

UNIVERSIDAD INTERNACIONAL DE LAS AMÉRICAS

ESCUELA DE INGENIERÍA INFORMÁTICA

**SISTEMA WEB PARA LA GESTIÓN DE PAGO DE NÓMINA
PARA LA EMPRESA ALL SPORT NUTRITION**

**MODALIDAD DE PROYECTO PARA OPTAR POR EL GRADO
DE BACHILLER EN INGENIERÍA INFORMÁTICA**

KENDALL SALAZAR SOTO

SEPTIEMBRE, 2025

ÍNDICE DE CONTENIDO

Capítulo I. Introducción	30
Planteamiento del problema	30
Objetivos	30
Objetivo general.	30
Objetivos específicos	30
Justificación.....	31
Viabilidad técnica.....	31
Viabilidad operativa.	32
Viabilidad económica.....	32
Viabilidad legal.	34
Proyecciones	35
Alcance funcional.....	35
Alcance metodológico.....	36
Alcance tecnológico.	37
Capítulo II. Marco referencial.....	39
Importancia de la digitalización en la gestión empresarial	39
Contexto costarricense en la adopción de sistemas web.	39
Rol estratégico de la automatización en la gestión de recursos humanos.	41
Conceptos fundamentales.....	42
Nómina.	42
Componentes de la nómina.....	42
Marco legal.	43
Requerimientos de software	44
Requerimientos funcionales.	45
Registro de empleados.....	45
Cálculo automático de planillas.....	46
Generación de reportes.	46
Control de usuarios y roles.	46
Requerimientos no funcionales.	47
Seguridad de la información.....	47

Usabilidad y accesibilidad.	47
Escalabilidad y rendimiento.	47
Bases de datos en sistemas de nómina	48
Definición de la base de datos relacional.	48
Herramientas de diseño de bases de datos.....	49
Integridad y seguridad de datos.....	50
Desarrollo de aplicaciones web.....	51
Ciclo de vida del Software.	52
Programación de funcionalidades.....	53
Gestión de usuarios.....	53
Gestión de planillas.....	53
Reportes.	53
Tecnologías relevantes.	54
Capítulo III. Marco metodológico	56
Enfoques de investigación.....	56
Enfoque cuantitativo.....	56
Enfoque cualitativo.....	57
Enfoque mixto.	58
Enfoque de investigación seleccionado	59
Tipos de investigación.....	60
Investigación descriptiva.	60
Investigación exploratoria.	61
Investigación explicativa.	62
Tipo de investigación por utilizar	63
Fuentes de información	64
Fuentes primarias.....	64
Fuentes secundarias.....	64
Fuentes terciarias.	65
Variables	66
Conceptuales.	66
Operacionales.	67

Instrumentales.	67
Población.....	69
Muestra.....	70
Instrumentos de recolección de datos	70
La observación.	70
La entrevista.	71
La encuesta.	71
Capítulo IV. Análisis de resultados	72
Análisis de la observación.....	72
Análisis de las entrevistas	74
Requerimientos del sistema.....	78
Requerimientos funcionales.	78
Requerimientos no funcionales.	80
Capítulo V. Propuesta	82
Casos de uso	82
Diseño del sistema.....	160
Arquitectura del sistema.	160
Arquitectura del software.	161
Diseño de entradas.....	162
Diseño físico de la base de datos	168
Diseño de procesos.	179
Diseño de salidas.	187
Diagramas UML	191
Programación	200
Entradas.	200
Salidas.	205
Procesos.....	207
Validaciones.	212
Módulos señalados en el alcance.....	215
Pruebas del sistema	221
Capítulo VI. Conclusiones y recomendaciones	234

Conclusiones	234
Recomendaciones.....	236
Implementación en un servidor local dedicado y el plan de respaldo de información.....	236
Ejecución de un plan piloto en entorno simulado de producción.....	237
Desarrollo e implementación de un plan de capacitación para el personal.	238
Promoción de una cultura organizacional orientada a la transformación digital.	238
Referencias.....	240
Apéndices.....	245
Apéndice A. Guía de observación.....	245
Apéndice B. Guía de observación aplicada a la empresa All Sport Nutrition	246
Apéndice C. Entrevista semiestructurada	249
Apéndice D. Entrevista semiestructurada aplicada al gerente general de la empresa	250
Apéndice E. Entrevista semiestructurada aplicada al administrador de la empresa	253

ÍNDICE DE TABLAS

Tabla 1 Detalles de posibles gastos por incurrir	33
Tabla 2 Módulos por implementar en el sistema	35
Tabla 3 Principales componentes de la nómina	43
Tabla 4 Variables de la investigación	68
Tabla 5 Requerimientos funcionales del sistema.....	78
Tabla 6 Requerimientos no funcionales del sistema.....	80
Tabla 7 Caso de uso para el inicio de sesión en el sistema.....	82
Tabla 8 Caso de uso para la recuperación y cambio de contraseña	83
Tabla 9 Caso de uso para la actualización de los datos del usuario.....	85
Tabla 10 Caso de uso para la creación de un usuario en el sistema.....	87
Tabla 11 Caso de uso para la eliminación de un usuario en el sistema	88
Tabla 12 Caso de uso para el acceso a los usuarios activos en el sistema.....	90
Tabla 13 Caso de uso para listar la asistencial del día	91
Tabla 14 Caso de uso para agregar un registro de asistencia.....	93
Tabla 15 Caso de uso para actualizar un registro de asistencia	95
Tabla 16 Caso de uso para eliminar un registro de asistencia	97
Tabla 17 Caso de uso para listar registros de horas extra	99
Tabla 18 Caso de uso para agregar un registro de horas extra.....	100
Tabla 19 Caso de uso para actualizar un registro de horas extra	102
Tabla 20 Caso de uso para eliminar un registro de horas extra	104
Tabla 21 Caso de uso para listar permisos de los empleados	106
Tabla 22 Caso de uso para agregar un registro de permisos solicitados.....	108

Tabla 23 Caso de uso para actualizar un registro de permisos solicitados	109
Tabla 24 Caso de uso para eliminar un registro de permisos solicitados	111
Tabla 25 Caso de uso para listar el registro de incapacidades	113
Tabla 26 Caso de uso para agregar un registro de incapacidades	114
Tabla 27 Caso de uso para actualizar un registro de incapacidades	116
Tabla 28 Caso de uso para eliminar un registro de incapacidades	118
Tabla 29 Caso de uso para listar registros de liquidaciones	119
Tabla 30 Caso de uso para agregar un registro de liquidación	121
Tabla 31 Caso de uso para actualizar un registro de liquidación.....	123
Tabla 32 Caso de uso para eliminar un registro de liquidación.....	125
Tabla 33 Caso de uso para listar la cantidad de días disponibles de vacaciones	127
Tabla 34 Caso de uso para listar solicitudes de vacaciones.....	128
Tabla 35 Caso de uso para agregar un registro de vacaciones.....	129
Tabla 36 Caso de uso para actualizar un registro de solicitud de vacaciones.....	131
Tabla 37 Caso de uso para actualizar un registro de solicitud de vacaciones.....	133
Tabla 38 Caso de uso del cálculo de aguinaldo	135
Tabla 39 Caso de uso del cálculo de salario bruto.....	137
Tabla 40 Caso de uso del cálculo de deducciones salariales	138
Tabla 41 Caso de uso del registro y aplicación de rebajos	140
Tabla 42 Caso de uso de realizar el cálculo de la nómina	142
Tabla 43 Caso de uso de generar reporte de empleados	143
Tabla 44 Caso de uso de generar un reporte de historial laboral	145
Tabla 45 Caso de uso de generar un reporte de beneficios del colaborador.....	146

Tabla 46 Caso de uso de generar un reporte de nómina del colaborador	148
Tabla 47 Caso de uso de generar un reporte de nómina del colaborador	150
Tabla 48 Caso de uso para generar un reporte de evaluaciones de rendimiento	151
Tabla 49 Caso de uso para listar las evaluaciones de rendimiento	153
Tabla 50 Caso de uso para agregar una evaluación de rendimiento	154
Tabla 51 Caso de uso para actualizar una evaluación de rendimiento	156
Tabla 52 Caso de uso para eliminar una evaluación de rendimiento.....	158
Tabla 53 Diccionario de datos de la tabla aguinaldo	168
Tabla 54 Diccionario de datos de la tabla asistencia	169
Tabla 55 Diccionario de datos de la tabla auditoría.....	169
Tabla 56 Diccionario de datos de la tabla departamento	170
Tabla 57 Diccionario de datos de la tabla detalles de evaluación.....	170
Tabla 58 Diccionario de datos de la tabla empleados.....	171
Tabla 59 Diccionario de datos de la tabla evaluaciones de rendimiento	172
Tabla 60 Diccionario de datos de la tabla horas extra	172
Tabla 61 Diccionario de datos de la tabla incapacidades	173
Tabla 62 Diccionario de datos de la tabla liquidaciones.....	174
Tabla 63 Diccionario de datos de la tabla métricas de rendimiento	174
Tabla 64 Diccionario de datos de la tabla nóminas	175
Tabla 65 Diccionario de datos de la tabla roles	176
Tabla 66 Diccionario de datos de la tabla permisos	176
Tabla 67 Diccionario de datos de la tabla puestos	177
Tabla 68 Diccionario de datos de la tabla saldo de vacaciones	177

Tabla 69 Diccionario de datos de la tabla usuarios.....	178
Tabla 70 Diccionario de datos de la tabla usuarios roles.....	178
Tabla 71 Diccionario de datos de la tabla vacaciones	179
Tabla 72 Caso de prueba para inicio de sesión exitoso	221
Tabla 73 Caso de prueba para inicio de sesión fallido.....	222
Tabla 74 Caso de prueba para registrar a una persona exitosamente	224
Tabla 75 Caso de prueba para registrar a una persona de manera fallida.....	226
Tabla 76 Caso de prueba para registrar la asistencia de forma exitosa	229
Tabla 77 Caso de prueba para registro fallido de la asistencia	231

ÍNDICE DE FIGURAS

Figura 1 Arquitectura cliente-servidor.....	160
Figura 2 Diseño de la pantalla de inicio de sesión.....	163
Figura 3 Diseño de la pantalla principal de entrada del sistema.....	163
Figura 4 Diseño de entrada para el ingreso de un empleado	165
Figura 5 Diseño de entrada para registrar una asistencia.....	166
Figura 6 Diseño de entrada para realizar la solicitud de un permiso	167
Figura 7 Modelo entidad-relación de la base de datos.....	168
Figura 8 Diagrama de flujo del proceso de inicio de sesión del sistema	180
Figura 9 Diagrama de flujo del registro de un empleado.....	181
Figura 10 Diagrama de flujo del registro de asistencia.....	182
Figura 11 Diagrama de flujo del registro de horas extra	183
Figura 12 Diagrama de flujo del registro de permisos.....	184
Figura 13 Diagrama de flujo del registro de solicitud de vacaciones	185
Figura 14 Diagrama de flujo del registro de incapacidad	186
Figura 15 Diagrama de flujo del cálculo de nómina.....	187
Figura 16 Diseño de la salida de datos personales del empleado	188
Figura 17 Diseño de salida del registro de asistencia semanal	189
Figura 18 Diseño de salida del historial de permisos	189
Figura 19 Diseño de salida de incapacidades registradas	190
Figura 20 Diseño de la salida de los comprobantes de pago de nómina.....	190
Figura 21 Diagrama de clases del prototipo de recursos humanos	191
Figura 22 Diagrama de secuencia para crear un empleado.....	192

Figura 23 Diagrama de secuencia para registrar la asistencia	193
Figura 24 Diagrama de secuencia para registrar horas extra	195
Figura 25 Diagrama de secuencia para registrar un permiso	196
Figura 26 Diagrama de secuencia para registrar una solicitud de vacaciones	197
Figura 27 Diagrama de secuencia para el cálculo de nómina	199
Figura 28 Extracto de código que se utiliza para registrar una asistencia desde la capa de lógica de negocio	201
Figura 29 Extracto de código que se utiliza para generar un período de nómina	202
Figura 30 Extracto de código que se utiliza para actualizar la información de un empleado.....	204
Figura 31 Extracto de código que se utiliza para la salida de datos respecto al historial de incapacidades	205
Figura 32 Extracto de código que se utiliza para la salida de las listas de nóminas	206
Figura 33 Extracto de código que se utiliza para la salida de horas extra de la semana.....	207
Figura 34 Extracto de código que se utiliza para el proceso de cálculo del impuesto sobre la renta	208
Figura 35 Extracto de código que se utiliza para el proceso de cálculo de las horas extra respecto a la nómina	209
Figura 36 Extracto de código que se utiliza para el proceso de cálculo del ajuste salarial por ausencias	210
Figura 37 Extracto de código que se utiliza para el proceso de construcción de un token JWT como medio de autenticación.....	211
Figura 38 Validaciones en el ámbito de DTO	213
Figura 39 Validaciones a nivel de controlador de la API	214
Figura 40 Validaciones a nivel de capa de negocio	215

Figura 41 Código para marcar asistencia.....	216
Figura 42 Código para registrar una horas extra.....	217
Figura 43 Código para registrar un permiso	218
Figura 44 Código para actualizar los datos de un usuario empleado.....	220

DEDICATORIA

Redacto la dedicatoria del presente trabajo con todo orgullo y, en primer lugar a Dios, por haberme brindado la oportunidad de llegar hasta acá y de, primeramente, haber cursado una carrera universitaria con el fin de convertirme en un profesional. Agradezco cada una de las bendiciones que me brindó durante todo el proceso y la sabiduría con la que me bendijo para tomar cada una de las decisiones en cada paso que di en el camino.

