

**UNIVERSIDAD INTERNACIONAL DE LAS
AMÉRICAS**

ESCUELA DE INGENIERÍA INFORMÁTICA

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

**PROTOTIPO FUNCIONAL PARA LA GESTIÓN DE
RECURSOS HUMANOS PARA EL SUPERMERCADO
HNOS PINEDA EN HOJANCHA,
GUANACASTE.**

**NOMBRE DE LA ESTUDIANTE:
JUSTIN CRUZ CASTRO**

SEDE CENTRAL, MAYO, 2025

Dedicatoria

Dedico este trabajo final de graduación, con profundo cariño y gratitud, a mis padres Roy Cruz Fonseca y Giselle Castro Sibaja, quienes han sido el motor y la base de cada uno de mis logros. Su apoyo incondicional, los consejos que siempre llegaron en el momento justo y la confianza que depositaron en mí, hicieron posible recorrer este camino académico con determinación y esperanza.

Cada esfuerzo, cada desvelo y cada paso firme que hoy me permiten culminar esta meta, no los he dado solo; los hemos dado juntos. Este logro no me pertenece únicamente a mí, sino que es el reflejo de todos los sacrificios familiares, del amor que nunca faltó y de la fortaleza que me transmitieron en los momentos más difíciles.

A ellos, que creyeron en mí, incluso cuando yo dudaba, les entrego con humildad este triunfo. Porque detrás de este título no solo hay estudios y dedicación, sino también abrazos, palabras de aliento y la certeza de que pase lo que pase, siempre los tendré a mi lado.

Agradecimientos

Antes de cerrar esta importante etapa, quiero expresar mi más sincero agradecimiento a Dios, por permitirme llegar hasta aquí con salud, claridad y propósito. Cada reto superado fue guiado por su mano y cada avance, una oportunidad para crecer.

Agradezco, profundamente, a todas las personas que formaron parte de este proceso académico: docentes que compartieron su conocimiento con pasión, compañeros que se convirtieron en amigos y en apoyo mutuo y, especialmente, a quienes, de una u otra forma, aportaron motivación para no rendirme en el camino.

A mi familia, no existen palabras suficientes para expresar mi agradecimiento. Gracias por acompañarme en los momentos de mayor presión, por comprender mis ausencias, por celebrar mis pequeños avances y por recordarme siempre, que los sueños se alcanzan con disciplina, humildad y perseverancia.

De manera especial, extiendo mi agradecimiento a mi tutor Santiago Astorga Pereira, por su guía constante, paciencia y compromiso durante el desarrollo de este proyecto. Su orientación fue clave para transformar una idea en un sistema funcional, estructurado y con valor académico real.

Este proyecto, más que una obligación universitaria, fue un viaje de crecimiento personal y profesional. Y hoy, al concluirlo, me siento profundamente agradecido por cada persona, aprendizaje y experiencia que formó parte del camino.

CONTENIDO

Dedicatoria	2
Agradecimientos.....	3
Carta de observaciones del lector	4
Carta de revisión filológica	5
Carta de aprobación de tutor.....	6
Declaración jurada del estudiante.....	17
Carta de solicitud de defensa.....	18
CAPÍTULO I: INTRODUCCIÓN	26
Planteamiento Del Problema	26
Objetivos.....	27
Objetivo General.	27
Objetivos Específicos.	27
Justificación.....	27
Viabilidad técnica.	28
Viabilidad operativa.....	29
Viabilidad económica.....	30
Viabilidad legal.....	32
Proyecciones.....	33
Alcance funcional.....	33
Alcance metodológico	36
Alcance tecnológico	37
CAPÍTULO II: MARCO REFERENCIAL.....	38
Sistema De Gestión De Recursos Humanos.....	38
Aspectos Generales De Los Sistemas De Gestión De Recursos Humanos.....	38
Ventajas.....	39
Impacto en la eficiencia y gestión	40
Funciones principales del sistema.	41
Legislación Laboral y Normativa.....	41
Código de Trabajo de Costa Rica.	41
Tecnologías Aplicables.....	43
Entornos de desarrollo.....	43
Seguridad Informática y Protección de Datos	45

Seguridad En Aplicaciones Web.....	47
Protocolos de seguridad.....	49
CAPITULO III: MARCO METODOLÓGICO	51
Enfoques De La Investigación.....	51
Enfoque cuantitativo.....	51
Enfoque cualitativo.....	51
Enfoque mixto	52
Enfoque de investigación seleccionado.....	52
Tipos de Investigación.....	53
Investigación descriptiva	53
Investigación exploratoria	54
Investigación explicativa	54
Investigación correlacional.....	54
Tipo de Investigación Seleccionado	55
Fuentes de información	56
Fuentes de información primaria.....	56
Fuentes de información secundaria	57
Fuentes de información terciaria	57
Variables	58
Variables conceptuales.....	58
Variables operacionales	59
Variables instrumentales.....	59
Población, Muestra y Cálculo de la Muestra.....	62
Población	62
Muestra.....	62
Fórmula.....	63
Recolección de datos	64
Análisis de datos.....	65
CAPITULO IV: ANÁLISIS DE RESULTADOS.....	66
Resultados.....	66
Resultados del cuestionario aplicado a empleados.....	66
Resultados de la entrevista al encargado administrativo	71
Relación de los resultados con el marco teórico.....	72

Resumen	72
CAPÍTULO V: PROPUESTA	73
Requerimientos.....	73
Requerimientos funcionales	73
1. Módulo de Nómina.....	73
2. Módulo de Horas Extra	74
3. Módulo de Vacaciones.....	74
4. Módulo de Aguinaldo.....	74
5. Módulo de Incapacidades.....	74
6. Módulo de Permisos	75
8. Módulo de Liquidación	75
9. Módulo de Evaluación de Desempeño	75
10. Mantenimientos generales	75
11. Consultas.....	76
12. Reportes.....	76
13. Seguridad.....	76
Requerimientos no funcionales	76
Análisis.....	76
Análisis detallado del software por desarrollar	77
Análisis detallado del hardware.....	81
Análisis detallado de las telecomunicaciones.....	81
Análisis detallado de herramientas técnicas	83
Conocimiento técnico del recurso humano	85
Casos de uso	86
Diseño.....	110
Arquitectura del sistema	110
Arquitectura del software	111
Diseño de entradas.....	113
Base de datos	116
Diccionario	117
Diseño de procesos.....	137
Diseño de salidas	140
Diagramas UML.....	142

Diagrama de clases.....	142
Diagrama de secuencia.....	145
Programación.....	146
Pruebas	159
CAPÍTULO VI: CONCLUSIONES Y RECOMENDACIONES	169
Conclusiones.....	169
Recomendaciones	170
Referencias	172
Apéndice.....	176
GUÍA DE ENTREVISTA.....	176
CUESTIONARIO.....	178
GUÍA DE OBSERVACIÓN	180

Tablas

Tabla 1	Descripción del hardware y software utilizado en el desarrollo del sistema.	29
Tabla 2	Costos de software	31
Tabla 3	Presupuesto para el desarrollo.	31
Tabla 4	Variables de investigación.....	60
Tabla 5	Entidad aguinaldo	117
Tabla 6	Entidad asistencia.....	118
Tabla 7	Entidad canton	119
Tabla 8	Entidad concepto_rebajo.....	119
Tabla 9	Entidad correo_electronico	120
Tabla 10	Entidad criterio.....	121
Tabla 11	Entidad Departamento.....	121
Tabla 12	Entidad dirección	122
Tabla 13	Entidad distrito.....	123
Tabla 14	Entidad empleado.....	123
Tabla 15	Entidad evaluacionrendimiento	124
Tabla 16	Entidad hora_extras	125
Tabla 17	Entidad horario_jornada.....	126
Tabla 18	Entidad incapacidad.....	127
Tabla 19	Entidad liquidación	128
Tabla 20	Entidad log_auditoria.....	129
Tabla 21	Entidad nomina	129
Tabla 22	Entidad nomina_detalle_rebajo	130
Tabla 23	Entidad permisos.....	131
Tabla 24	Entidad persona.....	132
Tabla 25	Entidad provincia.....	133
Tabla 26	Entidad puesto.....	134
Tabla 27	Entidad rol.....	134
Tabla 28	Entidad telefono	135
Tabla 29	Entidad usuario	136
Tabla 30	Entidad vacaciones.....	136
Tabla 31	Prueba uno	159
Tabla 32	Prueba dos.....	161
Tabla 33	Prueba tres.....	162
Tabla 34	Prueba cuatro	165
Tabla 35	Prueba cinco.....	166

Figuras

Figura 1 Diagrama de cascada.....	36
Figura 2 Distribución de respuestas sobre '1- ¿Con qué frecuencia recibe su salario de forma correcta y puntual?'	66
Figura 3 Distribución de respuestas sobre '2- ¿Con qué frecuencia ha identificado errores en los cálculos de sus horas extras?'	67
Figura 4 Distribución de respuestas sobre '3- ¿Qué tan clara es la información que recibe sobre sus días de vacaciones?'	67
Figura 5 Distribución de respuestas sobre '4- ¿Cómo califica el proceso de solicitud de permisos?'	68
Figura 6 Distribución de respuestas sobre '5- ¿Considera que sus horarios de trabajo son respetados por la organización?'	68
Figura 7 Distribución de respuestas sobre '6- ¿Tiene acceso a su historial de asistencia (entradas y salidas)?'	69
Figura 8 Distribución de respuestas sobre '7- ¿Considera que la evaluación de su desempeño es justa y objetiva?'	69
Figura 9 Distribución de respuestas sobre '8- ¿Qué tan eficientes considera los procesos administrativos actuales del supermercado?'	70
Figura 10 Distribución de respuestas sobre '9- ¿Ha recibido algún tipo de bonificación relacionada con su rendimiento laboral?'	70
Figura 11 Distribución de respuestas sobre '10- ¿Considera que la implementación de un sistema web mejoraría la gestión del personal?'	71
Figura 12 Flujo de comunicación.....	83
Figura 13 Diagrama de la arquitectura del sistema.....	110
Figura 14 Diagrama de la arquitectura del software.....	111
Figura 15 Pantalla de inicio de sesión.....	113
Figura 16 Pantalla de agregar incapacidades.....	114
Figura 17 Pantalla de Agregar Persona.....	114
Figura 18 Pantalla de Agregar Empleado.....	115
Figura 19 Pantalla de Inicio.....	115
Figura 20 Diagrama de base de datos.....	116
Figura 21 Diagrama de flujo del módulo de horas extras.....	138
Figura 22 Diagrama de flujo del módulo asistencia.....	139
Figura 23 Diseño de Salida de Asistencia.....	140
Figura 24 Diseño de Salida de Personas.....	140
Figura 25 Diseño de Salida de Empleados.....	140
Figura 26 Diseño de Salida de Direcciones.....	141
Figura 27 Diseño de Salida de Teléfonos.....	141
Figura 28 Diagrama de clases completo.....	142
Figura 29 Diagrama de clase parte izquierda.....	143
Figura 30 Diagrama de clases parte derecha.....	144
Figura 31 Diagrama de secuencia de Nomina.....	145
Figura 32 Diagrama de secuencia de Aguinaldo.....	145
Figura 33 Código Entradas.....	146

Figura 34 Código Salidas	147
Figura 35 Código Procesos.....	148
Figura 36 Código Validaciones	149
Figura 37 Módulo Empleados.	150
Figura 38 Módulo Asistencia.	150
Figura 39 Módulo Vacaciones.	151
Figura 40 Módulo Permisos.	152
Figura 41 Módulo Incapacidades.	153
Figura 42 Módulo Nómina.	154
Figura 43 Módulo Aguinaldo.	155
Figura 44 Módulo Liquidación.....	156
Figura 45 Módulo seguridad.	157
Figura 46 Módulo Horas extra.	158
Figura 47 Módulo Evaluar rendimiento.	158
Figura 48 Validación de inicio de sesión.....	160
Figura 49 Validación de fechas en el módulo Permisos	162
Figura 50 Validación Personas: rechazar menor de 18 años	164
Figura 51 Rechazar incapacidad cuando la fecha fin es menor a la fecha inicio	166
Figura 52 Bloquear creación de Persona con cédula de menos de 9 dígitos.....	167

CAPÍTULO I: INTRODUCCIÓN

Planteamiento Del Problema

El Supermercado Hnos. Pineda, ubicado en Hojancha, Guanacaste, es una empresa familiar con una historia de más de 55 años, que inició como un minisúper fundado por don Manuel Pineda. En el 2007 se fundó el supermercado, este pasó a la administración de su hijo Juan Manuel Pineda, quien se hizo cargo de los dos negocios. Es importante considerar que este negocio fue evolucionando hasta convertirse en un supermercado. Actualmente, cuenta con una planilla de 10 empleados, estos se han ganado el reconocimiento de los dueños y de la comunidad por su trato amable y compromiso con su trabajo.

A pesar de este crecimiento sostenido y del impacto positivo que genera en la zona, el supermercado enfrenta retos importantes en la gestión administrativa. La mayoría de sus procesos laborales como la gestión de planillas, vacaciones, incapacidades, permisos y evaluación de desempeño se realizan manualmente o mediante hojas de cálculo de Excel. Esta situación ha provocado complicaciones operativas que afectan la eficiencia del negocio.

Esta situación evidencia la necesidad de modernizar los procesos internos, mediante la implementación de un sistema de gestión de recursos humanos, que permita mejorar la precisión, transparencia y eficiencia del manejo laboral.

Entre los principales problemas se encuentran diversos aspectos críticos en la gestión actual del recurso humano. El control manual de la nómina salarial, realizado en Excel, causa errores que provocan quejas de los empleados, incluyendo cálculos incorrectos que los afectan. Debido a estos errores en los cálculos manuales, se generan dificultades en los pagos de liquidación, lo que provoca retrasos, pagos incompletos, daña el ambiente laboral y expone a posibles reclamaciones legales. Al no existir un sistema confiable, se generan valores incorrectos y retrasos en los pagos de aguinaldo, lo que puede causar enojo entre los empleados, especialmente en épocas, como el fin de año. La falta de control de las horas extras genera desbalances en los cálculos, afectando otros procesos contables, lo que ocasiona reclamos por parte de los empleados y afecta tanto la contabilidad, como la planificación financiera. La ausencia de un control claro de los días de vacaciones otorgados y su saldo provoca inconsistencias al gestionar derechos, lo que dificulta la planificación interna, genera asignaciones erróneas de descansos y puede afectar los derechos del trabajador por falta de información. La falta de un control adecuado de las incapacidades

tramitadas dificulta percibir su duración. La gestión verbal de permisos genera pérdida de datos importantes y problemas en la planificación, así como falta de control sobre los días otorgados y dificultades al momento de planificar turnos. La ausencia de un sistema para registrar entradas y salidas complica la evaluación de horarios y el cálculo salarial, al no contar con datos precisos de horarios, lo que genera conflictos entre encargados y empleados. Finalmente, al no existir un sistema de evaluación, no se han determinado criterios métricos de desempeño para cada empleado, lo que dificulta otorgar bonificaciones y reconocer esfuerzos.

Objetivos

Objetivo General.

Desarrollar un Sistema Web de Gestión de Recursos Humanos en el Supermercado Hnos Pineda en Hojancha, Guanacaste, mediante el uso de las herramientas Visual Studio Code, IntelliJ IDEA, React con TypeScript, Java con Spring Boot, Tailwind CSS para el diseño y MySQL para la base de datos.

Objetivos Específicos.

Analizar los requerimientos funcionales y no funcionales del sistema web.

Diseñar la base de datos y la arquitectura general del sistema web, considerando la estructura de datos, relaciones entre entidades y componentes funcionales, mediante el uso de herramientas como MySQL Workbench para modelado de datos.

Programar los módulos del sistema utilizando Java con Spring Boot, React con TypeScript y Tailwind CSS.

Realizar las pruebas de cada módulo para la verificación de su funcionalidad.

Justificación

Este proyecto tiene como objetivo principal crear un prototipo funcional que sirva como base para implementar un sistema de gestión de recursos humanos en el Supermercado Hnos. Pineda. La idea es automatizar varios procesos que, actualmente, se hacen de forma manual, como el manejo de la nómina, el control de asistencia, las vacaciones, las

incapacidades, los permisos, las liquidaciones, el cálculo del aguinaldo y también la evaluación del desempeño del personal.

Una de las ventajas de este sistema es que toda la información quedará guardada en una base de datos centralizada, lo cual no solo organiza mejor los datos, sino que también hace que todo sea más rápido y preciso, a la hora de hacer cálculos o consultar información.

El desarrollo del prototipo no solo permitirá estructurar, de manera clara, la organización interna del supermercado, sino que también contribuirá a mejorar la productividad del personal y el funcionamiento general del negocio.

Viabilidad técnica.

El desarrollo del Sistema de Gestión de Recursos Humanos para el Supermercado Hnos. Pineda, se evaluó si era factible implementarlo, técnicamente, en el entorno actual de la empresa. El sistema está pensado para que lo use, directamente, el encargado del área, por lo que no se requiere una red compleja ni múltiples accesos simultáneos. Se decidió hacerlo como una aplicación web, lo cual facilita su uso desde cualquier navegador, sin necesidad de instalaciones complicadas.

Actualmente, el supermercado cuenta con una computadora de escritorio que tiene lo necesario: un buen procesador, conexión a internet estable y suficiente memoria RAM.

En cuanto al desarrollo, se trabajará con herramientas accesibles como Visual Studio Code e IntelliJ IDEA. El diseño de la interfaz se hará con React y TypeScript, mientras que la parte del backend se programará en Java utilizando Spring Boot. También se usará Tailwind CSS para el estilo visual y MySQL como base de datos. Estas herramientas son bastante confiables, funcionan bien juntas y permiten desarrollar un sistema ligero pero eficiente.

Para crear el sistema, se cuenta con una computadora que tiene un procesador AMD Ryzen 5, 32 GB de RAM y un SSD de 1 TB, lo cual es más que suficiente para realizar pruebas, programar y ajustar todo sin contratiempos. Gracias a esto, se puede avanzar de forma fluida y sin demoras, asegurando que el sistema sea compatible con el entorno de trabajo del supermercado y que funcione como se espera.

Los detalles completos de los elementos de hardware y software utilizados, se presentan en la Tabla 1.

Tabla 1*Descripción del hardware y software utilizado en el desarrollo del sistema.*

CATEGORÍA	ENFOQUE	CARACTERÍSTICA
HARDWARE	Procesador	AMD Ryzen 5. Procesador de alto rendimiento adecuado para desarrollo.
	Memoria RAM	32 GB. Permite ejecutar múltiples procesos, simultáneamente, sin problemas.
	Almacenamiento	SSD de 1 TB. Ofrece velocidad y capacidad suficientes para el entorno de desarrollo.
SOFTWARE	Editor de código	Visual Studio Code. Editor ligero, multiplataforma.
	Entorno desarrollo	IntelliJ IDEA. IDE especializado en desarrollo con Java.
	Frontend	React con TypeScript. Librería eficiente para construir interfaces dinámicas.
	Backend	Spring Boot. Framework robusto para el desarrollo de aplicaciones en Java.
	Estilos	Tailwind CSS. Framework de CSS para diseño rápido y responsivo.
	Base de datos	MySQL. Sistema de gestión de bases de datos relacional, ampliamente utilizado.

*Fuente: Elaboración propia***Viabilidad operativa**

El sistema puede implementarse sin mayores complicaciones en el supermercado, ya que está pensado para que sea práctico, fácil de usar y accesible para todos los empleados.

No se necesitan conocimientos avanzados para manejarlo, con saber utilizar una computadora, un celular o una Tablet y saber los usos básicos de internet, es suficiente para

empezar a usarlo sin problemas. Esto facilita la adaptación de los trabajadores al sistema, sin necesidad de capacitaciones extensas o procesos complejos.

La idea principal no es reemplazar a nadie, sino hacer que el trabajo en el área de Recursos Humanos sea más ágil y organizado. Con este sistema, los encargados podrán llevar mejor el control de tareas y procesos, lo cual ayudará a reducir la carga de trabajo y a que todo funcione de forma más ordenada y eficiente. Al facilitar la gestión y mantener una estructura sencilla de uso, el sistema se convierte en una herramienta útil y adaptable a las necesidades del supermercado.

Por las razones mencionadas, se puede afirmar que el proyecto es operativamente viable, ya que puede implementarse sin complicaciones, es fácil de usar por el personal actual y mejora los procesos internos, sin alterar el funcionamiento habitual del supermercado.

Viabilidad económica

La viabilidad económica ha sido estudiada, detalladamente, para hacer una lista de los costos del prototipo. Principalmente, la empresa ya cuenta con el hardware necesario, por lo que no se requerirá inversión adicional en ese aspecto. En cuanto al software, se ha optado por tecnologías que cuentan con licencias gratuitas, lo cual reduce, considerablemente, los gastos relacionados con herramientas de desarrollo. Esto permite construir una solución funcional, sin incurrir en costos por adquisición de plataformas o programas.

Los detalles de las herramientas seleccionadas y su costo se presentan en la Tabla 2.

Tabla 2
Costos de software

Tecnologías	Costo
Visual Studio Code	¢0
IntelliJ IDEA	¢0
React con TypeScript	¢0
Java con Spring Boot	¢0
Tailwind CSS	¢0
MySQL	¢0
Total	¢0

Fuente: Elaboración propia

Además del software, se deben considerar los costos asociados al proceso de desarrollo. En relación con esto, el Ministerio de Trabajo y Seguridad Social (2025), menciona: “El salario mínimo para un programador de computación sin título, gana ¢15.983,96 por día” (p. 2). Tomando como base este salario diario, se calcula el costo por hora y con ello, se estima el presupuesto necesario para cada fase del desarrollo del sistema.

A continuación, se presenta el detalle de estos costos en la Tabla 3.

Tabla 3
Presupuesto para el desarrollo.

	Duración por días	Costo por hora en colones	Horas trabajadas en total (8 horas diarias)	Total, en colones
Planificación	10	¢1997,995	80 horas	¢159 836,9
Diseño	15	¢1997,995	120 horas	¢159 836,9
Implementación	15	¢1997,995	120 horas	¢159 836,9
Pruebas	5	¢1997,995	40 horas	¢159 836,9
Despliegue	5	¢1997,995	40 horas	¢159 836,9
Mantenimiento	10	¢1997,995	80 horas	¢159 836,9
Total	60		480 horas	¢959 037,6

Fuente: Elaboración Propia

Los costos se han estimado con base en una tarifa accesible y dentro de los estándares establecidos legalmente. Al contar con herramientas gratuitas y hardware ya disponible, el gasto principal recae en el tiempo de trabajo invertido en el desarrollo. Por estas razones, el proyecto es económicamente viable, ya que no representa una inversión elevada y permite construir una solución funcional con recursos controlados.

Viabilidad legal

Se ha evitado cualquier riesgo legal potencial y se asegura que el proyecto se desarrolla dentro del marco legal y con los permisos necesarios y para garantizar la validez legal del prototipo, es necesario cumplir con las siguientes leyes:

La Ley N.º 8148, que adiciona la Ley N.º 4573 sobre delitos informáticos, establece sanciones claras ante acciones como la violación de comunicaciones electrónicas, el fraude informático, la alteración de datos y el sabotaje digital. Esta ley protege la integridad de los sistemas informáticos y garantiza que los desarrolladores sean responsables del uso correcto de la información que se procesa y transmite a través del sistema.

La ley 6683, esta ley se encarga de proteger todo lo relacionado con las obras, interpretaciones, ejecuciones y grabaciones (fonogramas) que hayan sido creadas por autores, artistas, intérpretes o productores de Costa Rica, sin importar si viven dentro o fuera del país. En relación con el proyecto, esta normativa obliga a respetar los derechos sobre cualquier código fuente, diseño visual o contenido desarrollado y también, sobre cualquier herramienta externa utilizada bajo licencia.

La ley 8968, esta busca cuidar la información personal de las personas, lo que hace es asegurar que cada uno tenga control sobre sus propios datos y que se usen de forma responsable, siguiendo ciertos principios para proteger la privacidad. Esta ley es especialmente relevante en un sistema de gestión de recursos humanos, ya que se maneja información sensible como datos personales, registros laborales y datos financieros de los empleados. El sistema implementará medidas de seguridad para garantizar el acceso restringido a esta información y se asegurará el consentimiento informado, por parte de los usuarios, antes de cualquier tratamiento de datos.

El proyecto cumple con las leyes necesarias para su desarrollo legal, al respetar estas normativas, se asegura que el sistema sea seguro, confiable y que proteja tanto la información, como los derechos de las personas involucradas.

Proyecciones

El desarrollo del sistema tiene como objetivo principal modernizar y automatizar las tareas que, actualmente, se llevan a cabo de manera manual o con poca organización dentro del área de talento humano. Esta solución tecnológica está orientada a minimizar errores frecuentes en procesos como el cálculo de planillas, pagos de salarios, control de asistencia y manejo de permisos, al mismo tiempo que permite generar datos fiables que respalden la toma de decisiones estratégicas.

La implementación beneficiará. Por un lado, permitirá disminuir fallos administrativos. Por otro, contribuirá al bienestar del personal, brindando mayor transparencia y confianza en la gestión de planilla.

Alcance funcional

El alcance funcional del sistema es la descripción de las funcionalidades, características y capacidades que el prototipo debe incluir, según los objetivos de la empresa. Esta parte es fundamental para el éxito del sistema, ya que establece, claramente, lo que se espera lograr y cuáles son los límites del proyecto. El sistema tendrá la capacidad de gestionar las diferentes funciones de los siguientes módulos:

Gestionar Nóminas

Este módulo se encargará de gestionar, de manera integral, toda la información contable necesaria para el desarrollo preciso de la nómina salarial del supermercado. El administrador del sistema podrá calcular el pago a cada uno de los empleados de la empresa.

Gestionar Liquidación

Este módulo se encargará de calcular y gestionar las liquidaciones de los empleados de manera precisa, tomando en cuenta las condiciones específicas de cada caso, como días

trabajados, horas extras y otros factores, eliminando errores en los cálculos y asegurando pagos correctos y oportunos.

Calcular Aguinaldo

Este módulo se encargará de calcular, de manera automática, el aguinaldo anual de cada empleado, tomando en cuenta las variables necesarias y garantizando la precisión de los pagos, sin depender de cálculos manuales que puedan resultar en errores.

Gestionar Horas Extras

Este módulo se encargará de gestionar, de manera precisa, las horas extras trabajadas por cada empleado, permitiendo el cálculo automático de las horas adicionales trabajadas y su integración en los pagos de nómina, eliminando errores y desbalances en los cálculos manuales.

Gestionar Vacaciones

Este módulo se encargará de gestionar y registrar, de manera eficiente, los días de vacaciones de los empleados, permitiendo un control claro y accesible de las fechas solicitadas, las aprobadas y las utilizadas, evitando la pérdida de datos y errores en los cálculos de nómina.

Gestionar Incapacidades

Este módulo se encargará de gestionar y registrar las incapacidades de los empleados, asegurando el seguimiento adecuado de los días de incapacidad y su impacto en los pagos de nómina, sin que haya errores en su cálculo.

Gestionar Permisos

Este módulo se encargará de gestionar los permisos solicitados por los empleados, asegurando que cada permiso esté debidamente registrado y aprobado, evitando la pérdida de información que pueda afectar la precisión de los cálculos salariales.

Gestionar Asistencia

Este módulo se encargará de registrar y controlar, de manera precisa, la asistencia de los empleados, incluyendo horarios de entrada y salida, permitiendo un monitoreo adecuado y facilitando el cálculo exacto de las horas trabajadas, evitando errores en los pagos por asistencia irregular.

Evaluar rendimiento

Este módulo se encargará de evaluar y registrar el rendimiento de los empleados, proporcionando métricas claras sobre su desempeño laboral. Esto permitirá la toma de decisiones más informadas en cuanto a promociones, incrementos salariales y desarrollo profesional.

Mantenimientos

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

Consultas

Este módulo se encargará de generar información proporcionada.

Reportes

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

Seguridad

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

El sistema busca integrar y automatizar todas las funciones esenciales relacionadas con la gestión de recursos humanos del Supermercado Hnos Pineda, brindando una herramienta eficiente, precisa y adaptada a las necesidades específicas de la empresa. Cada módulo está diseñado para facilitar el trabajo administrativo.

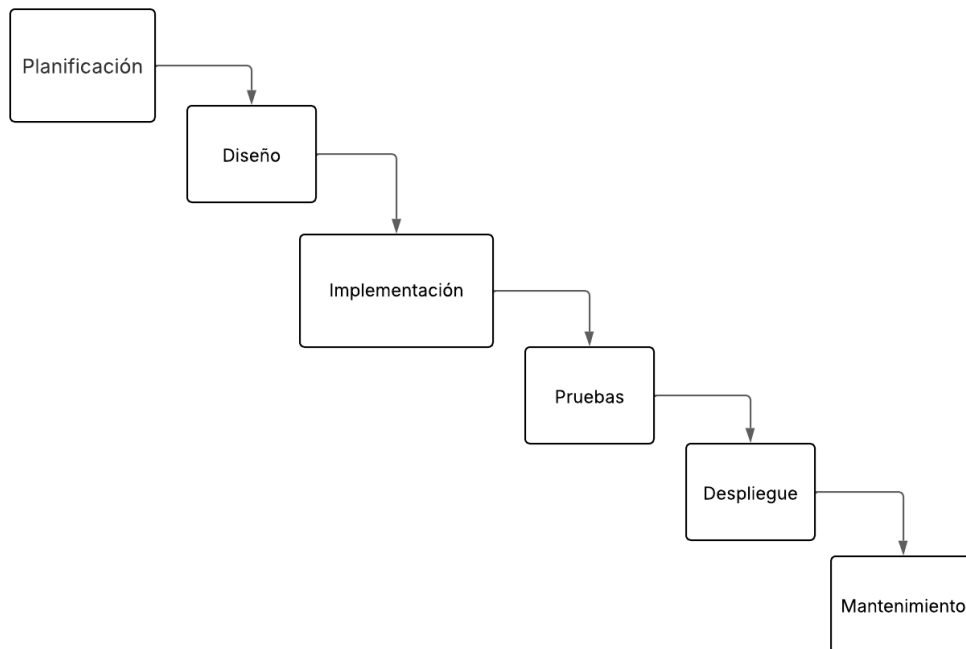
Alcance metodológico

El método que se utilizará es el Cascada. Según Amazon Web Services (s. f.),” el modelo de cascada dispone todas las fases, secuencialmente, de modo que las nuevas fases dependan del resultado de la fase anterior. Desde un punto de vista conceptual, el diseño fluye desde una fase a otra inferior, como en una cascada. Ventajas y desventajas, el modelo de cascada hace que la administración del proyecto sea muy estricta y proporciona un resultado tangible, al final de cada fase. Sin embargo, hay poco margen de cambio una vez que una fase se considera completa, ya que los cambios pueden afectar al tiempo de entrega, al costo y a la calidad del software. Por lo tanto, el modelo es más adecuado para pequeños proyectos de desarrollo de software, donde las tareas se pueden organizar y administrar fácilmente y los requisitos se pueden predefinir con precisión.”

Este método ayuda a mantener un control claro sobre lo que se quiere lograr en el proyecto y asegura que cada etapa se complete bien antes de pasar a la siguiente, por lo que el desarrollo del software sigue un orden paso a paso.

De acuerdo con esta información, este es el ciclo de vida del software:

Figura 1
Diagrama de cascada



Fuente: Elaboración propia

Alcance tecnológico

Se seleccionaron estas tecnologías para garantizar un rendimiento óptimo del sistema, ya sea en la carga de datos y en la interfaz del sistema web. Se va a llevar a cabo utilizando, Visual Studio Code e IntelliJ IDEA como entornos principales, aprovechando sus fuertes en el frontend y backend. Además, se va a utilizar Java, en conjunto con el framework Spring Boot.

En la interfaz se va a utilizar React con TypeScript, una combinación que permite desarrollar interfaces de usuario dinámicas, mantenibles y con una estructura clara. El diseño visual de la aplicación se realiza mediante Tailwind CSS, una librería de clases utilitarias que agiliza el proceso de maquetación y diseño, permitiendo construir interfaces modernas, responsivas y adaptadas a distintos dispositivos.

Para la gestión de los datos se utiliza MySQL ya que esta confiable. Esta es muy utilizada por los desarrolladores, esta es ideal para almacenar, consultar y administrar los datos del sistema, de manera segura y eficiente.

CAPÍTULO II: MARCO REFERENCIAL

Sistema De Gestión De Recursos Humanos

Sistema de gestión de recursos humanos para el Supermercado Hnos Pineda, ubicado en Hojancha, Guanacaste, representa una herramienta tecnológica clave para administrar de manera eficiente. Estas plataformas tecnológicas diseñadas para recopilar, almacenar y procesar información relacionada con las actividades de los Departamentos de Recursos Humanos de la empresa. Este sistema integra diversos módulos funcionales que permiten gestionar múltiples aspectos relacionados con los empleados, desde el control de asistencia y la elaboración de nómina, hasta la evaluación del desempeño y la asignación de turnos.

Aspectos Generales De Los Sistemas De Gestión De Recursos Humanos

Un sistema de recursos humanos se compone de un conjunto de procesos diseñados para automatizar y optimizar funciones clave como la nómina, el control de asistencia, la gestión de vacaciones, incapacidades, permisos, liquidaciones, el cálculo del aguinaldo y la evaluación del rendimiento de los empleados dentro de una organización.

