensaje indicando que hay un campo incompleto.
Requerimientos especiales	
No hay.	
Postcondiciones	
PC-01	El sistema muestra la lista de permisos y permite al usuario crear, editar, eliminar o ver el detalle de cada permiso.

Fuente: elaboración propia, 2025

Tabla 9*Caso de uso Calcular Deducciones.*

Prototipo: Sistema RH	
Número Caso de Uso: 5	Nombre del Caso de Uso: Calcular Deducciones
Fecha elaboración:	14/09/2025
Descripción Caso de Uso:	Calcular las deducciones de cada empleado.
Autor caso de uso:	Joselyn Raquel Alvarado Reyes.
Actores relacionados:	Recursos Humanos.
Precondiciones:	Se debe iniciar sesión con un usuario activo.
Flujo Básico	
1. El de recursos humanos ingresa al módulo "Calcular Deducciones".	
2. El sistema muestra la pantalla principal del módulo.	
3. El de recursos humanos debe seleccionar la pestaña de "Calcular Deducciones".	
4. El sistema muestra la pantalla para el cálculo de las deducciones y habilita una caja de texto.	
5. El de recursos humanos coloca el ID del empleado correspondiente.	
6. El de recursos humanos presiona el botón de "seleccionar empleado" para verificar que exista en el sistema (SF-01).	
7. El sistema muestra el monto total de las deducciones.	
8. Fin del caso de uso.	
Subflujos	
SF-01	El sistema valida que el ID ingresado se encuentre en los registros del sistema.
Flujos Alternos	
FA-01	En caso de que no se encuentre el ID, el sistema mostrara un mensaje que indique "El ID ingresado no existe", tambien mostrará un botón que redirija al administrador al formulario nuevamente.
Requerimientos especiales	
No hay.	
Postcondiciones	
PC-01	El sistema muestra el monto total de las deducciones calculadas para el empleado seleccionado.

Fuente: elaboración propia, 2025

Tabla 10*Caso de uso Calcular Aguinaldo.*

Prototipo: Sistema RH	
Número Caso de Uso: 6	Nombre del Caso de Uso: Calcular Aguinaldo
Fecha elaboración:	14/09/2025
Descripción Caso de Uso:	Calcular el aguinaldo de cada empleado.
Autor caso de uso:	Joselyn Raquel Alvarado Reyes.
Actores relacionados:	Recursos humanos.
Precondiciones:	Se debe iniciar sesión con un usuario activo.
Flujo Básico	
1.	El de recursos humanos ingresa al módulo "Calcular Aguinaldo".
2.	El sistema muestra la pantalla principal del módulo.
3.	El de recursos humanos debe seleccionar la pestaña de "Calcular Aguinaldo".
4.	El sistema muestra la pantalla para el cálculo del aguinaldo y habilita una caja de texto.
5.	El recursos humanos coloca el ID del empleado correspondiente.
6.	El de recursos humanos presiona el botón de "seleccionar empleado" para verificar que exista en el sistema (SF-01).
7.	El sistema muestra el monto total del aguinaldo.
8.	Fin del caso de uso.
Subflujos	
SF-01	El sistema valida que el ID ingresado se encuentre en los registros del sistema.
Flujos Alternos	
FA-01	En caso de que no se encuentre el ID, el sistema mostrara un mensaje que indique "El ID ingresado no existe", tambien mostrará un botón que redirija al administrador al formulario nuevamente.
Requerimientos especiales	
	No hay.
Postcondiciones	
PC-01	El sistema muestra el monto total del aguinaldo calculado para el empleado seleccionado.

Fuente: elaboración propia, 2025

Tabla 11*Caso de uso Calcular Liquidación.*

Prototipo: Sistema RH	
Número Caso de Uso: 7	Nombre del Caso de Uso: Calcular Liquidación
Fecha elaboración:	19/09/2025
Descripción Caso de Uso:	Calcular la liquidacion de cada empleado.
Autor caso de uso:	Joselyn Raquel Alvarado Reyes.
Actores relacionados:	Recursos Humanos.
Precondiciones:	Se debe iniciar sesión con un usuario activo.
Flujo Básico	
1. El de recursos humanos ingresa al módulo "Calcular Liquidación".	
2. El sistema muestra la pantalla principal módulo.	
3. El recursos humanos debe seleccionar la pestaña de "Calcular Liquidación".	
4. El sistema muestra la pantalla para el cálculo de la liquidacion y habilita una caja de texto.	
5.El recursos humanos coloca el ID del empleado correspondiente.	
6. El recursos humanos presiona el botón de "seleccionar empleado" para verificar que exista en el sistema (SF-01).	
7. El sistema muestra el monto total de la liquidación.	
8. Fin del caso de uso.	
Subflujos	
SF-01	El sistema valida que el ID ingresado se encuentre en los registros del sistema.
Flujos Alternos	
FA-01	En caso de que no se encuentre el ID, el sistema mostrara un mensaje que indique "El ID ingresado no existe", tambien mostrará un botón que redirija al administrador al formulario nuevamente.
Requerimientos especiales	
No hay.	
Postcondiciones	
PC-01	El sistema muestra el monto total de la liquidacion calculado para el empleado seleccionado.

Fuente: elaboración propia, 2025

Tabla 12

Caso de uso Tramitar Incapacidad.

Prototipo: Sistema RH	
Número Caso de Uso: 8	Nombre del Caso de Uso: Tramitar Incapacidad
Fecha elaboración:	19/09/2025
Descripción Caso de Uso:	Manejar las incapacidades de cada empleado.
Autor caso de uso:	Joselyn Raquel Alvarado Reyes.
Actores relacionados:	Recursos humanos.
Precondiciones:	Se debe iniciar sesión con un usuario activo.
Flujo Básico	
1. El de recursos humanos ingresa al módulo "Tramitar Incapacidad".	
2. El sistema muestra la pantalla principal módulo.	
3. El de recursos huamnos debe seleccionar la pestaña de "Tramitar Incapacidad".	
4. El sistema muestra la información de las incapacidades y las siguientes opciones: Agregar (SF-01), Editar (SF-02), Eliminar (SF-03) y Detalle (SF-04)	
8. Fin del caso de uso.	
Subflujos	
SF-01	1. El de recursos humanos debe seleccionar la opción de "Crear". 2. El sistema muestra varios campos vacios donde se debe agregar la infromación que se solicita. 3. El de recursos humanos completa los campos correspondientes y presiona el boton "agregar". 4. El sistema valida los datos (FA-01) y se registra en el sistema.
SF-02	1. El de recursos humanos debe seleccionar la opcion de "Editar". 2. El sistema muestra varios campos donde se puede editar la infomación. 3. El de recursos humanos realiza los cambios necesarios y presiona el botón de "guardar cambios". 4. El sistema valida los datos (FA-02) y se guarda la informacion.
SF-03	1. El usuario debe seleccionar la opcion de "Eliminar". 2. El sistema muestra la información correspondiente a eliminar. 3. El administrador presiona el botón "eliminar" y se realiza el borrado en el sistema.
SF-04	1. El usuario debe seleccionar la opcion de "Detalle". 2. El sistema muestra la información seleccionada por el de recursos humanos.
Flujos Alternos	
FA-01	El sistema valida que los campos enviados sean los correctos para poder insertar los permisos en la Base de Datos, de lo contrario mostrará un mensaje indicando que hay un campo incompleto o el permiso ya existe.
FA-02	El sistema valida que los campos enviados sean los correctos para poder insertar los cambios en la Base de Datos, de lo contrario mostrará un mensaje indicando que hay un campo incompleto.
Requerimientos especiales	
No hay.	
Postcondiciones	
PC-01	El sistema muestra la lista de incapacidades y permite al usuario crear, editar, eliminar o ver el detalle de cada solicitud.

Fuente: elaboración propia, 2025

Tabla 13

Caso de uso Marcar Asistencia.

Prototipo: Sistema RH	
Número Caso de Uso: 9	Nombre del Caso de Uso: Marcar Asistencia
Fecha elaboración:	14/09/2025
Descripción Caso de Uso:	Control de asistencia de los trabajadores.
Autor caso de uso:	Joselyn Raquel Alvarado Reyes.
Actores relacionados:	Usuario.
Precondiciones:	Se debe iniciar sesión con un usuario activo.
Flujo Básico	
1. El usuario presiona el botón marcar entrada.	
2. El usuario coloca su ID correspondiente.	
3. El usuario presiona el botón de "enviar" para verificar que exista dentro del sistema (SF-01).	
4. El sistema verificar si ya existe una asistencia abierta para esta persona, si no existe puede marcar la asistencia.	
5. Luego el usuario para marcar salida, presiona el botón de "marcar salida"	
6. El usuario coloca nuevamente su ID correspondiente.	
7. El sistema verifica que haya una asistencia activa.	
8. Fin caso de uso.	
Subflujos	
SF-01	El sistema valida que el ID ingresado se encuentre en los registros del sistema.
SF-02	El sistema valida que el ID ingresado no tenga una asistencia abierta.
Flujos Alternos	
FA-01	En caso de que no se encuentre el ID, el sistema mostrara un mensaje que indique "El ID ingresado no existe", tambien mostrará un botón que redirija al usuario al formulario nuevamente.
FA-02	En caso de que el ID ingeresado ya tenga una asistencica abierta, el sistema mostrará un mensaje que indique "Ya hay una asistencia abierta para esta persona", tambien mostrará un botón que redirija al usuario al inicio.
Requerimientos especiales	
	No hay.
Postcondiciones	
PC-01	La marca de asistencia queda guardada en el sistema.

Fuente: elaboración propia, 2025

Tabla 14*Caso de uso Mantenimientos*

Prototipo: Sistema RH	
Número Caso de Uso: 10	Nombre del Caso de Uso: Mantenimiento
Fecha elaboración:	17/09/2025
Descripción Caso de Uso:	Mantener la información actualizada de la empresa.
Autor caso de uso:	Joselyn Raquel Alvarado Reyes.
Actores relacionados:	Administrador.
Precondiciones:	El administrador debe ingresar al sistema e iniciar sesión.
Flujo Básico	
1. El administrador ingresa al sistema.	
2. Dentro del menú principal el administrador debe seleccionar la opción de mantenimiento.	
3. El sistema muestra las tablas las cuales se le puede realizar un mantenimiento.	
4. El administrador selecciona la tabla correspondiente.	
5. El sistema muestra la información de la tabla seleccionada y las siguientes opciones: Crear (SF-01), Editar (SF-02), Eliminar (SF-03).	
6. Fin del caso de uso.	
Subflujos	
SF-01	1. El usuario debe seleccionar la opción de "Crear". 2. El sistema muestra varios campos vacios donde se debe agregar la infomación que se solicita. 3. El administrador completa los campos correspondientes y presiona el boton "agregar".
SF-02	1. El usuario debe seleccionar la opcion de "Editar". 2. El sistema muestra varios campos con la información a editar. 3. El administrador realiza los cambios necesarios y presiona el botón de "guardar cambios".
SF-03	1. El usuario debe seleccionar la opcion de "Editar". 2. El sistema muestra la información correspondiente a eliminar, además de una alerta que dice ¿Está seguro de eliminar este registro? 3. El administrador presiona el botón "Eliminar".
Flujos Alternos	
	No hay.
Requerimientos especiales	
	No hay.
Postcondiciones	
PC-01	La información seleccionada queda creada, actualizada o eliminada en el sistema.

Fuente: elaboración propia, 2025

Tabla 15*Caso de uso Reportes*

Prototipo: Sistema RH	
Número Caso de Uso: 11	Nombre del Caso de Uso: Reportes
Fecha elaboración:	17/09/2025
Descripción Caso de Uso:	Obtener informacion de la Base de Datos.
Autor caso de uso:	Joselyn Raquel Alvarado Reyes.
Actores relacionados:	Administrador.
Precondiciones:	El administrador debe ingresar al sistema e iniciar sesión.
Flujo Básico	
1. El administrador ingresa al sistema.	
2. Dentro del menú principal el administrador debe seleccionar la opción de consultas.	
3. El sistema muestra las tablas las cuales se le puede realizar un mantenimiento.	
4. El administrador introduce el parametro y el sistema realiza la busqueda.	
5. El administrador selecciona el reporte a exportar en formato PDF.	
6. Fin del caso de uso.	
Subflujos	
	No hay.
Flujos Alternos	
	No hay.
Requerimientos especiales	
	No hay.
Postcondiciones	
PC-01	El reporte solicitado se genera y se descarga correctamente en formato PDF.

Fuente: elaboración propia, 2025

Diseño

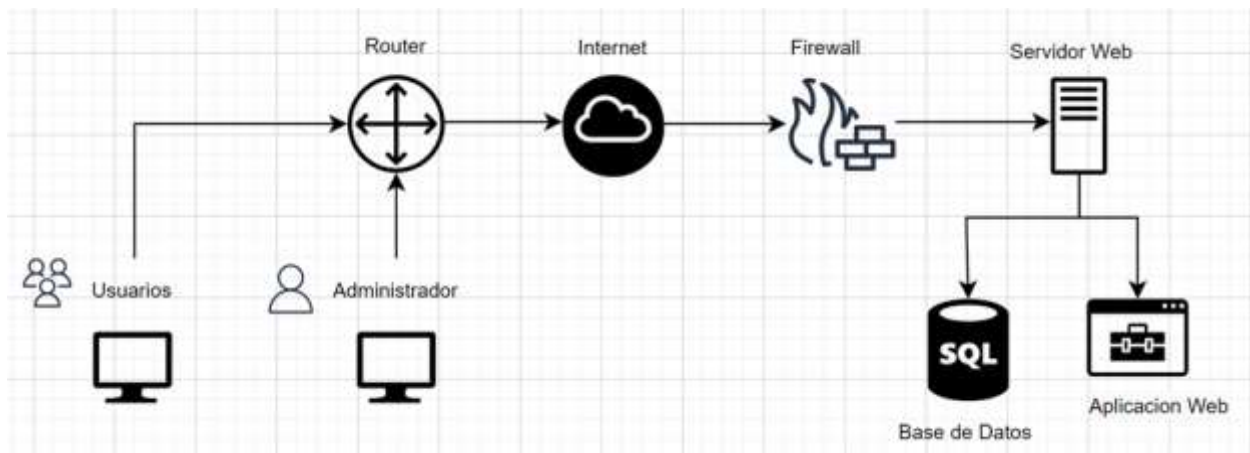
A continuación, se presentan figuras que muestran, de forma clara y detallada, la composición del sistema, sus componentes y los elementos esenciales para la implementación del prototipo. Estas ilustraciones tienen como objetivo facilitar la comprensión del funcionamiento y operación del prototipo, permitiendo que cualquier persona pueda entender cómo se organiza y se utiliza el sistema de manera práctica.

Arquitectura del sistema

La arquitectura del sistema representa la forma en que se gestiona y organiza su funcionamiento. En ella se destacan dos roles principales: el administrador y el empleado (usuario), cada uno con funciones específicas dentro del sistema. Además, se ilustra la interacción del sistema con la red, la cual integra componentes clave como el router y el firewall. El router permite la conexión a internet, garantizando el acceso y la comunicación de los distintos módulos, mientras que el firewall actúa como una barrera de seguridad que protege el sistema frente a posibles ataques externos y asegura la integridad de la información.

Figura 3

Arquitectura del sistema.