Le dedico en segunda instancia este trabajo de graduación a mi madre, Marisol Soto Mora y a mi padre Víctor Salazar Acosta, dos pilares en vida y todo lo que rodea a esta, les agradezco su apoyo incondicional, su amor y por tener fe de las acciones que tomo y hacia dónde me dirijo. Sin ellos nada de esto hubiera sido posible y han sido la pieza angular de este proceso.

Cabe mencionar, que agradezco a cada uno de los miembros de mi familia, especialmente a mi tío Carlos Soto Mora, mi abuela Margarita Mora Mora, mi tío Edwin Soto Mora y a mi otro tío Roy Soto Mora. Estas personas siempre me brindaron una mano cuando lo necesité, un consejo cuando no sabía hacia dónde caminar y un apoyo que siempre estuvo presente en las situaciones de adversidad y también de felicidad.

Reitero a cada una de las personas que menciono y también a las que me acompañaron en el proceso y su nombre no está en este documento, las más sinceras gracias por todo el apoyo brindado y con todo mi corazón les hago dedicatoria del esfuerzo con el que se realizó este proyecto y se culmina el proceso.

AGRADECIMIENTOS

Quiero comenzar expresando mi más sincero reconocimiento a la directora de la Escuela de Ingeniería Informática, Olda Bustillos Ortega, por su liderazgo comprometido y su vocación de servicio tanto a la carrera como a cada uno de los estudiantes que pertenece a la Escuela. Agradezco cada momento que ella estuvo en cercanía, la disposición de siempre escuchar y orientar cada paso que se daba, todo esto fue fundamental para tomar decisiones firmes en cada etapa del camino y seguir la misma línea.

Extiendo mi agradecimiento a cada uno de los docentes que fueron partícipes de mi formación, les agradezco de forma sincera su dedicación durante cada curso. Gracias por no solo enseñar y compartirnos sus conocimientos, sino también por brindarnos sabiduría a través de su experiencia, disciplina y pasión por formar profesionales. Sus enseñanzas dejaron una gran huella en mi persona y me ayudaron a crecer de forma personal y profesional.

A cada uno de los compañeros que me topé en cada paso que daba, cada curso al que asistí, gracias por caminar a mi lado. Compartimos grandes vivencias, largas jornadas de estudio, proyectos en común, ideas, desafíos y aprendizajes que hoy se convierten en recuerdos valiosos para mi alma. En cada uno de ellos, encontré no solo compañerismo, sino que descubrí lo valioso y certero que es el trabajar en equipo y unir fuerzas para concretar una meta en común.

Por último, agradezco enormemente a Dios, quien nunca me negó su mano, siempre se mantuvo a mi lado, fue gran fuente de fortaleza, sabiduría y esperanza, le agradezco por dotarme de la capacidad para salir victorioso de cada desafío al que me enfrentaba. Le doy las gracias de bendecirme con la salud, fe y determinación necesaria para culminar este grandioso logro.

CAPÍTULO I. INTRODUCCIÓN

Planteamiento del problema

La empresa en la que se realiza el desarrollo del sistema web lleva por nombre All Sport Nutrition y se encuentra ubicada en el cantón de Grecia, dentro del territorio de la provincia de Alajuela. Su propósito es satisfacer las insuficiencias nutricionales de las personas mediante la disposición de suplementos alimenticios al público, lo que incluye la venta de estos productos para el nicho del *fitness*, la salud y el bienestar en general.

En el estudio realizado en la empresa, se observó que esta presenta ciertos problemas que es necesario mencionar. A través del sistema se busca brindar una solución integral al proceso de nómina, el cual se realiza manualmente, lo que genera errores en los resultados contables debido a fallos en los cálculos y en la gestión de los datos que intervienen en la definición total de la nómina. La falta de un registro adecuado de los días de vacaciones asignados a cada colaborador, así como del saldo restante tras su disfrute, provoca inconsistencias cuando el Área de Recursos Humanos gestiona el uso de este derecho. La ausencia de un control adecuado sobre las incapacidades tramitadas por cada persona colaboradora impide tener una visión clara de su frecuencia y duración, lo que dificulta su gestión por parte del Área de Recursos Humanos.

Objetivos

Objetivo general. Desarrollar una aplicación web para la gestión de la remuneración en la empresa All Sport Nutrition.

Objetivos específicos

1. Analizar los requerimientos funcionales y no funcionales para el desarrollo del sistema web mediante el estudio de las necesidades de la empresa.
2. Diseñar el sistema web y las tablas de la base de datos para el almacenamiento correcto de datos mediante una herramienta de maquetado como MySQL Workbench.
3. Programar cada uno de los módulos para que el sistema web funcione correctamente mediante el uso de herramientas tecnológicas relevantes.

4. Probar cada una de las funcionalidades del sistema web para el funcionamiento eficaz del sistema mediante pruebas íntegras e individuales.

Justificación

El desarrollo del sistema web busca satisfacer las necesidades de una empresa cuyos procesos contables y de gestión de nómina no se encuentran automatizados ni son certeros en comparación con lo que se requiere para gestionar de manera integral la remuneración de las personas colaboradoras. De este modo, se logra realizar cálculos coherentes y precisos con los datos involucrados al gestionar la nómina individual de cada uno de los integrantes de la compañía All Sport Nutrition.

Se plantea esta solución para que la empresa que se mencionó obtenga una herramienta que le permita gestionar de manera automatizada el pago de salarios a las personas colaboradoras de forma justa y optimizada, lo que facilita una administración correcta de los recursos humanos y económicos y garantiza el cumplimiento de los lineamientos legales que se relacionan con este tema. Además, se busca mantener una convivencia organizacional sana y de acuerdo con los valores que definen a la compañía y que toda institución busca para ser más productiva y mejorar sus operaciones.

En cuanto a la viabilidad, Hernández Sampieri (2017) argumenta que:

La viabilidad de la investigación significa que es posible realizarla en un tiempo determinado y con los recursos disponibles. Es indispensable tener acceso al lugar o contexto donde se realizará la investigación (por ejemplo, un laboratorio o simulador) ya los datos e información que debemos recabar (p. 45).

Viabilidad técnica. Es posible llevar a cabo el desarrollo correcto del sistema web, ya que la empresa dispone de los recursos económicos suficientes para cubrir los gastos que se relacionan con el diseño, desarrollo, pruebas e implementación de este. Según lo analizado, la compañía cuenta con la estructura física necesaria en el ámbito tecnológico para hacer viable la propuesta, pues las instalaciones están adecuadas con los recursos de *hardware* requeridos.

Cabe destacar que el desarrollo de la solución programable está pensado para que sea económico y escalable a largo plazo, mediante el uso de VPS y las tecnologías en la nube con las

que se cuenta hoy en día, lo que permite evitar un gran gasto en *hardware*. Las licencias de herramientas como Visual Studio y el *hosting* por utilizar se encuentran dentro del presupuesto de la compañía. Por lo tanto, no se esperan problemas en el futuro en este aspecto.

En primera instancia, se planea ejecutar y desplegar la solución en el equipo VPS mencionado, cuyas características de *hardware* ofrecen una capacidad de ancho de banda inicial con una perspectiva de expansión futura de 1 TB mensual, entre 4 y 8 vCores de procesamiento de CPU, 8 a 16 GB de memoria RAM, 256 GB de almacenamiento en un disco de estado sólido y el sistema operativo es una distribución de Linux conocida como Ubuntu Server, específicamente la versión 22.04 LTS. En caso de ser necesario, se considera adquirir almacenamiento adicional a largo plazo a través de servicios en la nube como Amazon S3, Google Cloud Storage o Azure Blob Storage, ya que esto brinda una oportunidad significativa para realizar copias de seguridad automáticas, tanto de la base de datos como del sistema en general y de otra información si así se requiere.

Viabilidad operativa. Para el uso de este sistema, se espera que los usuarios involucrados no necesiten tener altos conocimientos tecnológicos para su uso correcto. Por ende, se prevé que cualquier colaborador que deba utilizar la solución tecnológica pueda emplearla de manera óptima y adecuada, sin que se requiera mucho tiempo de capacitación.

Por otra parte, está claro que si se habla de gestión y pagos de nómina, el personal debe tener estudios o habilidades que estén asociados a lo que se busca y, de esta manera, se logre aumentar los niveles de productividad y eficiencia en este proceso organizacional a través del sistema, mitigando los riesgos que implica el cálculo manual de estas cifras.

Asimismo, se espera que no se reduzca el personal de la institución, ya que lo que se pretende es que el personal agilice sus tareas, lo que les permite una mejor utilización del tiempo. Lo anterior tiene el fin de delegar sus funciones a otras actividades que no sean tan repetitivas o complejas, como los cálculos de una nómina y así reducir el margen de error humano cuando se trata la gestión manual de una operación tan delicada.

Viabilidad económica. A través de la Tabla 1 se busca mostrar el análisis de los gastos que se relacionan con el diseño, desarrollo y realización de la propuesta que se planteó. Se detalla cada uno de los gastos y el posible costo aproximado en el que incurriría la empresa:

Tabla 1
Detalles de posibles gastos por incurrir

Categorías de gasto	Descripción	Horas estimadas	Posible costo estimado
Análisis y diseño del sistema	Levantamiento de requerimientos, modelado de procesos, diseño de interfaces	10 h	No se estima un costo, puesto que es parte de las actividades del estudiante.
Desarrollo web	Programación del sistema (<i>back-end + front-end</i>), integración de módulos	30 h	No se estima un costo, puesto que es parte del estudiante.
Pruebas y control de calidad	Pruebas unitarias, de integración y de usuario	12 h	No se estima un costo, puesto que es parte de las actividades del estudiante.
Infraestructura en la nube (VPS)	Servidores virtuales, base de datos, copias de seguridad	-	\$500 – \$1,200 anuales, con base en los requerimientos y necesidad actuales del sistema.
Licencias de <i>software</i>	Visual Studio, herramientas de desarrollo, posibles librerías adicionales	-	\$300 – \$800
Hosting y dominio	Alojamiento web + nombre de dominio	-	\$150 – \$300 anuales
Capacitación de usuarios	Talleres básicos para el personal involucrado	5 h	No se estima un costo, puesto que es parte de las actividades del estudiante.
Soporte y mantenimiento	Actualizaciones, soporte técnico, corrección de errores	10 h	No se estima un costo, puesto que es parte de las actividades del estudiante.
Contingencias	Margen para imprevistos (10–15 % del presupuesto total)	-	

Cabe mencionar que el presente proyecto de desarrollo de la solución web se relaciona con un proyecto de graduación. Por lo tanto, los rubros correspondientes al análisis, diseño, desarrollo, pruebas, capacitación y soporte de la propuesta que se planteó no generan costos para la empresa. Esto se debe a que son actividades académicas que el estudiante debe ejecutar.

Se estima que el costo mínimo para la realización de la propuesta tecnológica comienza en la suma de \$1,250 USD y asciende a un costo máximo aproximado de \$2,800 USD. Esto se refiere a cifras aproximadas, considerando la fecha en la que se redactó este documento. Otros gastos que

se relacionan con el *hardware* no son necesarios, ya que la empresa dispone del equipo necesario para la implementación del sistema y las demás configuraciones que se deben realizar.