Según con Tweem (2024), contar con un software adecuado es fundamental para optimizar procesos, mejorar la eficiencia y garantizar el bienestar, tanto de los empleados como de la empresa en general. Entre las características principales se destacan:

Gestión de la información del personal: La capacidad de almacenar y gestionar, de manera segura, toda la información relacionada con los empleados es una característica fundamental. Esto incluye datos personales, historial laboral, formación, evaluaciones de desempeño y cualquier otra información relevante para la gestión de recursos humanos.

Automatización de procesos administrativos: Un buen software de recursos humanos debe permitir la automatización de tareas administrativas rutinarias como el envío de nóminas, control de asistencia, gestión de vacaciones y seguimiento de horarios. Esto no solo ahorra tiempo, sino que también reduce errores humanos y garantiza la precisión en los registros.

Gestión del rendimiento y evaluación de desempeño: La capacidad de llevar a cabo evaluaciones de desempeño periódicas y gestionar el rendimiento de los empleados es esencial para impulsar el desarrollo profesional y la motivación del personal. El software debe permitir la configuración de evaluaciones, el seguimiento del progreso y la generación de informes detallados.

Seguridad y cumplimiento normativo: La seguridad de la información y el cumplimiento de las normativas de protección de datos son aspectos fundamentales en la gestión de recursos humanos. El software debe garantizar la seguridad de los datos almacenados, cumplir con las regulaciones de privacidad y facilitar el cumplimiento de otras normativas laborales y fiscales. (párr. 3).

Las características principales anteriores demuestran la importancia de contar con herramientas tecnológicas adecuadas en la gestión del capital humano.

Ventajas

Además, según Ramon (2021), implementar un sistema web de recursos humanos ofrece múltiples ventajas como:

La utilización de soluciones tecnológicas permite encontrar perfiles más adecuados y que respondan a lo que están buscando las empresas. También se agilizan los procesos de selección, algo que ayuda a optimizar los tiempos para que aquellos que deben seleccionar candidatos puedan concentrarse en la evaluación de aptitudes o en aspectos más importantes.

Poder contar con información ayuda a mejorar los procesos en todos los niveles de la gestión de los recursos humanos, ya que se tiene un panorama general de la situación real de la empresa y sus diferentes áreas. Antes de la implementación de cualquier cambio, se pueden elaborar estadísticas acerca de los principales puntos a mejorar, de las debilidades y de lo que se debe cambiar.

Permiten también optimizar el tiempo para que ese sector pueda dedicarse a tareas más estratégicas y no sólo a la administración del personal (párr. 4).

Cabe resaltar que son ventajas de gran importancia, ya que ayuda a optimizar y automatizar para las gestiones en el sistema web.

Impacto en la eficiencia y gestión

Según McNamee (2024), al gestionar, automáticamente, la incorporación, los departamentos de recursos humanos pueden centrarse en los aspectos del bienestar y la satisfacción del personal durante ese proceso, creando una experiencia agradable para el nuevo empleado, destacando los siguientes beneficios:

Colaboración y planificación

Las herramientas de colaboración permiten una planificación precisa, ya que los miembros del equipo pueden comunicarse entre sí, para completar tareas, independientemente de su ubicación.

Regulación de cumplimiento

Los mejores sistemas de información de recursos humanos garantizan que las empresas cumplan con las leyes internacionales y locales y que todos los datos y políticas se actualicen, automáticamente, para garantizar el cumplimiento normativo.

Reducción de costes

Independientemente del tamaño de una empresa, la implementación de HRIS puede reducir, significativamente, los costos, esto se debe a que es más eficiente que los procesos manuales y múltiples sistemas distintos.

Eficiencia y aumento de la productividad.

Admite múltiples funciones relacionadas con recursos humanos en un solo sistema, lo que ayuda a las organizaciones a evitar desgaste, el agotamiento y la sobrecarga de trabajo. Esta mayor eficiencia y productividad beneficia a los empleados más que el uso de múltiples sistemas de recursos humanos dispares o procesos manuales complejos y obsoletos. (párr. 23).

La implementación de un prototipo de este tipo facilita la toma de decisiones más informadas en la gestión del personal. A partir de los datos registrados en el sistema, es posible identificar las fortalezas y debilidades de cada trabajador.

Funciones principales del sistema.

La función principal del sistema es calcular y organizar la nómina de todos los empleados, incluyendo la gestión de la liquidación al calcular pagos finales, según aspectos legales y financieros, el cálculo automatizado del aguinaldo, conforme a la obligación legal en Costa Rica. La incorporación de horas extras en el cálculo salarial, la gestión de vacaciones controlando los días acumulados, la consideración de incapacidades, ajustando la nómina por días no laborados. Según DocuSign (2024), “los sistemas de información de recursos humanos realizan una variedad de funciones para facilitar la gestión eficiente del talento humano, incluyendo la administración de nómina y beneficios”

Legislación Laboral y Normativa

Código de Trabajo de Costa Rica.

El Código de Trabajo de Costa Rica, junto con otras leyes complementarias como las que regulan la seguridad social y los impuestos sobre la renta, establece las bases legales que deben respetarse al automatizar procesos como la contratación, el control de jornada, el pago de salarios, las vacaciones, el cálculo de deducciones y la terminación de la relación laboral. A continuación, se citan algunos artículos relevantes del Código de Trabajo, cuya inclusión permite comprender los parámetros legales que deben ser incorporados en el diseño del sistema. Estas citas textuales permiten fundamentar, legalmente, los procesos automatizados y garantizar que se respeten los derechos y obligaciones de los trabajadores.

Artículo 18. Contrato individual de trabajo, sea cual fuere su denominación, es todo aquél en que una persona se obliga a prestar a otra, sus servicios o a ejecutarle una obra, bajo la dependencia permanente y dirección inmediata o delegada de ésta y por una remuneración de cualquier clase o forma.” (Código de Trabajo de Costa Rica, 2016, p. 12)

Este artículo es fundamental para establecer que el sistema debe manejar datos contractuales que evidencien la relación laboral, la subordinación y la remuneración pactada.

“Artículo 136. La jornada ordinaria de trabajo efectivo no podrá ser mayor de ocho horas en el día, de seis en la noche y de cuarenta y ocho horas por semana. Sin embargo, en los trabajos que, por su propia condición no sean insalubres o peligrosos, podrá estipularse una jornada ordinaria diurna hasta de diez horas y una jornada mixta hasta de ocho horas, siempre que el trabajo semanal no exceda de las cuarenta y ocho horas. Las partes podrán contratar, libremente, las horas destinadas a descanso y comidas, atendiendo a la naturaleza del trabajo y a las disposiciones legales.” (Código de Trabajo de Costa Rica, 2016, p. 45)

Este fragmento respalda la necesidad de configurar el sistema para registrar distintos tipos de jornada laboral, asegurando que no se excedan los límites legales.

“Artículo 147. Son hábiles para el trabajo, todos los días del año, excepto los feriados y los días de descanso semanal existentes por disposición legal o convenio entre las partes.” (Código de Trabajo de Costa Rica, 2016, p. 50)

“Artículo 148. Se considerarán días feriados y, por lo tanto, de pago obligatorio los siguientes: el 1 de enero, el 11 de abril, el jueves y Viernes Santos, el 1° de mayo, el 25 de julio, el 15 de agosto, [...]” (Código de Trabajo de Costa Rica, 2016, p. 51)

La integración de los días feriados y de descanso obligatorio es esencial para el cálculo preciso de salarios y para evitar conflictos laborales por errores en pagos o jornadas trabajadas.

“Artículo 153. Todo trabajador tiene derecho a vacaciones anuales remuneradas, cuyo mínimo se fija en dos semanas por cada cincuenta semanas de labores continuas, al servicio de un mismo patrono. En caso de terminación del contrato de trabajo antes de cumplir el período de las cincuenta semanas, el trabajador tendrá derecho, como mínimo, a un día de

vacaciones por cada mes trabajado, que se le pagará en el momento del retiro de su trabajo.”
(Código de Trabajo de Costa Rica, 2016, p. 54)

Este artículo sustenta la implementación de un módulo que permita gestionar, automáticamente, el cálculo y la asignación de vacaciones.

“Artículo 177. Todo trabajador tiene derecho a devengar un salario mínimo que cubra las necesidades normales de su hogar en el orden material, moral y cultural, el cual se fijará periódicamente, atendiendo a las modalidades de cada trabajo, a las particulares condiciones de cada región y de cada actividad intelectual, industrial, comercial, ganadera o agrícola.”
(Código de Trabajo de Costa Rica, 2016, p. 61)

Este principio legal justifica que el sistema incorpore una validación automática de salarios mínimos, según los parámetros definidos por el Ministerio de Trabajo y Seguridad Social.

El propósito de citar, directamente, estos artículos es garantizar que el sistema de gestión de recursos humanos esté alineado con los requerimientos legales vigentes en Costa Rica. El aporte de esta investigación consiste en traducir estos requerimientos jurídicos en especificaciones técnicas funcionales que orienten el desarrollo del software, asegurando que el mismo cumpla con los principios de legalidad, equidad y transparencia laboral.

Tecnologías Aplicables

El sistema web de gestión de recursos humanos será desarrollado utilizando tecnologías modernas que permiten una arquitectura robusta, escalable y segura. Las principales herramientas son:

Entornos de desarrollo.

Visual Studio Code es un editor de código fuente, ampliamente utilizado por su ligereza, personalización y compatibilidad con múltiples lenguajes. Su objetivo principal es facilitar la escritura, edición y depuración del código fuente. Además, ofrece funcionalidades como resaltado de sintaxis, autocompletado y control de versiones, lo cual mejora,

considerablemente, la productividad del desarrollador. Según Arsys (s. f.), una de sus funciones esenciales es “permitir editar el código usando los juegos de caracteres correctos y manejando la lista de archivos que forman parte de las aplicaciones o los sitios web. Además, ofrece diversas ayudas a los programadores, como el resultado de sintaxis o el autocompletado del código” (párr. 6).

Por su parte, **IntelliJ IDEA** es un entorno de desarrollo integrado (IDE), enfocado principalmente en Java, pero con soporte para otros lenguajes como Python, Kotlin y JavaScript. Esta herramienta destaca por su capacidad de análisis en tiempo real, refactorización inteligente y compatibilidad con sistemas de control de versiones como Git. De acuerdo con GoDaddy (2024), sus ventajas incluyen “refactorización del código, análisis estático en tiempo real, integración con Git y posibilidad de personalización, mediante plugins” (párr. 11). En este proyecto, IntelliJ IDEA se utiliza como IDE principal para la construcción del backend con Java y Spring Boot. Algunas de las tecnologías utilizadas:

React

Según Hutsulyak (2024), esta tecnología presenta grandes ventajas, a nivel de escalabilidad y productividad:

Being an extension of the JavaScript programming language, React brings many advantages. Products built with React are simple to scale, and what is convenient, a single language simultaneously used on the server/client/mobile side of things, resulting in outstanding productivity. Also, there are workflow patterns for convenient teamwork, and a readable and maintainable user interface code. (párr. 12)

Tailwind CSS

Como expone Idenautas Web Factory (s. f.), este enfoque ofrece eficiencia, tanto para diseñadores como desarrolladores:

Tailwind CSS es un framework de diseño web de código abierto que proporciona un conjunto completo de clases CSS predefinidas que se pueden utilizar para construir interfaces de usuario personalizadas. El enfoque de Tailwind es proporcionar una base sólida para diseñadores y desarrolladores que desean construir interfaces de usuario personalizadas de manera rápida y eficiente. (párr. 1).

Java y Spring Boot

En la lógica del servidor, el sistema utiliza Java como lenguaje principal, apoyado en el framework Spring Boot. Según lo expone IBM (2024), se trata de una plataforma clave en el desarrollo web:

Java es un marco ideal para desarrollar aplicaciones web, la base de un negocio digital en cualquier industria. Los servidores de aplicaciones Java son contenedores web para componentes Java, XML y servicios web, que interactúan con bases de datos y proporcionan contenidos web dinámicos. Los servidores de aplicaciones Java constituyen un entorno de despliegue estable para aplicaciones empresariales con funciones que incluyen gestión de transacciones, seguridad, agrupación en clústeres, rendimiento, disponibilidad, conectividad y escalabilidad. (párr. 11)

MySQL

Para la gestión de datos, se ha elegido el sistema de gestión de bases de datos MySQL, reconocido por su fiabilidad, alto rendimiento y licencia de código abierto. Esta tecnología es ideal para pequeñas y medianas empresas, ya que ofrece una solución robusta sin costos de licencia. Robledano (2019), menciona que:

“MySQL es una opción razonable para ser usado en el ámbito empresarial permitiendo gestionar catálogos de productos, clientes o contenidos de forma ordenada” (párr. 13).

La integración de estas tecnologías permite construir un sistema web moderno de tipo full-stack, donde React y Tailwind CSS en el cliente, junto con Java y Spring Boot en el servidor y MySQL como motor de base de datos, garantizan la cohesión entre funcionalidad, rendimiento y diseño.

Seguridad Informática y Protección de Datos

La seguridad informática constituye un aspecto fundamental para proteger la integridad, confidencialidad y disponibilidad de la información. La creciente digitalización de procesos empresariales ha incrementado la exposición a riesgos cibernéticos, lo que exige la implementación de tecnologías especializadas en defensa y prevención de ataques.

Según Fortinet (s. f.), la seguridad web consiste en un conjunto de medidas y herramientas diseñadas para proteger sitios y aplicaciones web contra amenazas como la inyección de código, el acceso no autorizado y la explotación de vulnerabilidades. Algunas de las principales tecnologías empleadas para garantizar la seguridad en sistemas web:

Firewall de aplicaciones web (WAF)

El WAF es una herramienta que ayuda a proteger las páginas web revisando todo lo que entra y sale entre la aplicación e Internet. Sirve para bloquear ataques comunes como los que intentan robar datos o dañar el sistema. No cubre todo, pero sí es clave para tener un sitio más seguro. Funciona en la parte del sistema que se encarga de manejar la información que llega. Normalmente se usa junto con otras herramientas de seguridad.

Escáneres de vulnerabilidades

Son programas que revisan, automáticamente, si un sistema o página web tiene fallas de seguridad, esto ayuda a detectar errores que podrían ser aprovechados por un hacker. Una vez que se encuentran, se pueden corregir antes de que causen problemas, es una forma rápida y útil de mantener todo más protegido, casi siempre se usan de forma regular en empresas.

Herramientas de descifrado de contraseñas

Estas herramientas permiten recuperar el acceso si alguien pierde u olvida su contraseña. Aunque suena peligroso, bien utilizadas pueden ayudar a no perder información importante, son útiles en situaciones de emergencia o fallos técnicos. Eso sí, deben usarse con cuidado y solo por personas autorizadas, son una especie de “plan B” de seguridad.

Herramientas de fuzzing

El fuzzing es una técnica donde se mandan datos aleatorios a un programa para ver si falla o tiene errores, así se pueden encontrar problemas de seguridad que, normalmente, no se ven, se puede usar en distintas etapas del desarrollo del software. Sirve mucho para mejorar el código antes de lanzar una aplicación. Es como hacerle pruebas de resistencia al sistema.

Pruebas de caja negra

Aquí se prueba un sistema sin saber cómo está hecho por dentro, solo viendo qué pasa cuando se le dan ciertos datos. Se usa para ver cómo responde el software frente a lo que haría un usuario normal, ayuda a encontrar errores en el uso cotidiano, también sirve para ver si funciona bien bajo presión. Es como probar un aparato sin abrirlo.

Pruebas de caja blanca

En estas pruebas sí se revisa cómo está hecho el software por dentro: su código, su estructura, entre otros, se hacen para detectar errores más técnicos y mejorar cómo funciona el programa. Es útil para asegurarse de que todo el sistema está bien conectado y que los datos fluyen correctamente. Se usa mucho cuando se quiere pulir el diseño y la seguridad, es como arreglar un carro sabiendo cómo funciona el motor.

Estas tecnologías son las que garantizan la seguridad web. Cada una cumple una función específica dentro del sistema y su implementación conjunta, permite a las organizaciones enfrentar, eficazmente, las amenazas modernas, proteger sus sistemas y salvaguardar la información crítica ante posibles ataques.

Seguridad En Aplicaciones Web.

En el desarrollo del sistema web de gestión de recursos humanos para el Supermercado Hnos Pineda, la seguridad informática adquiere una importancia crítica, ya que el sistema procesará información sensible como datos personales, registros de asistencia, salarios, incapacidades y evaluaciones de desempeño. Por tanto, proteger esta información, no solo es una necesidad operativa, sino también una obligación ética y legal.

La seguridad en aplicaciones web implica el uso de herramientas y prácticas diseñadas para proteger el sistema contra amenazas como ataques de **inyección SQL**, **cross-site scripting (XSS)**, **ataques de fuerza bruta** y accesos no autorizados. Estas amenazas podrían comprometer la integridad del sistema y exponer datos confidenciales, lo que puede traducirse en consecuencias legales y pérdida de confianza por parte de los usuarios.

De acuerdo con Fastly (2024), la seguridad moderna en aplicaciones web exige una visión integral, que combine herramientas tecnológicas, políticas internas, capacitación del

personal y medidas organizativas desde las primeras etapas del desarrollo del software. La seguridad ya no puede depender, exclusivamente, del área técnica, sino que debe implementarse desde la fase de codificación y mantenerse activa durante el despliegue, operación y mantenimiento del sistema.

Debe entenderse que hay múltiples formas de romper la seguridad del sistema, por lo cual, se debe estar conscientes de estas prácticas que realizan las personas no autorizadas.

Capacitación

Otro aspecto clave es la capacitación en ciberseguridad de los usuarios del sistema, incluyendo al personal administrativo del supermercado, un sistema seguro también depende del comportamiento consciente de quienes lo utilizan. Al brindar formación básica sobre amenazas comunes como el phishing o el programa maligno, se promueve una cultura de seguridad en la organización, lo que reduce el riesgo de incidentes por descuido humano.

Según Diaz (2023), “los empleados son el mayor activo que puede tener una empresa, no es menos cierto que también representan uno de los mayores riesgos, esto desde el punto de vista de ciberseguridad. ¿Cuál es la razón de esto? Muy simple, los delincuentes informáticos pueden vulnerar los sistemas informáticos de una empresa, al evitar ser detectados por los mecanismos de seguridad que los protegen. Sin embargo, hay una forma mucho más sencilla y rápida, que es a través del phishing o utilizando la ingeniería social, donde los atacantes usan a los empleados para que, involuntariamente, les faciliten el acceso a los sistemas de la organización.

Por esta razón, es de vital importancia para cualquier negocio, capacitar, oportunamente, a sus empleados en los riesgos asociados con la seguridad informática. De esa forma, pueden evitar que cualquier atacante pueda llegar hasta la información importante de la empresa, manteniéndola segura.”

Además, esto fortalece la confianza en el sistema, ya que los empleados sabrán que sus datos están protegidos y que existen protocolos para su resguardo y uso responsable.

Formas de proteger

La protección de los sistemas web se ha convertido en un desafío crucial, debido al aumento constante de amenazas cibernéticas que buscan vulnerar la seguridad de la información. Por ello, es necesario aplicar diversas estrategias que fortalezcan la defensa frente a estos riesgos. De acuerdo con Galicia (2025), algunas de las medidas más efectivas para proteger un sistema web incluyen:

Establecer contraseñas seguras.

Cambia el usuario “admin”, ya que existen ataques de fuerza bruta donde los hackers aplican “métodos de prueba y error” para descifrar la contraseña, utilizando combinaciones comunes (como “12345” o “password”) o programas automatizados (bots) que ingresan miles de combinaciones en segundos.

Adquirir y activar un certificado HTTPS, el cual codifica, a través de un cifrado de doble vía, los datos sensibles que los usuarios introducen en la página como números de cuentas bancarias o direcciones para que viajen de forma segura por internet y no puedan ser interceptados por hackers.

Realizar respaldos frecuentes del sitio web, para evitar la pérdida de información ante un eventual fallo causado por virus, ataques de hackers o errores humanos y prevenir la caída total del sitio, sin posibilidad inmediata de recuperación. (párr. 5)

Protocolos de seguridad.

Es importante introducir protocolos de seguridad informática estables para proteger la información. Datos y sistemas contra cualquier amenaza o vulnerabilidad. Estas actividades fueron implementadas, ayudan a proteger la integridad de los datos y se aseguran de que se mantenga la información integrando durante su ciclo de vida. Además, se mantiene la privacidad de la información, mientras se mantiene la confidencialidad de la información. Limita el acceso solo a personas autorizadas que ayuden a evitar difundir información. Datos delicados sobre individuos no autorizados.

De acuerdo con Chávez (2024), existen diferentes tipos de protocolos de seguridad que se utilizan a nivel empresarial para cumplir con estas funciones, entre los cuales destacan:

Protocolo TCP/IP

El Protocolo de Control de Transmisión/Protocolo de Internet (Transmission Control Protocol/Internet Protocol, TCP/IP) es un conjunto de reglas que permiten la comunicación entre dos equipos, a través de Internet. (Chávez, 2024).

Protocolo HTTP

El Protocolo de Transferencia de Hipertexto (HTTP) se encarga de la transferencia de información a través de la World Wide Web (WWW). Cuando un usuario ingresa una dirección URL en el navegador, la información es enviada al sitio web, mediante este protocolo. (Chávez, 2024).

Protocolo FTP

El Protocolo de Transferencia de Archivos (File Transfer Protocol, FTP) es utilizado para transferir archivos a través de redes conectadas por TCP/IP. Gracias al FTP, los usuarios pueden conectarse a un servidor para obtener archivos o información necesaria. (Chávez, 2024).

Protocolo SSH

El Protocolo Secure Socket Shell (Intérprete de Órdenes Seguro) ofrece una opción confiable para acceder de forma remota y segura a un ordenador, mediante un canal cifrado. (Chávez, 2024).

Protocolo DNS

También conocido como Sistema de Nombres de Dominio (Domain Name System), este protocolo asigna nombres de dominio a direcciones IP, permitiendo identificar cada URL que un usuario utiliza, a través de su respectiva dirección IP. (Chávez, 2024).

CAPITULO III: MARCO METODOLÓGICO

Enfoques De La Investigación

La investigación puede abordarse desde distintos enfoques metodológicos, los cuales permiten analizar un fenómeno de diferentes maneras, dependiendo de lo que se busca. Elegir un enfoque correcto es clave para alcanzar los objetivos de una investigación, ya que cada uno ofrece herramientas distintas para observar, analizar e interpretar la realidad. A continuación, se presentan los tres enfoques principales: cuantitativo, cualitativo y mixto.

Enfoque cuantitativo

El enfoque cuantitativo se basa en la recolección y análisis de datos numéricos, con el fin de explicar fenómenos, probar hipótesis o identificar relaciones entre variables. Es un enfoque más estructurado, ya que se diseña, desde el principio, con instrumentos bien definidos, como encuestas o cuestionarios y se aplica a muestras representativas para generalizar resultados. De acuerdo con Consultoría (s. f.), el enfoque cuantitativo es: “El enfoque cuantitativo busca probar hipótesis mediante la recolección y el análisis estadístico de datos numéricos. Se basa en la medición objetiva y en la generalización de resultados.” (párr. 4). Este enfoque es muy utilizado en las ciencias exactas, sociales y de la salud, especialmente cuando se necesita evidencia numérica y resultados generalizables. Además, su estructura permite hacer comparaciones claras y aplicar fórmulas estadísticas para validar los datos obtenidos. Aunque a veces se dice que puede dejar de lado el contexto, su ventaja es la precisión y la objetividad.

Enfoque cualitativo

El enfoque cualitativo es más flexible y se centra en la interpretación profunda de los fenómenos, estudiándolos desde la perspectiva de los participantes. En vez de trabajar con números, este enfoque se basa en experiencias, opiniones, emociones y significados. Utiliza técnicas como entrevistas abiertas, grupos focales, observaciones o análisis de contenido. Según Consultoría (s. f.), el enfoque cualitativo es: “El enfoque cualitativo se enfoca en comprender los significados y experiencias de los participantes. Utiliza métodos como entrevistas o análisis de contenido para interpretar la realidad desde una perspectiva subjetiva.” (párr. 5). Este enfoque es ideal cuando se quiere entender el “por qué” de un

comportamiento o fenómeno. Aunque no se puede generalizar con facilidad, sí se obtiene una comprensión profunda de casos particulares. Además, permite que el investigador adapte el proceso, a medida que surgen nuevas ideas, lo que enriquece el estudio.

Enfoque mixto

El enfoque mixto combina elementos del enfoque cuantitativo y cualitativo para lograr una comprensión más completa del fenómeno que se quiere estudiar. Este tipo de enfoque permite aprovechar lo mejor de ambos métodos, ya que por un lado se pueden obtener datos numéricos que dan una visión general y por otro lado, se exploran los significados y experiencias detrás de esos datos. Es muy útil cuando se quiere analizar una problemática desde diferentes perspectivas o validar resultados mediante el cruce de métodos. De acuerdo con Consultoría (s. f.), el enfoque mixto es: “El enfoque mixto combina elementos de los enfoques cuantitativo y cualitativo para obtener una comprensión más completa del objeto de estudio. Este enfoque permite analizar un fenómeno desde distintas perspectivas y validar los hallazgos, mediante la triangulación de métodos.” (párr. 6). Este enfoque se puede aplicar de manera secuencial, es decir, primero hacer la parte cuantitativa y luego la cualitativa (o viceversa), o también de forma simultánea. Lo más importante es que los resultados de ambos enfoques se integran para obtener un análisis más sólido. Al final, esta combinación ayuda a que los hallazgos sean más completos, confiables y enriquecidos, ya que se observan tanto los números, como los significados detrás de ellos.

En resumen, entender los enfoques de investigación ayuda bastante a elegir cómo investigar de forma más efectiva. En el caso de este proyecto de desarrollo de un sistema web de RRHH, conocer estas opciones permite tomar una mejor decisión para obtener datos más útiles y aplicables al diseño del sistema.

Enfoque de investigación seleccionado

Para el desarrollo de este sistema web de gestión de recursos humanos se decidió trabajar con un enfoque mixto, ya que parece el más adecuado para lo que se pretende lograr. Al principio se duda un poco entre usar sólo el cuantitativo, porque se piensa en datos, tiempos, rendimiento del sistema, pero también es clave entender cómo se sienten los

usuarios, qué cosas les resultan útiles, cuáles no, o qué problemas tienen en el día a día. Entonces, se toma la decisión de que lo mejor era combinar ambos enfoques y así tener una mirada más completa.

Con el enfoque cuantitativo se podrá recolectar datos más concretos, como cuántas tareas hace el área de RRHH por día, cuánto tiempo se pierde en trámites manuales, o incluso cuántas incidencias se repiten más seguido. Pero eso no alcanza. Por eso se suma la parte cualitativa, para poder entrevistar al personal, ver cómo usan las herramientas actuales y entender qué les gustaría que cambie.

Este enfoque mixto va a dar una base más sólida para diseñar el sistema, no solo desde lo técnico, sino también desde lo humano. Al fin y al cabo, el sistema no es solo para que funcione bien, sino para que realmente sea útil para quienes lo usan todos los días.

Tipos de Investigación

Existen distintos tipos de investigación que se pueden aplicar dependiendo del objetivo y el nivel de conocimiento que se tenga sobre el tema que se está investigando. Cada tipo tiene una función diferente y aporta a distintas etapas del proceso de indagación. Elegir, correctamente, el tipo de investigación permite plantear una metodología adecuada, así como interpretar de mejor manera los resultados. A continuación, se explican los cuatro tipos más utilizados: descriptiva, exploratoria, explicativa y correlacional.

Investigación descriptiva

La investigación descriptiva tiene como objetivo principal observar y describir, con detalle, las características de un fenómeno, situación o grupo de personas. Este tipo no busca explicar por qué ocurre algo, sino simplemente cómo es o cómo se presenta. Se enfoca más en el qué, el quién, el cuándo y el dónde, pero no en el por qué. Según Consultoría (s. f.), la investigación descriptiva es: “Tiene como objetivo describir con exactitud las características de un fenómeno o situación, sin enfocarse en las causas que lo originan. Es útil para conocer, a profundidad, un hecho.” (párr. 7). Este tipo es útil cuando se quiere tener un panorama general sobre un tema, ya que ofrece una base sólida de información que luego puede servir para investigaciones más profundas. Además, es frecuente en estudios sociales, educativos o de mercado, donde se necesita recopilar datos sobre comportamientos o actitudes.

Investigación exploratoria

La investigación exploratoria se aplica cuando hay poca información sobre un tema o problema. Sirve para familiarizarse con un fenómeno nuevo, generar ideas o preguntas de investigación y, muchas veces, es el primer paso antes de realizar estudios más estructurados. De acuerdo con Consultoría (s. f.), este tipo de investigación es: “Este tipo de investigación se utiliza cuando el problema aún no ha sido bien definido o estudiado. Busca explorar un fenómeno para identificar variables clave o generar hipótesis.” (párr. 8). Este tipo no pretende llegar a conclusiones definitivas, sino más bien abrir camino para futuros estudios. Por eso, su metodología suele ser flexible, permitiendo adaptar el enfoque a medida que se obtienen nuevos datos. Puede utilizar entrevistas abiertas, revisiones de literatura o análisis de casos para obtener un primer acercamiento.

Investigación explicativa

La investigación explicativa va más allá de describir o explorar un fenómeno; su objetivo es entender las causas que lo originan. Este tipo intenta responder a preguntas del tipo “¿por qué sucede esto?” y “¿cómo se relacionan estas variables?”. Implica análisis más complejos y uso de métodos rigurosos para establecer relaciones causa-efecto. Según Consultoría (s. f.), la investigación explicativa es: “Tiene como finalidad identificar las causas de un fenómeno y comprender las relaciones entre diferentes variables. Es un tipo de investigación profunda y estructurada.” (párr. 9). Este tipo es común en estudios científicos, donde es necesario comprobar teorías o hipótesis. Se apoya, frecuentemente, en experimentos, estudios longitudinales o análisis estadístico avanzado. Aunque requiere más tiempo y recursos, sus resultados son más precisos y útiles para tomar decisiones o desarrollar teorías.

Investigación correlacional

La investigación correlacional se enfoca en analizar si existe una relación entre dos o más variables. A diferencia de la explicativa, no intenta determinar cuál causa a la otra, sino si están asociadas de alguna forma. Por ejemplo, se puede estudiar si hay una relación entre el nivel de estrés y el rendimiento académico. De acuerdo con Consultoría (s. f.), este tipo de investigación es: “Se encarga de estudiar la relación o grado de asociación que existe entre

dos o más variables. No busca establecer una causa, sino observar cómo se relacionan entre sí.” (párr. 10). Este tipo de investigación es útil para generar hipótesis o identificar patrones que luego pueden ser profundizados con estudios explicativos. Utiliza métodos estadísticos, como coeficientes de correlación, para medir el grado de relación entre los datos recolectados.

Conocer estos tipos de investigación ayuda a decidir cómo abordar el estudio. Elegir bien permite organizar el trabajo de forma lógica y aprovechar mejor la información obtenida. Para este proyecto, se buscó entender bien el contexto actual de los procesos de RRHH, lo que hace que algunos tipos sean más útiles que otros.

Tipo de Investigación Seleccionado

En cuanto al tipo de investigación, se opta por una investigación descriptiva, porque lo que más interesa, en esta etapa, es conocer cómo están funcionando, actualmente, los procesos del área de RRHH en la organización. No se busca, todavía, explicar causas o hacer relaciones estadísticas profundas, sino más bien tener una idea clara de cómo se trabaja hoy, qué pasos se siguen, qué cosas funcionan bien y cuáles no tanto.

También se usa un poco el tipo exploratorio, sobre todo al principio del proyecto. La verdad es que no se tiene claro por dónde empezar, porque la gestión de RRHH es bastante amplia. Entonces se hace una primera investigación más general para poder identificar los puntos clave y, a partir de ahí, enfocarse más en describir cada parte del proceso, con mayor detalle.

Con esta combinación, será posible definir bien qué funciones debe tener el sistema, qué necesidades hay que cubrir y qué cosas podrían automatizarse o mejorarse. No es una investigación súper técnica ni científica en el sentido más estricto, pero sí es práctica y útil para lograr el objetivo final, que el sistema sea funcional y responda a lo que realmente se necesita.

Fuentes de información

Las fuentes de información son todos aquellos recursos de donde se puede obtener conocimiento o datos que ayudan a desarrollar un trabajo de investigación. Sin estas fuentes, sería casi imposible construir un proyecto con fundamentos, ya que quedaría solo con ideas propias, que podrían ser poco precisas o subjetivas. Por eso, es importante saber buscar información confiable que sirva de apoyo para entender mejor los temas, analizar situaciones o resolver problemas. Además, no solo se trata de encontrar información, sino de saber elegir bien las fuentes y revisar si realmente son útiles y confiables. Muchas veces, hay información que parece buena, pero en realidad no lo es, por eso hay que tener criterio al investigar.

Las fuentes se pueden dividir en tres grandes grupos: primarias, secundarias y terciarias. Cada una cumple un papel distinto y todas son importantes, según lo que se quiera investigar o el momento en que se usen dentro del proceso de investigación.

Fuentes de información primaria

Las fuentes primarias son aquellas que contienen información original, es decir, que no ha sido modificada, resumida o interpretada por alguien más. Se podría decir que es la información "en bruto", tal como fue creada o descubierta. Este tipo de fuente permite acceder, directamente, a los datos reales y concretos de un tema o suceso, por lo tanto, son bastante valiosas para quienes necesitan investigar a fondo y hacer un análisis propio, sin depender de la opinión o la visión de otros.