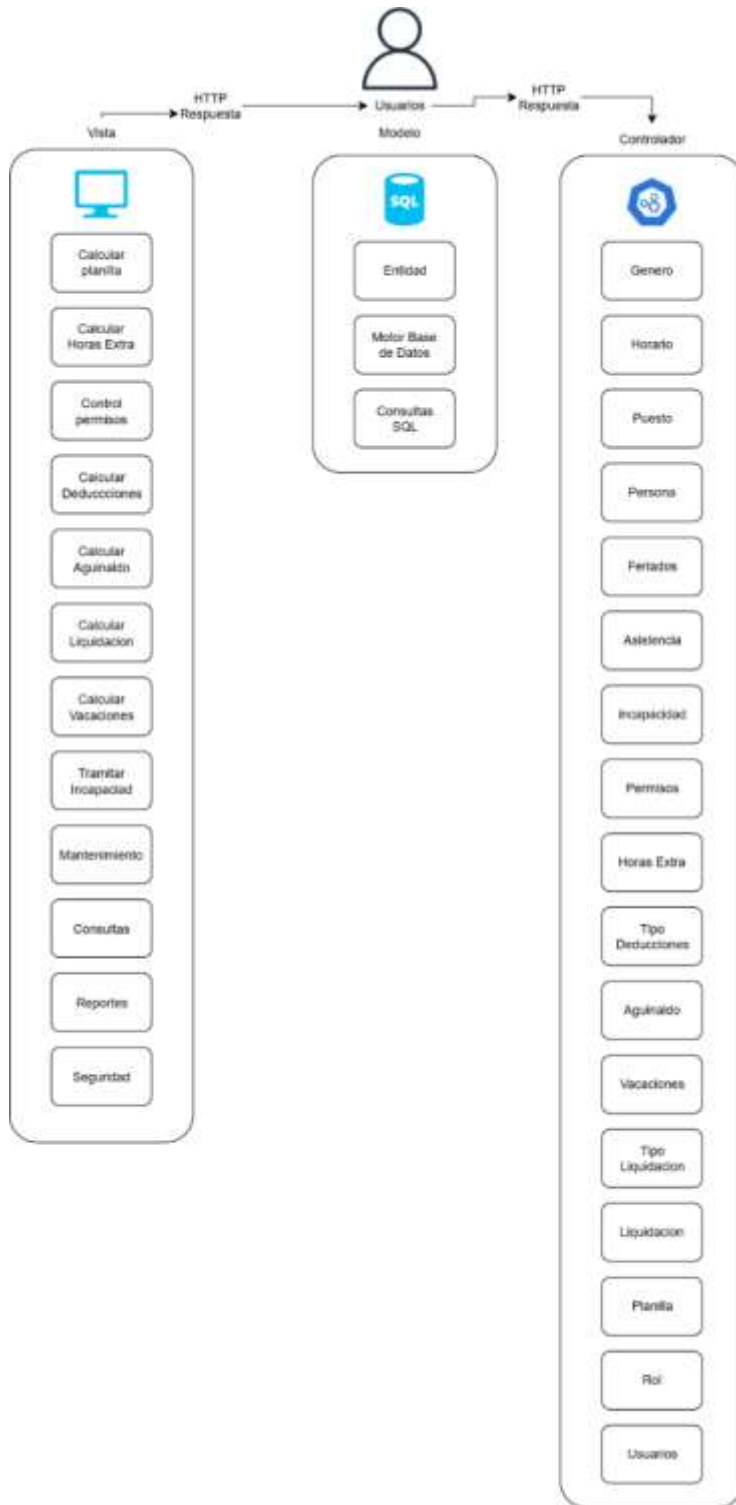
Fuente: Elaboración propia, 2025

Arquitectura del software

La arquitectura del prototipo implementa el patrón Modelo-Vista-Controlador (MVC), el cual divide el sistema en tres capas principales, cada una con su propia función específica. Esta organización permite separar la lógica de negocio, la presentación de la información y el control de las interacciones del usuario. Gracias a esta estructura modular, se facilita la incorporación de nuevo código y se simplifican las tareas de mantenimiento y actualización del sistema, asegurando un desarrollo más ordenado y eficiente. Para la implementación del prototipo se utilizaron Visual Studio como entorno de desarrollo, el lenguaje de programación C# para la lógica de la aplicación y el framework ASP.NET para la construcción de una aplicación web estructurada y escalable. Asimismo, se empleó SQL Server Management Studio para la gestión, consulta y administración de la base de datos, permitiendo un manejo eficiente de la información y facilitando las tareas de mantenimiento y control de los datos del sistema.

Figura 4

Arquitectura del Software



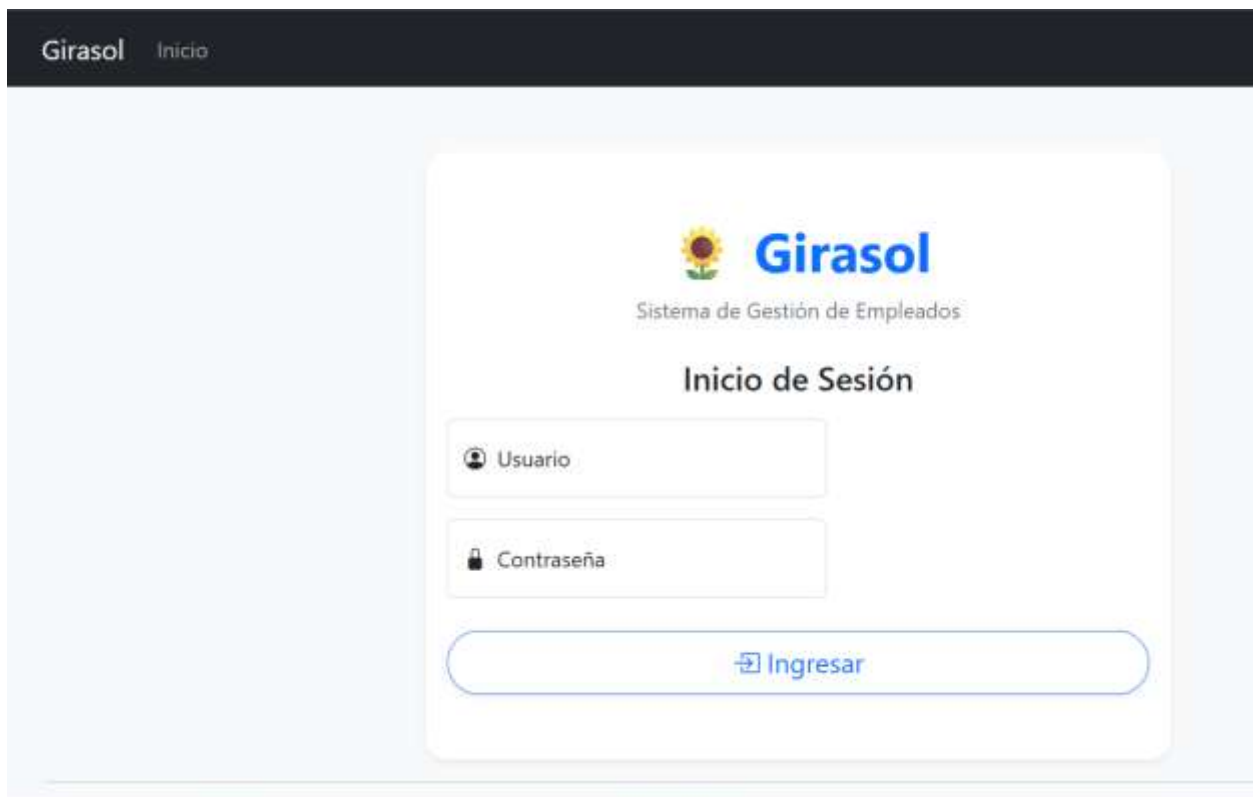
Fuente: Elaboración propia, 2025

Diseño de entradas

A continuación, se presentan las diferentes pantallas que conforman el prototipo. El propósito de este apartado es garantizar la satisfacción del usuario final, mostrando interfaces diseñadas para ser intuitivas, accesibles y visualmente agradables. De esta manera, se busca que la interacción del usuario con el sistema sea fluida, eficiente y cómoda, facilitando el uso de cada una de las funciones implementadas en el prototipo.

Figura 5

Inicio de Sesión



La pantalla de inicio de sesión funciona como punto de entrada al sistema, permitiendo que los usuarios se autenticquen de manera segura al ingresar las respectivas credenciales.

Fuente: Elaboración propia, 2025

Figura 6

Pantalla Empleados

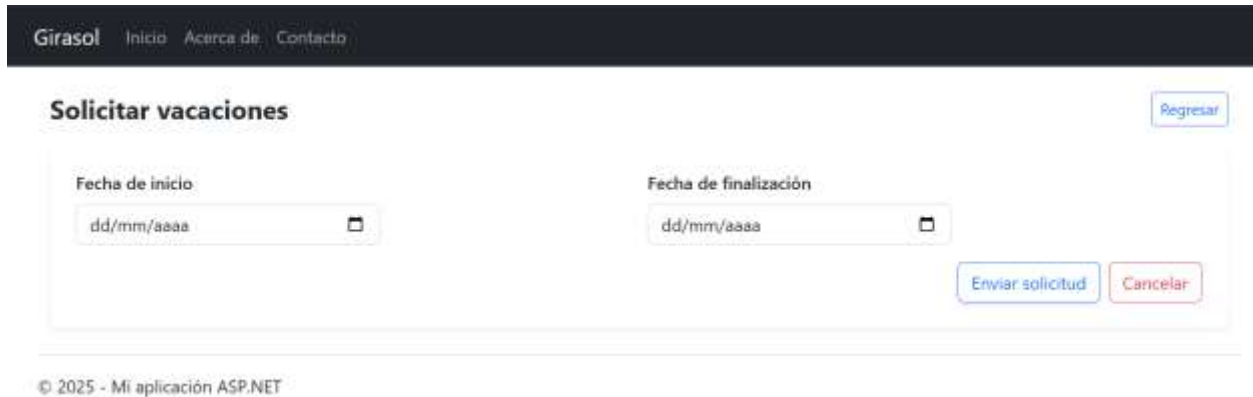


El panel del empleado permite al usuario acceder a los diferentes módulos del sistema, donde podrá realizar y gestionar sus solicitudes de forma sencilla y ordenada.

Fuente: Elaboración propia, 2025

Figura 7

Solicitar Vacaciones



The screenshot shows a web application interface for requesting vacation. At the top, a dark navigation bar contains the logo 'Girasol' and links for 'Inicio', 'Acerca de', and 'Contacto'. Below this, the main content area is titled 'Solicitar vacaciones' and includes a 'Regresar' button in the top right corner. The form consists of two date input fields: 'Fecha de inicio' and 'Fecha de finalización', both with a 'dd/mm/aaaa' placeholder and a calendar icon. At the bottom right of the form are two buttons: 'Enviar solicitud' (blue) and 'Cancelar' (red). A footer at the bottom left reads '© 2025 - Mi aplicación ASP.NET'.

La pantalla de vacaciones permite al usuario ingresar las fechas correspondientes para realizar su solicitud de vacaciones de forma sencilla.

Fuente: Elaboración propia, 2025

Figura 8

Solicitar Permiso

The screenshot shows a web application interface for requesting a permit. At the top, there is a dark navigation bar with the text 'Girasol Inicio Acerca de Contacto'. Below this, the main content area is titled 'Solicitud de permisos' and includes a 'Regresar' button in the top right corner. The form contains three input fields: a dropdown menu for 'Motivo del permiso' with the placeholder text 'Seleccione una opción', a date picker for 'Fecha de inicio' with the placeholder 'dd/mm/aaaa', and another date picker for 'Fecha de finalización' with the placeholder 'dd/mm/aaaa'. At the bottom right of the form are two buttons: 'Enviar solicitud' (blue) and 'Cancelar' (red). A footer at the bottom left reads '© 2025 - Mi aplicación ASP.NET'.

La pantalla de permisos permite al usuario ingresar la descripción y las fechas correspondientes para solicitar un permiso de manera sencilla.

Fuente: Elaboración propia, 2025

Figura 9

Solicitar Hora Extra

The screenshot shows a web application interface for requesting extra hours. At the top, there is a dark navigation bar with the text 'Girasol Inicio Acerca de Contacto'. Below this, the main content area is titled 'Solicitud de Horas Extra' and includes a 'Regresar' button in the top right corner. The form contains three input fields: a text area for 'Motivo' with the placeholder text 'Explique brevemente el motivo de las horas extra...', a date picker for 'Fecha' with the placeholder 'dd/mm/aaaa', and a text input for 'Cantidad de horas' with the placeholder 'Ejemplo: 2.5'. At the bottom right of the form are two buttons: 'Registrar' (blue) and 'Cancelar' (red). A footer at the bottom left reads '© 2025 - Mi aplicación ASP.NET'.

La pantalla de horas extra permite al usuario ingresar el motivo, la fecha y la cantidad de horas trabajadas para enviar la solicitud de forma sencilla.

Fuente: Elaboración propia, 2025

Diccionario de Datos

El diccionario de datos describe, de forma detallada, como está organizada la base de datos del sistema. Este documento incluye todas las tablas que la conforman, junto con sus columnas, los tipos de datos que utiliza cada una, así como claves primarias y foráneas que permiten relacionarlas entre sí.

Su objetivo principal es servir como guía práctica y de referencia, tanto para los desarrolladores como para los administradores de la base de datos, facilitando la comprensión del modelo de datos, asegurando la coherencia en el uso de la información y se promueve un manejo más ordenado y eficiente.

A continuación, se presentan las tablas que conforman el diccionario de datos del sistema.

Tabla 16

Tabla Persona

Persona					
Columna	Tipo dato	Tamaño	Nulos	Llaves	Descripcion
Cedula	INT	-	No null	Primary key	Cedula del empleado
Fecha_Creado	DATE	-	No null		Fecha en que se creo el empleado
Nombre	VARCHAR	50	No null		Nombre del empleado
Apellido1	VARCHAR	50	No null		Primer apellido del empleado
Apellido2	VARCHAR	50	No null		Segundo apellido del empleado
Fecha Nacimiento	DATE	-	No null		Fecha de nacimiento del empleado
Genero_idGenero	INT	-	No null	Foreing key	Genero del empleado
Estado Civil	VARCHAR	50	No null		Estado civil del empleado
Fecha Ingreso	DATE	-	No null		Fecha en que el empleado ingreso a la empresa
Cantidad Hijos	INT	-	No null		Si el empleado tiene hijos
Telefono	INT	-	No null		Telefono del empleado
Correo Electronico	VARCHAR	100	No null		Correo electronico del empleado
Estado	BIT	-	No null		Estado del empleado, activo o inactivo
Horario_idHorario	INT	-	No null	Foreing key	Horario del empleado
Puesto_idPuesto	INT	-	No null	Foreing key	Puesto del empleado

Fuente: elaboración propia, 2025

Tabla 17

Tabla Dirección

Direccion					
Columna	Tipo dato	Tamaño	Nulos	Llaves	Descripcion
idDireccion	INT	-	No null	Primary key	Numero identificador de la direccion
Provincia	VARCHAR	50	No null		Provincia de la direccion
Canton	VARCHAR	50	No null		Canton de la direccion
Distrito	VARCHAR	50	No null		Distrito de la direccion
Direccion	VARCHAR	100	No null		Direccion del empleado
Persona_Cedula	INT	-	No null	Foreing key	Cedula del empleado
Tipo_Direccion_idTipo_Direccion	INT	-	No null	Foreing key	Tipo de direccion

Fuente: elaboración propia, 2025

Tabla 18

Tabla Tipo Dirección

Tipo_Direccion					
Columna	Tipo dato	Tamaño	Nulos	Llaves	Descripcion
idTipo_Direccion	INT	-	No null	Primary key	Numero identificador de tipo direccion
Descripcion	VARCHAR	50	No null		Descripcion del tipo de direccion

Fuente: elaboración propia, 2025

Tabla 19

Tabla Vacaciones

Vacaciones					
Columna	Tipo dato	Tamaño	Nulos	Llaves	Descripcion
idVacaciones	INT	-	No null	Primary key	Numero identificador de las vacaciones
Fecha_Creado	DATE	-	No null		Fecha en que se creo la solicitud de vacaciones
Persona_Cedula	INT	-	No null	Foreing key	Cedula del empleado que toma las vacaciones
Fecha_Inicio	DATE	-	No null		Fecha de inicio de las vacaciones
Fecha_Fin	DATE	-	No null		Fecha de finalizacion de las vacaciones
Estado_Jefatura	VARCHAR	50	No null		Estado de la solicitud, aceptado o rechazado
Estado_RH	VARCHAR	50	No null		Estado de la solicitud, aceptado o rechazado, por parte de RRHH
Dias_Disponibles	INT	-	No null		Dias que tiene disponibles el empleado
Dias_Solicitados	INT	-	No null		Dias que solicita el empleado

Fuente: elaboración propia, 2025

Tabla 20

Tabla Tipo Liquidación

Tipo_Liquidacion					
Columna	Tipo dato	Tamaño	Nulos	Llaves	Descripcion
idTipo_Liquidacion	INT	-	No null	Primary key	Numero identificador del tipo de liquidacion
Descripcion	VARCHAR	100	No null		Descripcion del tipo de liquidacion

Fuente, elaboración propia, 2025

Tabla 21

Tabla Incapacidad

Incapacidad					
Columna	Tipo dato	Tamaño	Nulos	Llaves	Descripcion
idIncapacidad	INT	-	No null	Primary key	Numero identificador de la incapacidad
Fecha_Creado	DATE	-	No null		Fecha en que se creo la incapacidad
Persona_Cedula	INT	-	No null	Foreing key	Cedula del empleado
Comprobante	VARCHAR	100	No null		Comprobante de la incapacidad
Fecha_Inicio	DATE	-	No null		Fecha de inicio de la incapacidad
Fecha_Fin	DATE	-	No null		Fecha de finalizacion de la incapacidad
Tipo_Incapacidad	VARCHAR	100	No null		Tipo de incapacidad