Viabilidad legal. La viabilidad legal de este proyecto radica en el cumplimiento de la normativa legal establecida dentro del Código de Trabajo, la Caja Costarricense de Seguro Social (CCSS), el Instituto Nacional de Seguros (INS) y la normativa tributaria vigente. El proyecto debe tomar en consideración las limitaciones y normas determinadas por estos entes y seguir cada una de las regulaciones establecidas por el gobierno, de manera que el cálculo de salarios, deducciones y demás se haga de forma correcta y de acuerdo con lo que dicta la ley.

El proyecto espera automatizar el proceso general relacionado con los cálculos necesarios, mejorando la productividad de las personas colaboradoras que se involucrarán en estas acciones. De este modo, se optimizan mejor los recursos y los colaboradores confiarán en la forma en la que se les remunera, mientras que la empresa seguirá cada uno de los lineamientos establecidos por los reguladores.

En primer lugar, se consideran las disposiciones dictadas en la Ley n.º 8148, que hace referencia a los arts. 196 BIS, 217 BIS y 229 BIS del Código Penal, relativas al acceso indebido a sistemas informáticos y a la protección contra la manipulación indebida de datos. El sistema contempla tecnologías de autenticación y seguridad digital que previenen accesos no autorizados, protegiendo la información de la entidad y de las personas colaboradoras.

De igual manera, se asegura el cumplimiento de la Ley n.º 4573 de 2001, que trata sobre la sanción a los delitos informáticos, mediante la implementación de protocolos de seguridad, registros de actividad y encriptación de datos sensibles, lo que mitiga los riesgos de fraude y manipulación ilegal de información salarial.

Por otra parte, se cumple con lo establecido en la Ley de Derechos de Autor n.º 6683 (1982), ya que el sistema respeta las licencias de *software* que se utilizan en su desarrollo y evita el uso no autorizado de programas o herramientas que vulneren los derechos de propiedad intelectual.

Finalmente, se incorpora lo estipulado en la Ley n.º 8968 sobre la protección de la persona frente al tratamiento de sus datos personales y se asegura que toda la información relacionada con las personas colaboradoras (salarios, deducciones, aportes y demás datos privados) se gestione

bajo principios de confidencialidad, consentimiento informado y uso exclusivo para fines laborales y legales.

Proyecciones

Alcance funcional. El sistema web que se planteó toma en cuenta ciertos módulos para garantizar una gestión correcta, tanto administrativa como contable en la empresa. Cada uno de los módulos definidos representa una solución a una necesidad determinada, la cual se detecta en la etapa de levantamiento de requisitos, con el objetivo de que se asegure la optimización de cada uno de los recursos de la compañía y una integración correcta de los procesos internos. En la Tabla 2 se detalla cada uno de los módulos por desarrollar, los cuales están acompañados de una breve descripción de su funcionalidad principal:

Tabla 2
Módulos por implementar en el sistema

Nombre del módulo	Descripción del módulo
Gestionar nóminas	Este módulo permite administrar de forma integral la información contable requerida para llevar a cabo el cálculo preciso de la nómina del personal de la empresa.
Gestionar horas extra	Este módulo tiene la función de llevar un control adecuado del registro de las horas extra trabajadas por cada colaborador. La persona colaboradora debe enviar su solicitud a través del sistema a su jefatura inmediata. Si esta la aprueba, el sistema notificará a recursos humanos, quien a la vez informa al colaborador mediante el mismo sistema. En caso de que la Jefatura no apruebe la petición, el sistema notificará al colaborador que esta ha sido rechazada.
Gestionar vacaciones	Este módulo es responsable de administrar y calcular los días de vacaciones que le corresponden a cada colaborador, lo que lleva un historial del disfrute y del saldo de días disponibles. La persona colaboradora debe enviar su solicitud a través del sistema a su jefatura inmediata; si esta la aprueba, el sistema notificará a recursos humanos, quien luego informa al colaborador mediante la misma plataforma. En caso de que la petición sea rechazada por la Jefatura, el sistema notificará al colaborador sobre la negativa.
Calcular aguinaldo	Este módulo tiene la función de calcular el pago del aguinaldo para cada colaborador, considerando las normativas laborales vigentes aplicables a dicho concepto.
Gestionar incapacidades	Este módulo tiene la función de registrar las incapacidades gestionadas por cada colaborador, ya sea por enfermedad o accidente, lo que incluye la

	información relativa a la duración en días de cada incapacidad.
Gestionar permisos	Este módulo se encarga de gestionar el control de los permisos laborales solicitados por cada colaborador, registrando si dichos permisos se otorgan con o sin goce de salario. La persona colaboradora debe ingresar su solicitud a través del sistema y enviarla a su jefatura inmediata; si esta la aprueba, el sistema notificará a recursos humanos, quien a la vez informa al colaborador mediante la misma plataforma. En caso de que la petición sea rechazada por la Jefatura, el sistema notificará al colaborador sobre dicha decisión.
Gestionar asistencia	Este módulo es responsable de registrar y gestionar los horarios de ingreso y salida de cada colaborador.
Gestionar liquidación	Este módulo es responsable de ejecutar todo el proceso para gestionar el pago de la liquidación de cada colaborador, considerando las condiciones salariales establecidas en sus contratos laborales y las leyes nacionales aplicables.
Evaluar el rendimiento	Este módulo tiene la función de definir métricas de desempeño para cada colaborador, creando una herramienta de evaluación que permita valorar su rendimiento y facilite la asignación de diferentes bonificaciones.
Módulo de mantenimiento	Este módulo se encarga de realizar el borrado, inserción, modificación, actualización de datos
Módulo de consultas	Este módulo se encarga de generar información proporcionada de las diferentes tablas
Módulo de reportes	Este módulo se encarga de generar información proporcionada de las diferentes tablas y procesos, pero con un formato específico, según lo solicite la persona usuaria. Podrá ser impreso o por pantalla
Módulo de seguridad	Este módulo se encarga de realizar la autenticación de contraseñas y definición de perfiles.

Alcance metodológico. Este proyecto de investigación tiene como objetivo desarrollar un sistema basado en el desarrollo web, cuyo propósito es gestionar de forma adecuada el pago de nómina para la institución que se mencionó. En este caso se optó por utilizar la metodología ágil Scrum. Según Capuñay Uceda y Antón Pérez (2021), es posible definir SCRUM como: “La metodología de desarrollo ágil se refiere a una estrategia iterativa e incremental que involucra equipos autoorganizados que trabajan en colaboración para crear el software” (párr. 6), ya que este marco de trabajo permite dar una respuesta más flexible a los posibles cambios en los requerimientos y funcionalidades del sistema, lo que facilita la entrega de un producto de calidad y confianza.

A diferencia de otros enfoques que se utilizan en el ámbito profesional y catalogados como tradicionales, Scrum permite mantener una comunicación más fluida y constante entre los usuarios que emplean el aplicativo y las personas que se encargan del desarrollo de este. Tal como señalan Velasco *et al.* (2021): “Scrum gestiona estas iteraciones a través de reuniones diarias, uno de los elementos fundamentales de esta metodología” (p. 5). Esto permite que dicha solución evolucione conforme a los requisitos que la empresa exponga y, de este modo, se pueda brindar una mejor respuesta a esas necesidades.

Si se habla acerca del ciclo de vida del *software*, Scrum desempeña un papel muy importante en este tema, ya que cada *sprint* equivale a un pequeño ciclo que incluye acciones de análisis, diseño, implementación y pruebas de las nuevas características por desarrollar. Al tomar las palabras de Velasco *et al.* (2021):

Al usar la metodología Scrum/XP para el desarrollo de proyectos, permite introducir un conjunto de prácticas y métodos que mejoran el rendimiento de un equipo, que se refleja de forma notable en la evolución de un producto software y su calidad (p. 9).

De esta manera, la empresa no necesita esperar hasta el final del desarrollo del *software* para validar si todo está saliendo según lo planeado, sino que puede notar los avances de forma progresiva y decidir si el desarrollo del módulo cumple con lo solicitado. Además, identificar todo aquello que se encuentre erróneo o no alineado con los objetivos del proyecto.

Alcance tecnológico. La arquitectura de *software* representa la estructura a alto nivel de un sistema, además, suele ser una decisión clave, ya que debe satisfacer las necesidades del sistema, considerando los requerimientos funcionales y no funcionales del mismo (Tian, 2022, citado por Palomeque-Zambrano y Campoverde-Molina, 2023). El desarrollo del sistema web para el proyecto de pago de nómina fundamenta su construcción en un conjunto de tecnologías modernas que buscan cumplir con los requisitos de la institución, considerando la escalabilidad, la seguridad y la facilidad de mantenimiento, así como futuras mejoras para el sistema.

Como primer punto para mencionar, se utiliza un enfoque de desarrollo web, empleando un modelo cliente-servidor, con el cual se pretende que los usuarios puedan acceder al sistema desde cualquier navegador actual, sin necesidad de instalar paquetes u otras herramientas de *software* en sus dispositivos electrónicos. El diseño de las pantallas y otras interfaces se implementa mediante

tecnologías como React y otros *frameworks* con base en el lenguaje de programación web JavaScript y CSS3, los cuales garantizan una grata experiencia de usuario y, a la vez, compatibilidad con cientos de dispositivos. En cuanto al desarrollo de la lógica de negocio, se emplean C# y .NET como las principales herramientas, ya que ofrecen un gran rendimiento y versatilidad para mantener el desarrollo de una aplicación escalable y capaz.

En términos de almacenamiento y bases de datos, este aspecto se orienta al uso del motor SQL Server, el cual permite mantener la información estructurada de forma segura y organizada. El diseño de la base de datos garantiza la persistencia de la información incluso en caso de fallos técnicos e incluye la función de respaldos automáticos.

Con respecto a la infraestructura, los VPS en la nube desempeñarán un papel clave en el despliegue del *software* desarrollado, ya que ofrecen escalabilidad, disponibilidad y reducción de costos en comparación con mantener y adquirir un servidor local.

CAPÍTULO II. MARCO REFERENCIAL

Importancia de la digitalización en la gestión empresarial

El desarrollo tecnológico ha llevado a otro nivel la forma en la que las empresas gestionan sus procesos internos, logrando una transformación completa. En el contexto en el que se encuentra hoy en día, la globalización participa en los mercados crecientes y en la competencia que existe afuera. Por esto, cuando se habla de la digitalización empresarial, es importante mencionar que constituye un elemento fundamental para que las organizaciones crezcan y se mantengan sostenibles. Ames *et al.* (2023) explican que:

La digitalización empresarial no solo implicó la adopción de nuevas tecnologías, sino también un cambio profundo en la mentalidad organizacional. Las empresas que lograron adaptarse más rápido fueron aquellas que ya habían iniciado procesos de transformación antes de la pandemia (p. 63).

El que una empresa adopte una herramienta digital como solución a procesos mecánicos y, a veces, propensos a errores, le permite no solo optimizar sus operaciones, sino también aumentar la probabilidad de lograr una mayor capacidad de respuesta frente a la toma de decisiones y a los cambios en el contexto en el que pueda verse involucrada la compañía y los mercados. Por consiguiente, la digitalización en la gestión empresarial no debe entenderse únicamente como una tendencia tecnológica, sino como una estrategia esencial para la supervivencia y el crecimiento competitivo.

Contexto costarricense en la adopción de sistemas web. A través de los años, Costa Rica ha experimentado de forma acelerada un proceso de transformación digital. Este impulso no solo se ha dado por méritos del sector privado, sino que también han influido las políticas públicas orientadas a la mejora tecnológica. Según el Ministerio de Ciencia, Innovación, Tecnología y Telecomunicaciones (Micitt) (2022), el país se ha esforzado por posicionarse como un referente en tecnología dentro de América Latina, logrando atraer inversionistas extranjeros en el sector de servicios con base en tecnología y, de esta manera, proyectar la modernización digital de las pequeñas y medianas empresas que operan en el territorio nacional.

En el entorno empresarial, la necesidad de gestionar procesos de forma eficiente, competitiva y productiva ha provocado que el uso de sistemas web sea una realidad cada vez más presente en la actualidad. Por ende, el uso de estas herramientas digitales aumentó año tras año. Santander-Salmon *et al.* (2023) señalan que: “La resistencia al cambio organizacional es uno de los desafíos más persistentes en los procesos de transformación digital. Este fenómeno ocurre cuando empleados y líderes muestran actitudes de rechazo hacia las modificaciones en los métodos tradicionales de trabajo” (p. 42).