Algunos ejemplos claros de fuentes primarias serían: libros escritos por los propios autores sobre un tema específico, artículos científicos originales, documentos oficiales, actas, leyes, estadísticas, tesis de grado, entrevistas a expertos, cartas, diarios personales, fotografías, grabaciones, conferencias, videos documentales, etc. Todo lo que venga directamente de la persona o entidad que produjo esa información cuenta como fuente primaria.

Utilizar este tipo de fuente tiene varias ventajas, como poder analizar la información desde cero, sin que esté "contaminada" por otras opiniones. Pero también requiere que el investigador sea muy cuidadoso y crítico, porque hay que entender el contexto en el que se produjo esa información. Por ejemplo, una entrevista hecha hace 20 años puede tener ideas muy diferentes a las de hoy en día, y eso hay que tomarlo en cuenta.

Fuentes de información secundaria

Las fuentes secundarias, en cambio, ya no son originales, sino que son una especie de análisis, resumen o interpretación de las fuentes primarias. Aquí ya se presenta una visión más trabajada, con opiniones, reflexiones o conclusiones hechas por otros autores que ya han estudiado el tema antes. Estas fuentes son muy útiles porque permiten entender mejor que en las fuentes primarias, pueden ser más difíciles de interpretar.

Por ejemplo, si uno quiere investigar sobre la historia de un país, leer un libro de historia que ya analizó varios documentos antiguos (como cartas, leyes, discursos) sería una fuente secundaria. También lo son los artículos de revistas académicas que hacen revisión de otros estudios, los ensayos, las biografías, libros de texto, sitios web educativos y hasta videos explicativos que resumen temas complejos.

Estas fuentes ayudan a ampliar el conocimiento y a ver los temas desde diferentes ángulos. A veces, incluso comparan varias fuentes primarias entre sí o sacan conclusiones interesantes que tal vez no habría notado. Eso sí, como ya incluyen la opinión de otra persona, siempre es bueno revisar si ese autor o autora es confiable y no quedarse solo con una fuente secundaria, sino contrastarla con otras.

Fuentes de información terciaria

Por último, las fuentes terciarias son aquellas que no aportan información nueva ni análisis, sino que sirven como guía para encontrar otras fuentes. Se usan más bien al principio de una investigación, cuando todavía no sabe mucho del tema y necesita orientarse o tener una idea general antes de profundizar. También son muy útiles cuando se necesita ubicar información específica de forma rápida, como una definición o una fecha importante.

Ejemplos de fuentes terciarias son las enciclopedias, los diccionarios, los índices temáticos, las bibliografías, los manuales, las guías o incluso buscadores académicos que agrupan otras fuentes. Aunque no tienen contenido profundo, son muy útiles para comenzar una investigación, para repasar conceptos clave o para confirmar datos básicos.

Muchas veces estas fuentes ayudan a no perder tiempo buscando a ciegas. Por ejemplo, un índice bibliográfico puede indicar en qué página se habla de un tema en particular

y una enciclopedia puede dar un resumen breve, pero claro, para comenzar a entender un concepto nuevo.

Variables

Las variables son super importantes en cualquier investigación, porque ayudan a identificar lo que se va a estudiar o medir. Básicamente, una variable es como una característica que cambia o puede ser diferente, dependiendo de la situación, el tiempo, las personas, etc. Por ejemplo, si uno estudia el desempeño de los empleados, el “desempeño” sería una variable porque puede variar entre alto, medio o bajo. Según SalusPlay, (s. f.), “Una variable es una propiedad que puede fluctuar y cuya variación es susceptible de medirse u observarse. Son atributos o características que se miden en los sujetos de estudio. Como ejemplos de variables podemos citar: sexo, raza, tipo de población (urbana, rural), accesibilidad a los servicios de salud, número de hijos, peso en kilogramos, talla en centímetros, etc.” (párr. 1). Esta definición ayudó a entender que no es solo una palabra técnica, sino que es algo real que se puede observar y medir.

Variables conceptuales

Las variables conceptuales son como el primer paso para poder investigar algo. Son las ideas generales que ayudan a entender lo que se está tratando de estudiar. Por ejemplo, si alguien quiere investigar sobre la “autoestima” o la “motivación”, esas serían variables conceptuales porque son conceptos bastante amplios y no se pueden medir tan fácilmente. Son más abstractas y por eso se necesita primero definir las bien, como qué se entiende por ese término, qué incluye y qué no. A veces, diferentes personas pueden entender lo mismo de distintas formas, así que aclarar bien estas variables ayuda a que se hable de lo mismo en la investigación. Aunque no se puedan medir directamente, sirven como base para después crear variables más específicas que sí se puedan observar o contar.

Variables operacionales

Estas variables son como una versión más concreta de las conceptuales. Es decir, cuando ya se tiene claro qué se quiere estudiar, hay que buscar una manera de medirlo y ahí entran las variables operacionales. Se trata de ponerle medidas y definiciones claras a algo que puede ser un poco vago al principio. Por ejemplo, con la “autoestima”, se puede decidir medirla con una encuesta donde las personas respondan preguntas sobre cómo se sienten consigo mismas. Entonces, la variable operacional sería la puntuación en esa encuesta. Esto hace que el estudio sea más objetivo, porque cualquier otra persona podría aplicar la misma encuesta y obtener resultados comparables. Además, se tienen que establecer cosas como qué instrumentos se van a usar, qué pasos se van a seguir para medir y qué datos se van a recolectar, lo cual también ayuda a que la investigación sea más organizada y seria.

Variables instrumentales

Las variables instrumentales funcionan como herramientas dentro del proceso de investigación. No son exactamente lo que se quiere estudiar, sino que se usan para ayudar a medir o a recolectar información de forma más precisa. Pueden ser desde aparatos tecnológicos, como un termómetro en un experimento, hasta métodos como una entrevista estructurada o incluso otras variables que se escogen con un propósito específico para facilitar el análisis. Son clave para asegurar que los datos que se obtienen sean confiables, es decir que, si se repite el estudio, los resultados no cambien demasiado. También sirven para que los resultados tengan sentido y realmente se pueda decir algo útil con ellos. En resumen, sin estas variables, sería más difícil garantizar que lo que se midió refleja bien la realidad del fenómeno que se está investigando.

Tabla 4
Variables de investigación

Objetivo específico	Variable	Variable conceptual	Variable instrumental	Variable operacional
Analizar los requerimientos funcionales y no funcionales del sistema web.	Requerimientos	Según Northware (2022), “Los requerimientos son: Una descripción de lo que un programa de software en particular debe hacer. Actúan como pautas para que los desarrolladores creen un producto funcional que satisfaga las necesidades de los usuarios.” (párr. 3).	Guía de entrevista. Guía de Cuestionario. Guía de observación.	Entrevistas, encuestas. Observación.
Diseñar la base de datos y la arquitectura general del sistema web, considerando la estructura de datos, relaciones entre entidades y componentes funcionales, mediante el uso de herramientas como MySQL Workbench para modelado de datos.	Arquitectura del sistema.	De acuerdo con Communications (2024), “La arquitectura de 'software' de un sistema representa las decisiones de diseño relacionadas con su estructura y comportamiento global. En otras palabras, hace referencia a la forma en que se organizan los elementos en el proceso de creación de un 'software'.” (párr. 5).	MySQL Workbench.	Documento de análisis. Casos de uso.
Programar los módulos del sistema	Módulos del sistema.	De acuerdo con Colaboradores de Wikipedia,	Utilizando el entorno de desarrollo de	Diagrama Entidad-Relación

<p>utilizando Java con Spring Boot, React con TypeScript y Tailwind CSS.</p>		<p>(2023), “En programación, un módulo es una porción de un programa de ordenador. De las varias tareas que debe realizar un programa para cumplir con su función u objetivos, un módulo realizará, comúnmente, una de dichas tareas (o varias, en algún caso).” (párr. 1).</p>	<p>Visual Studio Code y IntelliJ IDEA.</p>	<p>Diagrama de Clases. Documentos de diseño.</p>
<p>Realizar las pruebas de cada módulo para la verificación de su funcionalidad.</p>	<p>Validación del sistema.</p>	<p>Según QBD Group (s. f.), “La Validación de Sistemas Informáticos (CSV) se define como el establecimiento de pruebas documentadas que proporcionan un alto grado de seguridad de que un proceso específico producirá de forma consistente un producto que cumple las especificaciones y características de calidad predeterminadas.” (párr. 4).</p>	<p>Pruebas funcionales.</p>	<p>Casos de prueba.</p>

Fuente: Elaboración propia

Población, Muestra y Cálculo de la Muestra

Población

La población en esta investigación está conformada por todos los empleados del Supermercado Hnos. Pineda, ubicado en Hojancha, Guanacaste. Actualmente, según los datos proporcionados por la empresa, se cuenta con un total de 10 colaboradores. Esta cantidad incluye, tanto al personal administrativo como operativo, que participa, directamente, en los procesos de Recursos Humanos, ya sea como usuarios del sistema o como beneficiarios de este. De acuerdo con Salusplay (s.f),” La población o universo es el conjunto de individuos u objetos de los que se desea conocer algo en una investigación. Es la totalidad de individuos o elementos, en los cuales, puede presentarse determinada característica que va a ser estudiada.” (párr. 1).

Por lo tanto, para este estudio se va a tomar como población total a los 10 empleados del supermercado, ya que todos forman parte del contexto que se desea analizar.

Muestra

La muestra es una parte seleccionada de la población que se estudia con el fin de obtener información que se pueda aplicar al grupo completo. En investigaciones donde hay muchas personas, es común trabajar solo con una muestra para ahorrar tiempo y recursos.

Sin embargo, en este proyecto, la muestra está directamente relacionada con los trabajadores del Supermercado Hnos Pineda, ya que son ellos quienes viven los procesos actuales de Recursos Humanos y, por lo tanto, su experiencia es clave para diseñar el sistema. Según Salusplay (s.f),” La muestra es el grupo de individuos que realmente se estudiará. Para que se puedan generalizar los resultados tiene que seleccionarse de modo que sea lo más representativa posible de la población destinataria y con una cantidad suficiente para obtener respuestas válidas.” (párr. 5).

Por lo tanto, al seleccionar a los trabajadores del Supermercado Hnos Pineda como muestra, se garantiza que la información recolectada refleje, fielmente, la realidad de los procesos de Recursos Humanos en la empresa, permitiendo diseñar un sistema que responda a sus necesidades reales.

Fórmula

Aunque en este caso no se aplica, debido al tamaño reducido de la población, es importante mencionar la fórmula que se utiliza, comúnmente, para calcular el tamaño de una muestra cuando se trabaja con una población finita. Esta fórmula resulta útil en investigaciones donde no es viable incluir a todos los miembros de la población, ya sea por razones de tiempo, costos o logística.

Según QuestionPro (s. f.), la siguiente fórmula es utilizada, frecuentemente, para este tipo de cálculos e incluye el significado de cada uno de sus elementos, así como los valores más comunes, según el nivel de confianza:

Ejemplo fórmula:

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

Donde:

n = Tamaño de muestra buscado

N = Tamaño de la población o universo

Z = Parámetro estadístico que depende el nivel de confianza (NC)

p = Probabilidad de que ocurra el evento estudiado (éxito)

q = (1 - p) = Probabilidad de que no ocurra el evento

e = Error de estimación máximo aceptado

Aquí están las puntuaciones Z para los niveles de confianza más comunes:

90% - Puntuación Z = 1,645

95% - Puntuación Z = 1.96

99% - Puntuación Z = 2.576

En este caso no se utilizará la fórmula, ya que el supermercado cuenta, únicamente, con 10 empleados en total. Debido a que la población es muy pequeña, se decidió trabajar

con un censo, lo que significa que se incluirá a todas las personas en la recolección de datos. Esto garantiza que se consideren todos los puntos de vista del personal y permite obtener información más completa y precisa, sin dejar a nadie por fuera.

Proceso para la Recolección y el Análisis de Datos

Recolección de datos

Primero que todo, hay que tener bien claro cuál es el objetivo de la investigación y qué método se va a usar para recolectar los datos. También es importante saber, exactamente, qué tipo de información se necesita y para qué se va a usar. En este caso, se planea usar dos técnicas principales: entrevistas y observaciones. Las entrevistas sirven para conseguir información directa de las personas involucradas, mientras que las observaciones ayudan a entender mejor el ambiente y cómo se comportan las personas en su entorno natural.

Teniendo eso en cuenta, para hacer las entrevistas es necesario preparar una guía con preguntas que cubran los temas importantes del estudio. En este caso, el tema central es el área de recursos humanos. Por otro lado, para las observaciones hay que hacer un esquema donde se especifique qué se va a observar y cómo se va a anotar esa información. Es súper importante elegir bien a las personas que van a participar, porque de ahí es de donde se sacará la información más valiosa para el trabajo.

Además, cuando ya se tienen los resultados, hay que validarlos bien para asegurarse de que todo lo que se encontró realmente tenga sentido y sea confiable. Una forma de hacer esto es usando la triangulación, que básicamente consiste en comparar datos de diferentes fuentes. También es útil tener opiniones de diferentes personas, ya que eso ayuda a detectar errores o puntos ciegos que uno mismo no ve. Todo esto hace que las conclusiones del estudio sean más objetivas y con más peso.

Análisis de datos

Para poder analizar bien los datos, hay que seguir varios pasos. Lo primero es transcribir las entrevistas que se hayan grabado, ya que tenerlas por escrito hace mucho más fácil analizarlas después. También es importante organizar todas las notas que se tomaron durante las observaciones, ya sea por orden de tiempo o por temas específicos, para que sea más sencillo manejarlas y entenderlas.

Luego de eso, viene la parte de codificar los datos. Esta codificación tiene dos etapas: la abierta y la axial. La codificación abierta sirve para identificar conceptos y ver qué propiedades o características tienen dentro de la información. Después, la codificación axial ayuda a conectar esas categorías principales con otras subcategorías relacionadas. Básicamente, es como poner todo en orden, donde un tema central tiene varios subtemas que se le relacionan.

Después de codificar, se hace el análisis temático. Esto consiste en agrupar las categorías encontradas en ideas más generales para poder encontrar patrones o tendencias que tengan sentido con lo que se está investigando. Este paso ayuda a dar forma a la información y a hacerla más fácil de interpretar.

Una vez hecho eso, es clave interpretar los datos, lo cual significa que hay que ver cómo los resultados se relacionan con los objetivos de la investigación. En este punto se trata de darle sentido a los patrones que salieron, pensando en lo que realmente significan y qué se puede concluir a partir de ellos. Si esta interpretación se hace bien, se puede sacar información muy útil y relevante.

Otro paso que no se puede dejar por fuera es validar los resultados. Esto es súper importante para asegurarse de que lo que se encontró es confiable y que de verdad representa la realidad que se está estudiando. Para validar se pueden usar varias estrategias, como repetir el estudio, pedir opiniones de otras personas expertas o usar diferentes métodos para comparar si se llega a las mismas conclusiones.

Por último, hay que presentar todo lo que se encontró en un informe. Este informe debe explicar cómo se recogieron y analizaron los datos. También tiene que mostrar los resultados de forma clara, incluyendo frases textuales de las entrevistas y ejemplos de las observaciones para apoyar lo que se dice. Una buena presentación ayuda a que los demás entiendan mejor la investigación y la consideren válida.

CAPITULO IV: ANÁLISIS DE RESULTADOS

Resultados

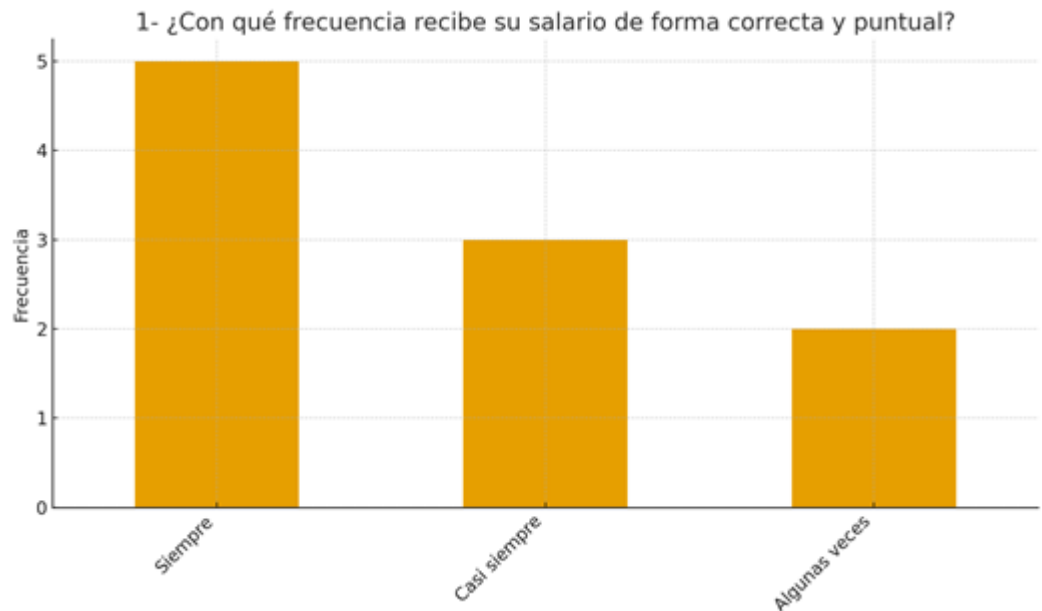
En este capítulo se discuten los resultados obtenidos a partir de la aplicación del cuestionario a los colaboradores del Supermercado Hnos. Pineda, así como la entrevista realizada al encargado administrativo. El análisis busca interpretar los datos en relación con los fundamentos teóricos planteados en el marco referencial, con el fin de sustentar la propuesta de un prototipo funcional para la gestión de recursos humanos.

Resultados del cuestionario aplicado a empleados

El cuestionario aplicado a los empleados tuvo como objetivo identificar las principales limitaciones del sistema actual de gestión de recursos humanos. A continuación, se presentan los resultados más relevantes en forma de gráficos y tablas, con el fin de visualizar, de manera clara, las percepciones de los colaboradores.

Figura 2

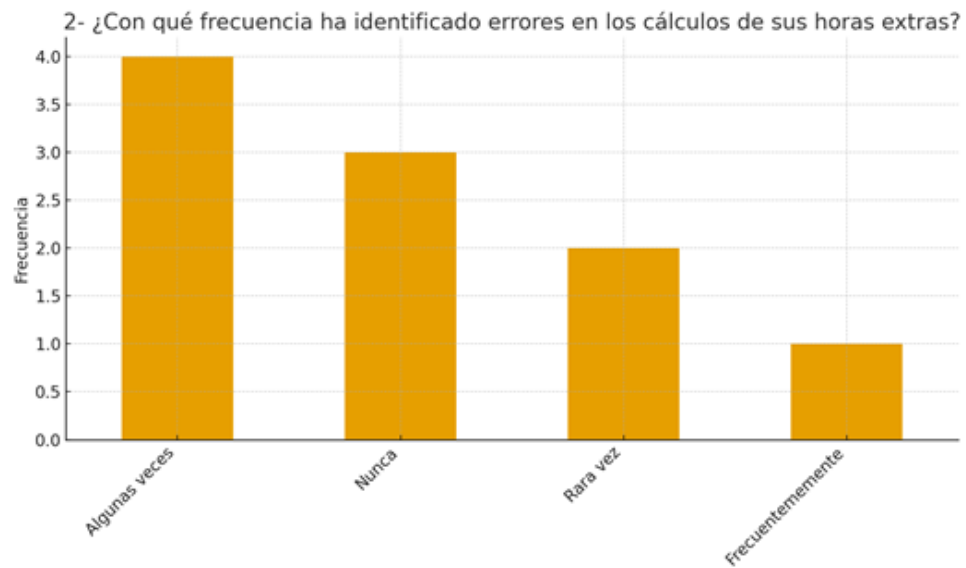
Distribución de respuestas sobre '1- ¿Con qué frecuencia recibe su salario de forma correcta y puntual?'



Fuente: Elaboración propia

Figura 3

Distribución de respuestas sobre '2- ¿Con qué frecuencia ha identificado errores en los cálculos de sus horas extras?'



Fuente: Elaboración propia

Figura 4

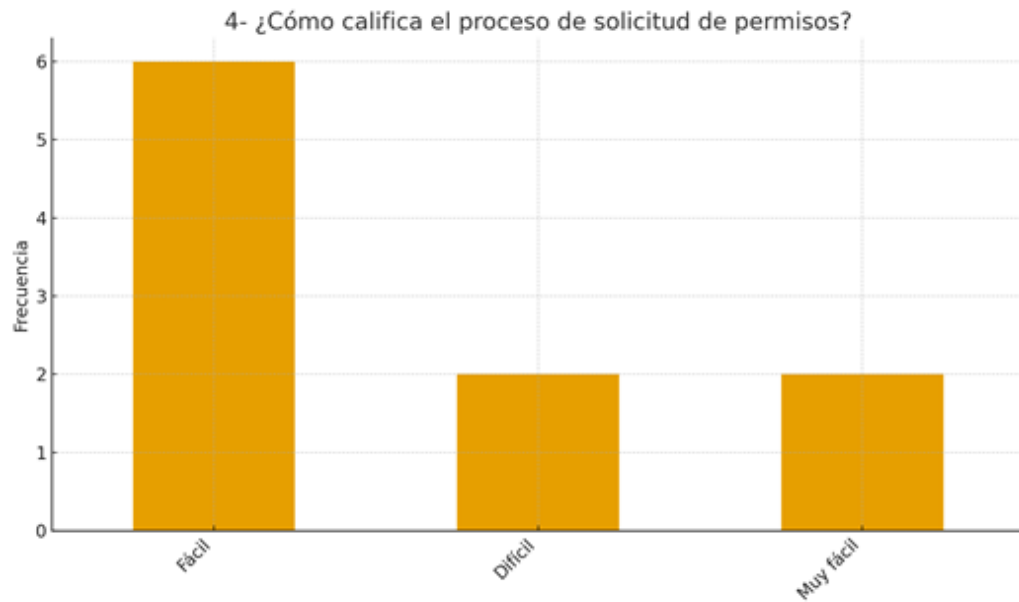
Distribución de respuestas sobre '3- ¿Qué tan clara es la información que recibe sobre sus días de vacaciones?'



Fuente: Elaboración propia

Figura 5

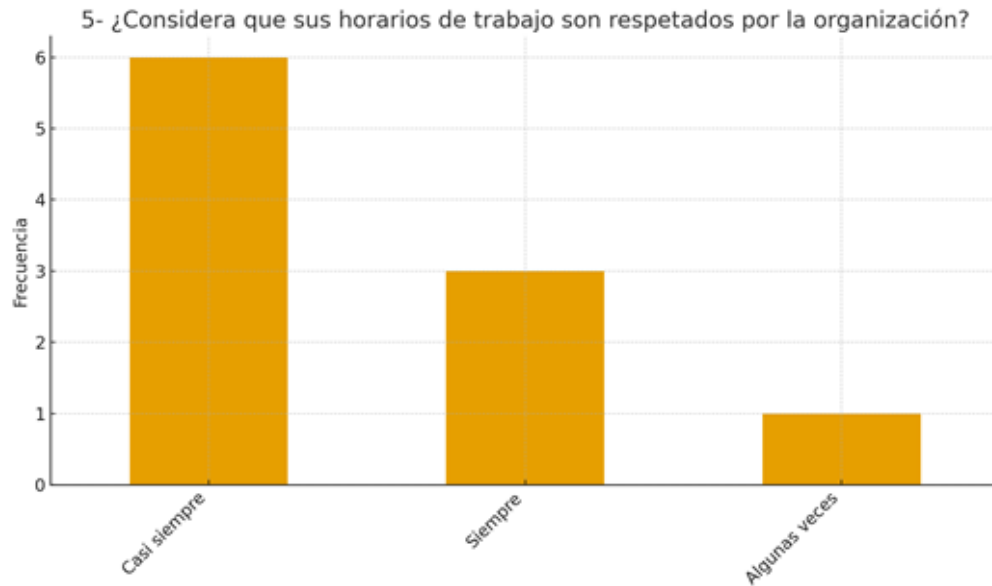
Distribución de respuestas sobre '4- ¿Cómo califica el proceso de solicitud de permisos?'



Fuente: Elaboración propia

Figura 6

Distribución de respuestas sobre '5- ¿Considera que sus horarios de trabajo son respetados por la organización?'



Fuente: Elaboración propia

Figura 7

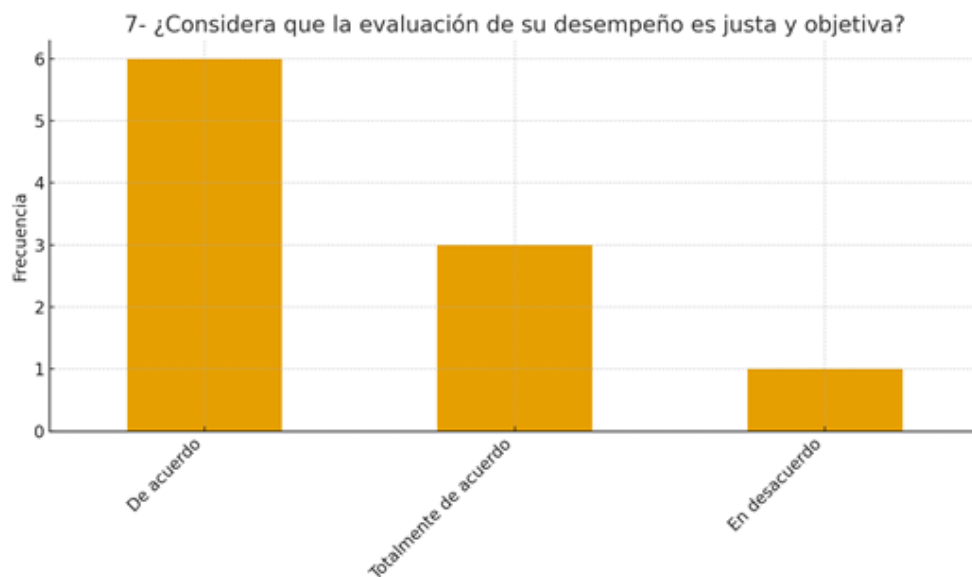
Distribución de respuestas sobre '6- ¿Tiene acceso a su historial de asistencia (entradas y salidas)?'.



Fuente: Elaboración propia

Figura 8

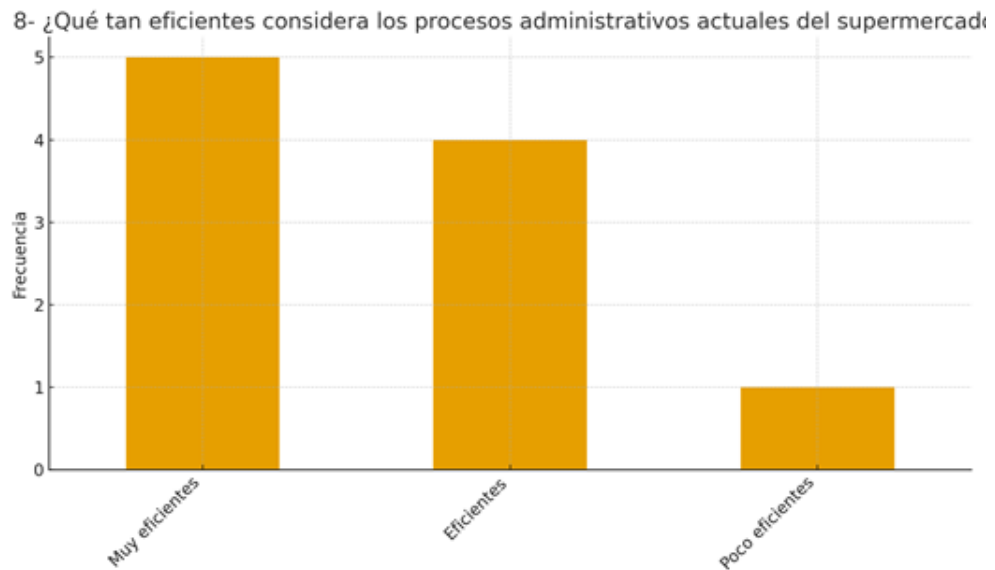
Distribución de respuestas sobre '7- ¿Considera que la evaluación de su desempeño es justa y objetiva?'.



Fuente: Elaboración propia

Figura 9

Distribución de respuestas sobre '8- ¿Qué tan eficientes considera los procesos administrativos actuales del supermercado?'



Fuente: Elaboración propia

Figura 10

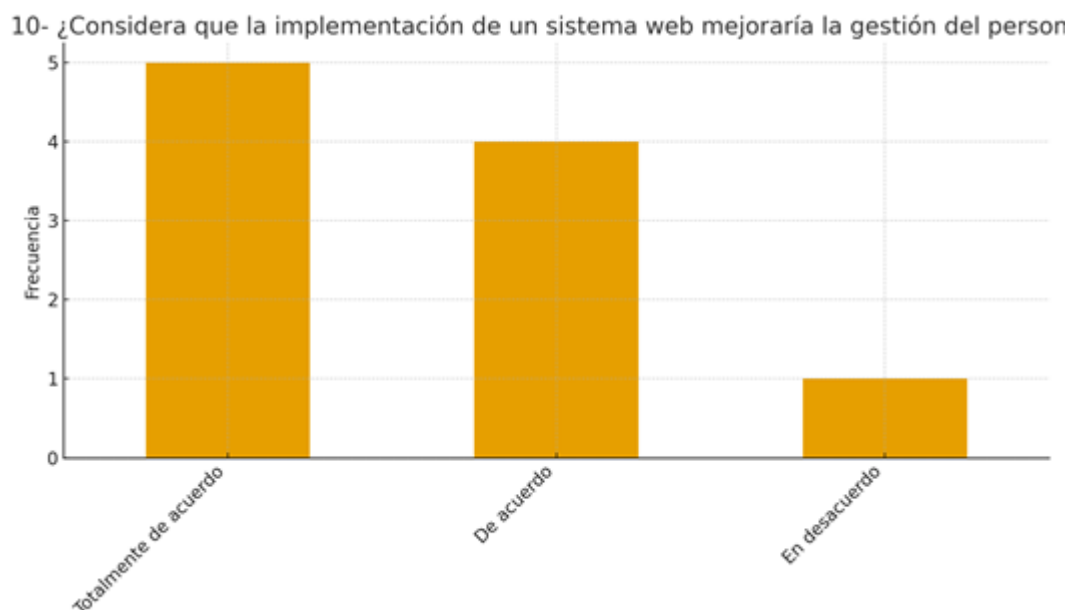
Distribución de respuestas sobre '9- ¿Ha recibido algún tipo de bonificación relacionada con su rendimiento laboral?'



Fuente: Elaboración propia

Figura 11

Distribución de respuestas sobre '10- ¿Considera que la implementación de un sistema web mejoraría la gestión del personal?'



Fuente: Elaboración propia

Los gráficos muestran que existe una percepción generalizada de que los procesos actuales, basados en registros manuales y hojas de Excel, generan dificultades en la organización de horarios, vacaciones, permisos y cálculo de planillas. De igual manera, los empleados manifestaron interés en contar con un sistema digital que les permita tener acceso claro y transparente a su información laboral.

Resultados de la entrevista al encargado administrativo

La entrevista realizada al encargado administrativo permitió profundizar en las dificultades cotidianas de la gestión manual del recurso humano. Entre los aspectos más relevantes se destacan:

- El control de las horas laborales resulta complejo, debido a los cambios de turnos y al registro manual en Excel.
- Las vacaciones se gestionan de manera manual, lo que dificulta la organización cuando varios colaboradores solicitan descanso al mismo tiempo.

- Se han presentado errores en el cálculo de salarios, en especial con horas extras y rebajos.
- Los permisos se solicitan de manera verbal, lo que genera riesgo de pérdida de datos importantes.
- No existe una evaluación formal de desempeño, únicamente se observa, empíricamente, el cumplimiento de funciones.

El encargado administrativo reconoció que el sistema actual 'funciona, pero no es lo más práctico', ya que demanda tiempo adicional y aumenta la probabilidad de errores en la gestión.

Relación de los resultados con el marco teórico

Los hallazgos obtenidos guardan relación directa con lo planteado en el marco referencial. Diversos autores sostienen que la gestión manual de los procesos de recursos humanos incrementa los errores y limita la eficiencia operativa. En este caso, tanto los empleados, como la administración, confirmaron que el uso de registros en Excel y procesos verbales ocasiona problemas de organización, retrasos en pagos y pérdida de información.

De acuerdo con la teoría, un sistema de gestión de recursos humanos debe automatizar la nómina, el control de permisos, vacaciones e incapacidades, además de establecer métricas claras de evaluación del desempeño. Los resultados obtenidos en el supermercado validan esta necesidad, ya que reflejan, precisamente, los problemas que la literatura advierte.

Resumen

En resumen, el análisis de resultados evidencia una brecha considerable entre la gestión manual actual y las necesidades reales del supermercado. Los empleados requieren claridad y acceso inmediato a su información, mientras que la administración necesita una herramienta que reduzca los errores y agilice los procesos. Los resultados obtenidos justifican la pertinencia de desarrollar el prototipo funcional de gestión de recursos humanos propuesto, ya que permitiría centralizar la información, mejorar la precisión de los cálculos y brindar mayor transparencia y satisfacción, tanto para los empleados, como para la administración.

CAPÍTULO V: PROPUESTA

Requerimientos

Requerimientos funcionales

Los requerimientos funcionales describen las acciones específicas que el sistema debe ser capaz de realizar para apoyar, correctamente, la gestión del talento humano dentro del Supermercado Hnos Pineda. Cada módulo fue definido tomando en cuenta los procesos administrativos actuales y las necesidades detectadas durante el análisis, con el propósito de reducir errores, agilizar tareas repetitivas y centralizar la información laboral del personal.