Fuente: elaboración propia, 2025

Tabla 22**Tipo Incapacidad**

Tipo_Incapacidad					
Columna	Tipo dato	Tamaño	Nulos	Llaves	Descripcion
idTipo_Incapacidad	INT	-	No null	Primary key	Numero indentificador del tipo de incapacidad
Descripcion	VARCHAR	100	No null		Descripcion del tipo de incapacidad

Fuente: elaboración propia, 2025

Tabla 23**Tabla Genero**

Genero					
Columna	Tipo dato	Tamaño	Nulos	Llaves	Descripcion
idGenero	INT	-	No null	Primary key	Numero identificador del genero
Des_Genero	VARCHAR	50	No null		Descripcion del genero

Fuente: elaboración propia, 2025

Tabla 24**Tabla Permisos**

Permisos					
Columna	Tipo dato	Tamaño	Nulos	Llaves	Descripcion
idPermisos	INT	-	No null	Primary key	Número identificador de permisos
Fecha_Creado	DATE	-	No null	Foreing key	Fecha en que se creo el permiso
Persona_Cedula	INT	-	No null	Foreing key	Cédula del empleado
Descripcion	VARCHAR	100	No null		Descripción de los permisos
Fecha_Inicio	DATE	-	No null		Fecha de inicio del permiso
Fecha_Fin	DATE	-	No null		Fecha de finalización del permiso
Tipo_Permiso	VARCHAR	100	No null		Tipo de permiso
Estado_Jefatura	VARCHAR	50	No null		Estado del permiso, aceptado o rechazado
Estado_RH	VARCHAR	50	No null		Estado de la solicitud, aceptado o rechazado, por parte de RRHH

Fuente: elaboración propia, 2025

Tabla 25**Tabla Planilla**

Planilla					
Columna	Tipo dato	Tamaño	Nulos	Llaves	Descripcion
idPlanilla	INT	-	No null	Primary key	Numero identificador de la planilla
Persona_Cedula	INT	-	No null	Foreing key	Cedula del empleado
Fecha Creado	DATE	-	No null		Fecha en que se creo la planilla
Salario Bruto	Decimal	10,2	No null		Salario bruto del empleado
Salario Neto	Decimal	11,2	No null		Salario neto del empleado
Pago Hora Extra	Decimal	10,2	No null		Pago total de las horas extras
Tipo_Deducciones_idTipo_Deducciones	INT	-	No null	Foreing key	Extraer el tipo de deducción
Pago Deducciones	Decimal	10,2	No null		Pago total de las deducciones
Pago Impuesto Renta	Decimal	10,2	No null		Impuesto de la renta que debe rebajarse
Impuesto_Renta_idTipo_Impuesto_Renta	INT	-	No null	Foreing key	Porcentaje de Impuesto de renta correspondiente
Pago Creditos fiscales	Decimal	10,2	No null		Pago total del credito fiscal
Pago Incapacidad	Decimal	10,2	No null		Pago total de la incapacidad.
Pago Permisos	Decimal	10,2	No null		Pago total del permiso.

Fuente: elaboración propia, 2025

Tabla 26*Tabla Aguinaldo*

Aguinaldo					
Columna	Tipo dato	Tamaño	Nulos	Llaves	Descripción
idAguinaldo	INT	-	No null	Primary key	Numero identificador del aguinaldo
Persona_Cedula	INT	-	No null	Foreing key	Cédula del empleado
Fecha Creado	DATE	-	No null		Fecha en que se creo la planilla
Pago Aguinaldo	Decimal	10,2	No null		Pago total del aguinaldo

Fuente: elaboración propia, 2025

Tabla 27*Tabla Usuarios*

Usuarios					
Columna	Tipo dato	Tamaño	Nulos	Llaves	Descripción
Persona_Cedula	INT	-	No null	Foreing key	Cedula del empleado
Fecha_Creado	DATE	-	No null		Fecha en que se creo el usuario
idUsuarios	INT	-	No null	Primary key	Numero identificador del usuario
Contraseña	VARCHAR	100	No null		Contraseña del usuario
Rol_idRol	INT		No null	Foreing key	Numero identificador del rol

Fuente: elaboración propia, 2025

Tabla 28*Tabla Rol*

Rol					
Columna	Tipo dato	Tamaño	Nulos	Llaves	Descripción
idRol	INT	-	No null	Primary key	Numero identificador del rol
Descripcion	VARCHAR	50	No null		Descripcion del rol

Fuente: elaboración propia, 2025

Tabla 29*Tabla Tipo Deducciones*

Tipo_Deducciones					
Columna	Tipo dato	Tamaño	Nulos	Llaves	Descripción
idTipo_Deducciones	int	-	No null	Primary key	Numero identificador del tipo de deducciones
Descripcion	VARCHAR	100	No null		Descripcion del tipo de deducciones
Porcentaje	DECIMAL	10,2	No null		Porcentaje de la deducción

Fuente: elaboración propia, 2025

Tabla 30*Tabla Hora Extra*

Hora_Extra					
Columna	Tipo dato	Tamaño	Nulos	Llaves	Descripcion
idHora_Extra	INT	-	No null	Primary key	Numero identificador de la hora extra
Fecha_Creado	DATE	-	No null		Fecha en que se creo la solicitud de horas extra
Persona_Cedula	INT		No null	Foreing key	Cedula del empleado
Fecha	DATE		No null		Fecha de cuando se realizo la hora extra
Motivo	VARCHAR	100	No null		Motivo de la hora extra
Cantidad_Horas	INT		No null		Cantidad de la horas extras realizadas
Estado_Jefatura	VARCHAR	100	No null		Estado del permiso, aceptado o rechazado
Estado_RH	VARCHAR	100	No null		Estado de la solicitud, aceptado o rechazado, por parte de RRHH

Fuente: elaboración propia, 2025

Tabla 31*Tabla Asistencia*

Asistencia					
Columna	Tipo dato	Tamaño	Nulos	Llaves	Descripcion
idAsistencia	INT	-	No null	Primary key	Numero identificador de la asistencia
Persona_Cedula	INT	-	No null	Foreing key	Cédula del empleado
Fecha	DATE	-	No null		Fecha del día de trabajo
Hora_Inicio	TIME	-	No null		Hora en que entro a trabajar
Hora_Fin	TIME	-	No null		Hora en que salio de trabajar
Feriatos_idFeriados	INT	-	No null	Foreing key	Si es un dia festivo

Fuente: elaboración propia, 2025

Tabla 32*Tabla Tipo Impuesto Renta*

Tipo_Impuesto_Renta					
Columna	Tipo dato	Tamaño	Nulos	Llaves	Descripcion
idTipo_Impuesto_Renta	INT	-	No null	Primary key	Número identificador del tipo impuesto de renta
Descripcion	VARCHAR	100	No null		Descripción del impuesto de renta
Porcentaje	DECIMAL	10,2	No null		Porcentaje del impuesto de renta

Fuente: elaboración propia, 2025

Tabla 33*Tabla Horario*

Horario					
Columna	Tipo dato	Tamaño	Nulos	Llaves	Descripcion
idHorario	INT	-	No null	Primary key	Número identificador del horario
Descripcion	VARCHAR	50	No null		Descripción del horario, dias de la semana
Des_Horario	VARCHAR	50	No null		Descripción del horario

Fuente: elaboración propia, 2025

Tabla 34*Tabla Feriados*

Feriados					
Columna	Tipo dato	Tamaño	Nulos	Llaves	Descripcion
idFeriados	INT	-	No null	Primary key	Número identificador del feriado
Dia	INT	-			Día del feriado
Mes	VARCHAR	50			Mes del feriado
Obligatorio	BIT	-	No null		Si es o no feriado obligatorio
Descripcion	VARCHAR	50	No null		Descripción del feriado

Fuente: elaboración propia, 2025

Tabla 35*Tabla Puesto*

Puesto					
Columna	Tipo dato	Tamaño	Nulos	Llaves	Descripcion
idPuesto	INT	-	No null	Primary key	Número identificador del puesto
Descripcion	VARCHAR	100	No null		Nombre del puesto
Salario	DECIMAL	10,2	No null		Salario del puesto

Fuente: elaboración propia, 2025

Tabla 36*Tabla Liquidación*

Liquidacion					
Columna	Tipo dato	Tamaño	Nulos	Llaves	Descripcion
idLiquidacion	INT	-	No null	Primary key	Numero identificador de la liquidacion
Fecha_Creado	DATE	-	No null		Fecha en que se creo la liquidacion
Persona_Cedula	INT	-	No null	Foreing key	Cedula del empleado
Fecha_Salida	DATE	-	No null		Fecha en que el empleado salio de la empresa
Preaviso	INT	-	No null		Si el empleado dio un preaviso cuando salio de la empresa
Cesantia	DECIMAL	10,2	No null		Si el empleado aplica para cesantia
PagoLiquidacion	DECIMAL	10,2	No null		Pago total de la liquidacion
PagoVacaciones	DECIMAL	10,2	No null		Pago total de las vacaciones
Aguinaldo_idAguinaldo	INT	10,2	No null	Foreing key	Extraer el pago total del aguinaldo de la tabla Aguinaldo
Tipo_Liquidacion_idTipo_Liquidacion	INT		No null	Foreing key	Tipo de liquidacion

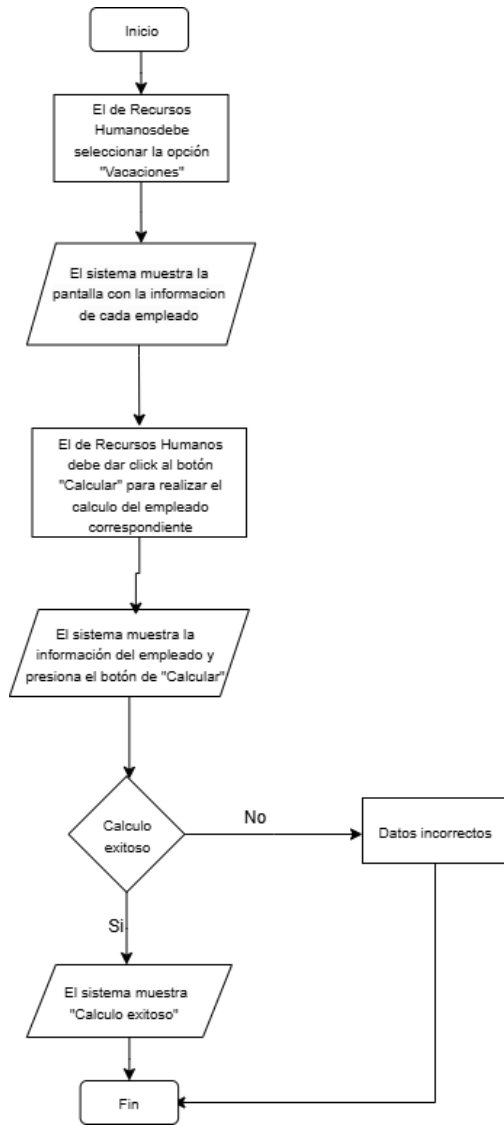
Fuente: elaboración propia, 2025

Diseño de Procesos

A continuación, se presentan los distintos diagramas de flujo que describen, de manera detallada, el funcionamiento de cada uno de los módulos de la aplicación, así como la relación entre el sistema y la base de datos. Esta relación resulta fundamental para comprender cómo se desarrolla cada proceso dentro del prototipo, desde la solicitud inicial hasta la obtención del resultado final. Los diagramas permiten visualizar la secuencia lógica de acciones y decisiones que ocurren en cada módulo, facilitando una comprensión más clara de la estructura y el comportamiento del sistema.