Empresas de distintos sectores, desde pequeños comercios hasta grandes compañías o industrias multinacionales, han tomado la iniciativa de implementar soluciones digitales no solo de forma local, sino también guiadas por tecnologías en la nube, lo que permite mejorar la administración en procesos como la facturación electrónica, la gestión de clientes, los inventarios e incluso las nóminas.

Ahora bien, para responder a la posible pregunta de qué aceleró el uso de sistemas digitales en los últimos años en Costa Rica, es posible mencionar lo establecido por la Dirección General de Tributación a partir de 2018: la obligatoriedad de la factura electrónica, lo cual provocó que las empresas, sin excepción, debieran adoptar aplicativos digitales para sus operaciones financieras. Por otra parte, Ames *et al.* (2023) señalan que: “La crisis sanitaria ha generado un profundo impacto en todas las esferas de la sociedad, incluyendo la forma en que las empresas operan y se adaptan a un entorno empresarial en constante cambio” (p. 62).

La pandemia vivida de la COVID-19 en el año 2020 causó que la digitalización tuviera cierta propulsión a las empresas en la necesidad de derivarse al teletrabajo y la gestión de forma remota de los procesos internos de cada empresa. Es por eso por lo que los sistemas web han demostrado ser no solo útiles, sino que también son herramientas imprescindibles para garantizar la competitividad de los negocios ante los mercados.

En síntesis, Costa Rica presenta no solo una gran viabilidad en el uso de los sistemas web como soluciones digitales a la gestión de los procesos internos de las empresas, sino que también es una forma de participar en la decisiva digitalización empresarial que se vive en el país y que continúa su ritmo de manera potenciada. Se argumenta esto porque existe la infraestructura tecnológica como la necesidad empresarial para conseguir optimizar y automatizar los procedimientos administrativos.

Rol estratégico de la automatización en la gestión de recursos humanos. Barrera *et al.* (2024) indican lo siguiente:

La automatización hoy en día es muy usual en el ámbito empresarial, dado el importante aporte que este brinda tanto a la productividad como a la eficiencia. Generalmente se adapta a la línea específica de procesos con los que cuentan las organizaciones, contribuyendo a que los trabajadores de las diversas áreas alcancen un óptimo nivel productivo, de modo que todo surja de forma rápida, con mejores resultados (p. 6204).

Es clave mencionar que uno de los procesos que más se ha beneficiado de la digitalización empresarial es la gestión del talento humano. Tiempo atrás, algunas de las actividades o tareas que debían realizar las personas que se encargan de esta gestión eran calcular los salarios, registrar la asistencia y las ausencias y verificar el cumplimiento de las obligaciones legales, entre otras. Ahora bien, cuando estas tareas se realizan de forma manual, existe la posibilidad de incurrir en errores, se consume más tiempo y suelen ser muy repetitivas, lo que provoca, de cierta manera, que los costos tiendan a elevarse.

De acuerdo con Segura (2025): “La gestión del talento humano, históricamente centrada en procesos administrativos y de soporte, se ha convertido en un eje estratégico clave para afrontar los desafíos de la competitividad, la innovación y la sostenibilidad organizacional” (párr. 1).

Actualmente, automatizar procesos permite incrementar la productividad, la efectividad en la ejecución y, además, reducir costos cuando sea posible, ya que utilizar un sistema que haga estas funciones representa una renovación estratégica. La gestión de recursos humanos, hoy en día, debe centrarse en el bienestar de cada colaborador de la empresa, sin olvidar su desarrollo profesional y personal, delegando las tareas monótonas y repetitivas a una solución tecnológica que se encargue de ellas. Gracias a esto, los departamentos responsables de gestionar el recurso humano pueden dedicar más tiempo a aquellas actividades que aportan mayor valor a la empresa, como la motivación y la retención del talento.

En cuanto a la gestión de nómina, automatizar este u otro proceso relacionado con esta área facilita el cálculo eficiente y eficaz de los salarios, las deducciones, los aportes patronales y los beneficios adicionales que presenten las personas colaboradoras, velando por el cumplimiento de lo que las leyes costarricenses indiquen. Además, contribuye a la posibilidad de generar reportes

que detallen información relevante, lo que permite una mayor transparencia interna y fomenta la confianza de los colaboradores en la ejecución de los procesos en la empresa.

Automatizar procesos no significa solamente que un proceso mecánico es más eficiente y productivo, sino que también permite que las empresas fortalezcan su competitividad empresarial, contribuye a mantener una sana convivencia entre las personas colaboradoras y logra que mejoren el pensamiento y el desarrollo innovador.

Conceptos fundamentales

Nómina. La nómina es uno de los ejes principales dentro de la gestión de recursos humanos, ya que constituye el registro detallado de los pagos y deducciones correspondientes a las personas colaboradoras de una empresa. Tal como lo plantean Hurtado Díaz y Samboni Bolaños (2021): “Se conoce por nómina los pagos mensuales o quincenales que el empleador o empresa debe realizar a los trabajadores que tiene vinculados mediante contrato de trabajo, pagos que comprenden el salario, comisiones, horas extras, recargos nocturnos” (p. 19). Por otra parte, Prats *et al.* (2021) señalan que: “La nómina es un parte fundamental para las empresas debido a que con ella se lleva un control sobre los pagos salariales que se realizan a sus empleados y así conocer de manera detallada el gasto que esta implica” (p. 6).

En general, la nómina tiene dos propósitos principales: por un lado, cumple la función de que la persona colaboradora reciba de forma justa y clara su remuneración económica. Por otro, brinda la posibilidad de que la empresa mantenga un control no solo contable, sino también administrativo y legal de todos los costos que se relacionan con el recurso humano. Por esto, no es posible afirmar que la nómina es solamente una lista o un reporte de pagos, sino que constituye un método que debe cumplirse con exactitud, de manera legal y manteniendo el criterio de confidencialidad.

Componentes de la nómina. La nómina está formada por una serie de elementos que hacen posible un reconocimiento justo y en sintonía con la legislación laboral actual. Mediante la Tabla 3 se muestran las principales partes que conforman la nómina:

Tabla 3
Principales componentes de la nómina

Nombre	Descripción
Salario base	Es la retribución fija que cada colaborador recibe por los servicios que se brindan, este se ve determinado por el contrato laboral y se ajusta al salario mínimo establecido por las entidades del Estado. En Costa Rica, el ente encargado de las regulaciones periódicas de los salarios mínimos es el Consejo Nacional de Salarios, el cual vela por el cumplimiento de estos de acuerdo con la categoría ocupacional del trabajador.
Deducciones	Las deducciones se definen como los descuentos aplicados sobre el monto del salario bruto del empleado. Entre estas se pueden encontrar los aportes obligatorios a la Caja Costarricense de Seguro Social (CCSS), cuotas al fondo de pensiones (ROP) y otros montos como impuestos, embargos judiciales o préstamos autorizados por el trabajador.
Prestaciones	Se refieren a los beneficios que puede obtener la persona colaboradora sin contar el salario, dichos beneficios pueden ser de carácter legal o existe la posibilidad de que estos sean dados por la organización. Algunos de los ejemplos más comunes son el aguinaldo, vacaciones, incentivos, bonos o beneficios en especie.
Cargas sociales	Son todos aquellos aportes que el patrono debe dar a diferentes instituciones del Estado, de manera prioritaria a la Caja Costarricense de Seguro Social (CCSS), el Instituto Nacional de Seguros (INS) y el Fondo de Desarrollo Social y Asignaciones Familiares (Fodesaf). Dichas cargas forman un sistema de financiamiento para distintos programas estatales y de seguridad social.

El balance conjunto de estos componentes asegura que la nómina cumpla su objetivo principal: brindar una remuneración al colaborador de forma justa, lo que promueve el cumplimiento de las garantías legales estipuladas por las leyes nacionales y el Código de Trabajo. Además de cumplir con las obligaciones no solo fiscales, sino también sociales.

Marco legal. Todo proceso o mecanismo relacionado con la nómina en Costa Rica está regulado por un marco jurídico sólido, cuyo propósito es proteger los derechos de cada uno de los trabajadores y regular las responsabilidades y obligaciones de los empleadores. El Código de Trabajo de Costa Rica (Ley n.º 2, 1943) constituye la base legal en materia laboral. Esta ley establece las disposiciones fundamentales que se relacionan con aspectos como el salario mínimo, las jornadas laborales, el pago de horas extra, las vacaciones, los despidos y las liquidaciones, así

como otros factores importantes que regulan la relación entre empleadores y los colaboradores de la empresa.

En lo que concierne a las contribuciones sociales, la Procuraduría General de la República (PGR) (s. f.) señala que la Ley Constitutiva de la Caja Costarricense de Seguro Social (CCSS) establece el marco legal que regula la administración y el financiamiento del sistema de seguridad social en Costa Rica, lo que incluye a los regímenes de salud y pensiones, así como las responsabilidades de los empleadores y trabajadores en el pago de las cuotas obrero-patronales.

El aguinaldo es otra de las obligaciones reguladas legalmente por la Ley n.º 2412. Esta ley estipula que el cálculo del aguinaldo equivale al promedio de los salarios ordinarios y extraordinarios devengados durante todo el año, debiendo pagarse como fecha límite el 20 de diciembre. Este beneficio constituye un derecho irrenunciable para las personas colaboradoras.

Por otra parte, las vacaciones están reglamentadas dentro del rango de artículos que abarcan del 154 al 161 del Código de Trabajo costarricense. Se estipula que todo colaborador que haya realizado sus tareas laborales de forma continua durante 50 semanas tiene derecho a un período de descanso remunerado de dos semanas. Con esto se busca que las personas colaboradoras descansen física y mentalmente, lo que contribuye a su bienestar general, con el objetivo de promover niveles óptimos de productividad y el equilibrio entre la vida personal y la vida laboral.

De manera resumida, la nómina en Costa Rica no solo representa una solución ante una necesidad administrativa, sino que también constituye un compromiso, tanto social como legal. Por esto, las empresas deben asumir la responsabilidad y el compromiso de calcular de forma precisa cada uno de los salarios de las personas colaboradoras de la organización, aplicar las deducciones obligatorias, reconocer las prestaciones y cumplir con las cargas sociales y demás disposiciones estipuladas por la ley. Es en este contexto donde se observa que utilizar un sistema web enfocado en la gestión de la nómina adquiere un papel muy importante como herramienta que facilita el cumplimiento de las normativas definidas y permite mantener la eficiencia en el manejo del recurso humano en la entidad.

Requerimientos de software

Cuando se desarrolla un sistema de información, los requerimientos de *software* participan activamente y resultan fundamentales como base para la estructura de la solución tecnológica que

se construye. En este caso es posible tomar las palabras de Zapana *et al.* (2021), quienes exponen lo siguiente:

Los requerimientos de software cumplen un papel primordial en el proceso de producción de software, ya que enfocan un área fundamental: la definición de lo que se desea producir. Su principal tarea consiste en la generación de especificaciones correctas que describan con claridad, sin ambigüedades, en forma consistente y compacta, el comportamiento del sistema; de esta manera, se pretende minimizar los problemas relacionados al desarrollo de sistemas (p. 2).

En cuanto al ejemplo de la empresa que se tomó para realizar el proyecto de investigación y el desarrollo del aplicativo web, llevar a cabo un estudio eficiente de los requerimientos fundamentales es importante y constituye uno de los principales pasos por ejecutar para garantizar que el sistema tecnológico de gestión de pago de nómina cumpla con las necesidades de la compañía, además, con los objetivos que se plantearon. De este modo, se logra la seguridad y eficiencia en el manejo de datos e información en el ámbito organizacional.

Requerimientos funcionales. Se pueden definir los requerimientos funcionales como todas aquellas operaciones definidas que el sistema debe poner en funcionamiento. Según Guerra (2021): “Los requerimientos funcionales son enunciados acerca de servicios que el sistema debe proveer, de cómo debería reaccionar el sistema a entradas particulares y de cómo debería comportarse el sistema en situaciones específicas” (p. 26). Al tomar el caso del sistema de gestión de nómina por desarrollar, las operaciones tienen como objetivo implementar la automatización como medio para procesar tareas administrativas importantes, enfocadas en el manejo del talento humano y el cálculo de las remuneraciones.