1. Módulo de Nómina

RF1: Registrar y mantener la información necesaria para el cálculo de la nómina quincenal.

RF1.1: El sistema debe permitir ingresar datos como salario base, jornadas, deducciones y bonificaciones.

RF1.2: Se debe contar con un formulario donde se asocie cada cálculo al empleado correspondiente.

RF2: Calcular, automáticamente, el salario bruto y neto en función de la asistencia, horas extra y rebajos legales.

RF2.1: El sistema debe aplicar rebajos obligatorios como CCSS y pólizas laborales.

RF2.2: El cálculo debe contemplar rebajos y beneficios adicionales definidos por la empresa (adelantos, incentivos, etc.).

RF2.3: La nómina deberá quedar en estado “pendiente” para su revisión antes de aprobación definitiva.

RF3: Generar reportes de nómina para cada periodo.

RF3.1: Los reportes deberán poder filtrarse por empleado o por periodo.

RF3.2: El sistema debe mostrar un resumen general con totales de salarios, deducciones y pagos efectuados.

2. Módulo de Horas Extra

RF4: Permitir el registro de horas extra trabajadas por cada empleado.

RF5: El encargado de RRHH podrá aprobar, rechazar o modificar los registros antes de ser enviados a nómina.

RF6: El sistema debe notificar, automáticamente, al empleado sobre la resolución de su solicitud.

RF7: Las horas extra aprobadas se sumarán, automáticamente, al cálculo salarial correspondiente.

3. Módulo de Vacaciones

RF8: Registrar solicitudes de vacaciones indicando fechas y cantidad de días.

RF9: Notificar al encargado para su aprobación o rechazo.

RF10: Descontar, automáticamente, los días aprobados del saldo disponible.

RF11: Mantener un historial de solicitudes anteriores.

RF12: Generar reportes individuales con días tomados, disponibles y pendientes.

4. Módulo de Aguinaldo

RF13: Calcular el aguinaldo en base al salario devengado en los últimos 12 meses.

RF14: Guardar el cálculo para cada empleado como respaldo.

RF15: Permitir la generación de reportes con detalle del cálculo y salario promedio.

5. Módulo de Incapacidades

RF16: Registrar incapacidades indicando tipo, fechas y entidad emisora (CCSS, INS, médico privado).

RF17: Guardar la información del tipo de incapacidad (médica, maternidad, riesgo laboral, etc.).

RF18: Generar reportes que indiquen días incapacidad, impacto en salarios y tipo de cobertura.

6. Módulo de Permisos

RF19: Permitir al empleado solicitar permisos con o sin goce salarial.

RF20: El encargado debe poder aprobar o rechazar solicitudes, quedando registro de decisión y fecha.

RF21: El sistema notificará a RRHH y al colaborador la resolución.

RF22: Los permisos sin goce deberán reflejarse en el cálculo de nómina.

7. Módulo de Asistencia

RF23: Registrar entradas y salidas por empleado.

RF24: Detectar, automáticamente, tardías, ausencias y salidas anticipadas.

RF25: Generar reportes por puntualidad, ausencias y asistencia general.

8. Módulo de Liquidación

RF26: Calcular, automáticamente, la liquidación con base en salario, vacaciones pendientes y normativa laboral.

RF27: Guardar desglose de pago y fechas correspondientes.

RF28: Generar reportes con el cálculo realizado.

9. Módulo de Evaluación de Desempeño

RF29: Configurar criterios de evaluación por puesto.

RF30: Registrar la evaluación asociada al empleado evaluado.

RF31: Generar reportes comparativos por empleado y por periodo.

10. Mantenimientos generales

RF32: Permitir insertar nuevos registros en tablas principales y de catálogo.

RF33: Permitir editar datos existentes.

RF34: Permitir eliminar datos obsoletos bajo control de auditoría.

11. Consultas

RF35: Permitir consultas filtradas de cualquier entidad del sistema.

RF36: Mostrar resultados de forma organizada, clara y exportable.

12. Reportes

RF37: Permitir generar reportes impresos o en formato PDF.

RF38: Los reportes deben permitir configurar encabezado, pie y columnas visibles.

13. Seguridad

RF39: Implementar autenticación de usuarios mediante credenciales seguras.

RF40: Asignar roles con permisos diferenciados (ej. Admin RRHH, Empleado, Supervisor).

Requerimientos no funcionales

RNF1 – Rendimiento:

El sistema debe responder en menos de 3 s por consulta, usando conexión estable.

RNF2 – Usabilidad:

La interfaz debe ser intuitiva, con diseño responsivo para PC y tablet.

RNF3 – Seguridad:

El sistema debe proteger datos personales conforme a Ley 8968 y buenas prácticas de cifrado.

RNF4 – Escalabilidad:

Debe poder ampliarse para más módulos o más empleados, sin deterioro del rendimiento.

Análisis

El Supermercado Hnos. Pineda, ubicado en Hojancha, Guanacaste, ha mostrado un crecimiento sostenido en los últimos años, consolidándose como un referente comercial en la comunidad. No obstante, la empresa enfrenta limitaciones significativas en la gestión

administrativa de sus recursos humanos, dado que muchos de los procesos se realizan de forma manual o con hojas de cálculo.

Esto genera problemas frecuentes como cálculos incorrectos en la nómina, retrasos en pagos de liquidaciones, errores en los aguinaldos, falta de control sobre incapacidades y permisos, ausencia de registros confiables de asistencia y horarios, así como la inexistencia de métricas objetivas para la evaluación del desempeño de los colaboradores.

Ante este panorama, se plantea la implementación de un sistema web de gestión de recursos humanos que automatice los procesos internos, centralice la información en una base de datos segura y proporcione herramientas que respalden la toma de decisiones. El sistema busca mejorar la eficiencia administrativa, garantizar la precisión de los cálculos, asegurar la transparencia en la gestión laboral y contribuir al bienestar de los colaboradores, mediante procesos más claros y confiables.

Análisis detallado del software por desarrollar

El sistema propuesto se desarrollará bajo un enfoque cliente–servidor, utilizando tecnologías modernas que aseguren robustez, escalabilidad y facilidad de uso.

Frontend: Se implementará con React y TypeScript, lo que permite la creación de interfaces dinámicas, modulares y fáciles de mantener. El diseño visual se apoyará en Tailwind CSS, garantizando una experiencia de usuario responsiva y adaptable a diferentes dispositivos.

Backend: El núcleo de la aplicación estará en Java con Spring Boot, framework que ofrece seguridad, estabilidad y escalabilidad. Los servicios se expondrán mediante controladores RESTful, que permitirán la comunicación eficiente con el frontend.

Base de datos: Se emplea MySQL como sistema gestor de bases de datos relacional. Su elección responde a su confiabilidad, compatibilidad con aplicaciones web y amplia documentación de soporte.

Módulos principales: El sistema abarcará las funciones críticas de la gestión del talento humano:

- Nóminas.
- Horas extra.
- Vacaciones.

- Aguinaldo.
- Incapacidades.
- Permisos.
- Asistencia.
- Liquidación.
- Evaluación de rendimiento.
- Mantenimientos, consultas, reportes y seguridad.

Cada módulo se integra en una estructura modular que facilita el mantenimiento y futuras extensiones del sistema.

Gestionar Nóminas

Este módulo es el eje principal del sistema. Permite administrar toda la información contable relacionada con la nómina de los empleados.

Su función es calcular, automáticamente, los salarios brutos y netos, de acuerdo con los datos del empleado, las horas trabajadas, los rebajos aplicables y los beneficios establecidos.

Además, almacena un registro histórico de cada nómina generada, lo que facilita el control y la trazabilidad de los pagos realizados.

Gestionar Liquidación

Este módulo automatiza el proceso de cálculo y gestión de liquidaciones laborales.

Toma en cuenta variables como los días trabajados, las horas extras y los periodos no laborados, garantizando exactitud y cumplimiento legal en el pago final a cada empleado.

Su implementación elimina los errores comunes asociados al cálculo manual y asegura una liquidación justa y oportuna.

Calcular Aguinaldo

El módulo de aguinaldo calcula, de manera automática, el monto correspondiente a cada empleado, según las disposiciones legales vigentes en Costa Rica.

Considera los ingresos obtenidos durante el periodo establecido por ley y realiza el cálculo, sin necesidad de intervención manual.

Con este módulo se evitan errores y retrasos en los pagos, fortaleciendo la transparencia de la gestión salarial.

Gestionar Horas Extras

Este módulo lleva el control exacto de las horas adicionales trabajadas por los empleados.

Registra las horas extras autorizadas y calcula, automáticamente, el monto correspondiente, según el tipo de jornada y el porcentaje de recargo.

Además, integra los resultados, directamente, con la nómina, reduciendo errores contables y asegurando la correcta compensación del personal.

Gestionar Vacaciones

El módulo de vacaciones administra, de forma precisa, los días de descanso de los empleados.

Permite registrar las solicitudes, aprobar o denegar las mismas y calcular el saldo disponible de vacaciones.

Con esto se evita la pérdida de información y los errores en el registro manual, garantizando que cada empleado reciba los días de descanso que le corresponden por ley.

Gestionar Incapacidades

Su propósito es llevar un control detallado de las incapacidades médicas registradas por los empleados.

El sistema almacena la información sobre el tipo de incapacidad, la duración y los días cubiertos, ajustando, automáticamente, los cálculos de la nómina, según corresponda.

De esta manera, se mantiene un seguimiento exacto de los periodos de ausencia por motivos de salud.

Gestionar Permisos

Este módulo permite gestionar, de manera formal y estructurada, las solicitudes de permisos laborales.

Cada permiso se registra con su respectiva fecha, motivo y estado (aprobado, pendiente o rechazado), garantizando control y transparencia en la planificación interna del supermercado. También ayuda a mantener un historial que facilita el control de ausencias justificadas.

Gestionar Asistencia

El módulo de asistencia registra las entradas y salidas de los empleados, controlando, de manera automática, la puntualidad y la presencia diaria. Estos datos se utilizan, posteriormente, para calcular el salario correspondiente y las horas efectivamente trabajadas.

El objetivo es mejorar la disciplina laboral y evitar inconsistencias en los pagos relacionados con la asistencia.

Evaluar Rendimiento

Este módulo evalúa el desempeño de los empleados, mediante criterios definidos por la administración. Permite registrar resultados de evaluaciones, asignar puntajes y generar informes que reflejen el rendimiento individual y grupal. Su aplicación contribuye a tomar decisiones informadas respecto a ascensos, incentivos o capacitaciones.

Mantenimientos

El módulo de mantenimientos es el responsable de permitir la creación, modificación, actualización y eliminación de datos en las diferentes entidades del sistema. Garantiza la consistencia de la información y permite realizar ajustes controlados, sin afectar la integridad de la base de datos.

Consultas

Este módulo brinda la posibilidad de obtener información específica del sistema mediante búsquedas personalizadas. Facilita la localización rápida de registros y datos históricos, sirviendo de apoyo para tareas administrativas y de supervisión.

Reportes

El módulo de reportes genera documentos e informes con la información contenida en el sistema, presentándolos de forma ordenada y exportable (PDF o Excel). Permite imprimir o visualizar los resultados, según las necesidades de la administración y contribuye al proceso de toma de decisiones gerenciales.

Seguridad

El módulo de seguridad gestiona la autenticación de usuarios y el control de accesos.

Implementa contraseñas cifradas y la asignación de roles (administrador, empleado e invitado), asegurando la protección de los datos confidenciales. Este componente es esencial para preservar la integridad, disponibilidad y confidencialidad de la información almacenada en el sistema.

Análisis detallado del hardware

El desarrollo y posterior uso del sistema requiere de una plataforma de hardware que cumpla con los requisitos mínimos para asegurar un funcionamiento óptimo.

Hardware disponible en la empresa: El supermercado cuenta con una computadora de escritorio de gama media, conexión a internet estable y periféricos básicos como impresoras y escáneres, suficientes para operar el sistema en la administración.

Hardware utilizado para el desarrollo: Procesador AMD Ryzen 5, de alto rendimiento, 32 GB de memoria RAM, que permite la ejecución de múltiples procesos simultáneamente, SSD de 1 TB, que ofrece rapidez en la lectura y escritura de datos.

Escalabilidad: En caso de que la empresa aumente su número de colaboradores o expanda operaciones, el sistema podrá escalar de manera efectiva.

Análisis detallado de las telecomunicaciones

El funcionamiento del sistema depende de una infraestructura de telecomunicaciones confiable que asegure la comunicación entre los diferentes componentes del software.

Conectividad actual: El supermercado cuenta con conexión a internet estable, lo que permite la operación del sistema, sin requerir infraestructura compleja.

Requisitos mínimos de red:

- Ancho de banda de al menos 10 Mbps para garantizar fluidez en el acceso.
- Router con capacidad de asignación de direcciones IP dinámicas.
- Configuración básica de firewall para evitar accesos no autorizados.

Seguridad en las comunicaciones:

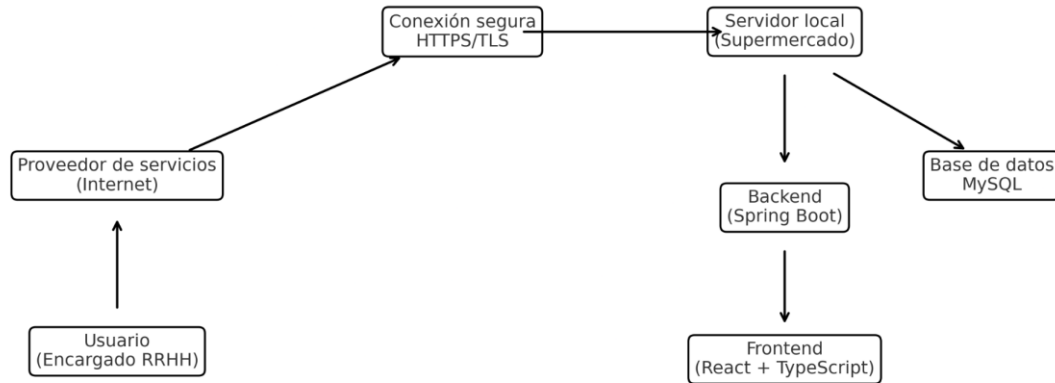
- Transmisión de datos bajo el protocolo HTTPS para proteger la información en tránsito.
- Uso de credenciales seguras y políticas de contraseñas robustas.
- Registro de accesos para monitorear intentos de conexión y prevenir vulnerabilidades.

Proyecciones futuras:

Si el supermercado llega a expandirse con nuevas sucursales, el sistema podría habilitarse de forma que permita centralizar la información y mantener el control de recursos humanos, desde diferentes ubicaciones.

La implementación será local y el acceso administrativo se realizará vía red interna y conexión a Internet del supermercado. La Figura 4 muestra el flujo de comunicación: usuario, proveedor de Internet, canal seguro HTTPS, servidor local, e integra los componentes software (Frontend React+TS, Backend Spring Boot y MySQL) para garantizar confidencialidad e integridad de los datos.

Figura 12
Flujo de comunicación



Fuente: Elaboración propia

Análisis detallado de herramientas técnicas

El desarrollo del sistema se sustenta en una pila tecnológica open source, lo que garantiza costo cero en licencias y alta compatibilidad.

Herramientas

Frontend

El sistema utiliza React junto con TypeScript para la construcción del entorno cliente. React es una biblioteca JavaScript desarrollada por Facebook que permite la creación de interfaces de usuario interactivas y reutilizables, mediante componentes modulares, facilitando el mantenimiento y la extensibilidad del sistema (Meta, 2023).

Por su parte, TypeScript aporta un tipado estático y herramientas de autocompletado, lo que reduce la cantidad de errores en tiempo de ejecución y mejora la legibilidad del código (Microsoft, 2023).

En cuanto al diseño visual, se emplea Tailwind CSS, un framework de utilidades que acelera la maquetación mediante clases predefinidas y fomenta una interfaz moderna y

consistente. Además, se incorpora Lucide Icons, un conjunto de íconos vectoriales open source que contribuyen a una experiencia de usuario más intuitiva (Lucide, 2024).

Backend

El servidor está desarrollado con Spring Boot, un framework basado en Java que permite la creación de servicios REST de forma rápida y estructurada. Su arquitectura modular, junto con la inyección de dependencias y la gestión automatizada de configuraciones, favorece la escalabilidad y la seguridad (Pivotal Software, 2023).

Para la capa de persistencia se utiliza Spring Data JPA, que implementa Hibernate como ORM (Object Relational Mapping). Esto posibilita mapear entidades de base de datos a clases Java, simplificando la creación de consultas personalizadas y reduciendo el código repetitivo (Red Hat, 2023).

Base de datos

El sistema de gestión de base de datos MySQL fue seleccionado por su robustez, portabilidad y compatibilidad con entornos empresariales. Su naturaleza open source lo convierte en una opción eficiente y económica para manejar la información estructurada de los módulos del sistema (Oracle, 2023).

Pruebas y desarrollo

Para las pruebas y validación de los servicios API REST se utiliza Postman, una herramienta que facilita el envío de solicitudes HTTP y la documentación de endpoints, garantizando una comunicación efectiva entre el frontend y el backend (Postman, 2024).

Como entornos de desarrollo (IDE) se emplean Visual Studio Code y IntelliJ IDEA, los cuales ofrecen integración con control de versiones, depuración, y extensiones personalizadas que mejoran la productividad del desarrollador (JetBrains, 2024; Microsoft, 2023).

Finalmente, se utiliza GitHub para el control de versiones y la gestión colaborativa del código fuente, lo que permite mantener un historial claro de cambios y facilitar la integración continua (GitHub, 2023).

Estas tecnologías garantizan que el sistema sea seguro, escalable y de mantenimiento sostenible, además de adaptarse a las necesidades reales de pequeñas y medianas empresas que buscan automatizar procesos de gestión del talento humano, mediante un software confiable, modular y accesible. Estas tecnologías garantizan que el sistema sea seguro, escalable y de mantenimiento sostenible, además de adaptarse a las necesidades reales de pequeñas y medianas empresas.

Conocimiento técnico del recurso humano

Para la correcta implementación y uso del sistema, el personal debe poseer un conjunto mínimo de competencias técnicas:

Rol y conocimientos requeridos

Administrador del sistema

Manejo básico de computadoras y navegadores, gestión de usuarios, exportación de reportes, conocimiento elemental de seguridad informática y respaldo de datos.

Encargado de Recursos Humanos

Dominio de los procesos laborales (vacaciones, incapacidades, nómina, liquidaciones). Capacidad de interpretar reportes y usar formularios digitales.

Soporte técnico

Conocimiento intermedio de instalación de software, conexión a bases de datos MySQL, mantenimiento preventivo y copias de seguridad.

El sistema está diseñado para ser intuitivo y accesible, por lo que no requiere personal especializado en informática. Sin embargo, una capacitación básica inicial es recomendable para garantizar un uso adecuado de las funcionalidades.

Casos de uso

Los casos de uso permiten representar, de manera sencilla, cómo las personas o actores interactúan con el sistema para realizar distintas tareas. Gracias a ellos se puede entender mejor el funcionamiento general y detectar posibles errores humanos o fallos en los procesos antes de que ocurran. Además, ayudan a que tanto los desarrolladores, como los usuarios, comprendan, claramente, qué se espera del sistema en cada situación. En esta sección se presentan los principales casos de uso del proyecto, mostrando los actores involucrados, sus objetivos y las acciones que realizan dentro del sistema.

Prototipo: Sistema Web de Gestión de RRHH - Supermercado Hnos. Pineda, ubicada en Hojancha, Guanacaste	
Número Caso de Uso: CU-PE-01	Nombre del Caso de Uso: Crear Persona
Fecha elaboración:	10/09/2025
Descripción Caso de Uso:	Permite registrar una nueva persona en el sistema incluyendo sus datos personales básicos.
Autor caso de uso:	Justin Cruz Castro
Actores relacionados:	Encargado de RRHH
Precondiciones:	<ul style="list-style-type: none"> • El usuario debe haber iniciado sesión. • El usuario debe de tener el rol. • Los parámetros de configuración deben de estar debidamente precargados, tales como: provincia, cantón y distrito.
Flujo Básico del caso de uso	
<ul style="list-style-type: none"> • El usuario accede al módulo Personas. • El usuario Selecciona la opción 'Agregar'. • El usuario Completa nombre, apellidos, cédula, fecha de nacimiento y sexo. • El usuario Selecciona provincia, cantón y distrito. • El usuario Agrega detalles de la dirección. • El usuario presiona el botón de Confirmar y guarda. • El sistema inicia el proceso de validación de los campos obligatorios y con el formato correcto. • Sistema valida que no exista la persona. • El sistema guarda el registro y confirma la creación. 	
Detallar el paso a paso del Flujo Básico	
Sub Flujos	

Corresponde a las diferentes opciones (alternativas funcionales) que un actor tiene al iniciar con el Flujo Básico.	
Sub Flujo registrar dirección	<ul style="list-style-type: none"> • Registrar dirección principal al crear persona. • Registrar teléfono principal al crear persona. • Registrar correo principal al crear persona.
Flujos Alternos	
Corresponde a lo que debe realizar el sistema ante posibles errores	
Flujo Alternativo No. 1 Campos obligatorios con formato	El sistema lanza una alerta de que se deben completar los campos obligatorios.
Flujo Alternativo No. 2 Campos obligatorios son vacíos	<ul style="list-style-type: none"> • El sistema lanza una alerta de que se deben completar los campos obligatorios. • El usuario corrige los datos. • El sistema regresa al flujo básico FB paso 6.
Flujo Alternativo No. 3 El sistema detecta que existe una persona	<ul style="list-style-type: none"> • El sistema lanza una alerta de que ya existe este registro previamente. • El usuario corrige los datos. • El sistema regresa al flujo básico FB paso 6.
Requerimientos especiales	
<ul style="list-style-type: none"> • Validación de cédula y fecha. • Registro en auditoría de creación. 	
Post-Condiciones	
<ul style="list-style-type: none"> • Persona creada con identificador único. • Registro disponible para consulta y demás acciones CRUD. 	

Prototipo: Sistema Web de Gestión de RRHH - Supermercado Hnos. Pineda, ubicada en Hojancha, Guanacaste	
Número Caso de Uso: CU-PE-02	Nombre del Caso de Uso: Editar Persona
Fecha elaboración:	25/09/2025
Descripción Caso de Uso:	Permite modificar datos personales de una persona.
Autor caso de uso:	Justin Cruz Castro
Actores relacionados:	Encargado de RRHH
Precondiciones:	<ul style="list-style-type: none"> • El usuario debe haber iniciado sesión. • El usuario debe de tener el rol. • Persona existente. • Permisos de edición.

Flujo Básico del caso de uso	
<ul style="list-style-type: none"> • 1. Buscar persona. • 2. Seleccionar 'Editar'. • 3. Actualizar datos. • 4. Guardar cambios. • 5. Validaciones de formato y obligatorios. • 6. Verificar cédula no duplicada si cambió. • 7. Confirmación de actualización. 	
Detallar el paso a paso del Flujo Básico	
Sub Flujos	
Corresponde a las diferentes opciones (alternativas funcionales) que un actor tiene al iniciar con el Flujo Básico.	
SF-01 Edición de contactos	El usuario agrega/edita/elimina teléfono o correo; el sistema valida formato.
Flujos Alternos	
Corresponde a lo que debe realizar el sistema ante posibles errores	
Flujo Alternativo No. 1 Campos obligatorios con formato	El sistema lanza una alerta de que se deben completar los campos obligatorios.
Flujo Alternativo No. 2 Campos obligatorios son vacíos	<ul style="list-style-type: none"> • El sistema lanza una alerta de que se deben completar los campos obligatorios. • El usuario corrige los datos. • El sistema regresa al flujo básico FB paso 6.
Flujo Alternativo No. 3 El sistema detecta que existe una persona	<ul style="list-style-type: none"> • El sistema lanza una alerta de que ya existe este registro previamente. • El usuario corrige los datos. • El sistema regresa al flujo básico FB paso 6.
Requerimientos especiales	
<ul style="list-style-type: none"> • Historial de cambios. • Auditoría de edición. 	
Post-Condiciones	
<ul style="list-style-type: none"> • Datos actualizados • Bitácora actualizada. 	

Prototipo: Sistema Web de Gestión de RRHH - Supermercado Hnos. Pineda, ubicada en Hojancha, Guanacaste	
Número Caso de Uso: CU-PE-03	Nombre del Caso de Uso: Eliminar Persona
Fecha elaboración:	25/09/2025
Descripción Caso de Uso:	Inactivar (borrado lógico) una persona sin dependencias bloqueantes.
Autor caso de uso:	Justin Cruz Castro
Actores relacionados:	Encargado de RRHH
Precondiciones:	<ul style="list-style-type: none"> • El usuario debe haber iniciado sesión. • El usuario debe de tener el rol. • Persona existente. • Sin empleado activo asociado. • Permisos de inactivación.
Flujo Básico del caso de uso	
<ul style="list-style-type: none"> • Localizar persona y elegir 'Eliminar. • Sistema verifica dependencias. • Solicitar confirmación. • Guardar cambios. • Confirmación de actualización. 	
Detallar el paso a paso del Flujo Básico	
Sub Flujos	
Corresponde a las diferentes opciones (alternativas funcionales) que un actor tiene al iniciar con el Flujo Básico.	
Sub Flujo registrar dirección	<ul style="list-style-type: none"> • Registrar dirección principal al crear persona.
Flujos Alternos	
Corresponde a lo que debe realizar el sistema ante posibles errores	
Flujo Alternativo No. 1: Empleado activo	<ul style="list-style-type: none"> • Deniega e indica inactivarlo/liquidarlo primero.
Flujo Alternativo No. 2: Restricción legal/contable	<ul style="list-style-type: none"> • Impide inactivar y muestra motivo.
Requerimientos especiales	
<ul style="list-style-type: none"> • Borrado lógico. • Auditoría de baja. 	
Post-Condiciones	
<ul style="list-style-type: none"> • Persona inactiva; no aparece en altas y listados por defecto. 	

Prototipo: Sistema Web de Gestión de RRHH - Supermercado Hnos. Pineda, ubicada en Hojancha, Guanacaste	
Número Caso de Uso: CU-PE-04	Nombre del Caso de Uso: Consultar Persona
Fecha elaboración:	25/09/2025
Descripción Caso de Uso:	Buscar y visualizar detalle de personas con filtros por cédula, nombre y estado.
Autor caso de uso:	Justin Cruz Castro
Actores relacionados:	Encargado de RRHH
Precondiciones:	<ul style="list-style-type: none"> • El usuario debe haber iniciado sesión. • El usuario debe de tener el rol.

Flujo Básico del caso de uso	
<ol style="list-style-type: none"> 1. Abrir Personas. 2. Definir filtros. 3. Ejecutar búsqueda. 4. Ver resultados paginados. 5. Abrir detalle para información completa. 	
Detallar el paso a paso del Flujo Básico	
Sub Flujos	
Corresponde a las diferentes opciones (alternativas funcionales) que un actor tiene al iniciar con el Flujo Básico.	
Sub Flujo listado	<ul style="list-style-type: none"> • Exportar listado
Flujos Alternos	
Corresponde a lo que debe realizar el sistema ante posibles errores	
Flujo Alternativo No. 1: Sin resultados	<ul style="list-style-type: none"> • Se informa y se sugieren ajustes de filtros.
Requerimientos especiales	
<ul style="list-style-type: none"> • Paginación/ordenamiento. • Exportación opcional. 	
Post-Condiciones	
<ul style="list-style-type: none"> • Consulta completada. 	

Prototipo: Sistema Web de Gestión de RRHH - Supermercado Hnos. Pineda, ubicada en Hojanca, Guanacaste	
Número caso de uso: CU-NO-01	Nombre del caso de uso: Generar nómina
Fecha de elaboración:	25/09/2025
Descripción del caso de uso:	Este caso de uso permite al administrador calcular la nómina salarial de todos los empleados activos para un periodo específico.
Autor del caso de uso:	Justin Cruz Castro
Actores relacionados:	Administrador.
Precondiciones:	<ul style="list-style-type: none"> • El administrador de nómina debe estar autenticado en el sistema. • Deben existir datos de empleados activos en el sistema. • No se debe haber calculado la nómina para el periodo solicitado.
Flujo básico del caso de uso	
<ul style="list-style-type: none"> • El administrador de nómina ingresa al apartado “Nómina” desde el menú lateral. • El sistema solicita que se indique el periodo correspondiente al cálculo. • El administrador proporciona el periodo requerido y confirma la selección. • El sistema comprueba que el periodo sea válido y que no exista un cálculo previo para ese rango. • Se filtran los empleados activos y se descartan aquellos que ya cuentan con liquidación registrada. • Para cada empleado activo, se determina el salario bruto, se aplican las deducciones y se obtiene el salario neto. 	

<ul style="list-style-type: none"> El sistema guarda la información generada en la base de datos y muestra al administrador una notificación de proceso completado. 	
Subflujos	
Subflujo manejo de empleados liquidados	El sistema consulta los registros de liquidación y excluye a estos empleados del cálculo.
Flujos alternos	
Flujo alternativo No. 1: Nómina ya calculada	Si ya existe una nómina calculada para el periodo el sistema muestra un mensaje de error e indica que la nómina ya fue calculada.
Requerimientos especiales	
El sistema debe cumplir con las normativas laborales vigentes en el país, respecto a las deducciones y cálculos de nómina, que garantiza que los cálculos de deducciones sean precisos.	
Post-Condiciones	
La nómina mensual ha sido generada y almacenada en el sistema, lo que permite futuras consultas y auditorías.	

Prototipo: Sistema Web de Gestión de RRHH - Supermercado Hnos. Pineda, ubicada en Hojancha, Guanacaste.	
Número caso de uso: CU-NO-02	Nombre del caso de uso: Consultar Nómina.
Fecha de elaboración:	25/09/2025
Descripción del caso de uso:	Este caso de uso permite al administrador visualizar y consultar la nómina salarial de todos los empleados activos para un periodo específico.
Autor del caso de uso:	Justin Cruz Castro
Actores relacionados:	Administrador.
Precondiciones:	<ul style="list-style-type: none"> El administrador de nómina debe estar autenticado en el sistema. Debe existir, al menos, una nómina calculada en el sistema para poder consultar.
Flujo básico del caso de uso	
<ul style="list-style-type: none"> El administrador de nómina accede desde el menú lateral a la opción “Nómina”. El administrador elige la funcionalidad “Consultar Historial de Nómina”. El sistema despliega el listado de periodos de nómina disponibles para revisión. El administrador selecciona el periodo específico que desea visualizar. El sistema carga y presenta la información del cálculo de la nómina, incluyendo montos de salario bruto, deducciones aplicadas y salario neto por empleado. El administrador tiene la posibilidad de exportar la información a un archivo en formatos como PDF o Excel. 	
Sub-Flujos	
Subflujo Consultar Nómina de un Empleado Específico	El administrador ingresa el nombre o ID de un empleado. El sistema muestra la información de la nómina para ese empleado en el periodo seleccionado.
Flujos alternos	

Flujo Alternativo No. 1: No se encuentran nóminas	Si no hay nóminas calculadas para el periodo seleccionado el sistema muestra un mensaje en el que indica que no se han encontrado registros.
Requerimientos especiales	
El sistema debe garantizar la confidencialidad de la información salarial y asegurar que solo los administradores autorizados tengan acceso a los datos.	
Post-Condiciones	
El administrador ha consultado el historial de nómina mensual y, si es necesario, ha exportado la información a un archivo externo.	

Prototipo: Sistema Web de Gestión de RRHH - Supermercado Hnos. Pineda, ubicada en Hojancha, Guanacaste.	
Número Caso de Uso: CU-HE-01	Nombre del caso de uso: Solicitar horas extra.
Fecha de elaboración:	25/09/2025
Descripción del caso de uso:	Este caso de uso permite a los empleados solicitar horas extra para su respectiva aprobación, por parte de su jefatura inmediata.
Autor caso de uso:	Justin Cruz Castro
Actores relacionados:	Empleado; Jefatura inmediata; RRHH.
Precondiciones:	<ul style="list-style-type: none"> • El empleado debe estar autenticado en el sistema. • Debe haber trabajado horas adicionales que desea registrar como horas extras.
Flujo básico del caso de uso	
<ul style="list-style-type: none"> • El empleado selecciona en el menú izquierdo la opción: "Horas Extra". • El sistema muestra la pantalla de solicitud de horas extra. • El usuario ingresa la fecha de solicitud y las horas trabajadas. • El usuario envía la solicitud de horas extra. • El sistema registra la solicitud con estado "Pendiente" y notifica. 	
Sub-Flujos	
Sub-Flujo Validar Rango de Fechas	El sistema verifica que la fecha ingresada esté dentro del rango permitido para la solicitud de horas extra.

Sub-Flujo Verificar Solicitudes Duplicadas	El sistema revisa si ya existe una solicitud para la misma fecha antes de proceder.
Flujos alternos	
Flujo alternativo No. 1: Fecha fuera de rango	Si la fecha de solicitud no está dentro del rango permitido el sistema muestra un mensaje de error y solicita una nueva fecha.
Flujo alternativo No. 2: Solicitud duplicada	Si ya existe una solicitud de horas extra para esa fecha, el sistema muestra un mensaje de error que indica que ya hay una solicitud registrada.
Requerimientos especiales	
El sistema debe permitir la validación de solicitudes de horas extra, según la normativa laboral vigente.	
Post-Condiciones	
La solicitud de horas extra ha sido registrada y está pendiente de aprobación por la jefatura.	