Figura 11

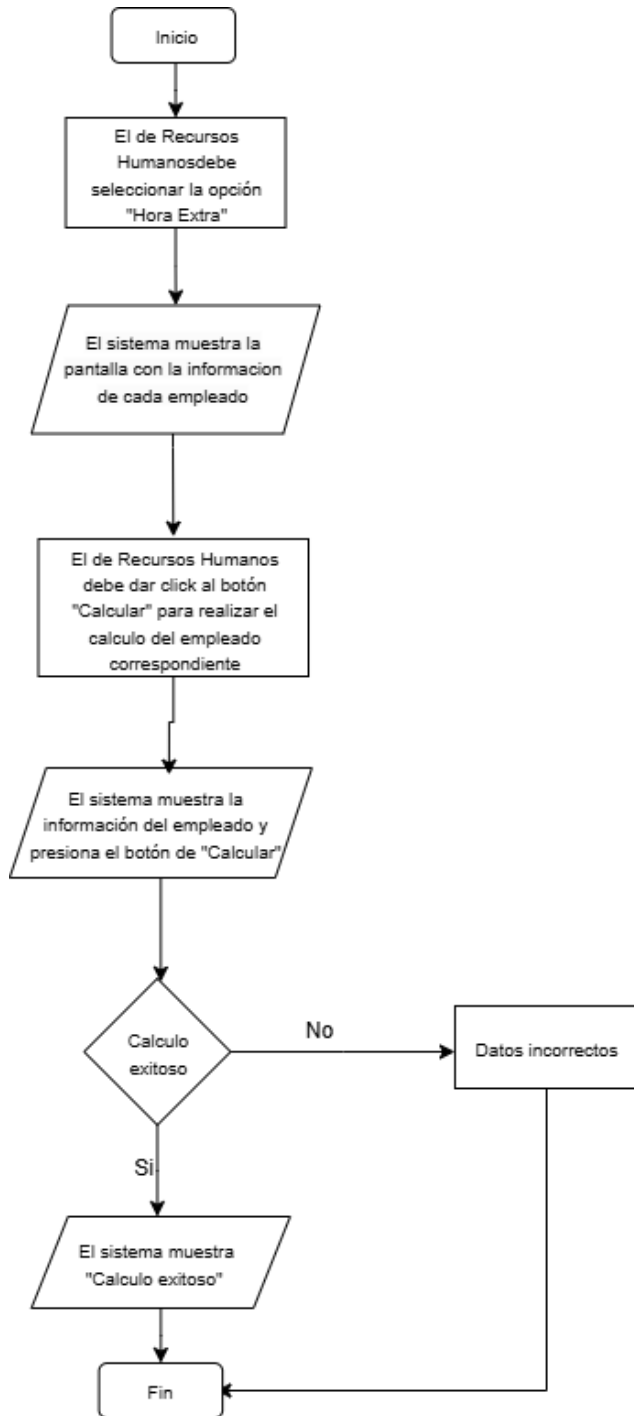
Diagrama de Flujo Calcular Vacaciones



Fuente: elaboración propia, 2025

Figura 12

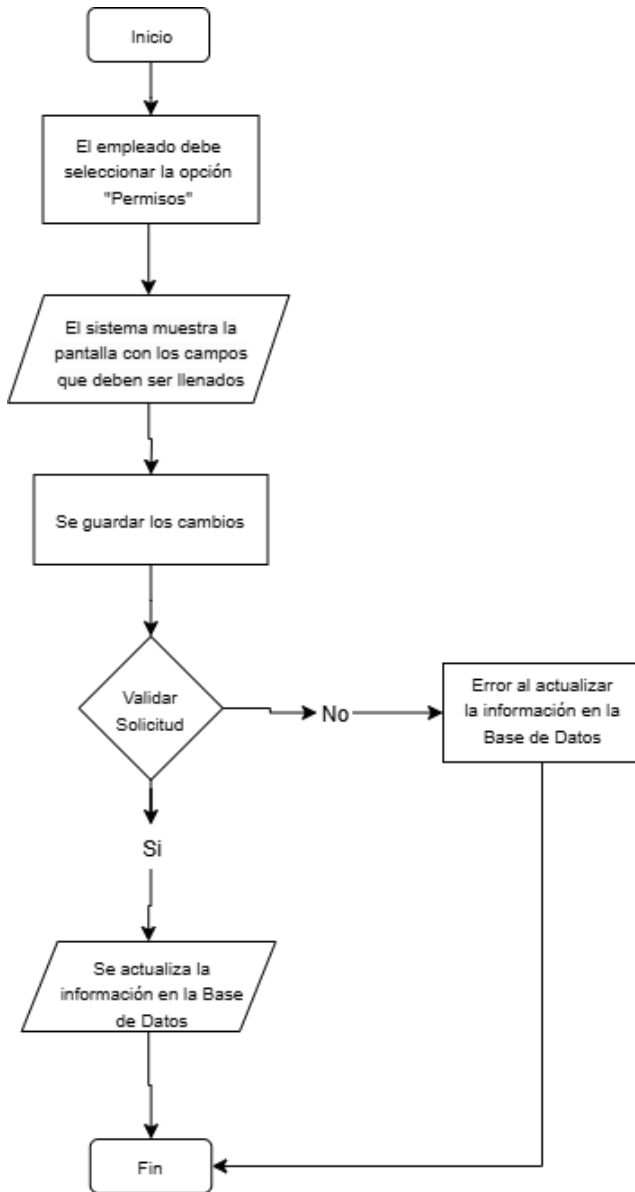
Diagrama de Flujo Calcular Hora Extras



Fuente: elaboración propia, 2025

Figura 13

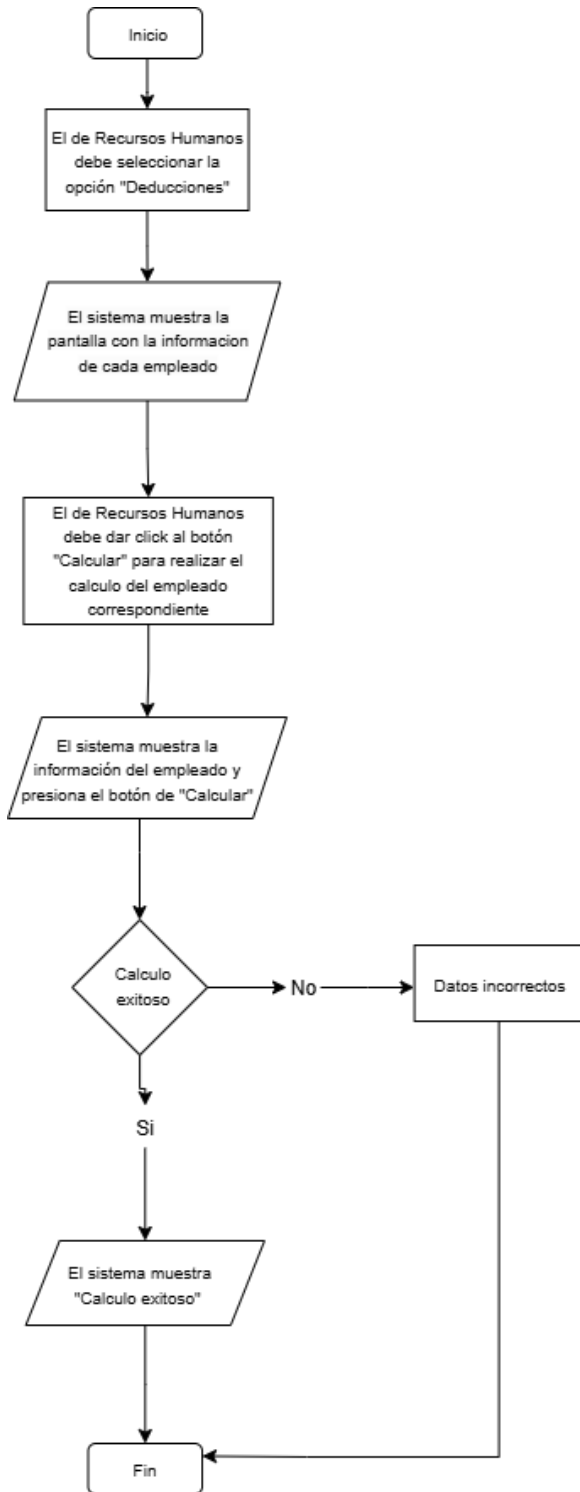
Diagrama de Flujo Control Permisos



Fuente: elaboración propia, 2025

Figura 14

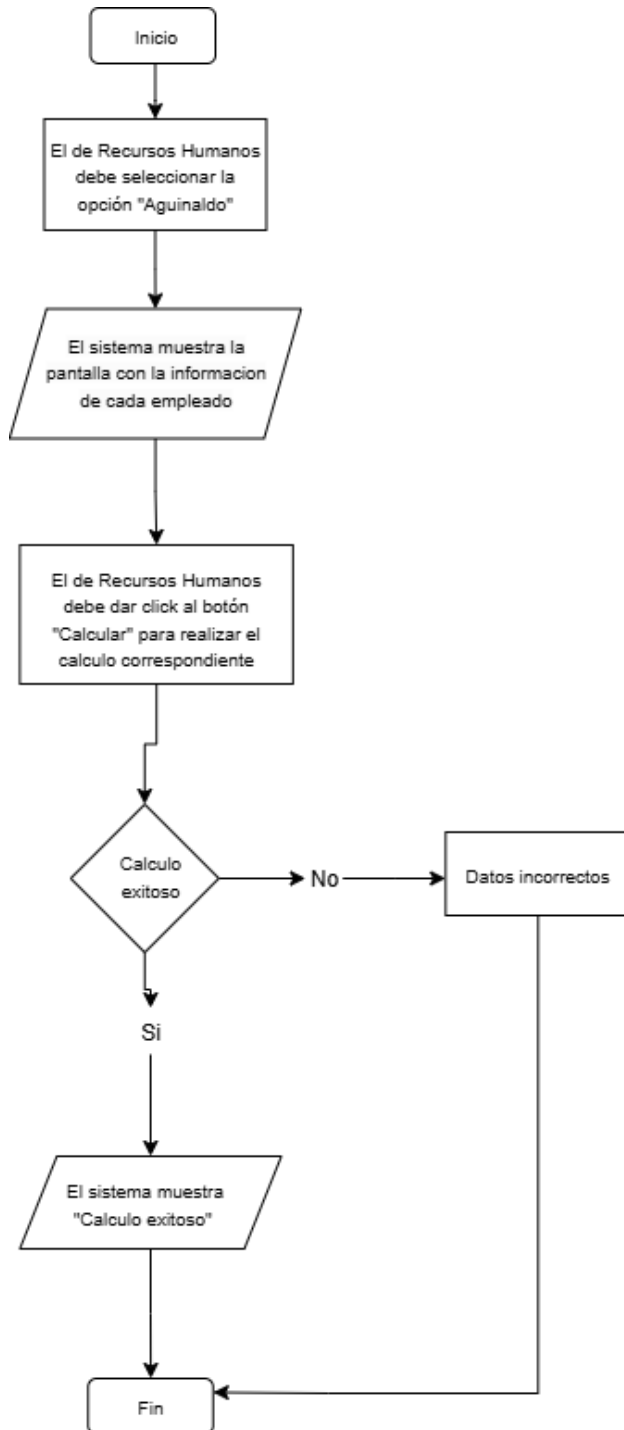
Diagrama de Flujo Calcular Deducciones



Fuente: elaboración propia, 2025

Figura 15

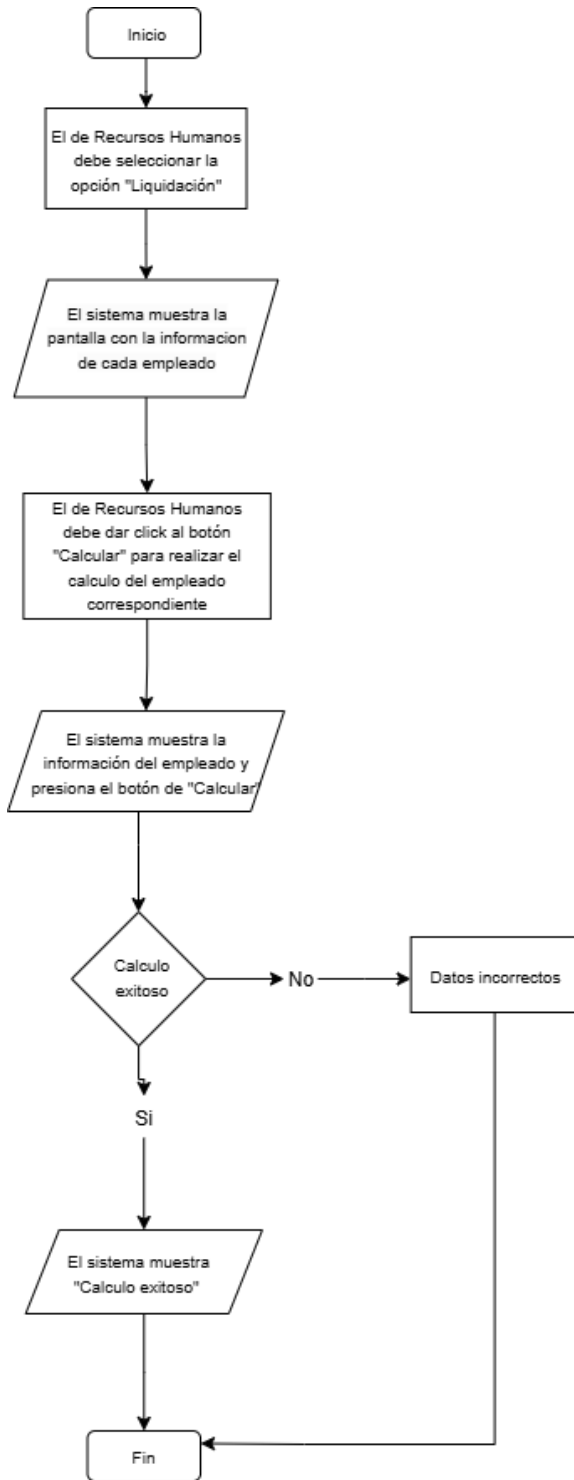
Diagrama de Flujo Calcular Aguinaldo



Fuente: elaboración propia, 2025

Figura 16

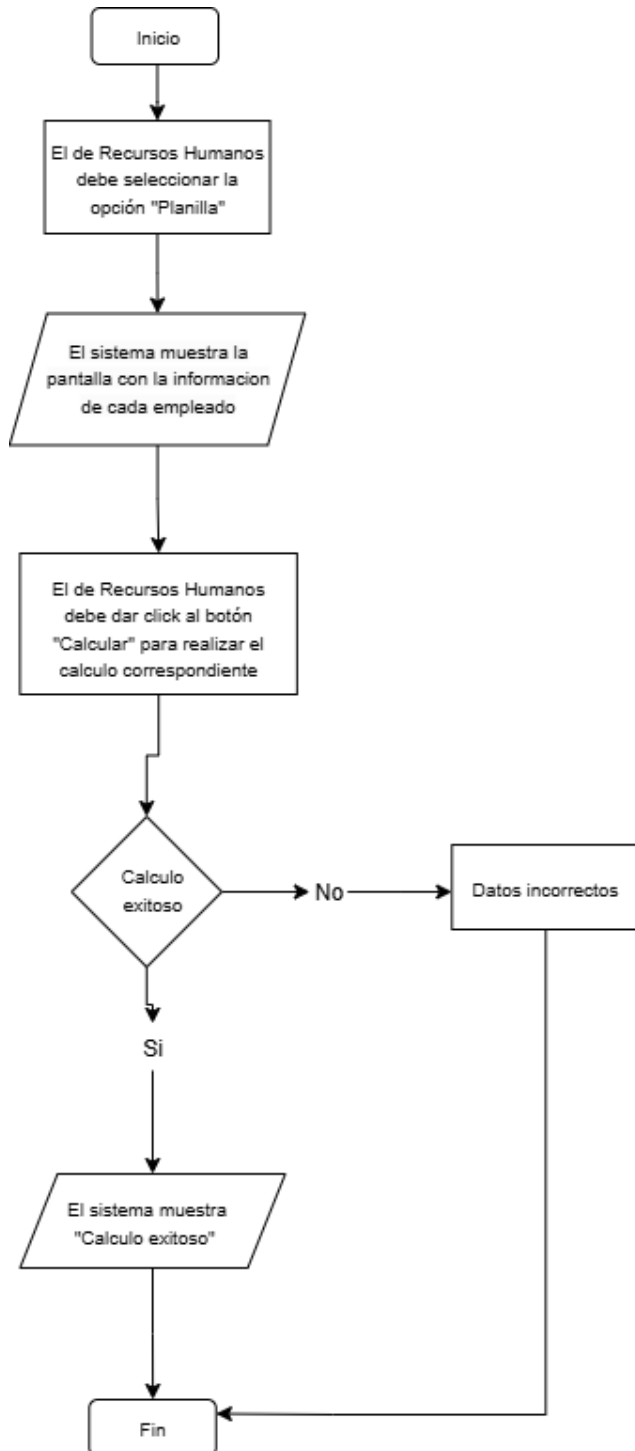
Diagrama de Flujo Calcular Liquidación



Fuente: elaboración propia, 2025

Figura 17

Diagrama de Flujo Calcular Planilla



Fuente: elaboración propia, 2025

Diseño de Salidas

A continuación, se presentan las principales salidas generadas por el prototipo. El propósito de este apartado es mostrar los resultados obtenidos, a partir del funcionamiento del sistema, evidenciando su capacidad para procesar la información y generar reportes o documentos útiles para la gestión del departamento de Recursos Humanos. Estas salidas permiten evaluar la efectividad del prototipo y garantizan la satisfacción del usuario final, al ofrecer información precisa, organizada y fácilmente interpretable.

Figura 18

Solicitar Vacaciones

Mis Solicitudes de Vacaciones Menu principal

Bienvenido, Elizabeth Alvarado Reyes
Días disponibles: 12

Crear solicitud

Fecha Salida	Fecha Ingreso	Días Solicitados	Jefatura	Recursos Humanos	Fecha Creado	Acciones
12/12/2025	12/12/2025	1	Pendiente	Pendiente	12/12/2025	Modificar Detalles

2025 - Gestión de Recursos Humanos Girasol

La pantalla de solicitudes de vacaciones permite al usuario visualizar el detalle y el estado de las solicitudes registradas en el sistema.

Fuente: elaboración propia, 2025

Figura 19

Solicitar Permisos

Fecha Inicio	Fecha Fin	Descripción	Tipo de Permiso	Jefatura	Recursos Humanos	Fecha Creado	Acciones
12/12/2025	12/12/2025	Por matrimonio del trabajador	Con goce salarial	Pendiente	Pendiente	12/12/2025	Modificar Detalles

2025 - Gestión de Recursos Humanos Girasol

La pantalla de solicitudes de permisos permite al usuario visualizar el detalle y el estado de las solicitudes registradas en el sistema.

Fuente: elaboración propia, 2025

Figura 20

Solicitar Horas Extra

Fecha Creado	Motivo	Fecha	Cantidad de Horas	Jefatura	Recursos Humanos	Acciones
12/12/2025	Temporada navideña	12/12/2025	2	Pendiente	Pendiente	Modificar Detalles

2025 - Gestión de Recursos Humanos Girasol

La pantalla de solicitudes de horas extra permite al usuario visualizar el detalle y el estado de las solicitudes registradas en el sistema

Fuente: elaboración propia, 2025

Figura 21

Agregar Incapacidad

Fecha Creado	Nombre	Primer Apellido	Segundo Apellido	Fecha Inicio	Fecha Fin	Tipo Incapacidad	Comprobante	Acciones
12/12/2025	Elizabeth	Alvarado	Reyes	11/12/2025	11/12/2025	Enfermedad		Modificar Detalles

2025 - Gestión de Recursos Humanos Girasol

La pantalla de incapacidades de empleados permite consultar las incapacidades registradas y su información asociada dentro del sistema.

Fuente: elaboración propia, 2025

Figura 22

Crear Empleado

Lista de Empleados [General PDF](#) [Nuevo empleado](#) [Menú principal](#)

Buscar por nombre, apellidos

Fecha Creado	Cédula	Nombre completo	Fecha de nacimiento	Estado civil	Hijos	Fecha de ingreso	Teléfono	Correo	Provincia	Cantón	Dictrito	Dirección
12/12/2025	116234098	Elizabeth Alvarado Reyes	01/12/2007	Soltero/a	0	01/12/2023	83631763	alvaradojos797@gmail.com	San José	San José	Carmen	300 sur pul Sar Rce
26/08/2025	117892309	Joselyn Beravides Solano	26/01/1994	Casado/a	2	26/08/2019	88675463	joselyn.ar24@gmail.com	San José	Acosta	San Ignacio de Acosta	300 sur pul Sar Rce
19/08/2025	116456789	Fabio Rodriguez Sanchez	19/01/1995	Casado/a	1	03/08/2020	74567809	pablo.ar27@gmail.com	San José	Desamparados	Desamparados	300 sur pul Sar Rce

2025 - Gestión de Recursos Humanos Girasol

La pantalla de lista de empleados permite consultar y visualizar la información de los empleados registrados en el sistema.