Registro de empleados. El sistema debe permitir ejecutar procesos como la creación, modificación y eliminación de los registros de empleados. Cada uno de los registros debe incluir información personal, como el nombre, identificación y dirección, así como datos laborales, como el puesto, salario, fecha de ingreso y departamento. Es importante que el aplicativo le permita al usuario almacenar información acerca de los beneficios y deducciones obligatorias con base en la ley y otros mecanismos que involucren estos cálculos. Además, es necesario destacar la importancia de este módulo, ya que constituye el corazón de la base de datos para generar posteriormente los cálculos en términos de nómina y demás.

Cálculo automático de planillas. Automatizar el cálculo de salarios es, si no la principal, una de las funcionalidades más importantes del sistema por desarrollar. El sistema debe considerar valores o datos como las horas laboradas, las deducciones por cargas sociales, impuestos, aguinaldo y vacaciones, para realizar el procesamiento correcto de las remuneraciones de cada una de las personas colaboradoras de la organización. Calcular la nómina de forma manual implica un alto margen de error, además, se considera una manera improductiva de llevar a cabo la tarea. Por esto, al automatizar computacionalmente el mencionado mecanismo de cálculo, se garantiza la exactitud de los montos y el cumplimiento de la normativa legal vigente en Costa Rica. Por ende, reducir el tiempo invertido en esta tarea y aumentar la transparencia en la manipulación de datos son algunos de los principales beneficios que ofrece la automatización.

Generación de reportes. Brindar la capacidad de generar reportes de forma detallada es una característica que se debe tener presente en el sistema, debido a que la persona usuaria final debe contar con información resumida y precisa de la nómina. Los reportes deben incluir el listado de planillas por período, comprobantes individualizados para cada colaborador y el resumen de las deducciones en el ámbito patronal. Por otra parte, es importante que al usuario se le permita exportar los reportes en formatos populares, como PDF o Microsoft Excel, lo que facilita la realización de auditorías internas y, en caso de que sea necesario, el envío de estos documentos a alguna entidad o institución, como la Caja Costarricense de Seguro Social (CCSS), sin contratiempos ni limitaciones tecnológicas o funcionales.

Control de usuarios y roles. Para que exista un uso correcto de la información es necesario que el sistema cuente con un mecanismo mediante el cual los usuarios puedan autenticarse y se gestione la asignación de roles con base en el acceso al que se ingresará y quién tiene dicho acceso. Un escenario de ejemplo que se puede plantear es el de un usuario administrador; se sabe con certeza que este es quien tiene el control y acceso absoluto a todas las funcionalidades o módulos del sistema. Por otro lado, un colaborador del Departamento de Recursos Humanos solo debe acceder a la gestión de planilla y reportes en el caso más común. Es claro que el acceso a los diferentes módulos según el rol debe ser estudiado y analizado para gestionar el uso correcto de los usuarios finales a las funcionalidades que se desarrollaron. Lo anterior tiene el fin de fortalecer el apartado de seguridad y mantener un eficiente flujo de trabajo de los departamentos con el sistema.

Requerimientos no funcionales. Todas aquellas características que no dictan exactamente las acciones o funciones que realiza el sistema, sino la forma en la que este debe ejecutarse, se conocen como requerimientos no funcionales. Según Guerra (2021): “Los requerimientos no funcionales son limitaciones sobre servicios o funciones que ofrece el sistema. Incluyen restricciones tanto de temporización y del proceso de desarrollo, como impuestas por los estándares” (p. 26). Estos son esenciales para lograr que el *software* mantenga una excelente calidad, sea amigable con el usuario y sea confiable.

Seguridad de la información. Implementar mecanismos robustos en términos de seguridad es importante, ya que es evidente que la información que el sistema manejará es sensible, puesto que entre esta se encuentran los datos personales de los colaboradores, además de datos de índole financiera. Entre estos mecanismos se pueden mencionar el cifrado de contraseñas; al ser un *software* basado en tecnología web, resulta fundamental utilizar protocolos HTTPS, así como establecer políticas de respaldo y recuperación de datos y controlar el acceso basado en roles, como se indicó. El cumplimiento de la Ley de Protección de la Persona Frente al Tratamiento de sus Datos Personales en Costa Rica también debe considerarse como marco normativo.

Usabilidad y accesibilidad. El usuario final debe contar con interfaces comprensibles y cómodas que hagan posible una interacción acogedora con cada una de las funcionalidades, lo que le permite ejecutar las acciones con base en las utilidades que presenta el aplicativo, independientemente del nivel de experiencia tecnológica que tenga. Por otra parte, el sistema responsivo es fundamental, ya que permite que las interfaces se adapten a las dimensiones del dispositivo, como computadoras, tabletas o teléfonos inteligentes, en los que se utilice el sistema, cumpliendo con los principios del diseño responsivo. No se debe pasar por alto a las personas que presenten una limitación visual o motora, por lo tanto, es necesario cumplir con buenas prácticas de diseño, siguiendo los lineamientos como las Pautas de Accesibilidad para el Contenido Web (World Wide Web Consortium [W3C], 2023).

Escalabilidad y rendimiento. De acuerdo con Colombero *et al.* (2024) es posible incorporar al conocimiento que:

A medida que el proyecto crece en tamaño y complejidad, las limitaciones de las arquitecturas monolíticas pueden volverse más evidentes. La necesidad de implementar y desplegar la aplicación completa en cada actualización puede resultar en tiempos de

desarrollo más largos y un ciclo de lanzamiento más lento. A su vez, la escalabilidad de una aplicación monolítica puede ser limitada, ya que agregar recursos adicionales al servidor único puede no ser suficiente para manejar un aumento en la carga de trabajo (p. 48).

Estas palabras de Colombero *et al.* (2024) indican que, a medida que una empresa se desarrolla y sus operaciones crecen, el sistema debe considerar este evento y adaptarse al progreso corporativo. Esto implica gestionar el aumento de las personas colaboradoras, las transacciones y la generación de reportes sin presentar posibles carencias en cuanto a eficiencia ni en la velocidad con que se procesan las operaciones. Por otro lado, los tiempos de respuesta deben estar optimizados, con el objetivo de que la realización de los cálculos de planillas o la generación de reportes se lleve a cabo de forma precisa en el menor tiempo posible, incluso en contextos en los que se presenten niveles altos de demanda en el sistema.

A partir de lo anterior es posible concluir que el apartado de requerimientos de *software* es uno de los primeros pasos para desarrollar el sistema web de gestión de nómina expuesto. A la vez, se define que los requisitos funcionales sirven como guía, con la finalidad de saber qué debe hacer el sistema en sí; por otra parte, los no funcionales garantizan la calidad y confiabilidad de la herramienta. Tener un conocimiento claro de ambos términos asegura que el sistema por desarrollar no solo tiene como objetivo cumplir con las necesidades que se estudian de la empresa, sino que también favorece la superación de posibles desafíos en el futuro.

Bases de datos en sistemas de nómina

Cuando se habla de pasos importantes en el desarrollo de soluciones informáticas, diseñar y administrar bases de datos es un aspecto crucial, ya que al gestionar información sensible o crítica, como la nómina de una empresa, resulta prioritario garantizar la confidencialidad, integridad y disponibilidad de los datos. Una base de datos funciona como el núcleo a través del cual se almacenan, organizan y procesan los datos de las personas colaboradoras y las remuneraciones de cada una. En relación con esto, la eficiencia de las operaciones que se ejecutan, así como la fiabilidad y la seguridad de la compañía, se logran mediante un sistema de gestión adecuado.

Definición de la base de datos relacional. Según Esquivel y Sevilla (2021, citando a Coronel, 2010 y Date, 2001), las bases de datos son parte fundamental de los sistemas informáticos. Coronel (2010) define una base de datos como:

Una estructura informática compartida e integrada que almacena una colección, mientras que Date considera a las bases de datos como una especie de armario electrónico para archivar; es decir, un depósito o contenedor de una colección de archivos de datos computarizados (p. 3).

En el ámbito empresarial, las bases de datos en su forma relacional predominan en el mercado, ya que una de sus características más destacables es la capacidad de estructurar la información de manera flexible y organizada.

Siguiendo la idea, una tabla es un conjunto de datos estructurados en filas y columnas; cada una de las filas de la tabla corresponde a un registro y cada columna recibe el nombre de un atributo. En el escenario de una base de datos para el sistema de nómina, existe la posibilidad de que haya una tabla denominada *empleado*, con atributos (columnas) como *id_empleado*, nombres, puesto, *salario_base*, entre otros atributos adicionales que pueden crearse dentro de la tabla. Esto significa que cada una de las filas de esta estructura de datos representaría la información de un trabajador específico de la empresa.

Las bases de datos relacionales utilizan claves primarias y claves foráneas. Las claves primarias (*primary key*) funcionan como identificadores únicos que permiten diferenciar de forma precisa un registro o fila dentro de una tabla; por ejemplo, se puede emplear el número de identificación de un empleado. Por otra parte, las claves foráneas (*foreign keys*) ofrecen al desarrollador la oportunidad de establecer una relación entre dos o más tablas diferentes, actuando como vínculos entre entidades. Por ejemplo, si se menciona la tabla *empleado* y, mediante una clave foránea, se relaciona con una segunda tabla como *Pago*, cada una de las transacciones que se relacionan con el salario se asociará al colaborador correspondiente. Por esto, definir relaciones permite evitar redundancias y favorece, tanto la consistencia como la escalabilidad del sistema.

Herramientas de diseño de bases de datos. El uso de herramientas que permiten al desarrollador diseñar una base de datos, crear los esquemas y visualizar las relaciones entre entidades es imprescindible. En este ámbito, MySQL Workbench es una de las herramientas que más se utilizan en el mundo actualmente para el modelado de bases de datos relacionales. Esta aplicación la desarrolló la empresa Oracle y posibilita que tareas como la generación de diagramas entidad-relación (ER), la normalización de tablas y el diseño de esquemas gráficos se realicen de una forma fácil y productiva.

Volviendo al contexto del sistema de nómina, modelar la base de datos para este programa facilita la identificación de las entidades clave como *empleado*, *Departamento*, *nómina*, *pago*, *deducción*, entre otras. El uso de los diagramas entidad-relación simplifica demostrar cómo un empleado está vinculado a un departamento, recibe pagos salariales en distintos períodos y está sujeto a deducciones determinadas, como las cargas sociales o las retenciones fiscales. El diseño gráfico de estas relaciones brinda una perspectiva más clara de la lógica detrás del sistema y cumple el papel de guía fundamental para el posterior desarrollo del código de los módulos que se implementan en la solución tecnológica.

Se desea ejemplificar de forma práctica ante el usuario, como una manera de condensar la información anteriormente expuesta, la creación de la tabla *Pago*, cuyos posibles atributos pueden ser: “id_pago” como clave primaria, “id_empleado” como clave foránea, “monto_bruto”, “deducciones” y “monto_neto”. Al tomar como referencia la clave foránea es lógico afirmar que esta tabla se relaciona con la entidad *empleado* mediante el atributo “id_empleado”. Esto garantiza que cada registro de pago se vincule con un trabajador que exista dentro de la otra tabla o, en general, en la base de datos. Por consiguiente, crear estas relaciones tiene como beneficio, entre otros, generar reportes automáticos y consolidar cálculos precisos sin duplicar la información.

Integridad y seguridad de datos. Trabajar con información confidencial relacionada con las personas colaboradoras dentro de una organización tiene como consecuencia factores críticos, como la integridad y seguridad de los datos en lo que respecta a la gestión de sistemas de nómina. Desde el punto de vista de la integridad es importante que los datos almacenados mantengan un grado de coherencia y consistencia en todo momento. Implementar restricciones, como claves primarias de origen único, claves foráneas válidas y diferentes reglas de validación, impide que la persona usuaria ingrese datos de forma incorrecta o duplicada, logrando la coherencia y consistencia de la información.

En cuanto a la seguridad, es conveniente implementar políticas y tecnologías que velen por la confidencialidad de la información. No todos los usuarios pueden ni deben acceder a toda la información almacenada en la base de datos. Por este motivo, los sistemas modernos deben incorporar mecanismos de gestión de accesos, de manera que solo aquellos usuarios a quienes se les defina la autorización puedan acceder a datos sensibles y, de ser necesario, modificarlos en la medida correspondiente. Cabe añadir que aplicar técnicas de cifrado y ponerlas en práctica es un

medio adecuado para reforzar la protección ante el acceso no autorizado o los ataques informáticos que puedan ocurrir. De acuerdo con Neninger *et al.* (2024):

Las empresas deben adaptarse a las infraestructuras tecnológicas locales y a las amenazas específicas de cada área. Por ejemplo, en regiones propensas a desastres naturales como terremotos o huracanes, contar con copias de seguridad fuera del sitio es una práctica esencial (p. 69).