Prototipo: Sistema Web de Gestión de RRHH - Supermercado Hnos. Pineda, ubicada en Hojancha, Guanacaste.	
Número caso de uso: CU-HE-02	Nombre del caso de uso: Crear horas extra
Fecha de elaboración:	25/09/2025
Descripción caso de uso:	Este caso de uso permite a la jefatura inmediata revisar y aprobar o rechazar las solicitudes de horas extra presentadas por los empleados.
Autor caso de uso:	Justin Cruz Castro
Actores relacionados:	Jefatura inmediata; Empleado; RRHH.
Precondiciones:	La jefatura debe estar autenticada en el sistema. Debe existir, al menos, una solicitud de horas extra pendientes de aprobación.
Flujo básico del caso de uso	
<ul style="list-style-type: none"> • La jefatura accede a la opción “Horas extra” desde el menú lateral. • El sistema presenta el listado de solicitudes de horas extra que aún están pendientes. • La jefatura elige una solicitud específica para su revisión. • El sistema despliega la información detallada de la solicitud, incluyendo fecha, empleado y cantidad de horas solicitadas. • La jefatura decide aprobar o rechazar la solicitud. • El sistema registra el nuevo estado y envía una notificación al empleado con la resolución. 	
Sub-Flujos	

Corresponde a las diferentes opciones (alternativas funcionales) que un actor tiene al iniciar el flujo básico.	
Sub flujo aprobar solicitud	La jefatura selecciona la opción aprobar. El sistema registra el estado como "Aprobado".
Sub-Flujo Rechazar Solicitud	La jefatura selecciona la opción rechazar. El sistema solicita un motivo para el rechazo y registra el estado como "Rechazado".
Flujos alternos	
Flujo alternativo No. 1: No hay solicitudes pendientes	Si no hay solicitudes pendientes, el sistema muestra un mensaje en el que indica que no hay solicitudes.
Requerimientos especiales	
Las notificaciones al empleado deben enviarse, de manera automática, tras la actualización del estado de su solicitud.	
Post-Condiciones	
La solicitud de horas extra ha sido actualizada con el nuevo estado y el empleado ha sido notificado.	

Prototipo: Sistema Web de Gestión de RRHH - Supermercado Hnos. Pineda, ubicada en Hojancha, Guanacaste.	
Número caso de uso: CU-VA-01	Nombre del caso de uso: Solicitar vacaciones
Fecha de elaboración:	25/09/2025
Descripción del caso de uso:	Este caso de uso permite al colaborador solicitar sus vacaciones por medio del sistema, con indicación de las fechas solicitadas.
Autor, caso de uso:	Justin Cruz Castro
Actores relacionados:	Empleado; Jefatura inmediata; RRHH.
Precondiciones:	<ul style="list-style-type: none"> • El empleado debe estar autenticado en el sistema. • El empleado debe tener saldo suficiente de días de vacaciones disponibles.
Flujo básico del caso de uso	

<ul style="list-style-type: none"> • El empleado ingresa a la opción “Vacaciones” desde el menú lateral. • El sistema despliega la pantalla “Solicitar Vacaciones”, permitiendo ingresar el rango de fechas deseado. • El empleado elige la fecha de inicio y finalización del periodo de vacaciones. • El empleado procede a confirmar la solicitud. • El sistema verifica que el colaborador cuente con días disponibles suficientes. • El sistema remite la solicitud a la jefatura correspondiente para su revisión y aprobación. • El sistema informa al empleado que la solicitud fue enviada y se encuentra en estado pendiente de aprobación. 	
Sub-Flujos	
Sub-Flujo Modificar fechas antes de enviar	El empleado desea cambiar las fechas antes de enviar la solicitud. El empleado modifica las fechas en la pantalla antes de confirmar la solicitud.
Flujos alternos	
Flujo alternativo No. 1: Saldo insuficiente de días	Si el colaborador no tiene suficientes días de vacaciones disponibles, el sistema muestra un mensaje de error e indica que no se puede aprobar la solicitud.
Requerimientos especiales	
El sistema debe calcular, automáticamente, los días disponibles y no permitir solicitudes que excedan el saldo de días.	
Post-Condiciones	
La solicitud de vacaciones ha sido registrada y se encuentra en espera de aprobación, por parte de la jefatura inmediata.	

Prototipo: Sistema Web de Gestión de RRHH - Supermercado Hnos. Pineda, ubicada en Hojancha, Guanacaste.	
Número Caso de Uso: CU-AG-01	Nombre del Caso de uso: Calcular Aguinaldo
Fecha de elaboración:	25/09/2025
Descripción Caso de Uso:	Este caso de uso permite al administrador del sistema realizar el cálculo del aguinaldo anual para todos los empleados del supermercado.
Autor del caso de uso:	Justin Cruz Castro

Actores relacionados:	Administrador.
Precondiciones:	<ul style="list-style-type: none"> • El administrador debe estar autenticado en el sistema. • Deben existir registros salariales de los empleados, correspondientes al año en curso. • Las políticas de cálculo del aguinaldo deben estar correctamente configuradas en el sistema.
Flujo básico del caso de uso	
<ul style="list-style-type: none"> • El administrador ingresa al módulo “Aguinaldo” desde el menú lateral del sistema. • En la sección “Calcular aguinaldo” se presenta el listado de los colaboradores activos. • El administrador selecciona la opción para iniciar el cálculo del aguinaldo. • El sistema procesa el cálculo considerando los salarios percibidos por cada empleado durante el año. • El sistema muestra un resumen con los montos generados y solicita la validación final. • El administrador confirma el resultado y procede a guardar la información. • El sistema envía una notificación a cada empleado con el monto correspondiente a su aguinaldo. 	
Sub-Flujos	
Sub-Flujo Visualización de administrador selecciona un colaborador,	El administrador selecciona un colaborador de la lista para ver los detalles del cálculo de su aguinaldo.
Flujos alternos	
Flujo alternativo No. 1: Error en los registros salariales	Si el sistema detecta que faltan datos salariales de algún empleado, muestra un mensaje de error y permite al administrador corregir los registros.
Requerimientos especiales	
El sistema debe realizar los cálculos de aguinaldo, de acuerdo con las normativas vigentes, que incluyen considerar todos los salarios devengados durante el año.	
Post-Condiciones	
El cálculo del aguinaldo ha sido realizado y registrado en el sistema.	

Prototipo: Sistema Web de Gestión de RRHH - Supermercado Hnos. Pineda, ubicada en Hojancha, Guanacaste.	
Número del Caso de Uso: CU-IN-01	Nombre del Caso de uso: Agregar la incapacidad.

Fecha de elaboración:	25/09/2025
Descripción del caso de uso:	Este caso de uso permite al administrador o al personal de Recursos Humanos registrar la información sobre las incapacidades de un colaborador, debido a una enfermedad o accidente.
Autor del caso de uso:	Justin Cruz Castro
Actores relacionados:	Administrador de Recursos Humanos.
Precondiciones:	<ul style="list-style-type: none"> • El administrador o el personal de Recursos Humanos debe estar autenticado en el sistema. • Debe existir un registro de colaboradores en el sistema.
Flujo básico del caso de uso	
<ul style="list-style-type: none"> • El administrador accede al módulo de "Incapacidades" desde el menú principal. • El sistema muestra un formulario para registrar una incapacidad. • El administrador completa el formulario con los datos correctos. • El administrador envía el formulario. • El sistema almacena la información de la incapacidad y genera un registro único. 	
Sub-Flujos	
Sub-Flujo Registro de datos adicionales	El administrador puede registrar información adicional como un documento adjunto (certificado médico) que respalde la incapacidad.
Flujos alternos	
Flujo alternativo No. 1: Datos incompletos	Si el administrador intenta enviar el formulario sin completar todos los campos obligatorios el sistema muestra un mensaje de error e indica los campos que faltan.
Requerimientos especiales	
El sistema debe permitir el registro de incapacidades, de acuerdo con las normativas laborales vigentes.	
Post-Condiciones	
La información de la incapacidad ha sido almacenada, correctamente, en el sistema.	

Prototipo: Sistema Web de Gestión de RRHH - Supermercado Hnos. Pineda, ubicada en Hojancha, Guanacaste.

Número de Caso de uso: CU-PM-01	Nombre del Caso de uso: Solicitar permiso
Fecha de elaboración:	25/09/2025
Descripción del caso de Uso:	Este caso de uso permite al colaborador solicitar un permiso laboral a su jefatura inmediata, por medio del sistema.
Autor caso de uso:	Justin Cruz Castro
Actores relacionados:	Jefatura inmediata; RRHH; Empleado.
Precondiciones:	<ul style="list-style-type: none"> • El colaborador debe estar autenticado en el sistema. • El colaborador debe tener permisos suficientes para solicitar permisos laborales.
Flujo básico del caso de uso	
<ul style="list-style-type: none"> • El colaborador selecciona la opción "Permisos" en el menú del sistema. • El sistema lleva al colaborador a la pantalla de "Solicitar Permiso." • El sistema solicita al colaborador que ingrese los detalles del permiso (tipo, fecha, duración, motivo). • El colaborador completa los detalles y envía la solicitud. • El sistema registra la solicitud y la envía a la jefatura inmediata para su aprobación. 	
Sub-Flujos	
Sub-Flujo Confirmar envío de solicitud	El sistema muestra un mensaje de confirmación de que la solicitud ha sido enviada correctamente.
Flujos alternos	
Flujo alternativo No. 1: Detalles del permiso incompletos	Si el colaborador no completa todos los campos requeridos el sistema muestra un mensaje de error para que complete la información antes de enviar la solicitud.
Requerimientos especiales	
El sistema debe cumplir con las normativas laborales vigentes en el país, respecto a la gestión de permisos laborales.	
Post-Condiciones	
La solicitud de permiso ha sido registrada en el sistema y está pendiente de aprobación.	

Prototipo: Sistema Web de Gestión de RRHH - Supermercado Hnos. Pineda, ubicada en Hojancha, Guanacaste.

Número de Caso de uso: CU-PM-02	Nombre del Caso de uso: Aprobar/Rechazar Permiso
Fecha elaboración:	25/09/2025
Descripción del Caso de uso:	Este caso de uso permite a la jefatura inmediata revisar y aprobar o rechazar las solicitudes de permisos presentadas por los colaboradores.
Autor caso de uso:	Justin Cruz Castro
Actores relacionados:	Jefatura Inmediata; RRHH; Empleado.
Precondiciones:	<ul style="list-style-type: none"> • La jefatura inmediata debe estar autenticada en el sistema. • Debe haber solicitudes de permiso pendientes de aprobación.
Flujo básico del caso de uso	
<ul style="list-style-type: none"> • La jefatura inmediata accede a la opción “Permisos” desde el menú del sistema. • El sistema despliega el listado de solicitudes de permiso que están pendientes de revisión. • La jefatura selecciona una de las solicitudes para evaluarla. • El sistema muestra la información completa de la solicitud: colaborador, tipo de permiso, fechas y el motivo indicado. • La jefatura decide si aprueba o rechaza la solicitud. • El sistema registra el nuevo estado y envía la notificación correspondiente al colaborador y al área de RRHH. 	
Sub-Flujos	
Sub-Flujo Confirmar la Decisión	El sistema solicita confirmación antes de registrar la decisión final.
Flujos alternos	
Flujo Alternativo No. 1: No hay solicitudes pendientes.	Si no hay solicitudes en la lista, el sistema informa que no existen permisos pendientes de aprobación.
Requerimientos especiales	
Las notificaciones automáticas deben enviarse a los colaboradores y a RRHH tras la decisión.	
Post-Condiciones	
La solicitud de permiso ha sido aprobada o rechazada y el estado actualizado en el sistema.	

Prototipo: Sistema Web de Gestión de RRHH - Supermercado Hnos. Pineda, ubicada en Hojanca, Guanacaste.

Número de caso de Uso: CU-AS-01	Nombre del caso de uso: Registrar asistencia
Fecha de elaboración:	25/09/2025
Descripción del Caso de uso:	Este caso de uso permite a cada colaborador registrar sus tiempos de entrada y salida.
Autor del caso de uso:	Justin Cruz Castro
Actores relacionados:	Colaborador.
Precondiciones:	<ul style="list-style-type: none"> El colaborador debe estar autenticado en el sistema.
Flujo básico del caso de uso	
<ul style="list-style-type: none"> El colaborador selecciona la opción "Asistencia" en el menú. El sistema muestra la pantalla para registrar asistencia. El colaborador ingresa la hora de entrada o salida o ambas. El colaborador confirma el registro de asistencia. El sistema almacena la información de asistencia y muestra un mensaje de éxito. 	
Sub-Flujos	
Sub-Flujo Ingreso de hora de entrada	El colaborador solo ingresa la hora de entrada y confirma.
Sub-Flujo Ingreso de hora de salida	El colaborador solo ingresa la hora de salida y confirma.
Flujos alternos	
Flujo Alternativo No. 1:	Si el colaborador intenta registrar una hora fuera del horario permitido el
Registro fuera de horario	sistema muestra un mensaje de error.
Requerimientos especiales	
El sistema debe permitir registrar tiempos de asistencia en formato de 24 horas.	
Post-Condiciones	
Los registros de asistencia han sido actualizados en el sistema.	

Prototipo: Sistema Web de Gestión de RRHH - Supermercado Hnos. Pineda, ubicada en Hojanca, Guanacaste.

Número Caso de Uso: CU-AS-02	Nombre del Caso de uso: Revisar asistencia
Fecha elaboración:	25/09/2025
Descripción Caso de Uso:	Este caso de uso permite al administrador revisar los registros de asistencia de todos los colaboradores de un período específico.
Autor caso de uso:	Justin Cruz Castro
Actores relacionados:	Administrador.
Precondiciones:	<ul style="list-style-type: none"> El administrador debe estar autenticado en el sistema.
Flujo básico del caso de uso	
<ul style="list-style-type: none"> El administrador selecciona la opción "Asistencia" en el menú. El sistema muestra la pantalla de revisión de asistencia. El administrador ingresa el período de revisión (fecha de inicio y fin). El administrador confirma la revisión. El sistema muestra la lista de asistencia de todos los colaboradores del período especificado. 	
Sub-Flujos	
El Sub-Flujo Selección de colaborador específico	El administrador puede seleccionar un colaborador específico para ver su asistencia.
Flujos alternos	
Flujo Alternativo No. 1: No se encuentra registros	Si no muestra un Hay registros de asistencia en el período especificado el sistema mensaje de error.
Requerimientos especiales	
El sistema debe permitir la exportación de registros de asistencia a un formato PDF.	
Post-Condiciones	
La información de asistencia ha sido revisada y presentada en pantalla.	

Prototipo: Sistema Web de Gestión de RRHH - Supermercado Hnos. Pineda, ubicada en Hojanca, Guanacaste.	
Número de Caso de uso: CU-LQ-01	Nombre del Caso de uso: Calcular la liquidación
Fecha de elaboración:	25/09/2025

Descripción del Caso de uso:	Este caso de uso le permite al administrador calcular la liquidación correspondiente a un colaborador específico, teniendo en cuenta las condiciones salariales y las leyes laborales vigentes.	
Autor caso de uso:	Justin Cruz Castro	
Actores relacionados:	Administrador de Recursos Humanos.	
Precondiciones:	<ul style="list-style-type: none"> • El administrador debe estar autenticado en el sistema. • Deben existir datos del colaborador en el sistema. • El colaborador debe estar en proceso de liquidación. 	
Flujo básico del caso de uso		
<ul style="list-style-type: none"> • El administrador ingresa a la opción “Liquidación” desde el menú principal. • El sistema despliega la interfaz destinada a la gestión de liquidaciones. • El administrador selecciona al colaborador para el cual desea generar la liquidación. • El sistema presenta la información salarial correspondiente a dicho colaborador. • El administrador presiona la opción “Calcular liquidación”. • El sistema procesa el cálculo y muestra el monto final, considerando deducciones y bonificaciones aplicables. • El administrador confirma la liquidación generada. 		
Sub-Flujos		
Sub-Flujo	Validación	de
datos	Si los datos del colaborador están incompletos el sistema muestra un mensaje de error y solicita la información faltante.	
Flujos alternos		
Flujo alternativo	No. 1:	Colaborador no Encontrado
		Si el colaborador no se encuentra en el sistema el sistema muestra un mensaje de error e indica que el colaborador no existe.
Requerimientos especiales		
El sistema debe cumplir con las leyes laborales vigentes en relación con los cálculos de liquidación.		
Post-Condiciones		
La liquidación del colaborador ha sido calculada y almacenada en el sistema.		

Prototipo: Sistema Web de Gestión de RRHH - Supermercado Hnos. Pineda, ubicada en Hojancha, Guanacaste.	
Número Caso de Uso: CU-ER-01	Nombre del Caso de uso: Registrar Evaluación de Rendimiento

Fecha elaboración:	25/09/2025
Descripción Caso de Uso:	Este caso de uso permite al administrador registrar las evaluaciones de rendimiento de los colaboradores para un periodo específico.
Autor caso de uso:	Justin Cruz Castro
Actores relacionados:	Administrador de Recursos Humanos; Colaborador.
Precondiciones:	<ul style="list-style-type: none"> • El administrador debe estar autenticado en el sistema. • Debe existir un colaborador asignado al proceso de evaluación.
Flujo básico del caso de uso	
<ul style="list-style-type: none"> • El administrador accede a la opción “Evaluaciones de desempeño” desde el menú. • El sistema despliega la pantalla correspondiente a las evaluaciones. • El administrador selecciona la función “Registrar evaluación”. • El sistema solicita que se ingrese la identificación del colaborador. • El administrador introduce la identificación del colaborador y confirma la acción. • El sistema muestra la plantilla de evaluación asociada a ese colaborador. • El administrador completa los campos de la evaluación. • El administrador procede a guardar la evaluación registrada. 	
Sub-Flujos	
Sub-Flujo Evaluación Incompleta	Si el administrador intenta guardar una evaluación incompleta, el sistema muestra un mensaje de error e indica los campos faltantes.
Flujos alternos	
Flujo alternativo No. 1:	Si el colaborador ingresado no existe en el sistema el sistema muestra un
Colaborador no encontrado	mensaje de error y solicita ingresar un ID válido.
Requerimientos especiales	
El sistema debe permitir la personalización de criterios de evaluación, según el puesto de trabajo. Las evaluaciones deben ser confidenciales y accesibles solo para el administrador y el colaborador evaluado.	
Post-Condiciones	
La evaluación de desempeño ha sido registrada y almacenada en el sistema.	

Prototipo: Sistema Web de Gestión de RRHH - Supermercado Hnos. Pineda, ubicada en Hojancha, Guanacaste.

Número de Caso de uso: CU-ER-02	Nombre del Caso de uso: Consultar resultados de las evaluaciones
Fecha elaboración:	25/09/2025
Descripción Caso de uso:	Este caso de uso permite a los colaboradores consultar los resultados de sus evaluaciones de desempeño por medio del sistema.
Autor caso de uso:	Justin Cruz Castro
Actores relacionados:	Empleado.
Precondiciones:	<ul style="list-style-type: none"> • El colaborador debe estar autenticado en el sistema. • Deben existir evaluaciones de desempeño registradas y aprobadas.
Flujo Básico del caso de uso	
<ul style="list-style-type: none"> • El colaborador inicia sesión en el sistema. • El colaborador selecciona la opción "Evaluaciones de desempeño" en el menú principal. • El sistema muestra una lista con las evaluaciones realizadas al colaborador, organizadas por fecha. • El colaborador selecciona la evaluación que desea consultar. • El sistema muestra el puntaje general, los comentarios del evaluador y cualquier recomendación o plan de acción sugerido. 	
Sub-Flujos	
Sub-Flujo Consultar Evaluaciones Pasadas	El colaborador puede revisar el historial de evaluaciones anteriores para comparar su progreso en diferentes periodos.
Flujos alternos	
Flujo alternativo No. 1: No hay evaluaciones disponibles	Si el colaborador no tiene evaluaciones registradas el sistema muestra un mensaje en el que indica que no hay resultados disponibles.
Requerimientos especiales	
El sistema debe proteger la confidencialidad de los datos de evaluación y limitar el acceso solo al colaborador evaluado y a su jefatura.	
Post-Condiciones	
El colaborador ha consultado los resultados de su evaluación de desempeño y puede descargar el archivo si lo desea.	

Prototipo: Sistema Web de Gestión de RRHH - Supermercado Hnos. Pineda, ubicada en Hojancha, Guanacaste.	
Número de Caso de Uso: CU-US-01	Nombre del Caso de uso: Autenticación de usuario
Fecha de elaboración:	25/09/2025
Descripción del Caso de uso:	Este caso de uso permite al usuario autenticarse en el sistema, mediante la verificación de sus credenciales.
Autor de caso de uso:	Justin Cruz Castro
Actores relacionados:	Usuario del sistema.
Precondiciones:	<ul style="list-style-type: none"> • El usuario debe tener una cuenta registrada en el sistema. • El sistema debe estar disponible y operativo.
Flujo básico del Caso de uso	
<ul style="list-style-type: none"> • El usuario accede a la pantalla de inicio de sesión. • El usuario ingresa su nombre de usuario y contraseña. • El sistema verifica las credenciales ingresadas. • Si las credenciales son válidas, el sistema concede acceso al usuario y lo redirige al menú principal. • Si las credenciales son inválidas el sistema muestra un mensaje de error. 	
Sub-Flujos	
Sub-Flujo Recuperar Contraseña	El usuario selecciona la opción "Recuperar Contraseña" y sigue el proceso para restablecerla.
Sub-Flujo Registro de Usuario	El usuario puede registrarse en el sistema si no tiene cuenta.
Flujos alternos	
Flujo Alternativo No. 1: Credenciales inválidas	Si el usuario ingresa credenciales incorrectas el sistema muestra un mensaje de error y permite reingresar la información.
Requerimientos especiales	
El sistema debe utilizar técnicas de encriptación para almacenar y verificar contraseñas.	
Post-Condiciones	
El usuario ha sido autenticado y tiene acceso al sistema.	

Prototipo: Sistema Web de Gestión de RRHH - Supermercado Hnos. Pineda, ubicada en Hojanca, Guanacaste.	
Número de Caso de uso: CU-EM-01	Nombre del Caso de uso: Registrar nuevo empleado
Fecha de elaboración:	25/09/2025
Descripción del Caso de uso:	Este caso de uso permite registrar un nuevo empleado en el sistema.
Autor caso de uso:	Justin Cruz Castro
Actores relacionados:	Administrador del sistema.
Precondiciones:	<ul style="list-style-type: none"> • El usuario debe estar autenticado en el sistema. • Debe existir al menos una Persona registrada sin empleado activo asociado. • Deben existir puestos y departamentos vigentes. • El usuario debe tener permisos para crear empleados.
Flujo básico del caso de uso	
<ul style="list-style-type: none"> • El usuario navega al módulo “Empleados” y selecciona la opción “Registrar”. • El sistema despliega un formulario con los campos requeridos: Persona, Puesto, Departamento, salario base, fecha de ingreso, tipo de contrato, etc. • El usuario selecciona la Persona a asociar (buscador / selector). • El sistema valida que dicha Persona no tenga un empleado activo registrado. • El usuario completa los campos restantes y confirma la creación. • El sistema valida los datos ingresados: <ul style="list-style-type: none"> • Mayoría de edad (\geq 18 años). • Puesto y Departamento activos. • Salario base válido. • Fechas coherentes. • El sistema crea el registro del empleado y lo marca como activo. • El sistema registra la acción en el Log de Auditoría (usuario, entidad, id, acción CREATE, fecha/hora, detalles). • El sistema muestra mensaje de confirmación y redirige al detalle del empleado. 	
Sub-Flujos	
Sub-Flujo	Validación de
datos	Si algún campo es inválido, el sistema muestra un mensaje de error y solicita la corrección.
Flujos alternos	

Flujo alternativo No. 1: Campos obligatorios no llenos	Si el administrador no llena todos los campos obligatorios el sistema muestra un mensaje de error e indica cuáles campos deben completarse.
Requerimientos especiales	
El sistema debe garantizar que cada empleado esté vinculado a una única Persona.	
Post-Condiciones	
<ul style="list-style-type: none"> • El empleado queda registrado como activo en el sistema. • La Persona queda asociada a su contrato laboral. 	

Prototipo: Sistema Web de Gestión de RRHH - Supermercado Hnos. Pineda, ubicada en Hojancha, Guanacaste.	
Número Caso de Uso: CU-EM-02	Nombre del Caso de uso: Consultar información de empleados
Fecha de elaboración:	25/09/2025
Descripción de Caso de uso:	Este caso de uso permite al usuario consultar información detallada de los empleados del sistema, como su nombre, puesto, departamento y estado (activo/inactivo).
Autor caso de uso:	Justin Cruz Castro
Actores relacionados:	Administrador del sistema; Usuario autorizado.
Precondiciones:	<ul style="list-style-type: none"> • El usuario debe estar autenticado en el sistema. • Deben existir empleados registrados en el sistema.
Flujo básico del caso de uso	
1. El usuario accede al módulo de "Consultas" en el menú principal.	
<ul style="list-style-type: none"> • Selecciona la opción de "Consultar Información de Empleados". • El sistema muestra una pantalla con los filtros opcionales (nombre, departamento, estado). • El usuario aplica los filtros deseados o consulta sin filtros. • El sistema muestra la lista de empleados que cumplen con los criterios seleccionados. • El usuario puede seleccionar un empleado de la lista para ver más detalles. • El sistema muestra información detallada del empleado seleccionado. 	
Sub-Flujos	
Sub-Flujo opcionales Aplicar filtros	El usuario selecciona los filtros que desea aplicar (por ejemplo, nombre)
Flujos alternos	

Flujo Alternativo No. 1: No hay empleados registrados	Si no existen empleados registrados el sistema muestra un mensaje para indicar que no hay empleados para mostrar.
Requerimientos especiales	
El sistema debe respetar los permisos de acceso y evitar que usuarios no autorizados consulten información confidencial.	
Post-Condiciones	
El usuario ha visualizado la información de los empleados consultados.	

Prototipo: Sistema Web de Gestión de RRHH - Supermercado Hnos. Pineda, ubicada en Hojancha, Guanacaste.	
Número de Caso de uso: CU-EM-03	Nombre del Caso de uso: Eliminar Empleado (borrado lógico)
Fecha de elaboración:	25/09/2025
Descripción del Caso de uso:	Este caso de uso permite inhabilitar (borrado lógico) a un empleado, marcándolo como inactivo
Autor caso de uso:	Justin Cruz Castro
Actores relacionados:	Administrador del sistema.
Precondiciones:	<ul style="list-style-type: none"> • El usuario navega al módulo “Empleados” y selecciona “Eliminar (Lógico)” sobre un empleado. • El sistema muestra un diálogo de confirmación, solicitando motivo de baja y fecha de baja. • El sistema marca el empleado como inactivo (borrado lógico). • El sistema registra la acción en el Log de Auditoría (usuario, entidad, id, acción DELETE_LOGICAL, detalle con motivo). • El sistema muestra mensaje de confirmación y actualiza la lista (filtro por estado).
Flujo básico del caso de uso	
<ul style="list-style-type: none"> • El administrador selecciona la opción "Mantenimientos" en el menú. • El sistema muestra la pantalla de registro de mantenimiento. • El administrador ingresa los detalles del mantenimiento. • El administrador hace clic en "Guardar". 	

<ul style="list-style-type: none"> El sistema confirma que el mantenimiento ha sido registrado exitosamente. 	
Sub-Flujos	
Sub-Flujo	Validación de
datos	Si algún campo es inválido el sistema muestra un mensaje de error y solicita la corrección.
Flujos alternos	
Flujo alternativo No. 1:	Campos obligatorios no llenos
	Si el administrador no llena todos los campos obligatorios el sistema muestra un mensaje de error e indica cuáles campos deben completarse.
Requerimientos especiales	
El sistema debe cumplir con las normativas de gestión de datos vigentes.	
Post-Condiciones	
Se ha registrado un nuevo mantenimiento en el sistema.	

Prototipo: Sistema Web de Gestión de RRHH - Supermercado Hnos. Pineda, ubicada en Hojanca, Guanacaste.	
Número Caso de Uso: CU-EM-04	Nombre del Caso de uso: Editar información de empleados.
Fecha de elaboración:	25/09/2025
Descripción de Caso de uso:	Este caso de uso permite a un usuario autorizado actualizar la información de un empleado.
Autor caso de uso:	Justin Cruz Castro
Actores relacionados:	Administrador del sistema; Usuario autorizado.
Precondiciones:	<ul style="list-style-type: none"> El usuario debe estar autenticado en el sistema. Deben existir empleados registrados en el sistema.
Flujo básico del caso de uso	

<ol style="list-style-type: none"> 1. El usuario navega al módulo “Empleados” y selecciona “Editar”. 2. El sistema muestra un listado de empleados. 3. El usuario selecciona un empleado para edición. <ol style="list-style-type: none"> a. El sistema despliega el formulario con los datos actuales del empleado. 4. El usuario modifica los campos necesarios (puesto, departamento, salario base). 5. El usuario confirma la operación con “Guardar cambios”. 6. El sistema persiste los cambios y registra la acción en el Log de Auditoría. 7. El sistema muestra mensaje de confirmación y retorna a la vista de detalle del empleado. 	
Sub-Flujos	
Sub-Flujo opcionales	Aplicar filtros
	El usuario selecciona los filtros que desea aplicar (por ejemplo, nombre)
Flujos alternos	
Flujo Alternativo No. 1: No hay empleados registrados	Si no existen empleados registrados el sistema muestra un mensaje para indicar que no hay empleados para mostrar.
Requerimientos especiales	
El sistema debe respetar los permisos de acceso y evitar que usuarios no autorizados consulten información confidencial.	
Post-Condiciones	
El usuario ha visualizado la información de los empleados consultados.	

Diseño

Arquitectura del sistema

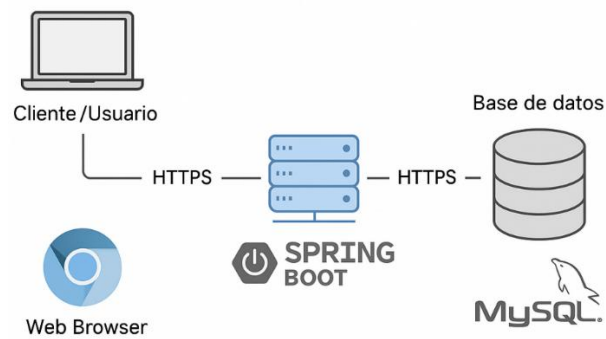
El sistema web de gestión de recursos humanos del Supermercado Hnos. Pineda adopta una arquitectura de tres capas, implementada bajo el modelo cliente-servidor.

El usuario interactúa mediante una interfaz web construida con React + TypeScript, que se comunica a través de solicitudes HTTP RESTful con el backend desarrollado en Spring Boot (Java). Este último procesa la lógica del negocio, ejecuta validaciones, gestiona la autenticación y realiza consultas hacia el motor de base de datos MySQL, donde se almacena toda la información de los empleados, planillas, asistencias, vacaciones y evaluaciones.

Figura 13

Diagrama de la arquitectura del sistema

Arquitectura general del sistema

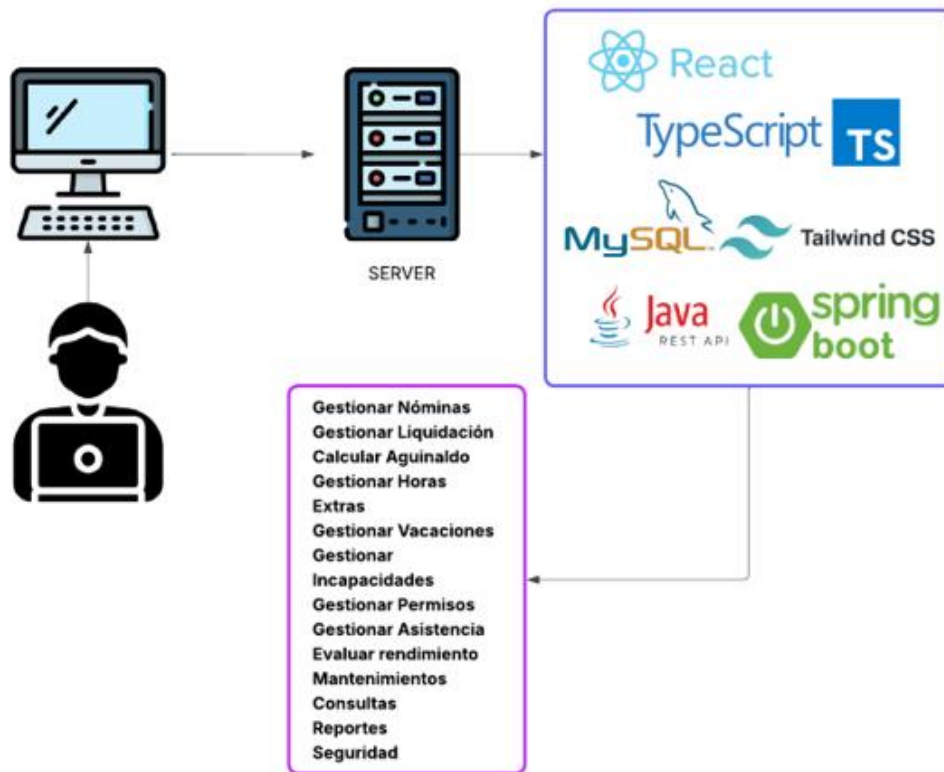


Fuente: Elaboración propia

Arquitectura del software

La arquitectura del software sigue el patrón MVC (Modelo-Vista-Controlador). Esto separa la lógica de negocio de la presentación, facilitando el mantenimiento, escalabilidad y futuras mejoras del sistema. Cada módulo es autónomo, con controladores REST dedicados, DTOs para transferencia de datos y validaciones, a nivel de servicio.