Fuente: elaboración propia, 2025

Diagramas UML

Figura 23

Diagrama de Clases



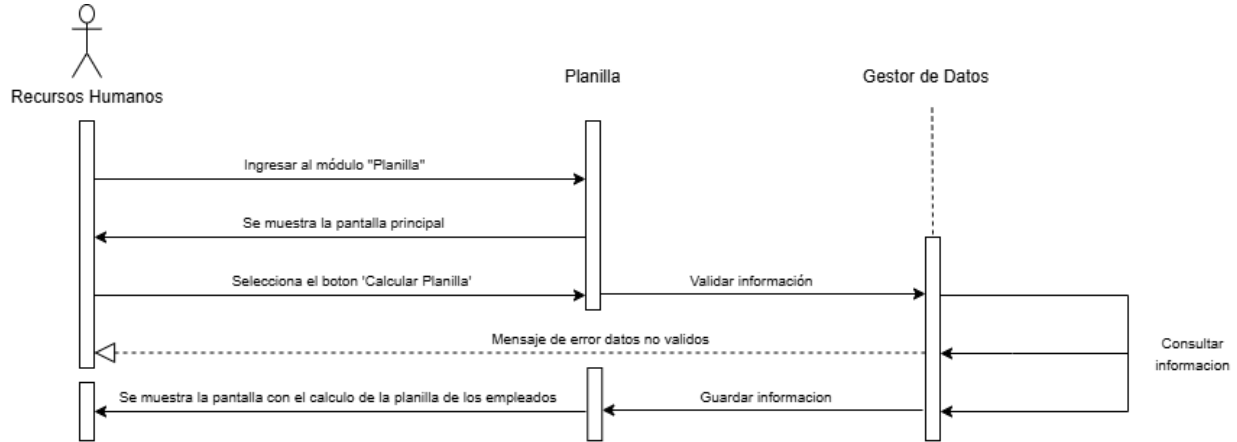
El diagrama de clases muestra los elementos principales del sistema, los datos que maneja cada uno y la forma en que se relacionan entre sí para gestionar la información de los empleados y los procesos de Recursos Humanos.

Fuente: elaboración propia, 2025

Diagrama de Secuencias

Figura 24

Generar Planilla

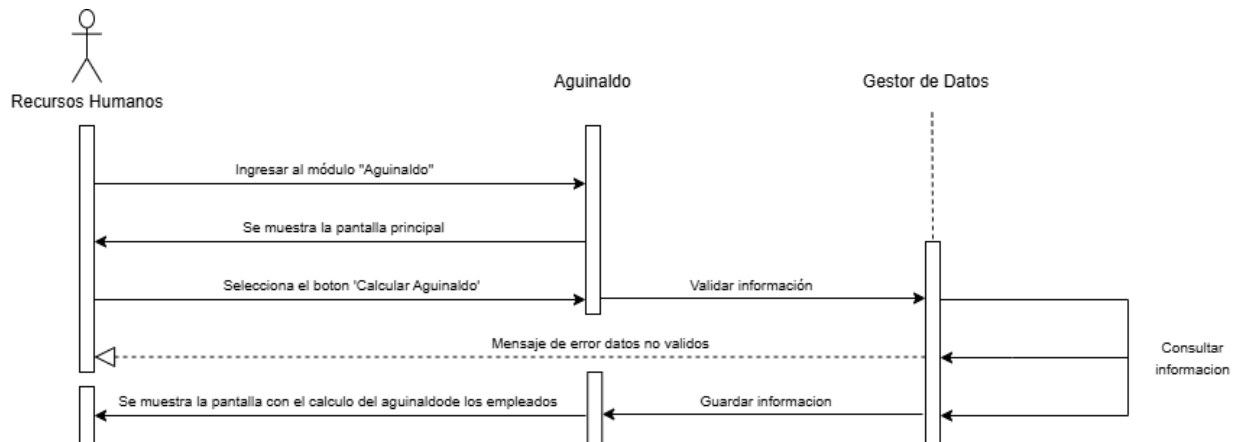


El diagrama de secuencia muestra cómo se realiza el cálculo de Planilla y la comunicación entre el área de Recursos Humanos, el sistema y el gestor de datos.

Fuente: elaboración propia, 2025

Figura 25

Calcular Aguinaldo

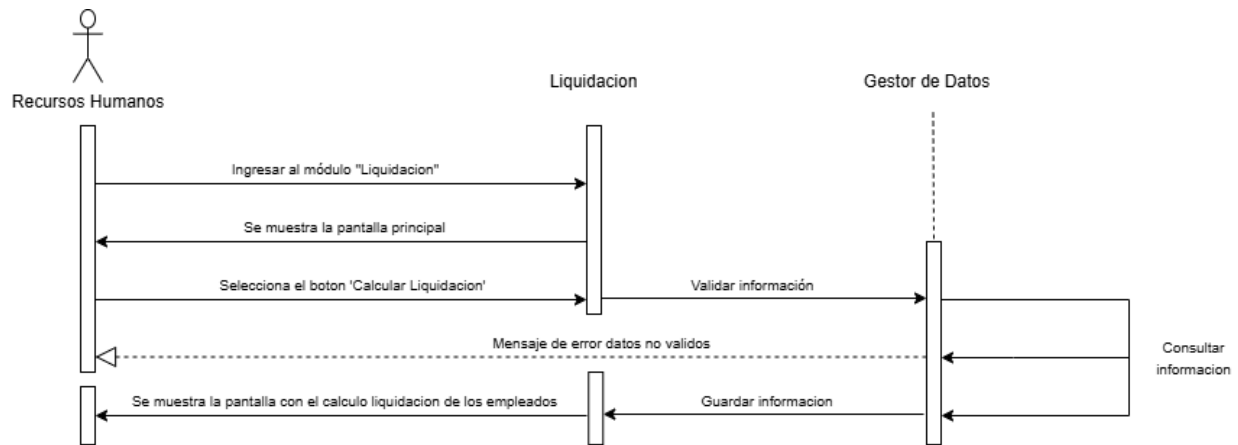


El diagrama de secuencia muestra cómo se realiza el cálculo del Aguinaldo y la comunicación entre el área de Recursos Humanos, el sistema y el gestor de datos.

Fuente: elaboración propia, 2025

Figura 26

Calcular Liquidación

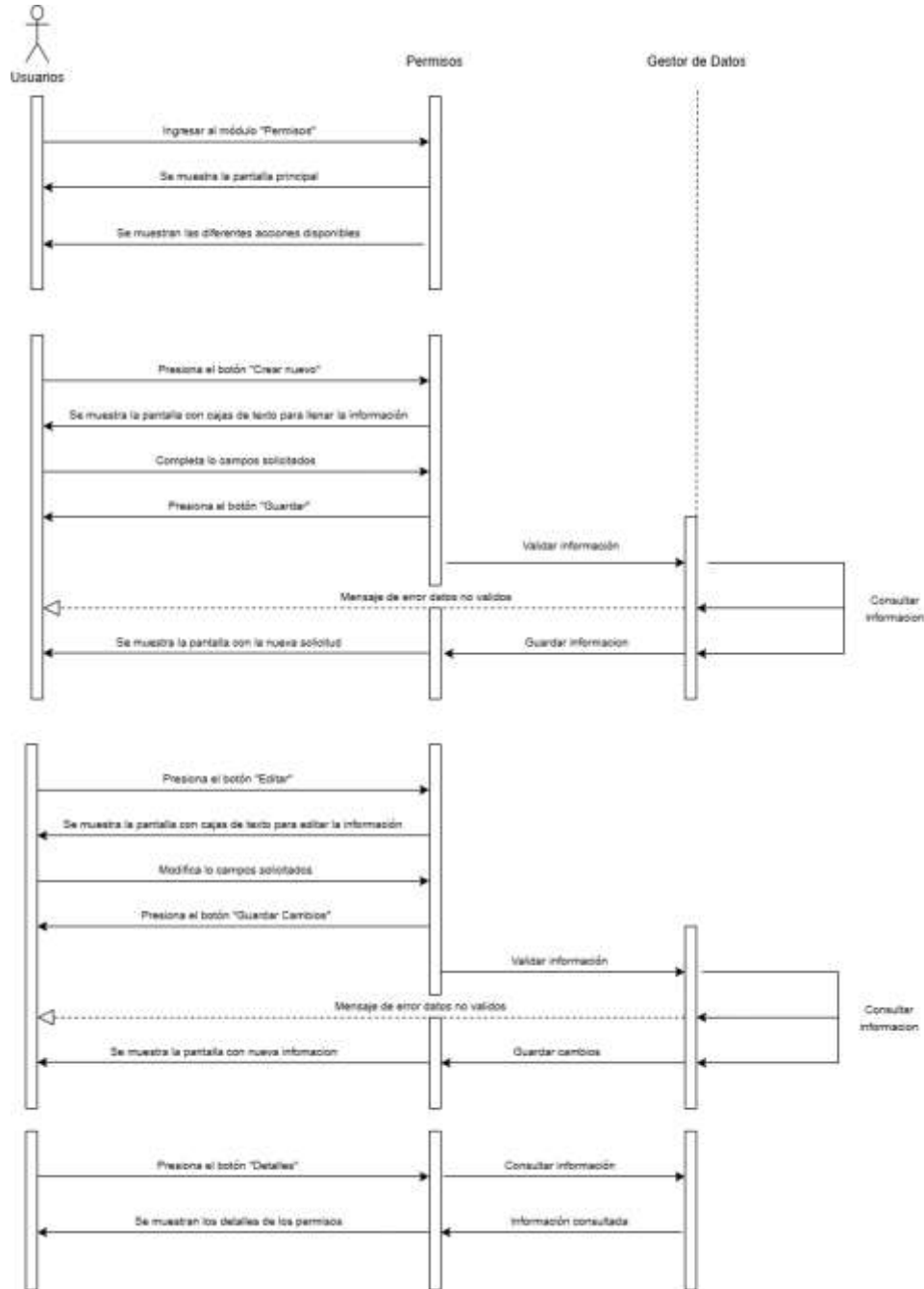


El diagrama de secuencia muestra cómo se realiza el cálculo de la Liquidación y la comunicación entre el área de Recursos Humanos, el sistema y el gestor de datos.

Fuente: elaboración propia, 2025

Figura 27

Control de Permisos

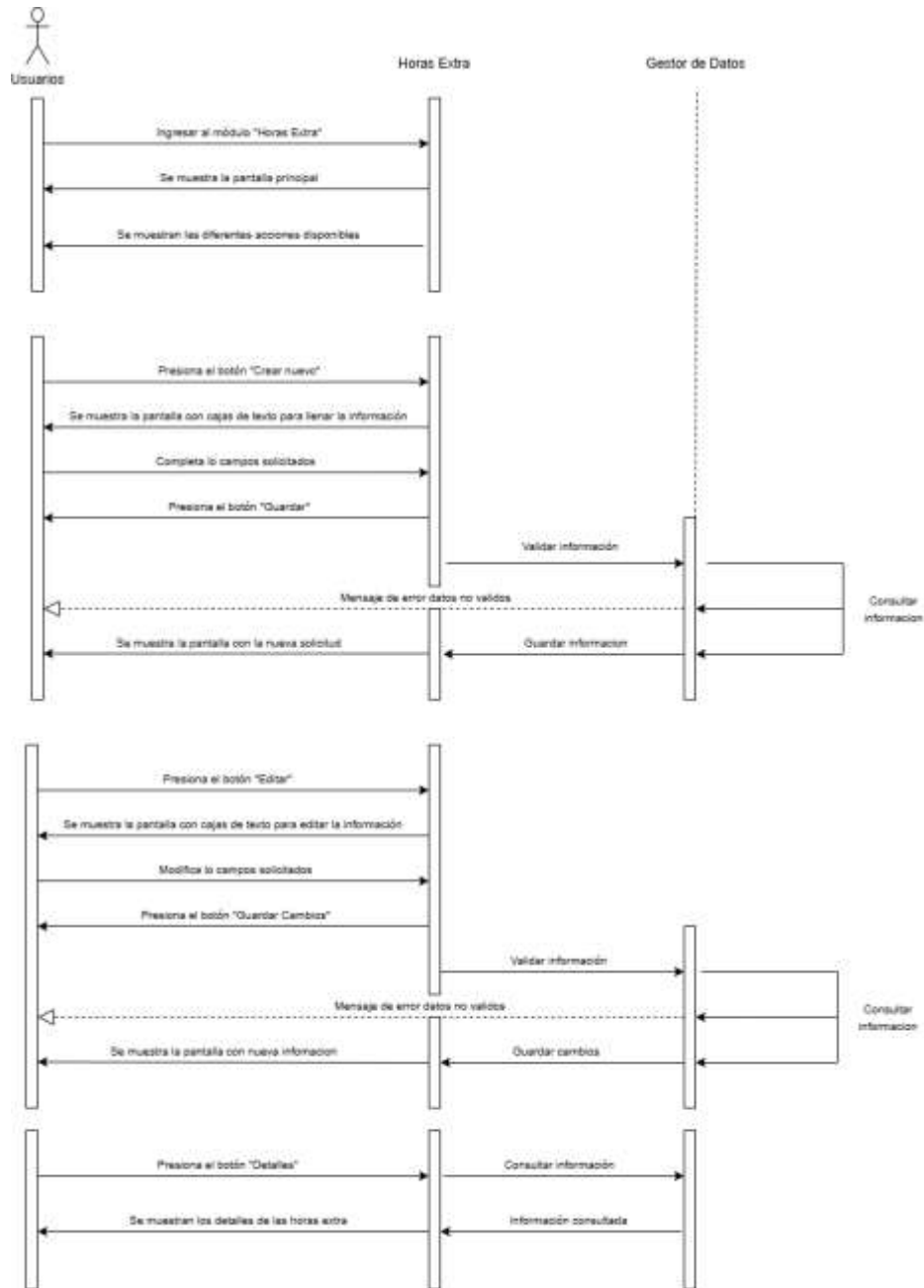


El diagrama de secuencia muestra cómo el usuario crea, consulta y edita permisos y cómo el sistema gestiona la información a través del gestor de datos.

Fuente: elaboración propia, 2025

Figura 28

Calcular Horas Extra

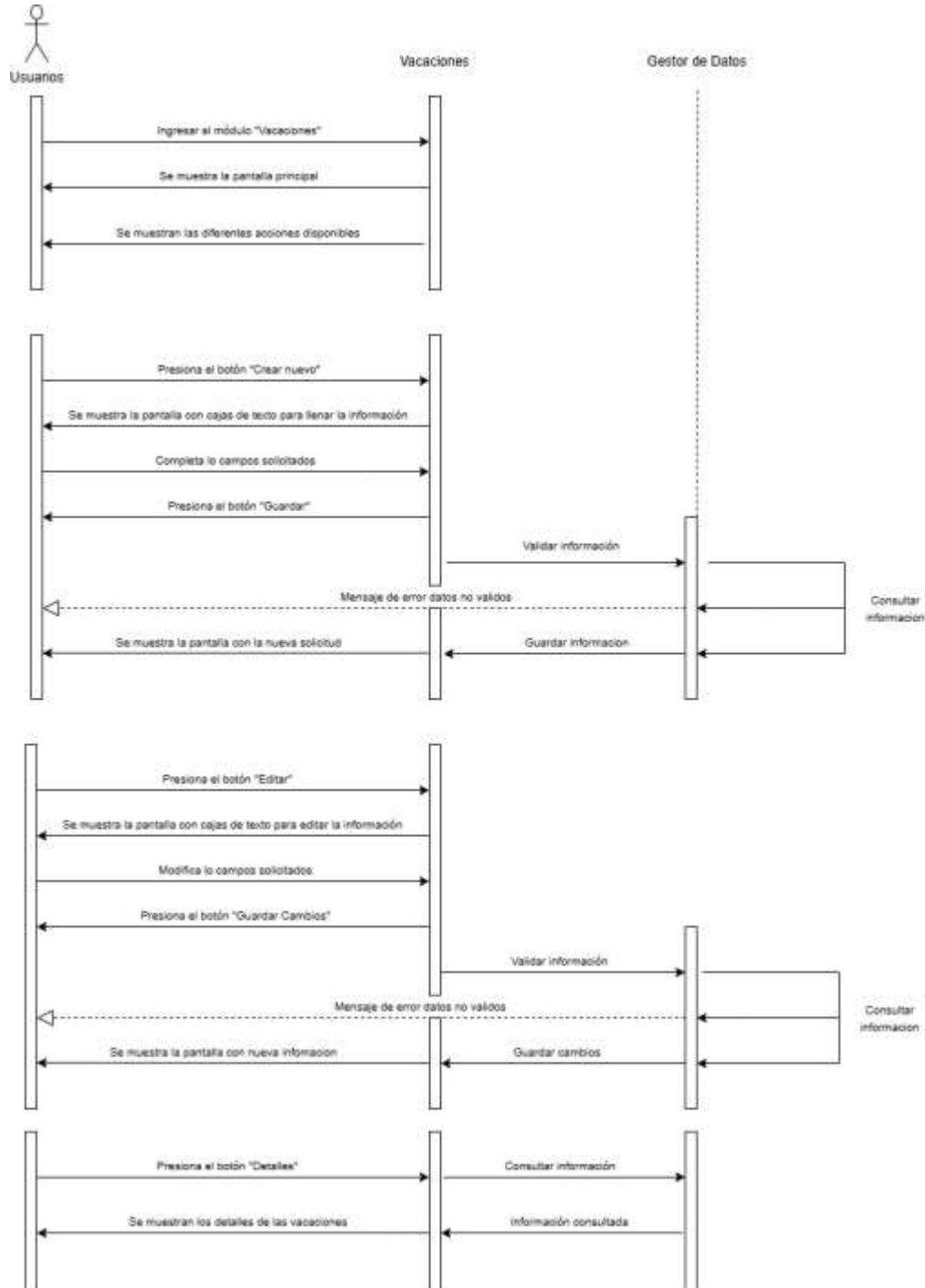


El diagrama de secuencia muestra cómo el usuario crea, consulta y edita horas extra y cómo el sistema gestiona la información a través del gestor de datos.