Esta apreciación de Neninger *et al.* (2024) no muestra el respaldo de datos como un aspecto adicional que deba cuestionarse. La nómina es un registro histórico de gran importancia, tanto para la organización como para las personas involucradas en ella. Por ende, establecer métodos de copias de seguridad de forma periódica facilita recuperar la información en caso de que el sistema presente algún tipo de fallo, ocurra un desastre natural o incluso se produzca un incidente de ciberseguridad. Es recomendable almacenar estas copias de seguridad en servidores externos o utilizar la tecnología de almacenamiento en la nube para incrementar la disponibilidad de los datos.

La base de datos de un sistema de nómina no es solo un lugar donde se almacenan datos o información, sino una de las partes más relevantes del sistema, por lo que debe planificarse cuidadosamente. Además, debe cumplir con reglas de precisión, seguridad y capacidad de crecimiento. Empresas como All Sport Nutrition pueden realizar un seguimiento de cuánto dinero pagan a sus socios mediante un sistema que utiliza tablas y reglas. Asimismo, pueden utilizar herramientas como MySQL Workbench para crear y modificar el sistema.

Desarrollo de aplicaciones web

El desarrollo de aplicaciones mediante el uso de tecnología web es un proceso estructural que incluye diferentes etapas para garantizar la calidad, la función y la adaptación del *software* en el entorno organizacional. De acuerdo con Espinosa-Hurtado (2021):

El desarrollo de aplicaciones web ha tenido un crecimiento exponencial a lo largo de los últimos años, esto se debe al hecho de que cada día son más las exigencias requeridas para la web que proporcionen a los usuarios mejores experiencias al momento de hacer uso de ellas (p. 133).

Al tomar el caso de un sistema de gestión de pagos de nómina basado en el desarrollo web, resulta determinante, al seleccionar un modelo de desarrollo adecuado, programar cada uno de los módulos funcionales que sirvan como solución a las necesidades existentes en la empresa y utilizar tecnologías actualizadas y modernas, formando parte de las orientaciones hacia un desempeño correcto, seguro, confiable y escalable.

Ciclo de vida del Software. El ciclo de vida del *software* (SDLC, por sus siglas en inglés) se define como una agrupación de diferentes fases que funcionan como una guía por seguir cuando la persona o el equipo involucrado en el desarrollo del sistema o la solución tecnológica debe definir la planificación, el diseño, el desarrollo, la implementación, las pruebas y otros aspectos necesarios para hacer posible el proyecto.

Con el paso de los años, se han desarrollado varios modelos SDLC, como los modelos en cascada y los modelos ágiles, que son los más utilizados en los proyectos de *software* de negocios y en el mercado en general. El modelo en cascada se caracteriza por un enfoque secuencial en el que cada paso debe completarse antes de iniciar el siguiente: análisis, diseño, implementación e inspección. Este modelo proporciona claridad en la planificación y una documentación cuidadosa del desarrollo de características, lo cual resulta útil en proyectos con requisitos definidos desde el principio. Sin embargo, su principal desventaja es su rigidez, ya que no puede adaptarse fácilmente a los cambios durante el proceso de desarrollo.

Por otro lado, modelos ágiles como Scrum o Kanban se basan en ciclos iterativos y graduales. Según Velasco *et al.* (2021):

La metodología SCRUM permite un desarrollo ágil autoorganizado que utiliza un enfoque incremental de procesos que permite una mejor forma de entregar los resultados. Sin embargo, se debe dimensionar correctamente el proyecto y también la correcta conformación de los equipos de trabajo (p. 435).

Este enfoque promueve la comunicación constante con el cliente y permite ajustar y proporcionar resultados parciales durante el proceso, los cuales pueden evaluarse antes de dar por finalizado el desarrollo del proyecto. La flexibilidad del enfoque ágil resulta particularmente útil en un contexto donde los requisitos pueden cambiar o verse alterados conforme transcurre el

tiempo o cuando se requiere realizar pruebas tempranas para garantizar que la función concuerde con las expectativas del usuario final.

En el caso del sistema de gestión de pago de nómina en la empresa All Sport Nutrition, se considera que un modelo ágil como Scrum es más apropiado porque permite incluir mejoras de acuerdo con las necesidades específicas de la compañía, los términos de trabajo aplicables a Costa Rica o cambios en la gestión administrativa. Además, la naturaleza modular del sistema facilita la entrega de características parciales, de manera que la empresa pueda conocer los avances realizados y enfrentar modificaciones de forma flexible en caso de que así lo requiera la empresa o la persona usuaria final. Esto tiene como consecuencia que la compañía pueda confirmar el progreso y asegurar que el producto final esté alineado con los objetivos estratégicos establecidos.

Programación de funcionalidades. En el desarrollo de *software*, el diseño modular es un principio fundamental porque segmenta el sistema en partes independientes que pueden ser programadas, probadas y mantenidas por sí mismas.

Gestión de usuarios. Administrar los perfiles de cada una de las personas colaboradoras y los niveles de acceso que tienen a las características del sistema y a la información almacenada en la base de datos es una funcionalidad muy importante por desarrollar para brindar protección al sistema en general. Asimismo, incluye la posibilidad de registrar empleados en la base de datos, actualizar los datos personales si así se desea y definir los roles de acceso (administrador, mantenimiento, recursos humanos, etc.). De igual forma, se deben incorporar mecanismos que permitan la correcta autenticación y autorización para los usuarios que vayan a usar el sistema, garantizando la seguridad y confidencialidad de la información.

Gestión de planillas. Este es uno de los principales componentes funcionales según la orientación u objetivo principal del sistema, ya que gracias a esta función se pueden calcular automáticamente los salarios, realizar las deducciones y los aportes a cargo del patrono, siguiendo los lineamientos estipulados por el Código de Trabajo de Costa Rica y la Caja Costarricense de Seguro Social. En este punto es relevante tomar en cuenta la generación de recibos de pago digitales y la programación periódica de pagos, enfocándose en la exactitud y eficiencia en el pago de salarios.

Reportes. La característica de generar informes o reportes que incluyan información o datos

relevantes y personalizables acerca de la nómina de la compañía, como los costos mensuales, las deducciones realizadas, las contribuciones del empleador y los datos estadísticos de remuneraciones, resulta vital para el análisis y la presentación de los datos, así como para brindar transparencia a las transacciones que se efectúan internamente en la empresa. Los informes, además de reforzar la honestidad y la franqueza en la administración, resultan útiles como insumo estratégico para la toma de decisiones económicas, administrativas y estratégicas.

Tecnologías relevantes. La elección de tecnologías apropiadas es un factor crucial para que un sistema web tenga éxito. En la actualidad, existen varios lenguajes de programación y *frameworks* que se distinguen por la manera en la que permiten a los desarrolladores escribir código y diseñar funcionalidades de forma productiva y eficiente. De acuerdo con Larrea *et al.* (2022):

Los lenguajes de programación han evolucionado junto con los sistemas hardware desde su inicio. Cada avance que los componentes hardware tenían hacía necesario el uso de lenguajes de programación que permitiera manejarlos. Desde la generación de instrucciones escritas en código máquina hasta el desarrollo de complejas aplicaciones web y móviles, los lenguajes de programación han jugado un papel fundamental en este camino de la era informática (p. 1604).

El PHP es un lenguaje muy popular; Laravel es uno de los *frameworks* más demandados en el entorno de este lenguaje, ya que facilita la implementación de sistemas sólidos y escalables, es uno de los lenguajes más comunes en lo que respecta al desarrollo web. Además, JavaScript resulta esencial para el desarrollo de funcionalidades en sistemas web y Node.js es una opción eficaz para el lado del servidor, más orientado al *back-end*. Por otra parte, C# proporciona un ambiente seguro y flexible para aplicaciones de negocios, junto con *frameworks* como .NET y sus paquetes NuGet, que hacen posible crear sistemas muy robustos con fines empresariales y de gran flujo transaccional.

En materia de administración de servidores, las herramientas Nginx y Apache suelen utilizarse con frecuencia para hospedar aplicaciones web. Por otro lado, navegadores como Google Chrome, Mozilla Firefox, Opera y Microsoft Edge garantizan que los usuarios puedan acceder a aplicativos orientados a la web en distintas plataformas.

Al combinar estas tecnologías, se garantiza que el sistema web de All Sport Nutrition no solo satisfaga los requisitos técnicos, sino que también brinde a los usuarios finales una experiencia segura y fluida, lo cual está en sintonía con las tendencias actuales de la digitalización de empresas.

CAPÍTULO III. MARCO METODOLÓGICO

El marco metodológico es una de las partes que, necesariamente, debe estar presente en toda investigación científica, ya que permite definir el camino que debe seguirse para alcanzar los objetivos establecidos y, a la vez, lograr que los resultados sean válidos. Gracias a este capítulo es posible conocer los métodos, enfoques y tipos de investigación que se centran en el desarrollo del proyecto del sistema web para la gestión de pago de nómina en la empresa que se seleccionó. A través de esta estructura metodológica, se establece cómo se realiza el proceso de recolección, análisis y validación de los datos, todos siendo un complemento fundamental, tanto para construir como para evaluar el sistema por desarrollar.

Enfoques de investigación

El enfoque de la investigación representa la manera en la que se aborda un problema científico, utilizando una orientación metodológica general que sirve de guía para realizar este abordaje. La elección de uno de los enfoques depende de qué se estudia, cuáles son los objetivos del proyecto y los datos que posiblemente se obtienen y, después, se analizan. En el presente proyecto, definir el enfoque de investigación constituye un paso crucial, ya que permite determinar cuáles estrategias se deben poner en práctica para recopilar datos confiables y garantizar que los resultados sean válidos.

Enfoque cuantitativo. Este enfoque se caracteriza por su orientación hacia la objetividad y la medición numérica, así como por el análisis estadístico de los datos recolectados. El propósito de este enfoque es describir, explicar o predecir fenómenos mediante el análisis de datos utilizando métodos numéricos y estadísticos.

Cabe destacar de este enfoque que presenta la capacidad de medir fenómenos observables. Si se toma como ejemplo el proyecto que se desarrolla, dicha capacidad se orienta a la posibilidad de recolectar y almacenar datos, como el tiempo que un colaborador del Departamento de Recursos Humanos u otro departamento relacionado tarda en generar la planilla antes y después de implementar la solución tecnológica en la investigación. Un segundo dato curioso e importante de analizar es la cantidad de errores que se pueden detectar o en los que puede incurrir una persona al realizar los cálculos de las remuneraciones. Es gracias al registro de datos de este tipo que se

pueden identificar patrones de comportamiento que, junto con el análisis estadístico, permiten determinar la eficiencia y mejora que puede aportar o ha aportado el sistema a la empresa.

Es necesario mencionar que uno de los puntos fuertes de este enfoque es el énfasis que se otorga a la validez y confiabilidad de los instrumentos que se utilizan en la etapa de recolección de datos. En un estudio de este tipo, la validez resulta indispensable; esta se garantiza mediante la elaboración de instrumentos que se alinean con los objetivos del proyecto. Por otro lado, al tomar una muestra del personal administrativo en este caso y aplicar los mismos instrumentos, se puede lograr la confiabilidad de los datos.

El enfoque cuantitativo, como tal, ofrece una base sobre la cual se utiliza una metodología adecuada para evaluar el impacto que tiene el sistema por desarrollar en la empresa. Esto se logra al recolectar los datos y realizar una medición objetiva de variables que se relacionan con la precisión y eficiencia del sistema, sin olvidar el nivel de satisfacción que pueda experimentar la persona usuaria frente a la herramienta. Utilizar este enfoque no solo facilita que el estudio científico cuente con rigurosidad, sino que también aporta a los resultados para determinar cuáles puntos de mejora deben acatarse, orientados a los procesos administrativos y tecnológicos de la compañía en el futuro próximo. De acuerdo con Faneite (2023), el enfoque cuantitativo puede entenderse como:

Llamado también racionalista-analítico o positivista-cuantitativo; surge a finales del siglo XIX e inicios del XX, proveniente de las ciencias naturales y trasladado luego a las ciencias sociales. Se caracteriza por su orientación hacia la predicción, la cual se basa en interpretaciones de hechos observados; su dirección es predecible. Además, es un proceso analítico que va de lo general a lo específico, del efecto a la causa (p. 84).