Figura 14
Diagrama de la arquitectura del software

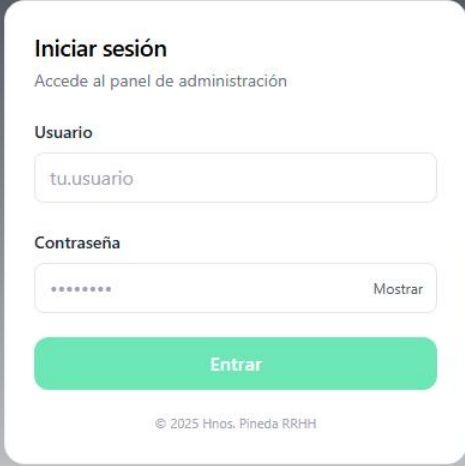


Fuente: Elaboración propia

Diseño de entradas

En esta sección se incluirán pantallas representativas del prototipo.

Figura 15
Pantalla de inicio de sesión



The image shows a login form titled "Iniciar sesión" (Log in) with the subtitle "Accede al panel de administración" (Access the administration panel). It features two input fields: "Usuario" (User) containing the text "tu.usuario" and "Contraseña" (Password) containing seven asterisks. A "Mostrar" (Show) link is positioned to the right of the password field. Below the fields is a prominent green "Entrar" (Log in) button. At the bottom of the form, there is a small copyright notice: "© 2025 Hnos. Pineda RRHH".

Fuente: Elaboración propia

Figura 16
Pantalla de agregar incapacidades

Crear Incapacidad ✕
Completá los datos y guardá para crear el registro.

Datos Usuario #2

Fecha de Inicio Fecha de Fin
dd/mm/aaaa dd/mm/aaaa

Días de Incapacidad (calculado)
0

Tipo de Incapacidad
Ej.: Enfermedad, Accidente...

Tipo de Permiso
Seleccione tipo...

Empleado
Seleccione un empleado...

Crear incapacidad

Fuente: Elaboración propia

Figura 17
Pantalla de Agregar Persona

Crear nueva Persona ✕

+ Dirección

Otro Activo

Provincia / Cantón / Distrito

Dirección #1

Tipo
Otro

Detalles
Ej: 100m sur del parque central

Principal

Provincia Cantón Distrito

Eliminar esta dirección

Datos de la Persona

Nombre

Apellido 1

Apellido 2

Cédula
101110111
Formato: 9 dígitos sin guiones (ej. 101110111)

Fecha de Nacimiento
dd/mm/aaaa

Sexo
Seleccione...

Limpiar **Crear persona**

Teléfonos + Teléfono

Otro Activo

Número
88888888

Tipo
Otro

Principal

Eliminar teléfono

Correos + Correo

Trabajo Activo

Correo
ejemplo@dominio.º

Tipo
Trabajo

Principal

Eliminar correo

Fuente: Elaboración propia

Figura 18
Pantalla de Agregar Empleado

Crear nuevo Empleado

Registra a una persona en la planilla activa.

FECHA DE INGRESO
08/05/2024

DEPARTAMENTO
Operaciones de Tienda

SALARIO BASE (€)
12236

PUESTO
Cajero/a

Activo

PERSONA
Sandra Valverde Castillo

Guardar Empleado

Fuente: Elaboración propia

Figura 19
Pantalla de Inicio

Supermercado Hnos Pineda
Panel — adminRRHH

INICIO

- Panel principal
- Reportes

OPERACIÓN DIARIA

- Asistencia
- Horas extra
- Permisos
- Incapacidades
- Vacaciones

COMPENSACIÓN Y LEGALES

- Nómina
- Aguinaldo
- Liquidaciones
- Evaluación de desempeño

CATÁLOGOS

- Empleados
- Personas
- Puestos
- Departamentos

Cerrar sesión

SUPERMERCADO HNOS PINEDA

EMPLEADOS 5 5% vs. mes pasado

EMPLEADOS ACTIVOS 5

Últimos salarios

Empleado (ID)	Periodo inicio	Periodo fin	Monto bruto	Monto neto	Estado
Sin registros					

Actividades

- adminRRHH creó Departamento #3
Creado departamento: nombre=Bodega y Abasto, activo=true
30/10/2025, 2:29:52 p. m.
- adminRRHH creó Departamento #2
Creado departamento: nombre=Operaciones de Tienda, activo=true
30/10/2025, 2:29:41 p. m.
- adminRRHH creó CorreoElectronico #4
Creado correo: olivcastro12@gmail.com (personald-5) principal=true
30/10/2025, 2:21:08 p. m.

Fuente: Elaboración propia

Diccionario

El presente diccionario de datos consiste en una descripción detallada de la estructura de las tablas que componen el prototipo funcional para la gestión del recurso humano en el Supermercado Hnos. Pineda. Este sistema integra un total de 26 entidades, cada una de las cuales ha sido concebida para almacenar y administrar, de manera eficiente, la información crucial relacionada con los empleados, la nómina, los permisos, las evaluaciones de desempeño y otros módulos fundamentales.

De seguido se presenta un análisis exhaustivo de la estructura de cada una de estas entidades, con detalle de los nombres de los campos, sus correspondientes tipos de datos, las restricciones impuestas y una descripción concisa de la función que desempeña cada campo dentro del sistema:

Tabla 5
Entidad aguinaldo

Campo	Tipo de dato	Nu lo	P K	FK (destino)	Valo r por defe cto	Extra
id_aguinaldo	INT	NO	P K			AUTO_INCRE MENT
activo	BIT(1)	NO				
anio	INT	NO				
base_calculo	DECIMAL	SÍ				
estado	VARCHAR (255)	NO				
fecha_pago	DATE	SÍ			NUL L	
monto	DECIMAL	SÍ				
observacion	VARCHAR (255)	SÍ			NUL L	
periodo_fin	DATE	NO				
periodo_inicio	DATE	NO				

empleado_id_empleado	INT	NO		empleado(id_empleado)		
----------------------	-----	----	--	-----------------------	--	--

Fuente: Elaboración propia.

Relaciones

Salientes (esta tabla → otras)

empleado_id_empleado → empleado(id_empleado) | 1:N]

Tabla 6
Entidad asistencia

Campo	Tipo de dato	Nu lo	P K	FK (destino)	Valo r por defe cto	Extra
id_asistencia	INT	NO	P K			AUTO_INCREMENT
empleado_id_empleado	INT	NO		empleado(id_empleado)		
fecha	DATE	NO				
hora_entrada	TIME	SÍ			NUL L	
hora_salida	TIME	SÍ			NUL L	
observacion	VARCHAR (500)	SÍ			NUL L	

Fuente: Elaboración propia.

Relaciones

Salientes (esta tabla → otras)

empleado_id_empleado → empleado(id_empleado) | 1:N]

Tabla 7
Entidad canton

Campo	Tipo de dato	Nu lo	P K	FK (destino)	Valor por defec to	Extra
id_canton	INT	NO	P K			AUTO_INCRE MENT
nombre_cant on	VARCHAR(100)	NO				
numero_cant on	INT	NO				
provincia_nu mero	INT	NO		provincia(número_pr ovincia)		
Activo	BIT(1)	NO				

Fuente: Elaboración propia.

Relaciones

Salientes (esta tabla → otras)

provincia_numero → provincia(numero_provincia) [fk_canton_provincia_numero | 1:N]

Entrantes (otras → esta)

distrito . canton_numero → esta (numero_canton) [fk_distrito_canton_numero | 1:N]

Tabla 8
Entidad concepto_rebajo

Campo	Tipo de dato	Nul o	P K	FK (destino)	Valor por defect o	Extra
id_concept o	INT	NO	P K			AUTO_INCREMEN T
activo	BIT(1)	NO				

nombre	VARCHAR(255)	NO				
tipo	VARCHAR(255)	NO				
valor	DECIMAL	SÍ				

Fuente: Elaboración propia.

Relaciones

Entrantes (otras → esta)

nomina_detalle_rebajo . concepto_id_concepto → esta (id_concepto) | 1:N]

Tabla 9
Entidad correo_electronico

Campo	Tipo de dato	Nul o	P K	FK (destino)	Valor por defec to	Extra
id_correo	INT	NO	P K			AUTO_INCREMENT
activo	BIT(1)	NO				
correo_electronico	VARCHAR(255)	NO				
es_principal	BIT(1)	NO				
tipo	VARCHAR(20)	NO				
persona_id_persona	INT	NO		persona(id_persona)		

Fuente: Elaboración propia.

Relaciones

Salientes (esta tabla → otras)

persona_id_persona → persona(id_persona) [fk_correo_electronico_persona | 1:N]

Tabla 10
Entidad criterio

Campo	Tipo de dato	Nul o	P K	FK (destino)	Valor por defect o	Extra
id_criterio	INT	NO	P K			AUTO_INCREMENT
activo	BIT(1)	NO				
descripcion	VARCHAR(255)	NO				
nombre_criterio	VARCHAR(100)	NO				

Fuente: Elaboración propia.

Relaciones

Entrantes (otras → esta)

evaluacionrendimiento . criterio_id_criterio → esta (id_criterio) | 1:N]

Tabla 11
Entidad Departamento

Campo	Tipo de dato	Nul o	P K	FK (destino)	Valor por defect o	Extra
id_departamento	INT	NO	P K			AUTO_INCREMENT
activo	BIT(1)	NO				
nombre_departamento	VARCHAR(255)	NO				

Fuente: Elaboración propia.

Relaciones

Entrantes (otras → esta)

puesto . departamento_id_departamento → esta (id_departamento) | 1:N]

empleado . departamento_id_departamento → esta (id_departamento) | 1:N]

Tabla 12
Entidad dirección

Campo	Tipo de dato	Nul o	P K	FK (destino)	Valor por defec to	Extra
id_direccion	INT	NO	P K			AUTO_INCREMENT
detalles_direccion	VARCHAR(255)	NO				
distrito_id_distrito	INT	NO		distrito(id_distrito)		
persona_id_persona	INT	NO		persona(id_persona)		
activo	BIT(1)	NO				
es_principal	BIT(1)	NO				
tipo	VARCHAR(50)	NO				

Fuente: Elaboración propia.

Relaciones

Salientes (esta tabla → otras)

distrito_id_distrito → distrito(id_distrito) [fk_direccion_distrito | 1:N] persona_id_persona → persona(id_persona) [fk_direccion_persona | 1:N]

Tabla 13
Entidad distrito

Campo	Tipo de dato	Nul o	P K	FK (destino)	Valor por defec to	Extra
id_distrito	INT	NO	P K			AUTO_INCREMENT
nombre_distrito	VARCHAR(100)	NO				
numero_distrito	INT	NO				
canton_numero	INT	NO		canton(numero_canton)		
activo	BIT(1)	NO				

Fuente: Elaboración propia.

Relaciones

Salientes (esta tabla → otras)

canton_numero → canton(numero_canton) [fk_distrito_canton_numero | 1:N]

Entrantes (otras → esta)

direccion . distrito_id_distrito → esta (id_distrito) [fk_direccion_distrito | 1:N]

Tabla 14
Entidad empleado

Campo	Tipo de dato	Nu lo	P K	FK (destino)	Valo r por defe cto	Extra
id_empleado	INT	N O	P K			AUTO_INCREMENT
estado	BIT(1)	N O				

fecha_ingreso	DATE	NO				
salario_base	INT	NO				
departamento_id_departamento	INT	NO		departamento(id_departamento)		
persona_id_persona	INT	NO		persona(id_persona)		
puesto_id_puesto	INT	NO		puesto(id_puesto)		

Fuente: Elaboración propia.

Relaciones

Salientes (esta tabla → otras)

departamento_id_departamento → departamento(id_departamento) [1:N]

puesto_id_puesto → puesto(id_puesto) [1:N]

persona_id_persona → persona(id_persona) [1:N]

Entrantes (otras → esta)

aguinaldo . empleado_id_empleado → esta (id_empleado) | 1:N]

asistencia . empleado_id_empleado → esta (id_empleado) | 1:N]

evaluacionrendimiento . empleado_id_empleado → esta (id_empleado) | 1:N]

usuario . empleado_id_empleado → esta (id_empleado) | 1:N]

hora_extras . empleado_id_empleado → esta (id_empleado) | 1:N]

horario_jornada . empleado_id_empleado → esta (id_empleado) | 1:N]

incapacidad . empleado_id_empleado → esta (id_empleado) | 1:N]

liquidacion . empleado_id_empleado → esta (id_empleado) | 1:N]

nomina . empleado_id_empleado → esta (id_empleado) | 1:N]

permisos . empleado_id_empleado → esta (id_empleado) | 1:N]

vacaciones . empleado_id_empleado → esta (id_empleado) | 1:N]

Tabla 15

Entidad evaluacionrendimiento

Campo	Tipo de dato	Nu lo	P K	FK (destino)	Valo r por	Extra
-------	--------------	----------	--------	--------------	---------------	-------

					defecto	
id_evaluacion	INT	NO	PK			AUTO_INCREMENT
criterio_id_criterio	INT	NO		criterio(id_criterio)		
empleado_id_empleado	INT	NO		empleado(id_empleado)		
fecha_evaluacion	DATE	NO				
resultado_evaluacion	VARCHAR(255)	NO				

Fuente: Elaboración propia.

Relaciones

Salientes (esta tabla → otras)

criterio_id_criterio → criterio(id_criterio) | 1:N]

empleado_id_empleado → empleado(id_empleado) | 1:N]

Tabla 16

Entidad hora_extras

Campo	Tipo de dato	Nulo	PK	FK (destino)	Valor por defecto	Extra
id_horas_extras	INT	NO	PK			AUTO_INCREMENT
estado	BIT(1)	NO				
fecha_fin	DATETIME(6)	NO				
fecha_inicio	DATETIME(6)	NO				

fecha_solicitud	DATE	NO				
observacion	VARCHAR (255)	SÍ			NUL L	
total_horas	DECIMAL	SÍ				
empleado_id_empleado	INT	NO		empleado(id_empleado)		
usuario_id_usuario	INT	NO		usuario(id_usuario)		

Fuente: Elaboración propia.

Relaciones

Salientes (esta tabla → otras)

usuario_id_usuario → usuario(id_usuario) | 1:N]

empleado_id_empleado → empleado(id_empleado) | 1:N]

Tabla 17
Entidad horario_jornada

Campo	Tipo de dato	Nu lo	P K	FK (destino)	Valo r por defe cto	Extra
id_horario	INT	NO	P K			AUTO_INCRE MENT
hora_entrada	TIME	NO				
hora_salida	TIME	NO				
tipo_jornada	VARCHAR (255)	NO				
empleado_id_empleado	INT	NO		empleado(id_empleado)		
activo	BIT(1)	NO				

Fuente: Elaboración propia.

Relaciones

Salientes (esta tabla → otras)

empleado_id_empleado → empleado(id_empleado) | 1:N]

Tabla 18
Entidad incapacidad

Campo	Tipo de dato	Nu lo	P K	FK (destino)	Valo r por defe cto	Extra
id_incapacidad	INT	N O	P K			AUTO_INCREMENT
activo	BIT(1)	N O				
dias_incapacidad	INT	N O				
estado	VARCHAR(255)	N O				
fecha_fin	DATE	N O				
fecha_inicio	DATE	N O				
tipo_incapacidad	VARCHAR(255)	N O				
empleado_id_empleado	INT	N O		empleado(id_empleado)		
usuario_id_usuario	INT	N O		usuario(id_usuario)		
tipo_permiso_salarial_incapacidad	VARCHAR(255)	N O				

Fuente: Elaboración propia.

Relaciones

Salientes (esta tabla → otras)

usuario_id_usuario → usuario(id_usuario) | 1:N]

empleado_id_empleado → empleado(id_empleado) | 1:N]

Tabla 19
Entidad liquidación

Campo	Tipo de dato	Nu lo	P K	FK (destino)	Valo r por defe cto	Extra
id_liquidacion	INT	NO	P K			AUTO_INCREMENT
activo	BIT(1)	NO				
estado	VARCHAR (255)	NO				
fecha_liquidacion	DATE	NO				
monto_liquidacion	DECIMAL	SÍ				
observacion	VARCHAR (255)	SÍ			NUL L	
periodo_fin	DATE	NO				
periodo_inicio	DATE	NO				
tipo_despido	VARCHAR (255)	NO				
empleado_id_empleado	INT	NO		empleado(id_empleado)		

Fuente: Elaboración propia.

Relaciones

Salientes (esta tabla → otras)

empleado_id_empleado → empleado(id_empleado) | 1:N]

Tabla 20
Entidad log_auditoria

Campo	Tipo de dato	Nulo	PK	FK (destino)	Valor por defecto	Extra
id_log	BIGINT	NO	PK			AUTO_INCREMENT
accion	VARCHAR(255)	NO				
detalle	TEXT	SÍ			NULL	
entidad	VARCHAR(255)	NO				
entidad_id	VARCHAR(255)	NO				
fecha_hora	DATETIME(6)	NO				
usuario_id	INT	SÍ			NULL	

Fuente: Elaboración propia.

Relaciones

No existen relaciones registradas para esta tabla.

Tabla 21
Entidad nomina

Campo	Tipo de dato	Nulo	PK	FK (destino)	Valor por defecto	Extra
id_nomina	INT	NO	PK			AUTO_INCREMENT

estado	VARCHAR (20)	NO				
fecha_cierre	DATE	SÍ			NUL L	
fecha_creacion	DATE	NO				
observaciones	TEXT	SÍ			NUL L	
periodo_fin	DATE	NO				
periodo_inicio	DATE	NO				
salario_bruto	DECIMAL	SÍ				
salario_netto	DECIMAL	SÍ				
empleado_id_empleado	INT	NO		empleado(id_empleado)		

Fuente: Elaboración propia.

Relaciones

Salientes (esta tabla → otras)

empleado_id_empleado → empleado(id_empleado) | 1:N]

Entrantes (otras → esta)

nomina_detalle_rebajo . nomina_id_nomina → esta (id_nomina) | 1:N]

Tabla 22

Entidad nomina_detalle_rebajo

Campo	Tipo de dato	Nu lo	P K	FK (destino)	Valor por defec to	Extra
id_detalle	INT	NO	P K			AUTO_INCRE MENT
activo	BIT(1)	NO				

base_calculo	DECIMAL	SÍ				
monto_rebajo	DECIMAL	SÍ				
concepto_id_concepto	INT	NO		concepto_rebajo(id_concepto)		
nomina_id_nomina	INT	NO		nomina(id_nomina)		

Fuente: Elaboración propia.

Relaciones

Salientes (esta tabla → otras)

concepto_id_concepto → concepto_rebajo(id_concepto) | 1:N]

nomina_id_nomina → nomina(id_nomina) | 1:N]

Tabla 23
Entidad permisos

Campo	Tipo de dato	Nu lo	P K	FK (destino)	Valo r por defe cto	Extra
id_permiso	INT	NO	P K			AUTO_INCREMENT
activo	BIT(1)	NO				
dias_permiso	INT	NO				
estado	VARCHAR (255)	NO				
fecha_fin	DATE	NO				
fecha_inicio	DATE	NO				
fecha_solicitud_permiso	DATE	NO				

motivo	VARCHAR (255)	SÍ			NUL L	
tipo_permiso_sal arial	VARCHAR (255)	NO				
empleado_id_em pleado	INT	NO		empleado(id_em pleado)		
usuario_id_usuari o	INT	NO		usuario(id_usuari o)		

Fuente: Elaboración propia.

Relaciones

Salientes (esta tabla → otras)

empleado_id_empleado → empleado(id_empleado) | 1:N]

usuario_id_usuario → usuario(id_usuario) | 1:N]

Tabla 24
Entidad persona

Campo	Tipo de dato	Nul o	P K	FK (destino)	Valor por defect o	Extra
id_persona	INT	NO	P K			AUTO_INCREMENT
apellido1	VARCHAR(25 5)	NO				
apellido2	VARCHAR(25 5)	NO				
cedula	VARCHAR(25 5)	NO				
fecha_nacimien to	DATE	NO				

nombre	VARCHAR(255)	NO				
sexo	VARCHAR(255)	NO				
activo	BIT(1)	NO				

Fuente: Elaboración propia.

Relaciones

Entrantes (otras → esta)

empleado . persona_id_persona → esta (id_persona) | 1:N]

correo_electronico . persona_id_persona → esta (id_persona)

[fk_correo_electronico_persona | 1:N]

direccion . persona_id_persona → esta (id_persona) [fk_direccion_persona | 1:N]

telefono . persona_id_persona → esta (id_persona) [fk_telefono_persona | 1:N]

Tabla 25

Entidad provincia

Campo	Tipo de dato	Nul o	P K	FK (destino)	Valor por defect o	Extra
id_provincia	INT	NO	P K			AUTO_INCREMENT
nombre_provincia	VARCHAR(100)	NO				
numero_provincia	INT	NO				
activo	BIT(1)	NO				

Fuente: Elaboración propia.

Relaciones

Entrantes (otras → esta)

canton . provincia_numero → esta (numero_provincia) [fk_canton_provincia_numero | 1:N]

Tabla 26
Entidad puesto

Campo	Tipo de dato	Nu lo	P K	FK (destino)	Valo r por defe cto	Extra
id_puesto	INT	N O	P K			AUTO_INCREMENT
activo	BIT(1)	N O				
nombre_puesto	VARCHA R(255)	N O				
departamento_id_d epartamento	INT	N O		departamento(id_de partamento)		

Fuente: Elaboración propia.

Relaciones

Salientes (esta tabla → otras)

departamento_id_departamento → departamento(id_departamento) | 1:N]

Entrantes (otras → esta)

empleado . puesto_id_puesto → esta (id_puesto) | 1:N]

Tabla 27
Entidad rol

Campo	Tipo de dato	Nul o	P K	FK (destino)	Valor por defect o	Extra

id_rol	INT	NO	P K			AUTO_INCREMENT
descripcion_rol	VARCHAR(255)	NO				
nombre_rol	VARCHAR(255)	NO				

Fuente: Elaboración propia.

Relaciones

Entrantes (otras → esta)

usuario . rol_id_rol → esta (id_rol) | 1:N]

Tabla 28
Entidad telefono

Campo	Tipo de dato	Nul o	P K	FK (destino)	Valor por defec to	Extra
id_telefono	INT	NO	P K			AUTO_INCREMENT
activo	BIT(1)	NO				
es_principal	BIT(1)	NO				
numero	INT	NO				
tipo	VARCHAR(50)	NO				
persona_id_persona	INT	NO		persona(id_persona)		

Fuente: Elaboración propia.

Relaciones

Salientes (esta tabla → otras)

persona_id_persona → persona(id_persona) [fk_telefono_persona | 1:N]

Tabla 29
Entidad usuario

Campo	Tipo de dato	Nu lo	P K	FK (destino)	Valo r por defe cto	Extra
id_usuario	INT	NO	P K			AUTO_INCRE MENT
activo	BIT(1)	NO				
contrasena	VARCHAR (255)	NO				
nombre_usuario	VARCHAR (50)	NO				
empleado_id_em pleado	INT	NO		empleado(id_em pleado)		
rol_id_rol	INT	NO		rol(id_rol)		

Fuente: Elaboración propia.

Relaciones

Salientes (esta tabla → otras)

rol_id_rol → rol(id_rol) | 1:N]

empleado_id_empleado → empleado(id_empleado) | 1:N]

Entrantes (otras → esta)

hora_extras . usuario_id_usuario → esta (id_usuario) | 1:N]

incapacidad . usuario_id_usuario → esta (id_usuario) | 1:N]

permisos . usuario_id_usuario → esta (id_usuario) | 1:N]

vacaciones . usuario_id_usuario → esta (id_usuario) | 1:N]

Tabla 30
Entidad vacaciones

Campo	Tipo de dato	Nu lo	P K	FK (destino)	Valo r por	Extra
-------	--------------	----------	--------	--------------	---------------	-------

					defecto	
id_vacaciones	INT	NO	PK			AUTO_INCREMENT
activo	BIT(1)	NO				
dias_tomados	INT	NO				
estado	VARCHAR(255)	NO				
fecha_fin_vacaciones	DATE	NO				
fecha_inicio_vacaciones	DATE	NO				
fecha_solicitud	DATE	NO				
empleado_id_empleado	INT	NO		empleado(id_empleado)		
usuario_id_usuario	INT	NO		usuario(id_usuario)		

Fuente: Elaboración propia.

Relaciones

Salientes (esta tabla → otras)

usuario_id_usuario → usuario(id_usuario) | 1:N]

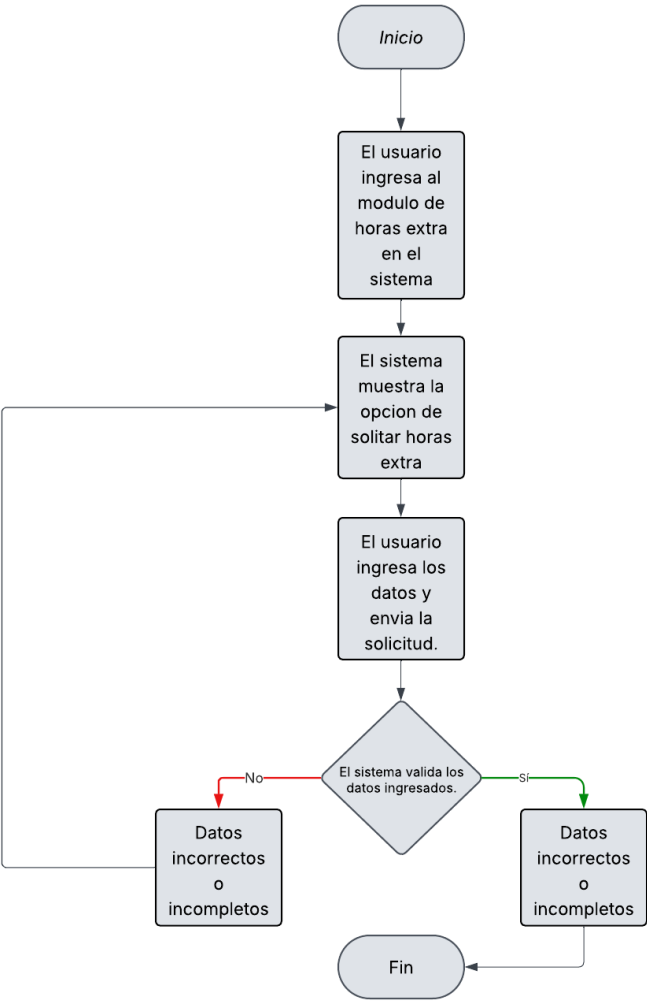
empleado_id_empleado → empleado(id_empleado) | 1:N]

Diseño de procesos

El diseño de procesos permite visualizar el funcionamiento general del sistema de gestión de recursos humanos propuesto para el Supermercado Hnos. Pineda. A través de los diagramas de flujo se busca representar de forma clara las etapas, decisiones y actores involucrados en los procedimientos más relevantes del área de Recursos Humanos. Estos diagramas constituyen una base fundamental para el desarrollo del sistema, ya que orientan

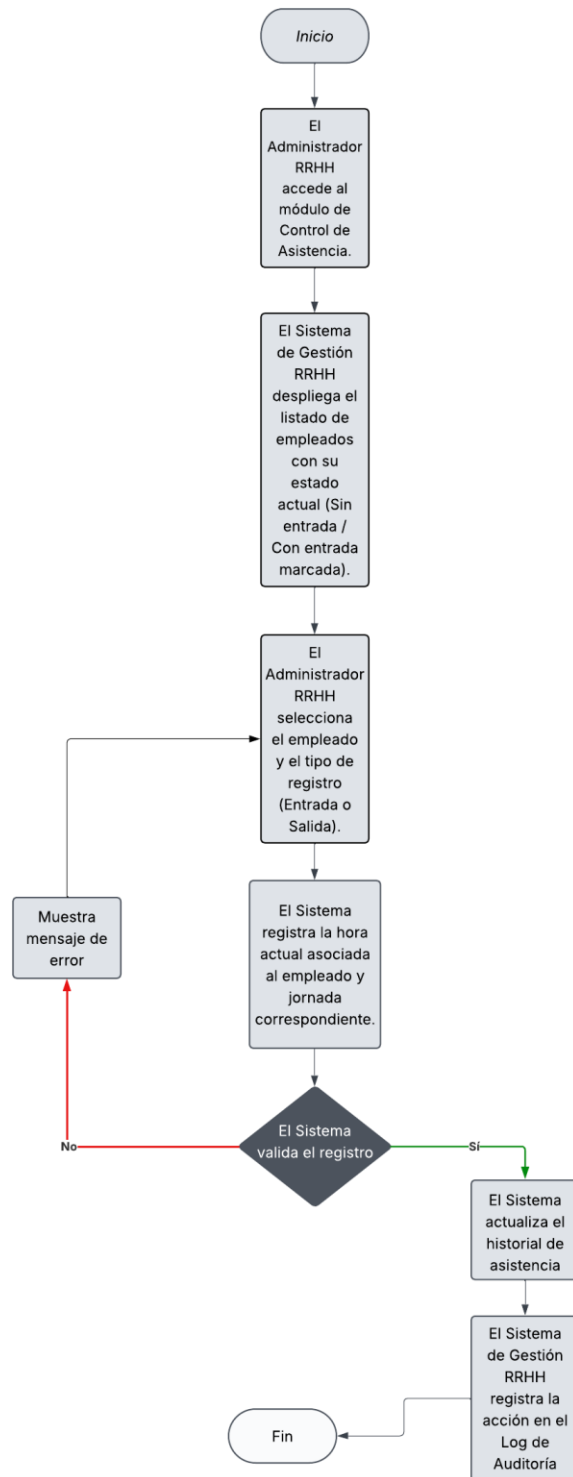
la codificación y garantizan que las funcionalidades respondan a los procesos reales de la organización.

Figura 21
Diagrama de flujo del módulo de horas extras.



Fuente: Elaboración propia

Figura 22
Diagrama de flujo del módulo asistencia



Fuente: Elaboración propia

Diseño de salidas

En esta sección se incluirán los diseños de salida de los datos.