Fuente: elaboración propia, 2025

Figura 29

Calcular Vacaciones

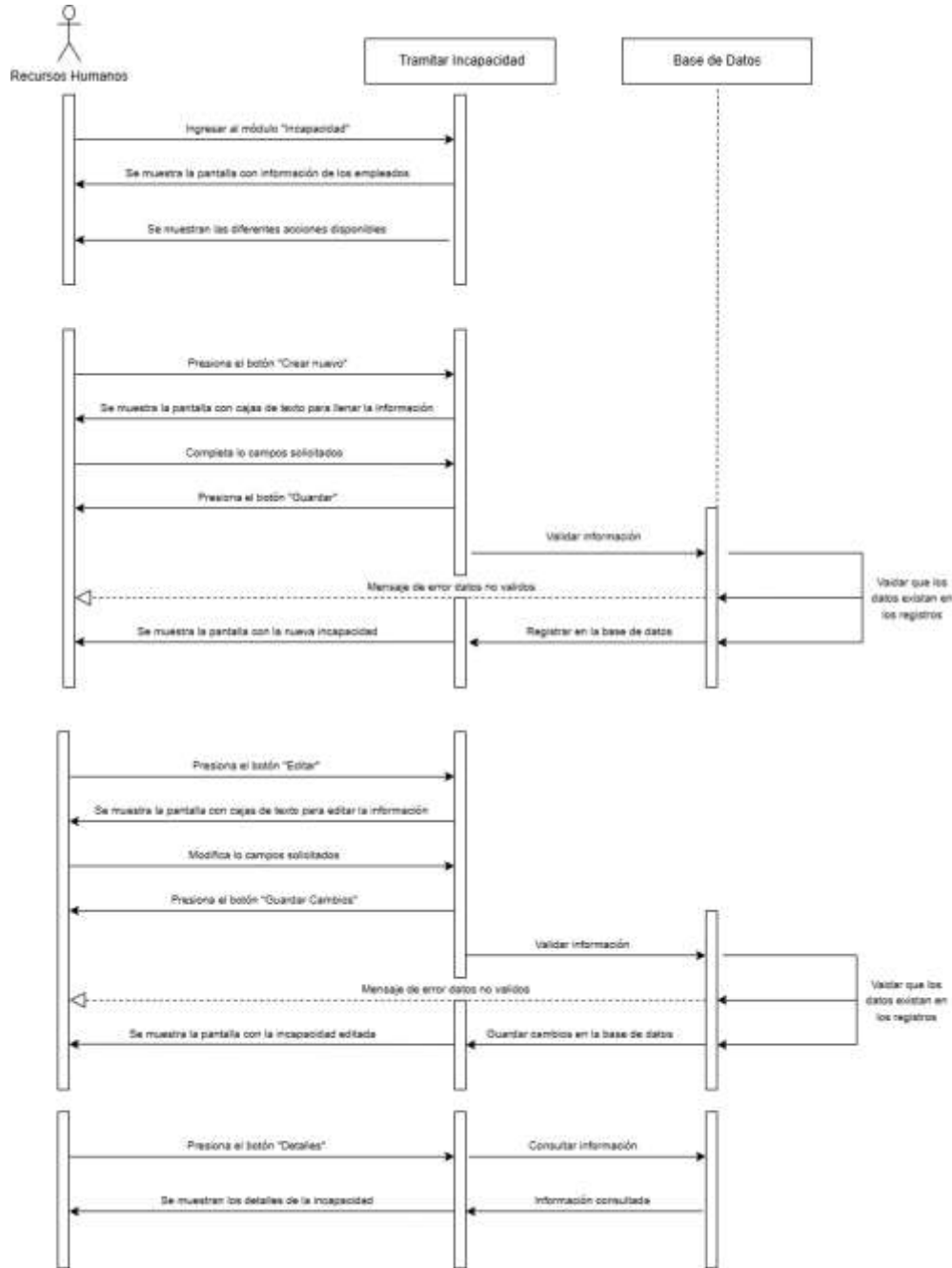


El diagrama de secuencia muestra cómo el usuario crea, consulta y edita vacaciones y cómo el sistema gestiona la información a través del gestor de datos.

Fuente: elaboración propia, 2025

Figura 30

Tramitar Incapacidad



El diagrama de secuencia muestra cómo Recursos Humanos crea, edita y consulta incapacidades y cómo el sistema gestiona la información mediante el gestor de datos.

Fuente: elaboración propia, 2025

PROGRAMACIÓN

A continuación, se presentan fragmentos representativos del código fuente del prototipo, relacionados con la gestión de datos en las distintas tablas de la base de datos. También se incluyen extractos de los controladores y vistas que conforman la arquitectura del sistema. En estos ejemplos se puede observar el proceso completo de creación, edición y visualización de datos, así como la lógica implementada en uno de los módulos específicos que integran el prototipo. Este apartado busca evidenciar el funcionamiento interno del sistema y la forma en que las diferentes capas del modelo interactúan para garantizar un flujo de información eficiente y coherente.

Figura 31

Controlador Iniciar Sesión: método Login

```
// Controller: Iniciar Login
[HttpPost]
[ValidateAntiForgeryToken]
public ActionResult Login(string IdUsuario, string Contraseña)
{
    // 1) Validaciones
    if (string.IsNullOrEmpty(IdUsuario) || string.IsNullOrEmpty(Contraseña))
    {
        TempData["Error"] = "Debe completar todos los campos.";
        return View("Index");
    }
    if (!int.TryParse(IdUsuario, out int cedula))
    {
        TempData["CedulaError"] = "La cédula debe ser un número válido.";
        return View("Index");
    }

    // 2) Hash de entrada (igual al usado al crear usuario)
    string hashEntrada = Sha256((Contraseña ?? "").Trim());

    // 3) Buscar usuario por cédula
    var usuario = GestorDB.GetUsuarioPorCedula(cedula); // asegúrate de tener este método
    if (usuario == null || !string.Equals(usuario.Contraseña, hashEntrada, StringComparison.OrdinalIgnoreCase))
    {
        TempData["CamposIncorrectos"] = "Cédula y/o contraseña incorrectas.";
        return View("Index");
    }

    // 4) Cargar datos de Persona para la sesión (nombre, etc.)
    using (var db = new PROYECTO_UIAEntities())
    {
        // Validar si ya marcó salida hoy
        DateTime hoy = DateTime.Now.Date;
        bool yaSalioHoy = db.Asistencia.Any(a =>
            a.Cedula == usuario.Cedula &&
            DbFunctions.TruncateTime(a.Fecha) == hoy &&
            a.Hora_Fin != TimeSpan.Zero);

        if (yaSalioHoy)
        {
            TempData["Error"] = "Ya marcó salida hoy. Podrá ingresar nuevamente mañana.";
            return View("Index");
        }

        var persona = db.Persona.FirstOrDefault(p => p.Cedula == usuario.Cedula);
        // guarda variables de sesión requeridas
        Session["Cedula"] = usuario.Cedula;
        Session["Rol"] = usuario.idRol;
        Session["Nombre"] = persona?.Nombre ?? "Usuario";
    }

    // 5) Redirigir según rol
    switch (usuario.idRol)
    {
        case 1: return RedirectToAction("Index", "PantallaAdministrador");
        case 2: return RedirectToAction("Index", "PantallaRecursosHU");
        case 3:
        default:
            return RedirectToAction("Index", "AsistenciaEmpleado");
    }
}

// == Helper SHA-256 (igual al usado al crear) ==
// referencia
private static string Sha256(string input)
{
    using (var sha = SHA256.Create())
    {
        var bytes = sha.ComputeHash(Encoding.UTF8.GetBytes(input ?? ""));
        var sb = new StringBuilder(bytes.Length * 2);
        foreach (var b in bytes) sb.Append(b.ToString("x2"));
        return sb.ToString();
    }
}
```

Fuente: elaboración propia, 2025

Figura 32

Controlador Permisos: vista Create

```

@model Proyecto2.Models.Permisos
ViewBag.Title = "Permisos";

<div class="container my-4">
  <!-- Encabezado -->
  <div class="d-flex justify-content-between align-items-center mb-3">
    <h4 class="fw-bold text-dark mb-0">
      <i></i> Solicitud de permisos
    </h4>
  </div>

  <!-- Formulario -->
  @using (Html.BeginForm())
  {
    @Html.AntiForgeryToken()

    <div class="card shadow-sm border-0 rounded-3 p-4">
      @Html.ValidationSummary(true, "", new { @class = "alert alert-danger" })

      <!-- Campo oculto: Tipo de permiso -->
      @Html.HiddenFor(model => model.Tipo_Permisos, new { id = "Tipo_Permisos" })

      <div class="row g-3">
        <!-- Descripción -->
        <div class="col-md-6">
          <label class="form-label fw-sembold">Motivo del permiso</label>
          <select id="Descripcion" name="Descripcion" class="form-select">
            <option value="">Seleccione una opción</option>
            <option value="Por matrimonio del trabajador" data-tipo="Con goce salarial">
              Por matrimonio del trabajador
            </option>
            <option value="Por fallecimiento de familiares directos" data-tipo="Con goce salarial">
              Por fallecimiento de familiares directos
            </option>
            <option value="Motivo personal" data-tipo="Sin goce salarial">
              Motivo personal
            </option>
          </select>
          @Html.ValidationMessageFor(model => model.Descripcion, "", new { @class = "text-danger" })
        </div>

        <!-- Fecha inicio -->
        <div class="col-md-6">
          <label class="form-label fw-sembold">Fecha de inicio</label>
          @Html.EditorFor(model => model.Fecha_Inicio, new { htmlAttributes = new { @class = "form-control", type = "date" } })
          @Html.ValidationMessageFor(model => model.Fecha_Inicio, "", new { @class = "text-danger" })
        </div>

        <!-- Fecha fin -->
        <div class="col-md-6">
          <label class="form-label fw-sembold">Fecha de finalización</label>
          @Html.EditorFor(model => model.Fecha_Fin, new { htmlAttributes = new { @class = "form-control", type = "date" } })
          @Html.ValidationMessageFor(model => model.Fecha_Fin, "", new { @class = "text-danger" })
        </div>

        <!-- Botones -->
        <div class="col-12 d-flex justify-content-end mt-3 gap-2">
          <input type="submit" value="Enviar solicitud" class="btn btn-outline-success btn-sm" />
          @Html.ActionLink("Volver", "Index", null, new { @class = "btn btn-outline-danger btn-sm" })
        </div>
      </div>
    </div>
  }
</div>

<!-- Scripts -->
@section Scripts {
  @Scripts.Render("~/bundles/jqueryval")

  <script src="https://code.jquery.com/jquery-3.6.0.min.js"></script>
  <script>
    $(function () {
      $("#Descripcion").on("change", function () {
        const tipo = $(this).find(":selected").data("tipo") || "";
        $("#Tipo_Permisos").val(tipo);
        $("#tipoDisplay").val(tipo);
      });
    });
  </script>

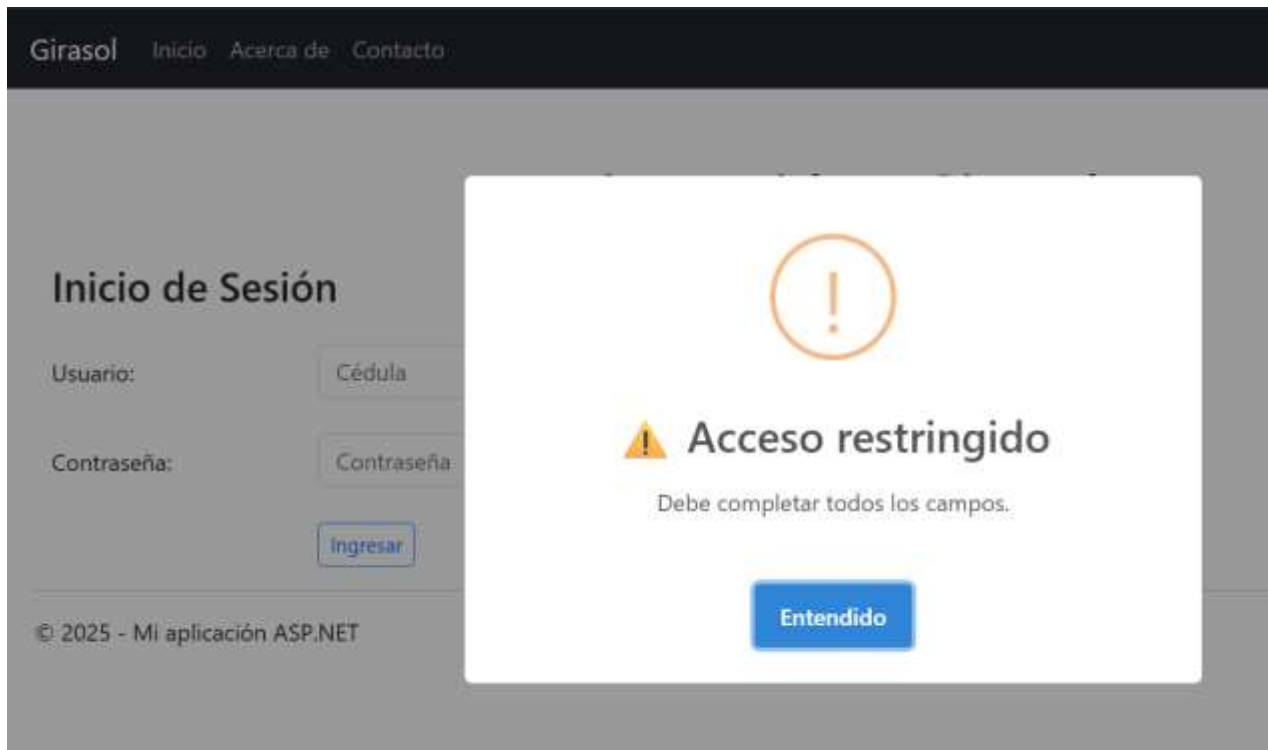
  <!-- Bootstrap Icons -->
  <link rel="stylesheet" href="https://cdn.jsdelivr.net/npm/bootstrap-icons@1.11.3/font/bootstrap-icons.css">