Enfoque cualitativo. A diferencia del enfoque cuantitativo, este tiene como fin comprender en profundidad los fenómenos, tanto sociales como humanos, a partir de la perspectiva de cada una de las personas participantes involucradas en el estudio. A través del mencionado enfoque, el estudio facilita la exploración de las percepciones, puntos de vista y experiencias en las que las personas pueden verse involucradas en diferentes situaciones. De acuerdo con Valle *et al.* (2022), los objetivos del enfoque cualitativo se centran en:

Los significados, las experiencias y las situaciones que atraviesan los sujetos; por este

motivo, la información recogida no suele ser cuantitativa. Sin embargo, sí se pueden usar datos numéricos que permitan distinguir diversidades (Jansen, 2012), categorizar o aproximarse a la información para luego profundizar en esta (Niño, 2011) (p. 91).

En el caso del presente estudio, el enfoque cualitativo sirve como una metodología de gran importancia, ya que permite comprender el punto de vista de las personas colaboradoras de la empresa respecto al proceso actual de gestión y pago de nómina, así como las expectativas que tienen sobre la implementación del nuevo sistema propuesto. Mediante técnicas como entrevistas a las personas participantes, actividades de observación y análisis de discurso, es posible resaltar los problemas cotidianos que enfrenta el personal de la empresa, así como identificar los puntos de mejora en términos de usabilidad, productividad, eficiencia y comunicación.

A través de este enfoque es posible obtener información de alto valor sobre los niveles del sistema web en la empresa, considerando que el ser humano, por su naturaleza, se caracteriza por la resistencia al cambio y es posible que, en primera instancia, se dude de la confianza que se deposita en el uso de la tecnología para realizar este tipo de procesos. Cabe mencionar un factor importante por considerar: el grado de familiaridad que tienen las personas colaboradoras actuales con la tecnología.

Tomar el enfoque cuantitativo para el estudio actual crea una gran ventana de posibilidades, ya que ofrece una amplia variedad de escenarios para analizar y permite obtener una visión amplia e integral del problema, su solución y las personas participantes involucradas. Esto se debe a que no solo se evalúa, con criterios matemáticos y estadísticos, la implementación de un sistema como este, sino que también resulta importante considerar el impacto que puede generar en el ámbito social y en términos de cultura organizacional en la empresa.

Enfoque mixto. Si frente a un estudio se considera utilizar, tanto el enfoque cuantitativo como el cualitativo, esta idea daría como resultado el enfoque mixto, en resumidas cuentas, ya que al juntar ambos enfoques se tiene la posibilidad no solo de recolectar datos y analizarlos a partir del criterio estadístico y matemático, sino que también se facilita la interpretación del usuario en el ámbito humano y su punto de vista.

A partir de las investigaciones aplicadas al ámbito tecnológico, como el presente proyecto de desarrollo de un sistema web, este enfoque resulta de gran provecho, ya que un proyecto

desarrollado no solo debe tener implicaciones técnicas en su funcionamiento, sino que también se debe considerar su aceptación social en una comunidad. Tal como señalan Sarango *et al.* (2025), el enfoque mixto se define como: “Diseño que integra fases o técnicas cuantitativas y cualitativas dentro de un mismo estudio. Combina medición estadística con interpretación contextual, triangulando resultados para obtener conclusiones más completas y válidas” (p. 4249).

Al tomar como ejemplo el sistema que se planteó sobre la gestión de pago de nómina, se puede utilizar el enfoque mixto como una forma de analizar el problema que resuelve desde dos perspectivas muy interesantes de estudiar. En primer lugar, analizar el impacto en la reducción de tiempo que genera una solución programada para automatizar procesos manuales y repetitivos, así como la disminución de errores cometidos por el personal en casos como el cálculo de nómina, corresponde a la primera dimensión que aborda este enfoque. Por otro lado, el componente cualitativo facilita la percepción que tiene la persona usuaria respecto a las funcionalidades del sistema y su uso en la empresa, así como los demás factores humanos que implica adoptar tecnologías que automatizan procedimientos y gestionan información sensible o confidencial.

Agrupar ambos enfoques y ponerlos en juego robustece la validez de los resultados por obtener, ya que los datos que se obtienen del análisis cuantitativo pueden estudiarse desde la percepción humana que ofrece el enfoque cualitativo. Lo anterior permite que la evaluación de los resultados sea integral y se contextualice con la realidad que vive cotidianamente la compañía.

Enfoque de investigación seleccionado

Para el presente proyecto, que lleva por nombre *Sistema web para la gestión de pago de nómina en la empresa All Sport Nutrition*, se seleccionó el enfoque mixto como una estrategia metodológica capaz de brindar un análisis eficaz para el estudio actual. Uno de los factores principales que influyeron en la elección de este enfoque y su incorporación al proyecto fue que permite combinar el análisis en el ámbito técnico, en lo relativo al desarrollo del sistema, con la comprensión de las experiencias y perspectivas de los usuarios finales que interactúan con la herramienta digital.

Como se anotó, el enfoque mixto fomenta combinar las ventajas que ofrece el enfoque cuantitativo, orientado a la medición y análisis matemático de los datos, con las del enfoque cualitativo, centrado en la interpretación de los componentes del factor humano y los fenómenos

sociales, lo que favorece una visión integral del problema que se investiga. Aplicar el enfoque mixto garantiza resultados de mayor validez y más robustos, al vincular la evidencia empírica con la comprensión contextual. De esta manera, el estudio no solo se centra en medir la eficiencia de la solución programada, sino que también valora el impacto que implementarla en términos sociales, respaldando que el sistema web sea funcional, resolutorio y oportuno en la compañía en la que se usa.

Tipos de investigación

El tipo de investigación hace referencia a la estrategia que se debe seguir al estudiar un fenómeno, definiendo el alcance, los objetivos y los métodos mediante los cuales se realiza la recolección de datos y el análisis posterior de estos. La escogencia del tipo de investigación constituye un paso crucial en la forma en la que se interpreta la información recogida y en la validez de los resultados. La selección adecuada del tipo de investigación contribuye a asegurar que los hallazgos sean, tanto confiables como pertinentes para la empresa.

En el contexto del proyecto actual de estudio, analizar qué tipo de investigación utilizar permite comprender de qué manera se lleva a cabo el proceso a partir del cual se gestiona la nómina, ayuda a explorar las necesidades y carencias del personal administrativo y a evaluar de qué forma impacta la implementación de esta clase de sistema desde el punto de vista de la eficiencia, precisión y satisfacción de los usuarios. Gracias a esto, se puede garantizar que el sistema no solo responda a los requerimientos técnicos, sino que también tome en cuenta el ámbito real y las expectativas de la empresa.

La elección correcta y la combinación de los tipos de investigación ofrecen un marco metodológico robusto que se centra, en general, en el proceso investigativo, desde describir el estado actual hasta evaluar el impacto del sistema web propuesto.

Investigación descriptiva. El objetivo principal que caracteriza a este tipo de investigación es brindar carácter, detalle y documentación a los componentes que forman parte del objeto en estudio, lo que evita la intervención o modificación de su desarrollo natural. Como lo definen Sarango *et al.* (2024), se trata de una: “Investigación que busca describir características de fenómenos o poblaciones, estableciendo asociaciones entre variables sin determinar causalidad” (p. 959).

En el caso del sistema web que se tomó como propuesta de objeto de estudio, la investigación descriptiva se aplica para conocer de forma específica cómo funciona un determinado proceso, en este caso, el proceso de nómina. Con base en lo anterior, se incluirían mecanismos como el registro de empleados, la elaboración de planillas, la gestión de deducciones, el cálculo de las nóminas y la generación de reportes. De manera similar, se puede identificar qué otras herramientas tecnológicas se utilizan actualmente y cuáles procesos manuales se realizan para determinar de qué forma generan ineficiencias o errores.

Realizar este tipo de investigación es determinante, ya que al aplicar esta metodología se puede obtener un diagnóstico claro y objetivo del estado actual de los procesos administrativos en la empresa. Conseguir esta información permite fundamentar los requerimientos que servirán como base para construir la solución tecnológica, establecer las prioridades durante el desarrollo y asegurar que el sistema se alinee firmemente con las necesidades reales de la empresa.

Tener a la mano el estado actual de la organización frente a los resultados que se esperan después de implementar el sistema es una pieza clave de la investigación descriptiva, ya que sirve como herramienta para comparar ambas informaciones y dar a conocer un diagnóstico sobre el estado resultante, con la finalidad de evaluar posibles mejoras, tanto en la eficiencia como en el margen de error. Otra de las características que se pueden aprovechar es la información cuantitativa y cualitativa, la cual posteriormente puede vincularse con otros tipos de investigación, con el objetivo de mantener un análisis de validez robusto y completo.

Investigación exploratoria. El propósito principal de la investigación exploratoria radica en examinar un fenómeno poco estudiado o sobre el cual la información existente es escasa o nula, con la intención de obtener una comprensión inicial que pueda servir como base para estudios posteriores de mayor profundidad o para investigaciones orientadas a analizar una solución concreta. En palabras de Sarango *et al.* (2024), se trata de una: “Investigación inicial que se realiza cuando el tema es poco entendido o no ha sido claramente definido. Busca identificar patrones, ideas o hipótesis” (p. 959).

En el contexto actual del objeto en estudio, tomar la investigación exploratoria como vehículo para conocer el problema existente en el proceso de gestión de nómina es esencial para detallar todos los aspectos o detalles que engloban esta situación. Mediante la puesta en práctica de técnicas como entrevistas, observaciones y consulta de documentos, se recopilan datos acerca

de los procedimientos actuales, los errores comunes que se cometen al calcular las planillas, las limitaciones del sistema actual ejecutado de forma manual y las expectativas que presenta la persona usuaria final respecto a la implementación del sistema web en la empresa.

Al emplear el mencionado tipo de investigación, es posible comprender los principales retos a los que la empresa se enfrenta, entre los que se incluye la carencia de automatización, la ejecución de tareas de forma duplicada y el margen de error humano al realizar las asignaciones. Por otra parte, ayuda a identificar correctamente las funcionalidades o características que debe cumplir el aplicativo como solución al problema en estudio.

El valor que aporta la investigación exploratoria se fundamenta en la capacidad que otorga al investigador para dirigir el rumbo del proyecto y orientarse hacia el problema de manera clara. Después de recolectar datos y analizar los resultados, la persona puede continuar con el diseño de una solución viable, que logre adaptarse a la situación de la empresa como forma de satisfacer las necesidades existentes.

Investigación explicativa. La investigación explicativa tiene como fin principal detectar las causas y efectos del objeto en estudio, relacionar un análisis entre las variables y brindar una comprensión más profunda sobre el origen de los hechos. Como lo señalan Hernández Sampieri *et al.* (2017):

Los estudios explicativos son más que la descripción de conceptos o fenómenos o el establecimiento de relaciones entre variables; más bien, están diseñados para determinar las causas de los eventos y fenómenos físicos o sociales. Como su nombre lo indica, su interés se centra en explicar por qué ocurre un fenómeno y en qué condiciones se manifiesta, o por qué se relacionan dos o más variables (p. 78).

A través de la comparación entre la forma en la que se realizan los procesos de manera tradicional y los resultados al implementar la solución tecnológica es posible analizar, mediante este tipo de investigación, cómo el sistema web puede influir en la eficiencia de las operaciones administrativas, la mitigación de errores cometidos por las personas colaboradoras u otras personas involucradas y la satisfacción laboral del recurso humano.

Este enfoque también aumenta la posibilidad de explicar los factores que influyen en la dificultad de adoptar el nuevo sistema, como el nivel de habilidad tecnológica requerido para

utilizarlo y el nivel actual de las personas colaboradoras de la empresa, la calidad de las capacitaciones que se imparten y que la persona usuaria recibe, así como la experiencia de usuario generada por el diseño de las interfaces y el flujo de trabajo del sistema a través del cual la persona usuaria interactúa y su percepción al manipular dichas pantallas. De este modo, la investigación no solo mide resultados, sino que también interpreta los motivos por los cuales el sistema produce mejoras o enfrenta resistencias en la compañía.