Figura 23
Diseño de Salida de Asistencia

Historial de Asistencia
Hoy: 2025-10-30

Empleado: Todos Desde: 30/10/2025 Hasta: 30/10/2025 Estado: Todos Tipo de jornada (filtro adicional): Todas

Hoy Esta semana Este mes

Diurna 1 Nocturna 1 Mixta 1

Jornada Diurna 1 Horas (cerradas): 08:00 • En turno: 0

Buscar rápido... Exportar CSV

Empleado	Horario planificado	Registro real	Acciones								
Fecha	Nombre	Cédula	Puesto	Departamento	Jornada	Entrada-Salida	Entrada	Salida	Horas	Estado	
2025-10-30	Roberto	90	Encargado/a de Bodega	Bodega y Abasto	DIURNA	06:00 - 14:00	06:00:00	14:00:00	08:00	Cerrado	Editar Reiniciar

n

Fuente: Elaboración propia

Figura 24
Diseño de Salida de Personas

Personas
Gestión de personas, contactos y estado activo/inactivo

Activos: 5 Inactivos: 3 Totales: 5

Agregar Persona Actualizar Datos

ID	Nombre	Apellido 1	Apellido 2	Cédula	Fecha de Nacimien...	Sexo	Dirección	Correos	Teléfonos	Estado	Acciones
5	Olivia	Castro	Pichardo	43	2000-04-10	F	Ver Direcciones	Ver Correos	Ver Teléfonos	Activo	Editar Eliminar
4	Arturo	Chinchilla	Arguedas	64	1998-11-10	M	Ver Direcciones	Ver Correos	Ver Teléfonos	Activo	Editar Eliminar
1	Juan	Pérez	González	12	1990-05-12	M	Ver Direcciones	Ver Correos	Ver Teléfonos	Activo	Editar Eliminar
3	Roberto	Retana	González	90	1995-03-12	M	Ver Direcciones	Ver Correos	Ver Teléfonos	Activo	Editar Eliminar
2	Sandra	Valverde	Castillo	10	2000-05-12	F	Ver Direcciones	Ver Correos	Ver Teléfonos	Activo	Editar Eliminar

Fuente: Elaboración propia

Figura 25
Diseño de Salida de Empleados

Empleados
Información laboral, estado y asignación de puesto

Agregar Empleado

ID	Fecha ingreso	Salario	Estado	Departamento	Puesto	Persona	Identificación	Acciones
2	2024-08-21	€13 307	Activo	Recursos Humanos	Administrador de Recurs...	Juan	12	Editar Eliminar
4	2025-03-04	€12 237	Activo	Bodega y Abasto	Encargado/a de Bodega	Roberto	90	Editar Eliminar
5	2023-02-09	€12 237	Activo	Bodega y Abasto	Repartidor/a (Delivery)	Arturo	64	Editar Eliminar
6	2025-01-30	€13 307	Activo	Operaciones de Tienda	Cajero/a	Olivia	43	Editar Eliminar
7	2024-05-08	€13 307	Activo	Operaciones de Tienda	Cajero/a	Sandra	10	Editar Eliminar

Fuente: Elaboración propia

Figura 26
Diseño de Salida de Direcciones

Direcciones registradas										Agregar Dirección	
Provincias, cantones, distritos y dirección principal											
ID	Detalles de dirección	Persona Asignada	Provincia	Cantón	Distrito	Tipo	Principal	Activo	Acciones		
1	Hojancha, Centro	Sandra Valverde	Guanacaste	Hojancha	Hojancha	Residencia	Si	Activo	Editar	Eliminar	
2	Bagaces, Centro	Justin TCR Cruz	Guanacaste	Bagaces	Bagaces	Residencia	Si	Inactivo	Editar	Eliminar	
4	Pacayas, Centro	Justin TCR Cruz	Cartago	Alvarado	Pacayas	Trabajo	Si	Inactivo	Editar	Eliminar	

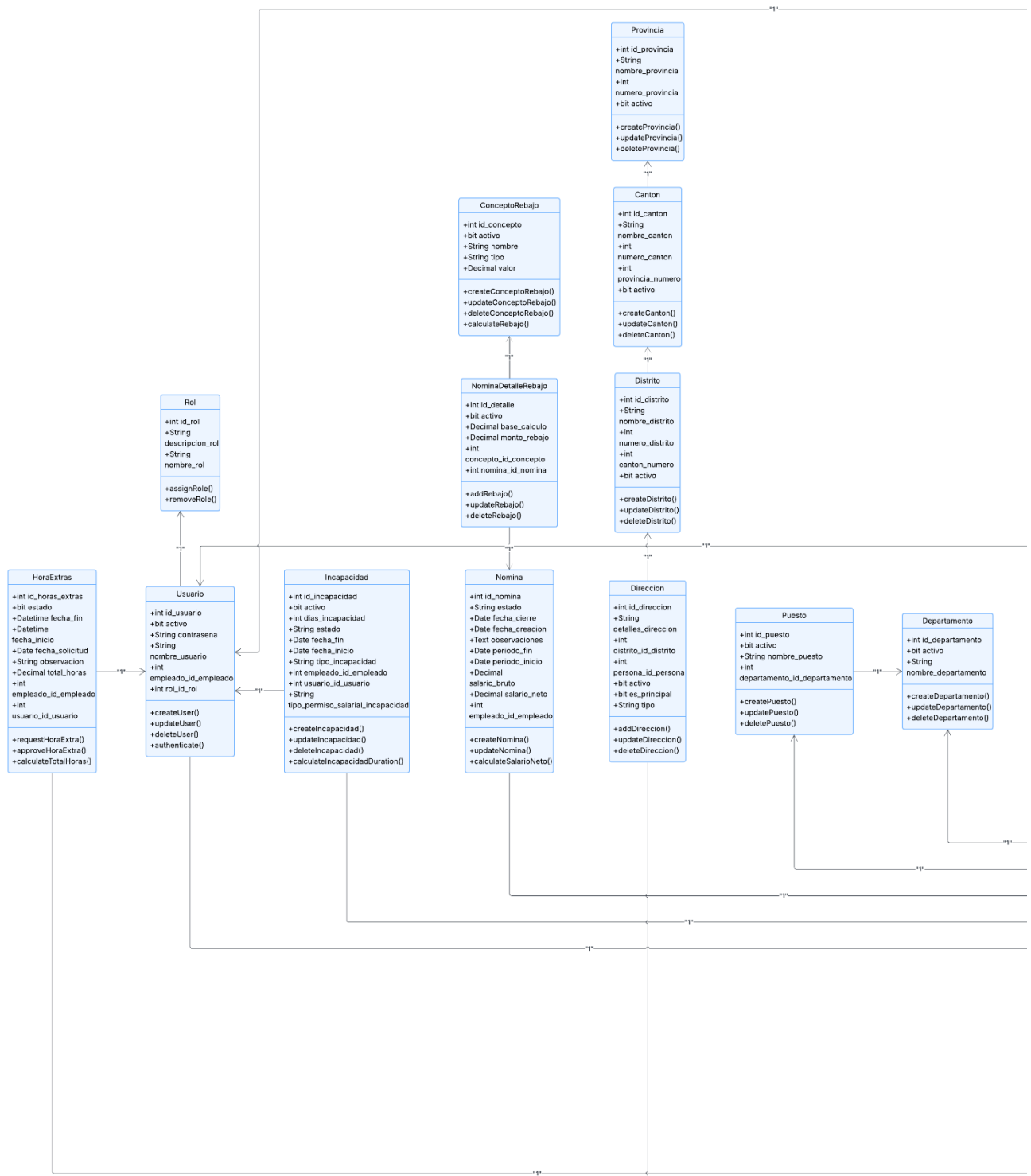
Fuente: Elaboración propia

Figura 27
Diseño de Salida de Teléfonos

Teléfonos							Agregar Teléfono	
Gestión de teléfonos asociados a personas								
ID	Número	Tipo	Principal	Activo	Persona	Acciones		
1	6332	Móvil	Si	Activo	Sandra Valverde Castillo	Editar	Eliminar	
2	4545	Móvil	Si	Activo	Roberto Retana González	Editar	Eliminar	
3	2512	Casa	Si	Activo	Arturo Chinchilla Arguedas	Editar	Eliminar	
4	8523	Móvil	Si	Activo	Olivia Castro Pichardo	Editar	Eliminar	

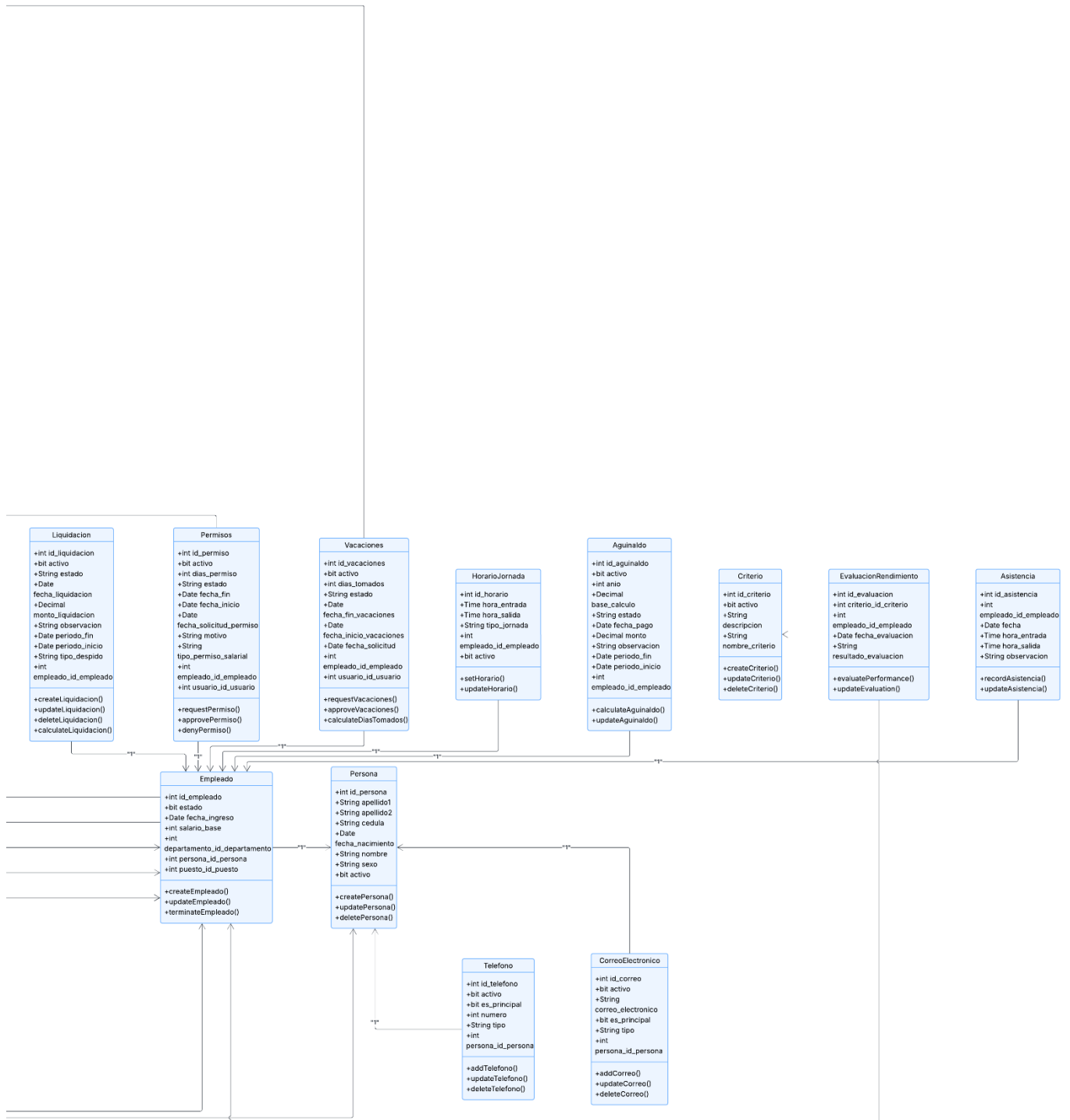
Fuente: Elaboración propia

Figura 29
Diagrama de clase parte izquierda



Fuente: Elaboración propia

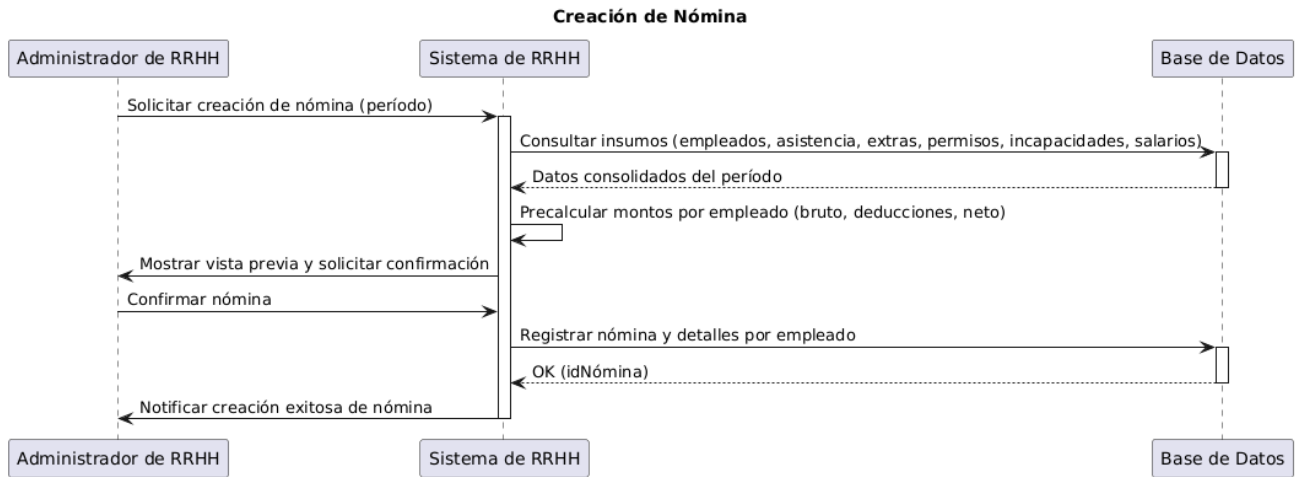
Figura 30
Diagrama de clases parte derecha



Fuente: Elaboración propia

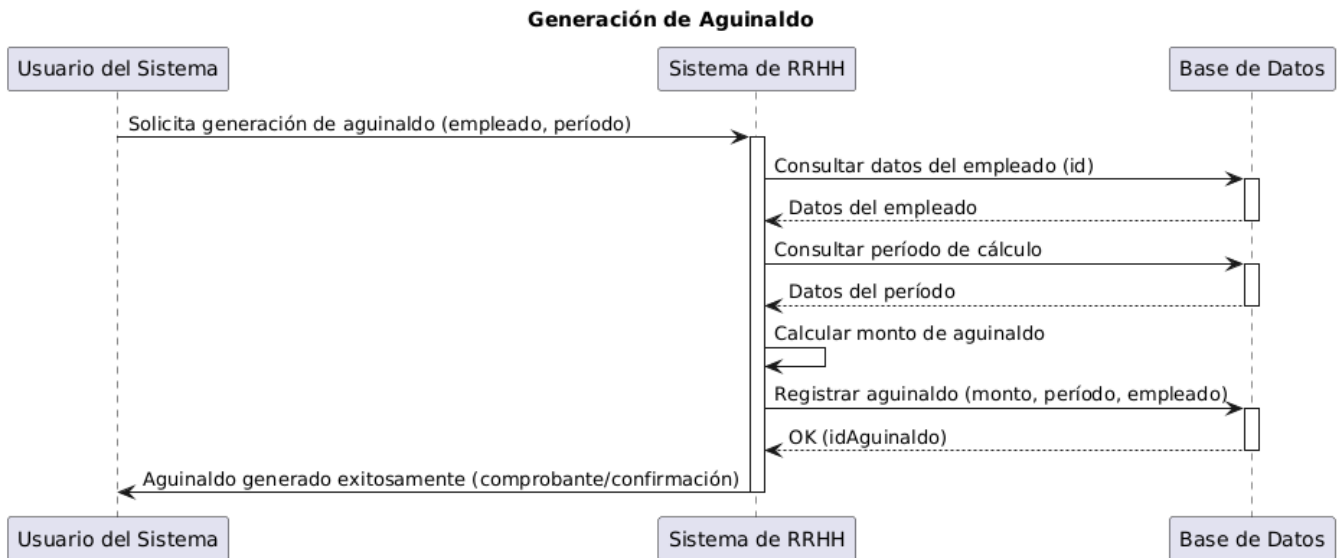
Diagrama de secuencia

Figura 31
Diagrama de secuencia de Nómina



Fuente: Elaboración propia

Figura 32
Diagrama de secuencia de Aguinaldo



Fuente: Elaboración propia

Programación

En esta sección se presentan fragmentos representativos del código fuente desarrollado para el sistema web de Gestión de Recursos Humanos. El objetivo no es mostrar la totalidad del software, sino ilustrar partes clave de la lógica implementada, buenas prácticas empleadas, estructura del proyecto, así como ejemplos de controladores, servicios, repositorios y componentes del frontend.

Cada fragmento de código expuesto forma parte del desarrollo real del sistema y se selecciona con la finalidad de evidenciar aspectos técnicos relevantes

Figura 33
Código Entradas

```
const handleSubmit = (e: FormEvent) => {
  e.preventDefault();
  setFormTouched(true);

  // Validar antes de construir
  const pe = validatePersona(persona);
  const de = validateDirecciones(direcciones);
  const te = validateTelefonos(telefonos);
  const ce = validateCorreos(correos);
  setPErrors(pe); setDirErrors(de); setTelErrors(te); setCorErrors(ce);
  if (Object.keys(pe).length || de.ites || de.principal || te.some(Boolean) || ce.some(Boolean)) return;

  // Construir payloads (sin cambiar tipos)
  const fixedDirs = (direcciones.some(d => d.esPrincipal) ? direcciones : [...direcciones[0], esPrincipal: true], ...direcciones.slice(1))
    .map(DireccionDB(d => ({ idDireccion: 0, tipo: d.tipo, esPrincipal: d.esPrincipal, detalles: d.detalles, idDistrito: d.idDistrito, idPersona: 0, activo: d.activo })));

  const fixedTels = (telefonos.some(t => t.esPrincipal) ? telefonos : [...telefonos[0], esPrincipal: true], ...telefonos.slice(1))
    .filter(t => Number(t.numero) > 0)
    .map(telefonoDB(t => ({ idtelefono: 0, numero: Number(t.numero), tipo: t.tipo || "Otro", activo: !!t.activo, esPrincipal: !!t.esPrincipal, idPersona: 0, nombrePersona: "", apellido1: "", apellido2: "" })));

  const fixedCors = (correos.some(c => c.esPrincipal) ? correos : [...correos[0], esPrincipal: true], ...correos.slice(1))
    .map(CorreoElectronicoDB(c => ({ idCorreo: 0, correoElectronico: (c.correoElectronico ?? "").trim(), tipo: c.tipo || "Trabajo", activo: !!c.activo, esPrincipal: !!c.esPrincipal, idPersona: 0, nombrePersona: "" })));

  // Entregar al padre / capa de API
  onCreate({ persona, direcciones: fixedDirs, telefonos: fixedTels, correos: fixedCors });
};
```

Fuente: Elaboración propia

Figura 34
Código Salidas

```
headerName: "Registro real",
marryChildren: true,
children: [
  {
    headerName: "Entrada",
    field: "horaEntrada",
    maxWidth: 120,
    valueFormatter: ({ value }) =>
      | value ? String(value).slice(0, 8) : "-",
  },
  {
    headerName: "Salida",
    field: "horaSalida",
    maxWidth: 120,
    valueFormatter: ({ value }) =>
      | value ? String(value).slice(0, 8) : "-",
  },
  { headerName: "Horas", field: "horasTrabajadas", maxWidth: 100 },
  {
    headerName: "Estado",
    field: "estado",
    maxWidth: 140,
    valueGetter: (p) => {
      const r = p.data as Row;
      const enTurno = r.fecha --- HOY && !!r.horaEntrada && !r.horaSalida;
      if (enTurno) return "EN_TURNO";
      return r.estado --- "CERRADO" ? "CERRADO" : "PENDIENTE";
    },
    valueFormatter: ({ value }) =>
      | value --- "CERRADO"
      | ? "Cerrado"
      | : value --- "EN_TURNO"
      | ? "En turno"
      | : "Pendiente",
    cellClass: (p) => {
      const v = p.value as string;
      if (v --- "CERRADO") return "bg-emerald-50 text-emerald-700";
      if (v --- "EN_TURNO") return "bg-sky-50 text-sky-700";
      return "bg-amber-50 text-amber-700";
    },
  },
],
],
}
```

Fuente: Elaboración propia

Figura 35
Código Procesos

```
@Service no usages new *
public class AguinaldoService {

    private final IngresoSalarialRepository ingresoRepo; 2 usages

    public AguinaldoService(IngresoSalarialRepository ingresoRepo) { no usages new *
        this.ingresoRepo = ingresoRepo;
    }

    public AguinaldoResultado calcularAguinaldo(Long empleadoId, int anioCierre) { 1 usage new *
        Objects.requireNonNull(empleadoId, message: "empleadoId requerido");

        LocalDate inicio = LocalDate.of(year: anioCierre - 1, Month.DECEMBER, dayOfMonth: 1);
        LocalDate fin = LocalDate.of(anioCierre, Month.NOVEMBER, dayOfMonth: 30);
        // Traer SOLO ingresos de NATURALEZA SALARIAL dentro del periodo legal
        List<IngresoSalarial> ingresos = ingresoRepo
            .findByEmpleadoIdAndFechaBetweenAndEsSalarialTrue(empleadoId, inicio, fin);
        BigDecimal totalDevengado = ingresos.stream()
            .map(IngresoSalarial::montoConsideradoParaAguinaldo) // ver nota abajo
            .reduce(BigDecimal.ZERO, BigDecimal::add);
        // Fórmula oficial: dividir entre 12, aunque la persona haya laborado menos de 12 meses
        BigDecimal aguinaldo = totalDevengado
            .divide(BigDecimal.valueOf(12), scale: 2, RoundingMode.HALF_UP);
        return new AguinaldoResultado(empleadoId, inicio, fin, totalDevengado, aguinaldo);
    }

    public AguinaldoResultado calcularAguinaldoCierreActual(Long empleadoId) { no usages new *
        int anioCierre = LocalDate.now().getYear();
        return calcularAguinaldo(empleadoId, anioCierre);
    }

    // ----- DTO resultado -----
    public record AguinaldoResultado( 3 usages new *
        Long empleadoId, no usages
        LocalDate periodoInicio, no usages
        LocalDate periodoFin, no usages
        BigDecimal totalDevengadoSalarial, no usages
        BigDecimal aguinaldo // monto final a pagar no usages
    ) {}
}
```

Fuente: Elaboración propia

Figura 36
Código Validaciones

```
// Persona: campos mínimos, cédula CR (9 dígitos) y mayoría de edad (>18)
const validatePersona = (p: PersonaDB): PersonaErrors => {
  const errs: PersonaErrors = {};
  if (!p.nombre?.trim()) errs.nombre = "El nombre es obligatorio.";
  if (!p.apellido?.trim()) errs.apellido = "El primer apellido es obligatorio.";
  if (!/^\d{9}$/.test(p.cedula || "")) errs.cedula = "Cédula: 9 dígitos (ej. 101110111).";
  if (!p.fechaNacimiento) errs.fechaNacimiento = "La fecha de nacimiento es obligatoria.";
  else if (!isAdult18(p.fechaNacimiento)) errs.fechaNacimiento = "Debe tener 18 años o más.";
  if (!p.sexo) errs.sexo = "Seleccione el sexo.";
  return errs;
};

// Direcciones: al menos una y exactamente una Principal + cascada completa
const validateDirecciones = (list: DireccionForm[]): DireccionErrors => {
  const errs: DireccionErrors = {};
  if (list.length === 0) { errs.item = "Debe registrar al menos una dirección."; return errs; }
  const principals = list.filter(d => d.esPrincipal).length;
  if (principals === 0) errs.principal = "Debe marcar una dirección como Principal.";
  if (principals > 1) errs.principal = "Solo una dirección puede ser Principal.";
  for (const d of list) {
    if (!d.idDistrito) { errs.item = "Cada dirección debe tener un Distrito válido."; break; }
    if (!d.numeroProvincia || !d.numeroCanton || !d.numeroDistrito) {
      errs.item = "Complete Provincia, Cantón y Distrito (números oficiales)."; break;
    }
  }
  return errs;
};
```

Fuente: Elaboración propia

Módulos de alcance:

Figura 37
Módulo Empleados.

```
// Crear un nuevo empleado
@PostMapping("/empleados") no usages  Justin Cruz
public Empleado createUser(@RequestBody Empleado empleado) {
    // Asegúrate de que no haya un empleado con el mismo ID si es necesario
    return empleadoRepository.save(empleado);
}

// Actualizar un empleado existente
@PutMapping("/empleados/{id}") no usages  Justin Cruz
public Empleado updateEmpleado(@PathVariable int id, @RequestBody Empleado empleado) {
    // Verifica si el usuario existe
    Optional<Empleado> existingEmpleado = empleadoRepository.findById(id);
    if (existingEmpleado.isPresent()) {
        // Si existe, actualiza los campos
        Empleado updatedEmpleado = existingEmpleado.get();
        updatedEmpleado.setDepartamento(empleado.getDepartamento());
        updatedEmpleado.setEstado(empleado.getEstado());
        updatedEmpleado.setFechaIngreso(empleado.getFechaIngreso());
        updatedEmpleado.setPersona(empleado.getPersona());
        updatedEmpleado.setPuesto(empleado.getPuesto());
        updatedEmpleado.setSalarioBase(empleado.getSalarioBase());
        return empleadoRepository.save(updatedEmpleado);
    }
    return null; // O podrías lanzar una excepción si el usuario no se encuentra
}
```

Fuente: Elaboración propia

Figura 38
Módulo Asistencia.

```
import { pinedaApi } from "../../datasources/pinedaApi.service";
import { AsistenciaCreatePayload, AsistenciaDB } from "../../modules/asistencia/asistencia.types";

export const useCreateAsistencia = () => {
    const createAsistencia = async (payload: AsistenciaCreatePayload): Promise<AsistenciaDB> => {
        // Si tu backend requiere anidar el empleado en el body, transforma aquí:
        const body = {
            fecha: payload.fecha,
            horaEntrada: payload.horaEntrada ?? null,
            horaSalida: payload.horaSalida ?? null,
            observacion: payload.observacion ?? null,
            empleado: { idEmpleado: payload.idEmpleado }, // <- anidar aquí si el endpoint lo exige
        };
        const { data } = await pinedaApi.post<AsistenciaDB>("/asistencias", body);
        return data;
    };

    return { createAsistencia };
};
```

Fuente: Elaboración propia

Figura 39
Módulo Vacaciones.

```
const handleSubmit = async (e: FormEvent) => {
  e.preventDefault();

  if (errorFecha) {
    alert(errorFecha);
    return;
  }
  if (!form.fechaInicioVacaciones || !form.fechaFinVacaciones) {
    alert("Debes indicar fecha de inicio y fin.");
    return;
  }
  if (!form.empleadoId || form.empleadoId <= 0) {
    alert("Debes seleccionar un empleado.");
    return;
  }

  const currentUserId = usuarioIdDefault ?? user?.idUsuario;
  if (!currentUserId) {
    alert("No hay usuario autenticado para asociar la creación.");
    return;
  }

  const payload: VacacionesCreatePayload = {
    fechaInicioVacaciones: form.fechaInicioVacaciones,
    fechaFinVacaciones: form.fechaFinVacaciones,
    diasTomados: Number(form.diasTomados || 0),
    fechaSolicitud:
      form.fechaSolicitud || new Date().toISOString().slice(0, 10),
    estado: form.estado, // <-- incluido en payload
    activo: !!form.activo,
    empleado: { idEmpleado: form.empleadoId },
    usuario: { idUsuario: Number(currentUserId) },
  };

  console.group("[CreateVacaciones] Submit");
  console.log("Payload:", payload);
  console.groupEnd();

  await onCreate(payload);

  setForm({
    fechaInicioVacaciones: "",
    fechaFinVacaciones: "",
    diasTomados: 0,
    fechaSolicitud: "",
    estado: "PENDIENTE",
    activo: true,
    empleadoId: 0,
  });
};
```

Fuente: Elaboración propia

Figura 40
Módulo Permisos.

```
const handleSubmit = async (e: FormEvent) => {
  e.preventDefault();

  if (errorFecha) {
    alert(errorFecha);
    return;
  }

  if (!form.fechaInicio || !form.fechaFin) {
    alert("Debes indicar fecha de inicio y fin.");
    return;
  }

  if (!form.empleadoId || form.empleadoId <= 0) {
    alert("Debes seleccionar un empleado.");
    return;
  }

  if (!user?.idUsuario) {
    alert("No hay usuario autenticado para asociar la creación.");
    return;
  }

  const payload: PermisosCreatePayload = {
    tipoPermisoSalarial: form.tipoPermisoSalarial,
    fechaSolicitudpermiso: form.fechaSolicitudpermiso,
    fechaInicio: form.fechaInicio,
    fechaFin: form.fechaFin,
    estado: form.estado || "PENDIENTE",
    activo: !!form.activo,
    motivo: form.motivo,
    diasPermiso: Number(form.diasPermiso || 0),
    empleado: { idEmpleado: form.empleadoId },
    usuario: { idUsuario: user.idUsuario as number },
  };

  console.group("[CreatePermisos] Submit");
  console.log("Payload:", payload);
  console.groupEnd();

  await onCreate(payload);
}
```

Fuente: Elaboración propia

Figura 41
Módulo Incapacidades.

```
// Crear una nueva incapacidad
@PostMapping("/incapacidades") no usages  ± Justin Cruz
public Incapacidad createIncapacidad(@RequestBody Incapacidad incapacidad) {
    return incapacidadRepository.save(incapacidad);
}

// Actualizar una incapacidad existente
@PutMapping("/incapacidades/{id}") no usages  ± Justin Cruz
public Incapacidad updateIncapacidad(@PathVariable int id, @RequestBody Incapacidad incapacidad) {
    Optional<Incapacidad> existingIncapacidad = incapacidadRepository.findById(id);
    if (existingIncapacidad.isPresent()) {
        Incapacidad updatedIncapacidad = existingIncapacidad.get();

        updatedIncapacidad.setFechaInicio(incapacidad.getFechaInicio());
        updatedIncapacidad.setFechaFin(incapacidad.getFechaFin());
        updatedIncapacidad.setDiasIncapacidad(incapacidad.getDiasIncapacidad());
        updatedIncapacidad.setTipoIncapacidad(incapacidad.getTipoIncapacidad());
        updatedIncapacidad.setEstado(incapacidad.getEstado());
        updatedIncapacidad.setActivo(incapacidad.getActivo());
        updatedIncapacidad.setTipoPermisoSalarialIncapacidad(incapacidad.getTipoPermisoSalarialIncapacidad());

        if (incapacidad.getEmpleado() != null) {
            updatedIncapacidad.setEmpleado(incapacidad.getEmpleado());
        }
        if (incapacidad.getUsuario() != null) {
            updatedIncapacidad.setUsuario(incapacidad.getUsuario());
        }

        return incapacidadRepository.save(updatedIncapacidad);
    }
    return null;
}
```

Fuente: Elaboración propia

Figura 42
Módulo Nómina.

```
0 public class NominaService {
7     public Quincena calcularQuincena( no usages new *
8         BigDecimal baseQuincenal = salarioBaseMensual
9             .divide(new BigDecimal( val: "2"), scale: 2, RoundingMode.HALF_UP);
10
11         BigDecimal salarioDiario = salarioBaseMensual
12             .divide(new BigDecimal( val: "30"), scale: 5, RoundingMode.HALF_UP);
13
14         BigDecimal rebajoIncapDias13 = salarioDiario
15             .multiply(new BigDecimal(incap.dias1a3()))
16             .multiply(new BigDecimal( val: "0.50"))
17             .setScale( newScale: 2, RoundingMode.HALF_UP);
18
19         BigDecimal rebajoIncapDia4Mas = salarioDiario
20             .multiply(new BigDecimal(incap.dias4omas()))
21             .setScale( newScale: 2, RoundingMode.HALF_UP);
22
23         BigDecimal rebajosAusencias = nvl(rebajosPermisosSinGoce)
24             .add(rebajoIncapDias13)
25             .add(rebajoIncapDia4Mas);
26
27         // 3) Bruto del periodo (cotizable típico)
28         BigDecimal bruto = baseQuincenal
29             .add(nvl(totalExtrasQuincena))
30             .subtract(rebajosAusencias)
31             .max(BigDecimal.ZERO)
32             .setScale( newScale: 2, RoundingMode.HALF_UP);
33
34         // 4) Deducciones obrero (SEM + IVM + BP) - parametrizables
35         BigDecimal totalTasaObrero = tasaSEM.add(tasaIVM).add(tasaBP);
36         BigDecimal deduccionesCCSS = bruto.multiply(totalTasaObrero)
37             .setScale( newScale: 2, RoundingMode.HALF_UP);
38
39         BigDecimal renta = BigDecimal.ZERO;
40         BigDecimal otras = BigDecimal.ZERO;
41
42         // 5) Neto
43         BigDecimal neto = bruto.subtract(deduccionesCCSS).subtract(renta).subtract(otras)
44             .max(BigDecimal.ZERO)
45             .setScale( newScale: 2, RoundingMode.HALF_UP);
46
47         return new Quincena(
```

Fuente: Elaboración propia

Figura 43
Módulo Aguinaldo.

```
public class AguinaldoService {
    private final IngresoSalarialRepository ingresoRepo; 2 usages

    public AguinaldoService(IngresoSalarialRepository ingresoRepo) { no usages new *
        this.ingresoRepo = ingresoRepo;
    }

    public AguinaldoResultado calcularAguinaldo(Long empleadoId, int anioCierre) { 1 usage new *
        Objects.requireNonNull(empleadoId, message: "empleadoId requerido");

        LocalDate inicio = LocalDate.of(year: anioCierre - 1, Month.DECEMBER, dayOfMonth: 1);
        LocalDate fin = LocalDate.of(anioCierre, Month.NOVEMBER, dayOfMonth: 30);

        // Traer SOLO ingresos de NATURALEZA SALARIAL dentro del periodo legal
        List<IngresoSalarial> ingresos = ingresoRepo
            .findByEmpleadoIdAndFechaBetweenAndEsSalarialTrue(empleadoId, inicio, fin);

        BigDecimal totalDevengado = ingresos.stream()
            .map(IngresoSalarial::montoConsideradoParaAguinaldo) // ver nota abajo
            .reduce(BigDecimal.ZERO, BigDecimal::add);

        // Fórmula oficial: dividir entre 12, aunque la persona haya laborado menos de 12 meses
        BigDecimal aguinaldo = totalDevengado
            .divide(BigDecimal.valueOf(12), scale: 2, RoundingMode.HALF_UP);

        return new AguinaldoResultado(empleadoId, inicio, fin, totalDevengado, aguinaldo);
    }

    public AguinaldoResultado calcularAguinaldoCierreActual(Long empleadoId) { no usages new *
        int anioCierre = LocalDate.now().getYear();
        return calcularAguinaldo(empleadoId, anioCierre);
    }

    // ----- DTO resultado -----
    public record AguinaldoResultado( 3 usages new *
        Long empleadoId, no usages
        LocalDate periodoInicio, no usages
        LocalDate periodoFin, no usages
        BigDecimal totalDevengadoSalarial, no usages
        BigDecimal aguinaldo // monto final a pagar no usages
    ) {}
}
```