```

Fuente: elaboración propia, 2025

Figura 34

Validación Inicio de Sesión



Fuente: elaboración propia, 2025

Figura 35

Generar Planilla

```
[HttpPost]
[ValidateAntiForgeryToken]
0 referencias
public ActionResult Calcular()
{
    const int ID_TIPO_DEDUCCIONES = 1;
    const decimal HORAS_MES = 260m; // Ajustado segun la empresa (6 días * 10h * 4.333)
    const int HORAS_POR_DIA = 10; // Jornadas de 10 horas
    var DIA_LIBRE = DayOfWeek.Monday;

    DateTime hoy = DateTime.Now;
    DateTime inicioPeriodo = new DateTime(hoy.Year, hoy.Month, 1);
    DateTime finPeriodo = new DateTime(hoy.Year, hoy.Month, DateTime.DaysInMonth(hoy.Year, hoy.Month));

    // Tipo de deducción por defecto
    var tipoDed = db.Tipo_Deducciones.FirstOrDefault(t => t.idTipo_Deducciones == ID_TIPO_DEDUCCIONES);
    if (tipoDed == null)
    {
        ModelState.AddModelError("", $"No existe Tipo_Deducciones id={ID_TIPO_DEDUCCIONES}.");
        return RedirectToAction("Index");
    }

    var personas = db.Persona.ToList();
    var idsPuestos = personas.Select(p => p.idPuesto).Distinct().ToList();
    var puestos = db.Puesto.Where(x => idsPuestos.Contains(x.idPuesto))
        .ToDictionary(x => x.idPuesto, x => x);

    // Evitar duplicar planillas en el mismo periodo
    var yaTienePlanilla = db.Planilla
        .Where(pl => pl.Fecha_Creado >= inicioPeriodo && pl.Fecha_Creado <= finPeriodo)
        .Select(pl => pl.Cedula)
        .Distinct()
        .ToHashSet();

    var nuevos = new List<Planilla>();

    foreach (var persona in personas)
    {
        if (yaTienePlanilla.Contains(persona.Cedula)) continue;
        if (!puestos.TryGetValue(persona.idPuesto, out var puesto)) continue;

        // Validar que tenga al menos 1 mes de antigüedad para calcular
        DateTime fechaIngreso = persona.Fecha_Ingreso; // ya es DateTime (no nullable)
        var mesesTrabajados = ((hoy.Year - fechaIngreso.Year) * 12) + hoy.Month - fechaIngreso.Month;
    }
}
```

Fuente: elaboración propia, 2025

Figura 36

Calcular Vacaciones

```
[HttpPost]
[ValidateAntiForgeryToken]
0 referencias
public ActionResult Create([Bind(Include = "idVacaciones,Fecha_Inicio,Fecha_Fin")] Vacaciones vacaciones)
{
    // Tomar cédula desde la sesión (evita spoofing)
    if (Session["Cedula"] == null)
        return RedirectToAction("Index", "Home");

    vacaciones.Cedula = Convert.ToInt32(Session["Cedula"]);

    // ===== Validaciones básicas de fechas =====
    if (vacaciones.Fecha_Inicio == default(DateTime))
        ModelState.AddModelError("Fecha_Inicio", "Debe ingresar la fecha de inicio.");

    if (vacaciones.Fecha_Fin == default(DateTime))
        ModelState.AddModelError("Fecha_Fin", "Debe ingresar la fecha de fin.");

    if (vacaciones.Fecha_Inicio != default && vacaciones.Fecha_Fin != default)
    {
        if (vacaciones.Fecha_Fin.Date < vacaciones.Fecha_Inicio.Date)
            ModelState.AddModelError("Fecha_Fin", "La fecha fin no puede ser anterior a la fecha inicio.");

        // Bloquear si inicio/fin son feriados obligatorios
        if (EsFeriadoObligatorioDb(vacaciones.Fecha_Inicio))
            ModelState.AddModelError("Fecha_Inicio", "La fecha de inicio no puede ser un feriado obligatorio.");

        if (EsFeriadoObligatorioDb(vacaciones.Fecha_Fin))
            ModelState.AddModelError("Fecha_Fin", "La fecha de fin no puede ser un feriado obligatorio.");

        // Bloquear si el rango incluye algún feriado obligatorio
        for (var d = vacaciones.Fecha_Inicio.Date; d <= vacaciones.Fecha_Fin.Date; d = d.AddDays(1))
        {
            if (EsFeriadoObligatorioDb(d))
            {
                ModelState.AddModelError("", $"El rango incluye un feriado obligatorio: {d:dd 'de' MMMM}.");
                break;
            }
        }
    }
}
```

Fuente: elaboración propia, 2025

Figura 37

Calcular Horas Extra

```
[HttpPost]
[ValidateAntiForgeryToken]
public ActionResult Create([Bind(Include = "IdHora_Extra,Motivo,Fecha,Cantidad_Horas,Estado,Jefatura,Estado_RH,Cedula")] Hora_Extra hora_Extra)
{
    // Tomar cédula desde la sesión (evita spoofing)
    if (Session["Cedula"] == null)
        return RedirectToAction("Index", "Home");

    hora_Extra.Cedula = Convert.ToInt32(Session["Cedula"]);

    if (!ModelState.IsValid)
    {
        ViewBag.Cedula = new SelectList(db.Persona, "Cedula", "Nombre", hora_Extra.Cedula);
        return View(hora_Extra);
    }

    hora_Extra.Estado_Jefatura = "Pendiente";
    hora_Extra.Estado_RH = "Pendiente";

    db.Hora_Extra.Add(hora_Extra);
    db.SaveChanges();

    return RedirectToAction("Index");
}
```

Fuente: elaboración propia, 2025

Figura 38

Control Permisos

```
[HttpPost]
[ValidateAntiForgeryToken]
public ActionResult Create([Bind(Include = "IdPermisos,Descripcion,Fecha_Inicio,Fecha_Fin,Tipo_Permisos,Estado_Jefatura,Estado_RH,Cedula")] Permisos permisos)
{
    // Tomar cédula desde la sesión (evita spoofing)
    if (Session["Cedula"] == null)
        return RedirectToAction("Index", "Home");

    permisos.Cedula = Convert.ToInt32(Session["Cedula"]);

    if (!ModelState.IsValid)
    {
        ViewBag.Cedula = new SelectList(db.Persona, "Cedula", "Nombre", permisos.Cedula);
        return View(permisos);
    }

    permisos.Estado_Jefatura = "Pendiente";
    permisos.Estado_RH = "Pendiente";

    db.Permisos.Add(permisos);
    db.SaveChanges();

    return RedirectToAction("Index");
}
```

Fuente: elaboración propia, 2025

Figura 39

Calcular Deduciones

```
// --- 3) Deduciones ---
decimal baseDeduccion = salarioBruto + pagoHorasExtra;
decimal pagoDeduciones = 0m;

var pPorc = tipoDed.GetType().GetProperty("Porcentaje");
if (pPorc?.GetValue(tipoDed) is object porcObj && decimal.TryParse(porcObj.ToString(), out var porcentaje))
    pagoDeduciones += Math.Round(baseDeduccion * (porcentaje / 100m), 2, MidpointRounding.AwayFromZero);

var pFijo = tipoDed.GetType().GetProperty("MontoFijo");
if (pFijo?.GetValue(tipoDed) is object fijoObj && decimal.TryParse(fijoObj.ToString(), out var montoFijo))
    pagoDeduciones += Math.Round(montoFijo, 2, MidpointRounding.AwayFromZero);
```

Fuente: elaboración propia, 2025

Figura 40

Calcular Aguinaldo

```
[HttpPost]
[ValidateAntiForgeryToken]
0 referencias
public ActionResult Calcular()
{
    DateTime ahora = DateTime.Now;

    // Período legal: 01-Dic (año-1) a 30-Nov (año actual)
    DateTime inicioPeriodo = new DateTime(ahora.Year - 1, 12, 1);
    DateTime finPeriodo = new DateTime(ahora.Year, 11, 30, 23, 59, 59);

    // Todas las cédulas
    var cedulas = db.Persona.Select(p => p.Cedula).ToList();

    foreach (var cedula in cedulas)
    {
        // Obtener datos de la persona para verificar antigüedad
        var persona = db.Persona.FirstOrDefault(p => p.Cedula == cedula);
        if (persona == null) continue;

        DateTime fechaIngreso = persona.Fecha_Ingreso;

        // Al menos 1 mes de ingreso, no calcular aguinaldo
        var mesesTrabajados = ((ahora.Year - fechaIngreso.Year) * 12) + ahora.Month - fechaIngreso.Month;
        if (mesesTrabajados < 1 || (mesesTrabajados == 1 && ahora.Day < fechaIngreso.Day))
            continue;

        // Planillas del periodo para la persona
        var planillas = db.Planilla.Where(p => p.Cedula == cedula
            && p.Fecha_Creado >= inicioPeriodo
            && p.Fecha_Creado <= finPeriodo);

        // Ajusta estas líneas si tu Salario_Neto YA incluye horas extra (en ese caso, no sumes PagoHoras_Extra)
        decimal totalBruto = planillas.Select(p => (decimal?)p.Salario_Bruto).Sum() ?? 0m;
        decimal totalExtras = planillas.Select(p => (decimal?)p.PagoHoras_Extra).Sum() ?? 0m;

        decimal totalDevengado = totalBruto + totalExtras;
        decimal montoAguinaldo = Math.Round(totalDevengado / 12m, 2, MidpointRounding.AwayFromZero);

        // Upsert de Aguinaldo dentro del periodo
        var existente = db.Aguinaldo.FirstOrDefault(a => a.Cedula == cedula
            && a.Fecha_Creado >= inicioPeriodo
            && a.Fecha_Creado <= finPeriodo);
```

Fuente: elaboración propia, 2025

Figura 41

Registrar Asistencia

```
[HttpPost]
[ValidateAntiForgeryToken]
0 referencias
public ActionResult MarcarEntrada()
{
    if (Session["Cedula"] == null) return RedirectToAction("Index", "Home");

    int cedula = Convert.ToInt32(Session["Cedula"]);
    DateTime ahora = DateTime.Now;
    DateTime hoy = ahora.Date;

    var reg = db.Asistencia.FirstOrDefault(a => a.Cedula == cedula &&
        DbFunctions.TruncateTime(a.Fecha) == hoy);

    if (reg == null)
    {
        // Nuevo registro con Hora_Fin = 00:00:00
        reg = new Asistencia
        {
            Cedula = cedula,
            Fecha = ahora.Date,
            Hora_Inicio = ahora.TimeOfDay, // hora del sistema
            Hora_Fin = TimeSpan.Zero, // 00:00:00 hasta salida
            idFeriados = ObtenerFeriadoIdNotNull(hoy)
        };
        db.Asistencia.Add(reg);
    }
    else if (reg.Hora_Inicio == TimeSpan.Zero)
    {
        // Ya existe registro, pero sin entrada
        reg.Hora_Inicio = ahora.TimeOfDay;
        reg.Hora_Fin = TimeSpan.Zero;
        db.Entry(reg).State = EntityState.Modified;
    }

    db.SaveChanges();

    return Json(new
    {
        success = true,
        message = $"Hora de entrada marcada exitosamente: {ahora:HH:mm:ss}"
    });
}
```

Fuente: elaboración propia, 2025

Figura 42

Calcular Liquidación

```
[ApiController]
[ValidateAntiForgeryToken]
//Validar
public ActionResult Create([Bind(Include = "IdLiquidacion, Fecha_Salida, Puesto, Cesantia, Pago_Liquidacion, Pago_Vacaciones, IdLiquidación, IdTipo_Liquidacion, Cedula")] Liquidacion liquidacion)
{
    if (ModelState.IsValid)
    {
        // la fecha de salida igual a la actual
        liquidacion.Fecha_Salida = DateTime.Now;

        // Obtener datos basicos del empleado
        var persona = db.Persona.FirstOrDefault(p => p.Cedula == liquidacion.Cedula);
        if (persona == null)
        {
            ModelState.AddModelError("", "No se encontró la persona asociada.");
            return View(liquidacion);
        }

        // Obtener puesto desde la tabla Puesto
        var puesto = db.Puesto.FirstOrDefault(p => p.IdPuesto == persona.IdPuesto);
        if (puesto == null)
        {
            ModelState.AddModelError("", "No se encontró el puesto asociado a la persona.");
            return View(liquidacion);
        }

        decimal salarioMensual = puesto.Salario; // * salario promedio de Puesto
        DateTime fechaIngreso = persona.Fecha_Ingreso;
        DateTime fechaSalida = liquidacion.Fecha_Salida;

        // Calcular años trabajados (para cesantia)
        int añosTrabajados = fechaSalida.Year - fechaIngreso.Year;
        if (fechaSalida.Month < fechaIngreso.Month)
        {
            añosTrabajados -= (fechaSalida.Month < fechaIngreso.Month && fechaSalida.Day < fechaIngreso.Day)
        }

        // Calcular días de vacaciones pendientes
        var vacaciones = db.Vacaciones.FirstOrDefault(v => v.Cedula == liquidacion.Cedula);
        int diasPendientes = vacaciones != null ? vacaciones.Dias_Disponibles : 0;

        // Cálculo de vacaciones
        decimal pagoVacaciones = Math.Round((salarioMensual / 30) * diasPendientes, 2, MidpointRounding.AwayFromZero);
    }
}
```

Fuente: elaboración propia, 2025

Figura 43

Tramitar Incapacidad

```
[DataContract]
[ValidateAntiForgeryToken]
<summary>
public ActionResult Create([Bind(Include = "IdIncapacidad, Fecha_Inicio, Fecha_Fin, Tipo_Incapacidad, Cedula, Comprobante")] Incapacidad incapacidad, HttpPostedFileBase archive)
{
    // Si subieron archivo, validar y guardar
    if (archive != null && archive.ContentLength > 0)
    {
        var permitidas = new[] { ".pdf", ".jpg", ".jpeg", ".png" };
        var ext = Path.GetExtension(archive.FileName).ToLowerInvariant();

        if (!permitidas.Contains(ext))
            ModelState.AddModelError("Comprobante", "Solo se permiten PDF/JPG/PNG.");

        var maxBytes = 10 * 1024 * 1024; // 10 MB
        if (archive.ContentLength > maxBytes)
            ModelState.AddModelError("Comprobante", "El archivo no puede superar 10 MB.");

        // Si NO hay errores de archivo, guardar archivos y asignar la ruta
        if (!ModelState.ContainsKey("Comprobante"))
        {
            var raiz = ConfigurationManager.AppSettings["RUTA_COMPROBANTES"];
            if (string.IsNullOrEmpty(raiz))
                raiz = Server.MapPath("~/Uploads/Comprobantes");

            var subcarpeta = Path.Combine("Incapacidad", DateTime.Now.ToString("yyyy"), DateTime.Now.ToString("MM"));
            var carpetaDestino = Path.Combine(raiz, subcarpeta);
            Directory.CreateDirectory(carpetaDestino);

            var nombreArchivo = $"{DateTime.Now:yyyyMMddHHmmssfff}_{Guid.NewGuid():N}{ext}";
            var rutaFisica = Path.Combine(carpetaDestino, nombreArchivo);
            archive.SaveAs(rutaFisica);

            // Ruta para BD
            string rutaParaBD;
            var rootSite = Server.MapPath("~/");
            if (raiz.StartsWith(rootSite))
            {
                var relativa = rutaFisica.Substring(rootSite.Length).TrimStart('\\', '/');
                rutaParaBD = "/" + relativa.Replace("\\", "/");
            }
            else
            {
                rutaParaBD = rutaFisica;
            }
        }
    }
}
```

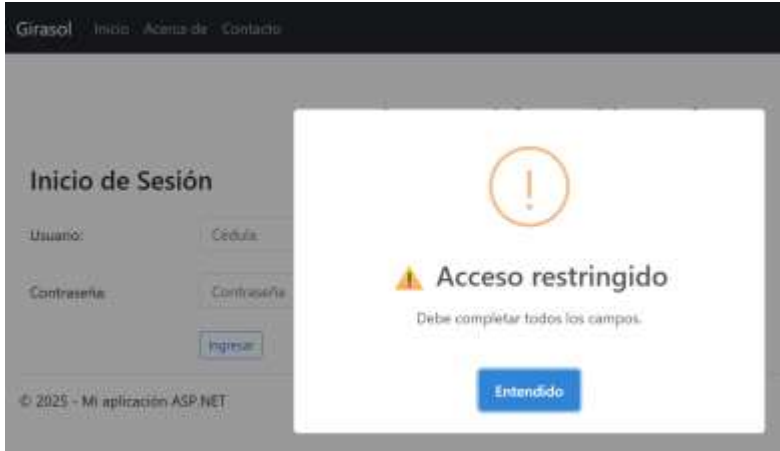
Fuente: elaboración propia, 2025

PRUEBAS

En esta sección se realizarán distintas pruebas a los módulos principales del prototipo, con el propósito de verificar su correcto funcionamiento y garantizar que la aplicación maneje los posibles errores de forma efectiva. Además, se validará la precisión de la información ingresada, asegurando así un alto nivel de calidad, confiabilidad y estabilidad en el sistema antes de su implementación final.

Tabla 35


Prueba 1

Prueba 1			
Nombre:	Prueba inicio de Sesión		
Modulo:	Seguridad		
Caso a probar	Resultado deseado	Resultado obtenido	Estado
Comprobar que el sistema no permita al usuario acceder si no ingresa las credenciales.	El sistema no debe permitir ingresar al usuario, sin antes haber escrito las credenciales.	El sistema no permite que el usuario ingrese y muestra un aviso "Debe completar todos los campos."	Satisfacctoria
Evidencia			
			

Fuente: elaboración propia, 2025


Tabla 36

Prueba 2

Prueba 2			
Nombre:	Prueba solicitar vacaciones		
Modulo:	Calcular Vacaciones		
Caso a probar	Resultado deseado	Resultado obtenido	Estado
Comprobar que el sistema muestre un mensaje cuando se solicite mas días de lo que tiene disponibles.	El sistema muestra un aviso al usuario "No tiene días suficientes. Disponible: {diasDisponibles}".	El sistema muestra un aviso al usuario "No tiene días suficientes. Disponible: {diasDisponibles}".	Satisfactoria
Evidencia			
			

Fuente: elaboración propia, 2025

Tabla 37*Prueba 3*

Prueba 3			
Nombre:	Prueba marcar hora de entrada		
Modulo:	Registrar Asistencia		
Caso a probar	Resultado deseado	Resultado obtenido	Estado
Comprobar que el sistema muestre un mensaje cuando se marque la hora de entrada.	El sistema muestra un aviso al usuario "Hora de entrada marcada exitosamente: {ahora:HH:mm:ss}".	El sistema muestra un aviso al usuario "Hora de entrada marcada exitosamente: {ahora:HH:mm:ss}".	Satisfactoria
Evidencia			
			

Fuente: elaboración propia, 2025

Tabla 38

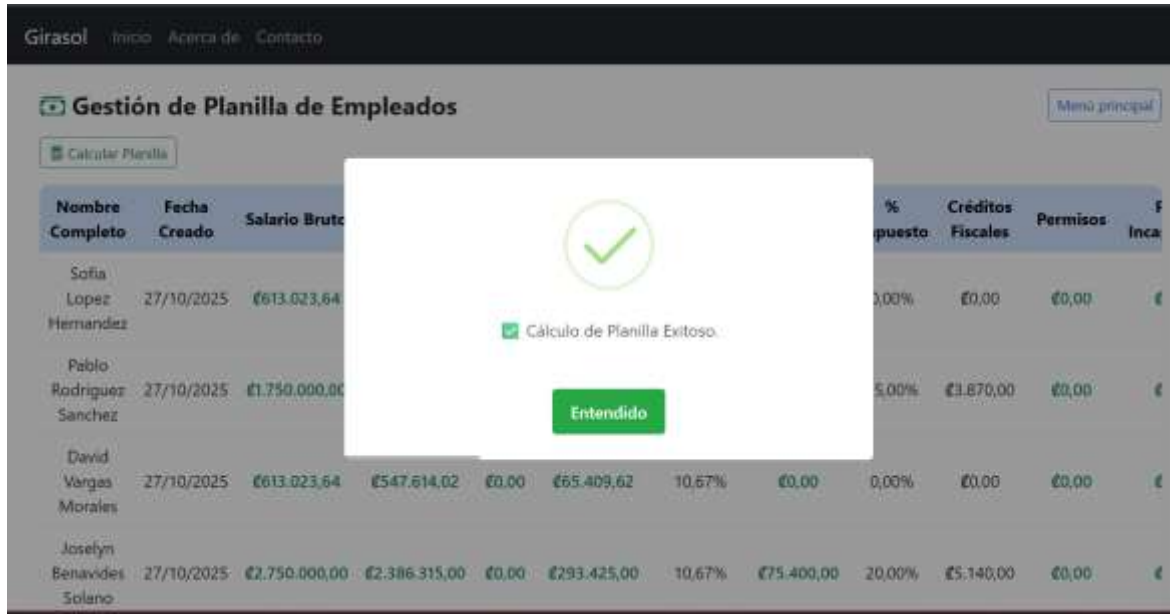
Prueba 4

Prueba 4			
Nombre:	Prueba calcular aguinaldo		
Modulo:	Calcular Aguinaldo		
Caso a probar	Resultado deseado	Resultado obtenido	Estado
Comprobar que el sistema muestre un mensaje cuando se calcule el aguinaldo para todos los empleados.	El sistema muestra un aviso al usuario "Calculo de Aguinaldo Exitoso".	El sistema muestra un aviso al usuario "Calculo de Aguinaldo Exitoso".	Satisfactoria
Evidencia			
Antes			
Despues			

Fuente: elaboración propia, 2025

Tabla 39

Prueba 5

Prueba 5																																						
Nombre:	Prueba crear planilla																																					
Modulo:	Generar Planilla																																					
Caso a probar	Resultado deseado	Resultado obtenido	Estado																																			
Comprobar que el sistema muestre un mensaje cuando se calcule la planilla para todos los empleados.	El sistema muestra un aviso al usuario "Calculo de Aguinaldo Exitoso".	El sistema muestra un aviso al usuario "Calculo de Aguinaldo Exitoso".	Satisfactoria																																			
Evidencia																																						
 <p>The screenshot shows the 'Gestión de Planilla de Empleados' interface. A modal window displays a green checkmark and the message 'Cálculo de Planilla Exitoso' with an 'Entendido' button. The background table lists employees with their names, creation dates, and various financial and tax-related data points.</p> <table border="1"> <thead> <tr> <th>Nombre Completo</th> <th>Fecha Creado</th> <th>Salario Bruto</th> <th>% puesto</th> <th>Créditos Fiscales</th> <th>Permisos</th> <th>F Inca</th> </tr> </thead> <tbody> <tr> <td>Sofia Lopez Hernandez</td> <td>27/10/2025</td> <td>€613.023,64</td> <td>3,00%</td> <td>€0,00</td> <td>€0,00</td> <td>€</td> </tr> <tr> <td>Pablo Rodriguez Sanchez</td> <td>27/10/2025</td> <td>€1.750.000,00</td> <td>5,00%</td> <td>€3.870,00</td> <td>€0,00</td> <td>€</td> </tr> <tr> <td>David Vargas Morales</td> <td>27/10/2025</td> <td>€613.023,64</td> <td>10,67%</td> <td>€0,00</td> <td>€0,00</td> <td>€</td> </tr> <tr> <td>Joselyn Benavides Solano</td> <td>27/10/2025</td> <td>€2.750.000,00</td> <td>10,67%</td> <td>€75.400,00</td> <td>€5.140,00</td> <td>€</td> </tr> </tbody> </table>				Nombre Completo	Fecha Creado	Salario Bruto	% puesto	Créditos Fiscales	Permisos	F Inca	Sofia Lopez Hernandez	27/10/2025	€613.023,64	3,00%	€0,00	€0,00	€	Pablo Rodriguez Sanchez	27/10/2025	€1.750.000,00	5,00%	€3.870,00	€0,00	€	David Vargas Morales	27/10/2025	€613.023,64	10,67%	€0,00	€0,00	€	Joselyn Benavides Solano	27/10/2025	€2.750.000,00	10,67%	€75.400,00	€5.140,00	€
Nombre Completo	Fecha Creado	Salario Bruto	% puesto	Créditos Fiscales	Permisos	F Inca																																
Sofia Lopez Hernandez	27/10/2025	€613.023,64	3,00%	€0,00	€0,00	€																																
Pablo Rodriguez Sanchez	27/10/2025	€1.750.000,00	5,00%	€3.870,00	€0,00	€																																
David Vargas Morales	27/10/2025	€613.023,64	10,67%	€0,00	€0,00	€																																
Joselyn Benavides Solano	27/10/2025	€2.750.000,00	10,67%	€75.400,00	€5.140,00	€																																

Fuente: elaboración propia, 2025

Referencias

- Ashikuzzaman, D. (23 de diciembre de 2023). *lisedunetwork*. Obtenido de lisedunetwork:
<https://www.lisedunetwork.com/sources-of-information/>
- Bhandari, P. (14 de enero de 2025). *scribbr*. Obtenido de scribbr:
<https://www.scribbr.com/methodology/qualitative-research/>
- Calderón, J. H. (09 de agosto de 2024). *jaimerodriguez.mx*. Obtenido de jaimerodriguez.mx:
<https://www.jaimerodriguez.mx/proyectos-de-investigacion/observacion-que-es-y-para-que-sirve-en-una-investigacion/>
- Chimphondah, N. (07 de abril de 2025). *thehumancapitalhub*. Obtenido de thehumancapitalhub:
<https://www.thehumancapitalhub.com/articles/functions-of-human-resources-how-the-department-works>
- Cuadrado, G. C. (06 de septiembre de 2024). *openwebinars.net*. Obtenido de openwebinars.net:
<https://openwebinars.net/blog/que-es-un-lenguaje-de-programacion/>
- Fernández, E. C. (30 de marzo de 2023). *tokioschool*. Obtenido de tokioschool:
<https://www.tokioschool.com/noticias/entorno-desarrollo-integrado/>
- Franzolini, D. (27 de enero de 2023). *blog.hubspot.es*. Obtenido de blog.hubspot.es:
<https://blog.hubspot.es/website/tipos-aplicaciones-web>
- González, G. (02 de junio de 2020). *lifeder*. Obtenido de lifeder:
<https://www.lifeder.com/variables-de-investigacion/>
- Gupta, R. (21 de febrero de 2025). *turing*. Obtenido de turing:
<https://www.turing.com/blog/software-architecture-patterns-types>
- Hassan, M. (26 de marzo de 2024). *researchmethod.net*. Obtenido de researchmethod.net:
<https://researchmethod.net/research-approach/>
- Hurtado, A. M. (20 de septiembre de 2024). *siepsi*. Obtenido de siepsi:
<https://siepsi.com.co/2024/09/20/entrevistas-cualitativas-definicion-tipos-y-recomendaciones/>
- Islam, Q. N. (09 de julio de 2025). *builtin*. Obtenido de builtin:
<https://builtin.com/articles/integrated-development-environment>
- Jervis, T. M. (27 de agosto de 2020). *lifeder*. Obtenido de lifeder:
<https://www.lifeder.com/investigacion-descriptiva/>

Kirvan, P. (14 de noviembre de 2024). *techtarget.com*. Obtenido de techtarget.com:
<https://www.techtarget.com/searchsoftwarequality/definition/waterfall-model>

Kosinski, M. (30 de septiembre de 2024). *ibm*. Obtenido de ibm:
<https://www.ibm.com/think/topics/database>

Kowalski, M. (09 de abril de 2025). *udc.libguides*. Obtenido de udc.libguides:
<https://udc.libguides.com/blog/Primary-Secondary-and-Tertiary-Sources-The-Recipe-for-Research>

Larque, J. (26 de mayo de 2025). *hiberus*. Obtenido de hiberus:
<https://www.hiberus.com/crecemos-contigo/tipos-de-bases-de-datos-cuales-hay-y-porque-es-importante-elegirlos-bien/>

Lima, G. (09 de diciembre de 2022). *aluracursos*. Obtenido de aluracursos:
<https://www.aluracursos.com/blog/aprenda-todo-sobre-el-ide-entorno-de-desarrollo-integrado>

Microsoft. (12 de junio de 2025). *learn.microsoft*. Obtenido de learn.microsoft:
<https://learn.microsoft.com/en-us/ssms/sql-server-management-studio-ssms>

Nwonah, R. (19 de agosto de 2024). *medium*. Obtenido de medium:
<https://medium.com/%40nwonahr/understanding-the-model-view-controller-mvc-pattern-97c6e057d96a>

O'Reilly, D. (01 de noviembre de 2024). *forbes*. Obtenido de forbes:
<https://www.forbes.com/advisor/business/what-is-human-resources/>

Parada, M. (23 de noviembre de 2019). *openwebinars.net*. Obtenido de openwebinars.net:
<https://openwebinars.net/blog/que-es-sql-server/>

Ramírez, F. (19 de mayo de 2023). *itsoftware*. Obtenido de itsoftware:
<https://itsoftware.com.co/content/lenguaje-programacion-csharp/>

Saavedra, J. A. (24 de julio de 2023). *ebac.mx*. Obtenido de ebac.mx:
<https://ebac.mx/blog/lenguaje-de-programacion>

Salminen, M. (27 de julio de 2025). *hostinger*. Obtenido de hostinger:
<https://www.hostinger.com/tutorials/what-is-web-application>

Silva, L. (27 de enero de 2023). *blog.hubspot.es*. Obtenido de blog.hubspot.es:
<https://blog.hubspot.es/website/ventajas-desventajas-aplicacion-web>

Staff, C. (06 de diciembre de 2024). *coursera*. Obtenido de coursera:
<https://www.coursera.org/articles/c-sharp>

Suarez, E. (09 de noviembre de 2023). *expertouniversitario.es*. Obtenido de
expertouniversitario.es: <https://expertouniversitario.es/blog/tipos-de-investigacion/>

Apéndice A
GUÍA DE ENTREVISTA

Entidad:	Girasol
Nombre del entrevistado:	Luis Portuguez Azofeifa
Puesto del entrevistado:	Administrador
Nombre del estudiante:	Joselyn Alvarado Reyes
Fecha de la entrevista:	lunes 11 de agosto del 2025
Lugar o medio de la entrevista:	Tienda Girasol

Preguntas:

Pregunta 1 ¿Actualmente posee un sistema para el manejo de Recursos Humanos?

Pregunta 2 ¿Cómo se llevan los registros de la empresa?

Pregunta 3 ¿Qué método utiliza para el cálculo de planilla?

Pregunta 4 ¿Cómo se comunican los resultados de la planilla a los colaboradores?

Pregunta 5 ¿Cuál es el proceso actual para calcular las vacaciones?

Pregunta 6 ¿Cómo es el proceso para realizar una solicitud de vacaciones?

Pregunta 7 ¿Quién se encarga de reportar las horas extras de los trabajadores?

Pregunta 8 ¿Cuál es el proceso actual para calcular las horas extras?

Pregunta 9 ¿Cuáles son los aspectos que incluyen en el cálculo de liquidaciones?

Pregunta 10 ¿Cuál es el proceso actual para calcular las liquidaciones?

Pregunta 11 ¿Cuál es la fecha límite para realizar el pago del aguinaldo a los empleados?

Pregunta 12 ¿Cómo se tramitan las incapacidades de los trabajadores actualmente?

Apéndice B

GUÍA DE OBSERVACIÓN

Entidad:	Girasol
Dirección física de la entidad:	De la rotonda la Y griega 350mts Sur hacia carretera a Desamparados, a la par de la bomba Delta.
Fecha de la actividad de observación:	jueves 13 de agosto del 2025
Nombre del estudiante:	Joselyn Alvarado Reyes

No	Aspectos por observar	Cumple	No cumple	Oportunidad de mejora	Detalle de observación
1	Ver que exista un sistema para el manejo de Recursos Humanos	Sí		Sistema automatizado	Existe un manejo en Excel de la información y de los pagos, lo que puede ocasionar errores y retrasos.
2	Observa cómo se llevan los registros de la empresa	Sí		Sistema automatizado	Se realiza de forma manual, lo que puede ocasionar errores en el pago.
3	Ver el método que utiliza para el cálculo de planilla	Sí		Cálculo automatizado	Se realiza con fórmulas en Excel, lo que puede ocasionar errores en el pago.
4	Observar cómo se comunican los resultados de la planilla a los colaboradores	Sí		Notificación automatizada	El encargado debe notificar a cada colaborador, lo que ocasiona retrasos.
5	Ver el proceso actual para calcular las vacaciones	Sí		Cálculo automatizado	Se realiza con fórmulas en Excel, lo que puede ocasionar errores en el pago.
6	Ver quien se encarga de aceptar o rechazar las solicitudes de vacaciones para los empleados	Sí		Sistema automatizado	El encargo debe revisar cada solicitud, lo que ocasiona retrasos.
7	Observar quién se encarga de reportar las horas extras de los trabajadores	Sí		Notificación automatizada	El encargo debe notificar, lo que ocasiona retrasos.
8	Ver cuál es el proceso actual para calcular las horas extras	Sí		Cálculo automatizado	Se realiza con fórmulas en Excel, lo que puede ocasionar errores en el pago.
9	Observar cuáles son los aspectos que incluyen en el cálculo de liquidaciones	Sí		Sistema automatizado	El encargado debe verificar que todos los aspectos estén reflejados en la liquidación.
10	Observar quién es el encargado de validar y aprobar las liquidaciones de los colaboradores	Sí		Notificación automatizada	El encargo debe notificar a cada colaborador, lo que ocasiona retrasos.
11	Ver que se cumpla la fecha límite para realizar el pago del aguinaldo a los empleados	Sí		Sistema automatizado	El encargado debe verificar que se cumpla la fecha límite establecida.
12	Observar cómo se gestionan las incapacidades de los trabajadores actualmente	Sí		Sistema automatizado	Se realiza por medio de formularios, lo que puede ocasionar errores al ser registrados manualmente.

Fuente: elaboración propia, 2025