Fuente: Elaboración propia

Figura 44
Módulo Liquidación.

```
public class LiquidacionService {
    public ResultadoLiquidacion generarLiquidacion( no usages new *
        // Salario base de cálculo (convención: diario = mensual / 30)
        BigDecimal salarioDiario = salarioMensualPromedio
            .divide(new BigDecimal( val: "30"), scale: 5, RoundingMode.HALF_UP);
        // 1) Preaviso (solo si patrono debe y NO lo otorgó)
        int diasPreaviso = 0;
        if (!preavisoOtorgado && tipo == TipoCese.DESPIDO_CON_RESP_PATRONAL) {
            diasPreaviso = diasPreavisoSegunAntiguedad(fechaIngreso, fechaSalida);
        }
        BigDecimal montoPreaviso = salarioDiario.multiply(new BigDecimal(diasPreaviso))
            .setScale( newScale: 2, RoundingMode.HALF_UP);
        // 2) Cesantia (solo si despido con responsabilidad patronal)
        BigDecimal diasCesantia = BigDecimal.ZERO;
        if (tipo == TipoCese.DESPIDO_CON_RESP_PATRONAL) {
            diasCesantia = diasCesantiaSegunArt29(fechaIngreso, fechaSalida);
        }
        // tope legal: máximo 8 años (la función ya lo respeta)
        BigDecimal montoCesantia = salarioDiario.multiply(diasCesantia)
            .setScale( newScale: 2, RoundingMode.HALF_UP);
        // 3) Vacaciones pendientes (días * diario)
        BigDecimal montoVacaciones = salarioDiario
            .multiply(new BigDecimal(vacacionesPendientesDias))
            .setScale( newScale: 2, RoundingMode.HALF_UP);
        // 4) Aguinaldo proporcional (del 1/dic al cese ÷ 12)
        BigDecimal aguinaldoProporcional = nvl(devengadoSalarialDec1_a_Salida)
            .divide(new BigDecimal( val: "12"), scale: 2, RoundingMode.HALF_UP);
        BigDecimal total = montoPreaviso
            .add(montoCesantia)
            .add(montoVacaciones)
            .add(aguinaldoProporcional)
            .setScale( newScale: 2, RoundingMode.HALF_UP);
        return new ResultadoLiquidacion(
            empleadoId, tipo, fechaIngreso, fechaSalida,
            salarioMensualPromedio.setScale( newScale: 2, RoundingMode.HALF_UP),
            diasPreaviso, montoPreaviso,
            diasCesantia, montoCesantia,
            vacacionesPendientesDias, montoVacaciones,
            aguinaldoProporcional, total
        );
    }
}
```

Fuente: Elaboración propia

Figura 45
Módulo seguridad.

```
public class SecurityConfig {
    private final JwtAuthFilter jwtFilter; 2 usages
    private final UserDetailsService uds; 2 usages
    public SecurityConfig(JwtAuthFilter jwtFilter, UserDetailsService uds) { no usages  Justin Cruz
        this.jwtFilter = jwtFilter;
        this.uds = uds;
    }
    @Bean 1 usage  Justin Cruz
    public PasswordEncoder passwordEncoder() {
        return new BCryptPasswordEncoder();
    }
    @Bean 1 usage  Justin Cruz
    public AuthenticationProvider authenticationProvider() {
        var provider = new DaoAuthenticationProvider();
        provider.setUserDetailsService(uds);
        provider.setPasswordEncoder(passwordEncoder());
        return provider;
    }
    @Bean no usages  Justin Cruz
    public AuthenticationManager authenticationManager(AuthenticationConfiguration cfg) throws Exception {
        return cfg.getAuthenticationManager();
    }
    @Bean no usages  Justin Cruz
    public SecurityFilterChain filterChain(HttpSecurity http) throws Exception {
        http
            .csrf(csrf -> csrf.disable())
            .cors(c -> c.configurationSource(corsConfigurationSource()))
            .sessionManagement(s -> s.sessionCreationPolicy(SessionCreationPolicy.STATELESS))
            .authorizeHttpRequests(reg -> reg
                // permitir login y health sin token
                .requestMatchers(...patterns: "/api/auth/login", "/actuator/health").permitAll()
                // permitir preflight OPTIONS sin autenticación
                .requestMatchers(org.springframework.http.HttpMethod.OPTIONS, ...patterns: "**").permitAll()
                // todo lo demás requiere estar autenticado con JWT
                .anyRequest().authenticated()
            )
            .authenticationProvider(authenticationProvider())
            .addFilterBefore(jwtFilter, UsernamePasswordAuthenticationFilter.class);
        return http.build();
    }
}
```

Fuente: Elaboración propia

Figura 46
Módulo Horas extra.

```
const handleSubmit = async (e: FormEvent) => {
  e.preventDefault();
  const next = validateForm(form);
  setErrors(next);
  if (Object.keys(next).length > 0) {
    scrollToFirstError(next);
    return;
  }

  const payload: HorasExtraCreatePayload = {
    fechaSolicitud: form.fechaSolicitud,
    fechaInicio: ensureSeconds(form.fechaInicio),
    fechaFin: ensureSeconds(form.fechaFin),
    estado: !!form.estado,
    empleado: { idEmpleado: form.empleadoId },
    usuario: { idUsuario: Number(user!.idUsuario) },
    observacion: form.observacion.trim(),
    totalHoras: Number(form.totalHoras) || 0,
  };

  console.group("[CreateHorasExtra] Submit");
  console.log("State:", form);
  console.log("Payload:", payload);
  console.groupEnd();

  await onCreate(payload);

  // Reset, dejando fechaSolicitud en el día actual
  setForm({
    fechaSolicitud: todayLocalDate(),
    fechaInicio: "",
    fechaFin: "",
  });
};
```

Fuente: Elaboración propia

Figura 47
Módulo Evaluar rendimiento.

```
import { EvaluacionRendimientoCreatePayload } from "../../modules/evaluacionRendimiento/evaluacionRendimiento.types";
import { pinedaApi } from "../../datasources/pinedaApi.service";

export const useCreateEvaluacionRendimiento = () => {
  const createEvaluacionRendimiento = async (
    newEvaluacionRendimiento: EvaluacionRendimientoCreatePayload
  ): Promise<EvaluacionRendimientoCreatePayload> => {
    try {
      const response = await pinedaApi.post<EvaluacionRendimientoCreatePayload>({
        "/evaluacionRendimiento",
        newEvaluacionRendimiento
      });
      return response.data;
    } catch (error) {
      console.error("Error creating evaluacion-rendimiento:", error);
      throw error;
    }
  };

  return { createEvaluacionRendimiento };
};
```

Fuente: Elaboración propia

Pruebas

A continuación, se muestran las pruebas realizadas

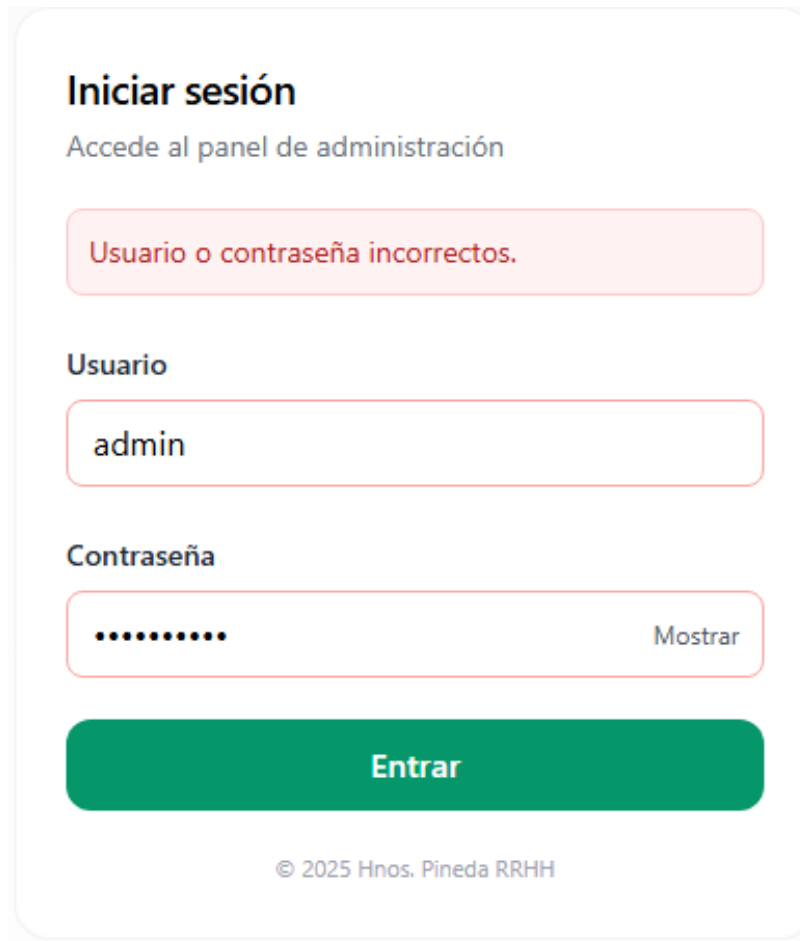
Tabla 31
Prueba uno

Campo	Descripción / Ejemplo
ID del Caso de Prueba	CT-001
Nombre del Caso de Prueba	Validar inicio de sesión con credenciales incorrectas
Modulo / Funcionalidad	Seguridad / Autenticación (Login)
Tipo de Prueba	Sistema / Aceptación
Prioridad	Alta
Precondiciones	<ul style="list-style-type: none">• Usuario “admin” existe, activo y con rol asignado.• Password hash almacenado en BD (BCrypt).
Datos de entrada	<ul style="list-style-type: none">• Usuario: admin• Contraseña: Admin#2025
Pasos por seguir	<ul style="list-style-type: none">• Abrir la aplicación.• Ingresar usuario y contraseña.• Presionar el botón "Iniciar sesión".
Resultado esperado	<ul style="list-style-type: none">• El sistema no permite el acceso.• Se muestra el mensaje: "Usuario o contraseña incorrectos."• No se genera token JWT.• No redirige al panel principal.
Resultado Obtenido	Sistema muestra el mensaje “Usuario o contraseña incorrectos.” y no permite avanzar.
Estado de la Prueba	APROBADO (PASS)

Fuente: Elaboración propia

Figura 48

Validación de inicio de sesión



Iniciar sesión
Accede al panel de administración

Usuario o contraseña incorrectos.

Usuario
admin

Contraseña
..... Mostrar

Entrar

© 2025 Hnos. Pineda RRHH

The image shows a login form with a title 'Iniciar sesión' and a subtitle 'Accede al panel de administración'. Below the subtitle is a red error message box containing the text 'Usuario o contraseña incorrectos.'. There are two input fields: 'Usuario' with the value 'admin' and 'Contraseña' with masked characters '.....'. A 'Mostrar' link is next to the password field. A green 'Entrar' button is at the bottom. A copyright notice '© 2025 Hnos. Pineda RRHH' is at the very bottom.

Fuente: Elaboración propia

Tabla 32
Prueba dos

Campo	Descripción / Ejemplo
ID del Caso de Prueba	CT-003
Nombre del Caso de Prueba	Rechazo cuando fecha fin < fecha inicio y aceptación válida
Modulo / Funcionalidad	Permisos / Crear solicitud
Tipo de Prueba	Validación / Aceptación
Prioridad	Media
Precondiciones	Usuario autenticado con permisos de RRHH.
Datos de entrada	Intento 1 (inválido): inicio 12-11-2025, fin 10-11-2025
Pasos por seguir	<ul style="list-style-type: none"> • Abrir “Crear Permiso”. • Cargar intento 1 y presionar “Guardar”.
Resultado esperado	Resultado correcto que se espera obtener si el sistema funciona adecuadamente. Ejemplo: El sistema redirige al panel principal y muestra el nombre del usuario.
Resultado Obtenido	El sistema mostró validación y no permitió guardar hasta corregir la fecha.
Estado de la Prueba	Aprobado (PASS), Fallido (FAIL), Bloqueado (BLOCKED).

Fuente: Elaboración propia

Figura 49
Validación de fechas en el módulo Permisos

Crear Permiso ✕

Completá los datos y guardá para crear el registro.

Datos Usuario #2

Tipo de Permiso: Estado: Activo

Fecha de Solicitud: Fecha de Inicio: Fecha de Fin: La fecha final debe ser igual o posterior a la fecha de inicio.

Días de Permiso (calculado):

Motivo:

Empleado:

Fuente: Elaboración propia

Tabla 33
Prueba tres

Campo	Descripción / Ejemplo
ID del Caso de Prueba	CT-011
Nombre del Caso de Prueba	Impedir registro de persona menor de 18 años.
Modulo / Funcionalidad	Personas / Validaciones de fecha de nacimiento.
Tipo de Prueba	Sistema / Validación
Prioridad	Alta
Precondiciones	Usuario autenticado con permisos de RRHH.

Datos de entrada	fechaNacimiento > fecha máxima permitida (hoy – 18 años).
Pasos por seguir	<ul style="list-style-type: none"> • Abrir Personas → Crear. • Ingresar nombre y apellidos válidos. • Ingresar una fecha de nacimiento que represente 7 años. • Presionar Crear persona.
Resultado esperado	Se muestra error “Debe ser mayor de edad (18+)”.
Resultado Obtenido	<p>El sistema detectó que la persona tenía menos de 18 años y mostró el mensaje de validación.</p> <p>El botón "Crear persona" quedó deshabilitado y no permitió registrar el dato.</p>
Estado de la Prueba	APROBADO (PASS)

Fuente: Elaboración propia

Figura 50

Validación Personas: rechazar menor de 18 años

Datos de la Persona

Nombre


Apellido 1

Apellido 2

Cédula


Formato: 9 dígitos sin guiones (ej. 101110111)

Fecha de Nacimiento

Debe ser mayor de edad (18+). Edad actual: 7 años.

Sexo

Fuente: Elaboración propia

Tabla 34
Prueba cuatro

Campo	Descripción / Ejemplo
ID del Caso de Prueba	CT-INC-005
Nombre del Caso de Prueba	Rechazar incapacidad cuando la fecha fin es menor a la fecha inicio.
Modulo / Funcionalidad	Incapacidades / Crear incapacidad
Tipo de Prueba	Sistema / Validación
Prioridad	Alta
Precondiciones	<ul style="list-style-type: none"> • Usuario autenticado con permisos de RRHH.
Datos de entrada	<ul style="list-style-type: none"> • Fecha inicio: 09-09-2025 • Fecha fin: 08-08-2025
Pasos por seguir	<ul style="list-style-type: none"> • Abrir Incapacidades → Crear. • Ingresar Fecha inicio 09-09-2025 • Ingresar Fecha fin 08-08-2025. <p>Presionar Guardar.</p>
Resultado esperado	<ul style="list-style-type: none"> • Se muestra mensaje de validación: “La fecha final debe ser igual o posterior a la fecha de inicio.” • El sistema no guarda el registro; no se crea folio de incapacidad.
Resultado Obtenido	
Estado de la Prueba	

Fuente: Elaboración propia

Figura 51

Rechazar incapacidad cuando la fecha fin es menor a la fecha inicio

Crear Incapacidad ✕

Completá los datos y guardá para crear el registro.

Usuario #2

Datos

Fecha de Inicio: 09/09/2025

Fecha de Fin: 08/08/2025

La fecha final debe ser igual o posterior a la fecha de inicio.

Días de Incapacidad (calculado): 0

Tipo de Incapacidad: Ej.: Enfermedad, Accidente...

Tipo de Permiso: Seleccione tipo...

Empleado: Seleccione un empleado...

Crear incapacidad

Fuente: Elaboración propia

Tabla 35

Prueba cinco

Campo	Descripción / Ejemplo
ID del Caso de Prueba	CT-PER-001
Nombre del Caso de Prueba	Bloquear creación de Persona con cédula de menos de 9 dígitos.
Modulo / Funcionalidad	Personas / Crear persona
Tipo de Prueba	Sistema / Validación
Prioridad	Alta
Precondiciones	Usuario autenticado con permisos de RRHH.
Datos de entrada	<ul style="list-style-type: none">Nombre: María

	<ul style="list-style-type: none"> • Apellido1: Rojas • Apellido2: Vargas • Cédula: 10111011 (8 dígitos — inválida) • Fecha de nacimiento: 05-10-1995 • Sexo: Femenino
Pasos por seguir	<ul style="list-style-type: none"> • Abrir Personas → Crear. • Completar el formulario con los datos indicados. • Presionar Crear persona.
Resultado esperado	Se muestra mensaje de validación: “Cédula: 9 dígitos (ej. 101110111)”.
Resultado Obtenido	
Estado de la Prueba	

Fuente: Elaboración propia

Figura 52

Bloquear creación de Persona con cédula de menos de 9 dígitos

Datos de la Persona

Nombre

Apellido 1

Apellido 2

Cédula

Formato: 9 dígitos sin guiones (ej. 101110111)
Cédula: 9 dígitos (ej. 101110111).

Fecha de Nacimiento

Sexo
 ▼

Fuente: Elaboración propia

CAPÍTULO VI: CONCLUSIONES Y RECOMENDACIONES

Conclusiones

En el desarrollo del prototipo funcional del Sistema Web de Gestión de Recursos Humanos para el Supermercado Hnos Pineda, se logró realizar un análisis detallado de los requerimientos que permitieron comprender, de manera clara, las necesidades reales de la empresa. La recolección de datos mediante entrevistas, cuestionarios y observación directa fue fundamental para identificar los principales problemas operativos, entre ellos la dependencia de hojas de Excel, errores frecuentes en planillas y falta de control en asistencia y vacaciones. Dicho análisis sirvió como base para definir los módulos prioritarios del sistema.

El diseño de la arquitectura del prototipo se realizó siguiendo criterios de escalabilidad, simplicidad y bajo costo, tomando en cuenta las limitaciones tecnológicas actuales del supermercado. Gracias a esto, se logró estructurar un sistema modular que puede ampliarse en el futuro, sin necesidad de rehacer todo el software. La correcta definición de la base de datos y de los procesos internos permitió construir un soporte sólido para las etapas posteriores del desarrollo.

La fase de programación del prototipo se ejecutó respetando los lineamientos establecidos durante el análisis y diseño. Las herramientas utilizadas (React con TypeScript, Spring Boot, Tailwind CSS y MySQL) permitieron implementar funcionalidades alineadas con lo que el cliente realmente necesita. Como resultado, se obtuvo un prototipo funcional que automatiza procesos clave como la nómina, el control de asistencia, el cálculo de aguinaldo y la gestión de vacaciones, evitando errores manuales y mejorando la trazabilidad de los datos.

Durante la fase de pruebas, se realizaron validaciones funcionales, de interfaz y de lógica interna, lo que permitió detectar inconsistencias antes de la implementación definitiva. Esto permitió corregir fallos de forma anticipada y asegurar que el sistema responda, adecuadamente, a los escenarios más frecuentes dentro del supermercado. Las pruebas también confirmaron que el prototipo es intuitivo y puede ser utilizado sin necesidad de conocimientos técnicos avanzados.

En síntesis, el desarrollo del prototipo se llevó a cabo de manera ordenada, siguiendo las etapas del modelo en cascada y cumpliendo los objetivos planteados al inicio del proyecto.

Recomendaciones

Se recomienda realizar una capacitación inicial para los colaboradores que utilizarán el sistema web de Recursos Humanos, especialmente en los módulos relacionados con asistencia, nómina, vacaciones y rebajos, con el fin de asegurar que el personal se adapte correctamente al uso de la plataforma. Esta formación puede ser organizada por el encargado de RRHH del supermercado y desarrollarse en un lapso aproximado de dos semanas, mediante sesiones prácticas que permitan a los trabajadores familiarizarse con las funciones básicas y evitar errores durante la transición desde los registros manuales.

Se sugiere designar a una persona encargada de brindar soporte técnico básico al sistema dentro del supermercado, ya sea un colaborador actual con conocimientos tecnológicos o alguien contratado específicamente para esta tarea. Su función sería dar seguimiento al funcionamiento general del sistema, coordinar ajustes con el desarrollador cuando haga falta y mantener el orden en la administración de usuarios, lo cual resultará más eficiente que depender únicamente de ayuda externa cada vez que surja un inconveniente.

Se recomienda implementar lineamientos internos de seguridad digital para el uso del sistema, incluyendo la obligación de crear contraseñas seguras, evitar el préstamo de credenciales entre compañeros y activar el cierre de sesión automático después de cierto periodo de inactividad. Estas medidas pueden definirse y comunicarse desde el área de RRHH con apoyo del responsable técnico y resultan esenciales para cumplir con lo establecido en la Ley 8968 sobre protección de datos personales.

Es aconsejable establecer un sistema de respaldo periódico de la base de datos, preferiblemente de manera mensual o quincenal, almacenando la copia en un dispositivo o ubicación distinta al equipo principal para prevenir la pérdida definitiva de información en caso de fallos eléctricos, virus o errores humanos. Esta tarea puede ser supervisada por el encargado TIC, quien además debería comprobar regularmente que los respaldos puedan restaurarse sin problemas.

Se propone realizar encuestas o evaluaciones periódicas entre los usuarios del sistema para conocer su nivel de satisfacción, identificar dificultades en el uso de la plataforma y recoger sugerencias de mejora. Estas mediciones pueden aplicarse cada cuatro meses desde el área de RRHH y deben contemplar aspectos como facilidad de uso, claridad de la interfaz

y utilidad de los módulos ofrecidos, lo que permitirá adaptar el sistema a la experiencia real de quienes lo utilizan.

Finalmente, se sugiere implementar un plan básico de seguridad informática que incluya pequeñas sesiones de sensibilización sobre riesgos digitales, tales como el uso indebido de contraseñas, acceso desde dispositivos no autorizados o intentos de suplantación. Este proceso preventivo, a cargo del responsable del sistema, fortalecerá la protección de los datos personales y salariales del personal y reducirá la posibilidad de incidentes derivados del desconocimiento del usuario.

Referencias

- Arsys. (s. f.). *¿Qué es Visual Studio Code y cuáles son sus ventajas?*
<https://www.arsys.es/blog/que-es-visual-studio-code-y-cuales-son-sus-ventajas#:~:text=Por%20supuesto%2C%20el%20prop%C3%B3sito%20principal,aplicaciones%20o%20los%20sitios%20web.>
- Chavez, J. J. S. (2024, 29 abril). *¿Cuáles son los protocolos de seguridad informática?*
<https://www.deltaprotect.com/blog/protocolos-seguridad-informatica>
- Colaboradores de Wikipedia. (2023, 13 enero). *Módulo (informática)*. Wikipedia, la Enciclopedia Libre.
[https://es.wikipedia.org/wiki/M%C3%B3dulo_\(inform%C3%A1tica\)#:~:text=En%20programaci%C3%B3n%20un%20m%C3%B3dulo%20es,varias%2C%20en%20alg%C3%BAn%20caso\).](https://es.wikipedia.org/wiki/M%C3%B3dulo_(inform%C3%A1tica)#:~:text=En%20programaci%C3%B3n%20un%20m%C3%B3dulo%20es,varias%2C%20en%20alg%C3%BAn%20caso).)
- Communications. (2024, 14 mayo). *Arquitectura de «software» o sistemas: qué es y ejemplos*. BBVA NOTICIAS. <https://www.bbva.com/es/innovacion/arquitectura-de-software-o-sistemas-que-es-y-ejemplos/>
- Consultoría, J. (s. f.). *Conoce los enfoques y tipos de investigación para tesis*.
<https://jpconsultoriatesis.com/>. <https://jpconsultoriatesis.com/blog/tipos-de-investigacion-para-tesis/>
- De Contenidos de GoDaddy, E. (2024, 8 octubre). *IntelliJ IDEA: Ventajas para el desarrollo*. GoDaddy Resources - Spain.
<https://www.godaddy.com/resources/es/crearweb/intellij-idea-que-es>
- Diaz, L. (2023, 12 diciembre). *¿Por qué es importante capacitar al personal de tu empresa en seguridad informática?* Hillstone Networks.
<https://www.hillstonenet.lat/blog/seguridad-de-la-red/importante-capacitar-al-personal-de-tu-empresa-en-seguridad-informatica/>
- DocuSign. (2024, abril 1). *¿Qué es un sistema de información de recursos humanos?*
<https://www.docuSign.com/es-mx/blog/sistema-de-informacion-recursos-humanos>
- Fastly. (2024). *¿Cómo definir la seguridad de las aplicaciones web?* | Fastly.
<https://www.fastly.com/es/learning/security/what-is-web-application-security>
- Galicia, F. P. L. (2025, 27 enero). *Cómo proteger tu sitio web de hackers y ataques*. GoDaddy Resources - LATAM.

- <https://www.godaddy.com/resources/latam/seguridad/como-proteger-sitio-web-de-amenazas>
- GitHub. (2023). *About GitHub*. GitHub, Inc. <https://github.com>
- Hutsulyak, O. (2024, 13 agosto). *Why Use React for Web Development: 10 Reasons To Apply*. Blog | TechMagic. <https://www.techmagic.co/blog/why-we-use-react-js-in-the-development>
- IBM. (2024, 29 noviembre). *Java*. IBM. <https://www.ibm.com/mx-es/topics/java#:~:text=En%20concreto%2C%20la%20tecnolog%C3%ADa%20Java,y%20proporcionan%20contenidos%20web%20din%C3%A1micos>.
- JetBrains. (2024). *IntelliJ IDEA: The Java IDE for professional developers*. JetBrains. <https://www.jetbrains.com/idea/>
- Lucide. (2024). *Lucide Open Source Icons*. Lucide Icons Project. <https://lucide.dev>
- McNamee, L. (2024, July 22). *Beneficios de los sistemas de información de recursos humanos (HRIS) para las pequeñas empresas*. YourShortlist. <https://yourshortlist.com/es/beneficios-de-los-sistemas-de-informacion-de-recursos-humanos-hris-para-las-pequenas-empresas/>
- Meta. (2023). *React – A JavaScript library for building user interfaces*. Meta Platforms. <https://react.dev>
- Microsoft. (2023a). *TypeScript Documentation*. Microsoft Corporation. <https://www.typescriptlang.org/>
- Microsoft. (2023b). *Visual Studio Code Documentation*. Microsoft Corporation. <https://code.visualstudio.com/docs>
- Ministerio de Trabajo y Seguridad Social. (10 de diciembre del 2024). *Lista de salarios mínimos por ocupación*. https://www.mtss.go.cr/temas-laborales/salarios/Documentos-Salarios/lista_salarios_2025.pdf
- Ministerio de Trabajo y Seguridad Social. (25 de enero del 2016). *Lista de salarios mínimos por ocupación*. https://www.mtss.go.cr/elministerio/marco-legal/documentos/Codigo_Trabajo_RPL.pdf
- Northware. (2022, 29 julio). *Requerimientos en el desarrollo de software y aplicaciones*. Northware. <https://www.northware.mx/blog/requerimientos-en-el-desarrollo-de-software-y->

- [aplicaciones/#:~:text=requerimientos%20de%20software%20son%20simplemente,las%20necesidades%20de%20los%20usuarios.](#)
- Oracle. (2023). *MySQL Documentation*. Oracle Corporation. <https://dev.mysql.com/doc/>
- Pivotal Software. (2023). *Spring Boot Reference Documentation*. Spring Framework Project. <https://spring.io/projects/spring-boot>
- Postman. (2024). *Postman API Platform Documentation*. Postman, Inc. <https://www.postman.com>
- QBD Group. (s. f.). | *Validación de software y sistemas informáticos* [https://www.qbdgroup.com/es/servicios/software-solutions-services/validacion-de-software-y-sistemas-informaticos/#:~:text=La%20Validaci%C3%B3n%20de%20Sistemas%20Inform%C3%A1ticos%20\(CSV\)%20se%20define%20como%20el,y%20caracter%C3%ADsticas%20de%20calidad%20predeterminadas.](https://www.qbdgroup.com/es/servicios/software-solutions-services/validacion-de-software-y-sistemas-informaticos/#:~:text=La%20Validaci%C3%B3n%20de%20Sistemas%20Inform%C3%A1ticos%20(CSV)%20se%20define%20como%20el,y%20caracter%C3%ADsticas%20de%20calidad%20predeterminadas.)
- QuestionPro*. (s. f.). *Tamaño de la muestra. Qué es y cómo calcularla*. *QuestionPro* <https://www.questionpro.com/es/tama%C3%B1o-de-la-muestra.html>
- ¿Qué es el SDLC? - *Explicación del ciclo de vida del desarrollo de software* - AWS. (s. f.). Amazon Web Services, Inc. <https://aws.amazon.com/es/what-is/sdlc/>
- Ramon, J. (2021, octubre 27). *Cómo la tecnología influye en el departamento de Recursos Humanos*. José Ramón Valbuena Coach Y Mentor. <https://aceleratucarrera.com/como-la-tecnologia-influye-en-el-departamento-de-recursos-humanos-2/>
- Red Hat. (2023). *Hibernate ORM Documentation*. Red Hat, Inc. <https://hibernate.org/orm/>
- Robledano, A. (2019, 24 septiembre). *Qué es MySQL: Características y ventajas*. OpenWebinars.net. <https://openwebinars.net/blog/que-es-mysql/>
- SalusPlay*. (s. f.). *La Muestra y la Población de estudio* | <https://www.salusplay.com/apuntes/apuntes-metodologia-de-la-investigacion/tema-5-la-muestra-y-la-poblacion-de-estudio>
- SalusPlay*. (s. f.). | *Las variables de investigación*. <https://www.salusplay.com/apuntes/apuntes-metodologia-de-la-investigacion/tema-2-las-variables-de-investigacion>

Seguridad web y seguridad de sitios web | Fortinet. (s. f.). Fortinet.

<https://www.fortinet.com/lat/resources/cyberglossary/what-is-web-security>

Tweem. (2024, febrero 28). *Cuáles son las características clave de un software de recursos humanos para tu empresa.* [https://tweem.com/detalle-](https://tweem.com/detalle-noticia/dynacontent/caracteristicas-clave-software-recursos-humanos.html)

[noticia/dynacontent/caracteristicas-clave-software-recursos-humanos.html](https://tweem.com/detalle-noticia/dynacontent/caracteristicas-clave-software-recursos-humanos.html)

Uso de Tailwind CSS para el desarrollo web - Idenautas Web Factory. (s. f.).

<https://www.idenautas.com/blog/uso-de-tailwind-css-para-el-desarrollo-web/>

Apéndice

GUÍA DE ENTREVISTA

Organización: Supermercado Hnos Pineda

Nombre del Entrevistado: Colaborador del supermercado

Cargo: Encargado administrativo

Preguntas para el Encargado Administrativo

¿Cuáles son las funciones principales que usted realiza dentro del supermercado?

¿Qué tan fácil o difícil le resulta llevar el control de las horas laborales?

¿Cómo se manejan actualmente las vacaciones dentro de la empresa?

¿Ha tenido algún problema con el cálculo de su salario?

¿Cómo solicitan ustedes los permisos o incapacidades?

¿Considera que el sistema actual (manual o Excel) es eficiente?

¿Le gustaría que se automatizaran algunos procesos? ¿Cuáles?

¿Cómo se lleva a cabo la evaluación del desempeño en su puesto?

¿Cree que un sistema web ayudaría a mejorar la organización del trabajo?

¿Qué tipo de información le gustaría tener siempre disponible como empleado?

¿Cuáles son los errores más comunes que ha notado en la gestión de RRHH?

¿Siente que su trabajo es reconocido de manera justa?

¿Qué tan accesible cree que sería un sistema digital para su uso diario?

¿Cómo se manejan actualmente las horas extras?

¿Qué cambios considera más urgentes en la forma de gestionar el personal?

CUESTIONARIO

En el marco de una investigación sobre Sistema de gestión de Recursos Humanos, le invitamos a completar este cuestionario. Su participación es de gran importancia para comprender cómo el tema en estudio influye en la actividad de la organización.

Este cuestionario es confidencial. Sus respuestas solo se utilizarán con fines de investigación y no serán compartidas con ninguna otra persona o institución. Completar el cuestionario tomará aproximadamente 10 minutos

Marque con una “X” la opción que mejor representa su experiencia. Las respuestas son confidenciales y utilizadas únicamente con fines de investigación.

1. ¿Con qué frecuencia recibe su salario de forma correcta y puntual?
 Siempre Casi siempre Algunas veces Nunca
2. ¿Con qué frecuencia ha identificado errores en los cálculos de sus horas extras?
 Nunca Rara vez Algunas veces Frecuentemente
3. ¿Qué tan clara es la información que recibe sobre sus días de vacaciones?
 Muy clara Algo clara Poco clara Nada clara
4. ¿Cómo califica el proceso de solicitud de permisos?
 Muy fácil Fácil Difícil Muy difícil
5. ¿Considera que sus horarios de trabajo son respetados por la organización?
 Siempre Casi siempre Algunas veces Nunca
6. ¿Tiene acceso a su historial de asistencia (entradas y salidas)?
 Sí, de forma clara y actualizada Sí, pero con limitaciones No tengo acceso No sé si existe
7. ¿Considera que la evaluación de su desempeño es justa y objetiva?
 Totalmente de acuerdo De acuerdo En desacuerdo Totalmente en desacuerdo
8. ¿Qué tan eficientes considera los procesos administrativos actuales del supermercado?
 Muy eficientes Eficientes Poco eficientes Ineficientes

9. ¿Ha recibido algún tipo de bonificación relacionada con su rendimiento laboral?
 Sí, frecuentemente Sí, ocasionalmente No, nunca No sé si existen bonificaciones
10. ¿Considera que la implementación de un sistema web mejoraría la gestión del personal?
 Totalmente de acuerdo De acuerdo En desacuerdo Totalmente en desacuerdo

GUÍA DE OBSERVACIÓN

Nombre de la Empresa: Supermercado Henos Pineda

Actividad de la Empresa: Comercio minorista / Venta de productos de consumo

OBJETIVO: Observar y evaluar las actividades relacionadas con la gestión del recurso humano dentro del Supermercado Hnos. Pineda, ubicado en Hojancha, Guanacaste, con el fin de identificar oportunidades de mejora para el desarrollo del prototipo funcional del sistema de gestión de RRHH.

N#	Aspectos por observar	Cumple	No Cumple	Oportunidad de mejora	Detalle de Observación
1	Flujo de trabajo y procesos actuales.				
2	Uso de la tecnología.				
3	Organización y manejo de la información.				
4	Mecanismo de control de vacaciones e incapacidades.				
5	Procedimientos para solicitud de permisos laborales.				
6	Registro de horas extra y cálculo salarial.				
7	Evaluación del desempeño del personal.				
8	Gestión y control de asistencia.				