Tipo de investigación por utilizar

Para el desarrollo del proyecto actual del sistema web para la gestión de pago de nómina, se decidió utilizar la investigación descriptiva como tipo de investigación principal. Se ha elegido este tipo debido a que permite realizar un estudio detallado de la situación actual en la que la empresa se encuentra respecto a su problemática principal, el proceso de manejo de nómina, sin que se requiera experimentaciones ni centrarse en procesos matemáticos o estadísticos de alta complejidad.

La investigación descriptiva se centra en observar, registrar y analizar los atributos de un fenómeno según su comportamiento en el contexto en que este se encuentra. Con base en el ámbito actual del proceso de gestión de pago de nómina, se estudian problemas como los errores en los cálculos, la duplicidad de tareas y la demora en la realización de los pagos e identifican las oportunidades de mejora que pueden ser atendidas al implementar la solución programada.

Este tipo de investigación ofrece grandes oportunidades en estudios realizados en ingeniería de *software* y desarrollo de sistemas de información. Esto se debe a que proporciona buenas bases para definir los requerimientos del aplicativo, diseñar la solución del sistema y evaluar el impacto que tiene la implementación de este sin recurrir a métodos muy complejos. Por otro lado, facilita presentar los resultados de manera clara y comprensible. Además, de los objetivos que ayudan a tomar decisiones de impacto y para seguir el rumbo definido por la entidad.

La escogencia de la investigación descriptiva se justifica por su facilidad de aplicación, su pertinencia con los objetivos del estudio y su capacidad para ofrecer un análisis preciso de la realidad organizacional. Gracias a este enfoque, el proyecto fundamentará el diseño y desarrollo del sistema web de nómina de manera práctica, objetiva y coherente con las necesidades de All Sport Nutrition.

Fuentes de información

En toda investigación científica, las fuentes de información constituyen recursos fundamentales para recolectar los datos, obtener nuevos conocimientos y recopilar evidencias que conforman el estudio, de manera que este pueda sustentarse sobre una base teórica. Dichas fuentes son de vital importancia, ya que representan los cimientos para comprender la realidad del problema, analizar sus causas y, a partir de esta información, construir una solución adecuada que sirva para resolver el problema en estudio.

Al tomar como ejemplo el estudio actual del sistema web para la gestión de pago de nómina, realizar una selección adecuada de las fuentes de información y aplicarlas de manera correcta permite que el estudio tenga mayor validez y se logre una confiabilidad de más alto grado en los resultados.

Fuentes primarias. De acuerdo con Hernández Sampieri (2017), se pueden definir las fuentes de información primaria de la siguiente manera:

Las fuentes primarias o referencias son libros, antologías, artículos de publicaciones periódicas, monografías, tesis y disertaciones, documentos oficiales, reportes de asociaciones, trabajos presentados en conferencias o seminarios, artículos periodísticos, testimonios de expertos, documentales, videgrabaciones en diferentes formatos, foros, páginas y sitios en internet, etc. (p. 59).

A partir de la apreciación de Hernández Sampieri (2017), las fuentes primarias tienen como característica principal compartir información original y directa sobre lo que se pone bajo estudio. Las fuentes que se utilizan en este caso no son interpretadas ni modificadas por otros autores, lo que resalta la autenticidad del evento investigado.

Estas fuentes son de gran importancia, ya que la información que proporcionan es actualizada, precisa y se mantiene dentro de un contexto real del estudio, lo que permite realizar un diagnóstico claro y útil para iniciar el desarrollo de la solución al problema que se investiga.

Fuentes secundarias. Las fuentes secundarias no solo recopilan información, sino que también son piezas clave en términos de interpretación y análisis de los datos extraídos de las fuentes primarias. Dichas fuentes se elaboran por especialistas en un área concreta, investigadores

o instituciones que se dedican por estudiar ciertos temas desde diferentes perspectivas. Entre estas fuentes de información se pueden encontrar recursos como libros, tesis, informes de grado técnico, artículos científicos y documentos académicos sobre el tema por tratar. Según Maranto Rivera y González Fernández (2015):

Este tipo de fuentes son las que ya han procesado información de una fuente primaria. El proceso de esta información se pudo dar por una interpretación, un análisis, así como la extracción y reorganización de la información de la fuente primaria (p. 3).

En el marco de toda investigación por realizar, las fuentes de nivel secundario permiten crear el fundamento teórico sobre el cual se construye el estudio. Esto abarca diferentes conceptos y terminologías de acuerdo con lo que se planea estudiar. En el caso actual, el fundamento teórico toma definiciones de la ingeniería de *software*, el diseño de bases de datos, los lenguajes de programación que se pretenden usar, términos de seguridad informática y ciberseguridad y hace hincapié en la mención de las normativas de carácter laboral vigentes en Costa Rica, que se relacionan con la gestión de nómina en la empresa.

Fuentes terciarias. Cuando se habla sobre las fuentes terciarias, se pueden destacar atributos como la facilitación del acceso, la localización y la organización de todos aquellos recursos que provienen de los dos tipos de fuentes anteriores: las primarias y las secundarias. Entre estas fuentes terciarias, se pueden mencionar varios recursos de los que es posible extraer información relevante para el estudio, entre estos se encuentran los catálogos bibliográficos, las bases de datos académicas, los catálogos de biblioteca, los índices, entre otros. Con base en las ideas que presentan Maranto Rivera y González Fernández (2015):

Este tipo de fuentes son las que recopilan fuentes de información primarias o secundarias. Estas fuentes son utilizadas para buscar datos o para obtener una idea general sobre algún tema, algunas son bibliografías, almacenes, directorios, donde se encuentran la referencia de otros documentos, que contienen nombres, títulos de revistas y otras publicaciones (p. 3).

En el presente trabajo se han empleado plataformas digitales como Google Académico, Scielo y diferentes repositorios de documentos científicos provenientes de universidades y otros centros de investigación profesional. El objetivo es localizar investigaciones de otros autores para

construir un fundamento robusto que aporte validez al eje principal sobre el cual se desarrolla este proyecto.

La combinación de fuentes primarias, secundarias y terciarias permite definir una base robusta de información que respalda el desarrollo del proyecto. Por otro lado, las fuentes primarias tienen como objetivo ofrecer datos reales y específicos de la empresa, las secundarias aportan los fundamentos teóricos y técnicos y las terciarias facilitan localizar y validar la información que se utiliza. En conjunto, estas fuentes aseguran la fiabilidad, coherencia y rigor científico de la investigación y garantizan que el sistema web propuesto se diseñó con base en información verificable, actual y relevante para la gestión de nómina en All Sport Nutrition.

Variables

Las variables son una pieza fundamental que se encuentra en toda investigación científica. Estas permiten al investigador medir, analizar y comprender los fenómenos que están bajo estudio. De acuerdo con Rodríguez *et al.* (2021), se pueden definir las variables como:

Las variables constituyen el eje transversal de todo el proyecto de investigación desde el planteamiento de la idea que lo origina; de ahí que cabe preguntarse qué variables se pretenden estudiar, cómo se las identifica, de qué forma se miden o cómo se relacionan (p. 11).

En el caso del proyecto actual, la identificación de variables resulta esencial para evaluar el impacto del sistema propuesto en la eficiencia del Área Administrativa, la reducción de errores cometidos por las personas colaboradoras de la empresa y la satisfacción de los usuarios finales que utilicen la herramienta. Al contar con una definición más clara en el ámbito conceptual, operacional e instrumental de las variables, se busca respaldar una medición precisa y objetiva de los resultados en el proceso.

Conceptuales. Este tipo de variables se centra en hacer referencia a las definiciones del fundamento teórico de los fenómenos que son estudiados. Además, se basa en tomar el objeto al que se aplica el estudio y analizar los conceptos, tanto generales como abstractos, que lo conforman, sin necesidad de entrar en los detalles de su medición.

Las variables conceptuales representan las definiciones teóricas de los conceptos que intervienen en una investigación, las cuales deben validarse siguiendo la línea que conduce al planteamiento del problema, las hipótesis y el diseño del estudio. En este sentido: “Se validan las definiciones conceptuales y operacionales de las variables del estudio” (Hernández Sampieri, 2017, p. 148). Estas variables representan los aspectos teóricos clave que sustentan el propósito general del estudio y orientan la formulación de los objetivos específicos.

Operacionales. La principal característica que define a las variables operacionales es la capacidad de traducir todos aquellos conceptos de nivel teórico en elementos que primero pueden observarse y luego medirse. Es decir, a través de estos métodos se establece el comportamiento de las variables en el contexto real. Las variables operacionales se refieren a la forma en la que las variables del estudio pueden medirse o apreciarse mediante instrumentos, fórmulas, algoritmos u observación directa y expresan los datos en números o categorías codificadas.

En este sentido: “Las variables medibles (por un sistema, aparato, equipo, fórmula, algoritmo, etc.) o apreciadas mediante los sentidos (como la observación)” (Hernández Sampieri, 2017, p. 148), por lo que constituyen su dimensión operacional.

Instrumentales. Las variables instrumentales se relacionan con las herramientas y los procedimientos que se utilizan para recolectar y analizar los datos correspondientes a las variables operacionales. En este proyecto se emplean diversos instrumentos que garantizan la validez y la confiabilidad de la información.

Las variables instrumentales representan la dimensión de las variables que se vinculan con los instrumentos o procedimientos mediante los cuales se obtienen los datos empíricos. En este sentido: “Las variables medibles (por un sistema, aparato, equipo, fórmula, algoritmo, etc.) o apreciadas mediante los sentidos (como la observación)” (Hernández Sampieri, 2017, p. 147), por lo que reflejan su carácter instrumental.

Estos instrumentos permiten obtener información confiable, la cual sirve para analizar el cumplimiento de los objetivos y validar las hipótesis del trabajo. Enseguida, se presentan las variables de estudio para el presente proyecto sobre el desarrollo de un sistema de gestión de pago de nómina, mediante la Tabla 4.

Tabla 4
Variables de la investigación

Objetivo específico	Variable	Variable conceptual	Variable instrumental	Variable operacional
Analizar los requerimientos funcionales y no funcionales para el desarrollo del sistema web por medio del estudio de las necesidades de la empresa.	Requerimientos del sistema	Según Zapana <i>et al.</i> (2021): "Su principal tarea consiste en la generación de especificaciones correctas que describan con claridad, sin ambigüedades, en forma consistente y compacta, el comportamiento del sistema" (p. 44).	Entrevistas	Guía de entrevistas Guía de observación
Diseñar el sistema web y las tablas de la base de datos para el almacenamiento correcto de datos por medio de herramienta de maquetado como MySQL Workbench.	Diseño del sistema y base de datos	De acuerdo con Araujo-Inastrilla <i>et al.</i> (2023) se puede afirmar sobre las bases de datos que: "Las BD son un conjunto de datos almacenados en memoria externa que están organizados mediante una estructura de datos" a los que se puede acceder de forma rápida y su diseño responde a la necesidad de: "Satisfacer los requisitos de información de una empresa u otro tipo de organización" (p. 3).	Documentos de Análisis Casos de uso	MySQL Workbench Figma
Programar cada uno de los módulos para el funcionamiento correcto del sistema web a través de herramientas tecnológicas relevantes.	Desarrollo y codificación del sistema	Interpretando lo analizado por Larrea <i>et al.</i> (2022), se puede rescatar de los lenguajes de programación el desarrollo de sistemas como: "Desde la generación de instrucciones escritas en código	Diagramas UML Diagrama E-R	Codificación mediante C# con .NET y MySQL, integrados en el entorno de desarrollo Visual Studio.

		máquina hasta el desarrollo de complejas aplicaciones web y móviles, los lenguajes de programación han jugado un papel fundamental en este camino de la era informática” (p. 1604).		
Probar cada una de las funcionalidades del sistema web para el funcionamiento eficaz del sistema por medio de pruebas íntegras e individuales.	Validación y pruebas del sistema	A través de lo que expresa Mamani (2023), se menciona que: “Las pruebas de software son seguramente la actividad más común de control de calidad realizada en los proyectos de desarrollo o mantenimiento de aplicaciones y sistemas” (p. 4).	Listas de verificación. Guías de prueba. <i>Software de Testing.</i>	Casos de prueba

Población

La población representa el conjunto de personas, individuos u objetos en su totalidad que comparten características en común y que se relacionan con el contexto de lo que se estudia. Según Chero-Pacheco (2024), se puede pensar que la población: “Expresa el conjunto de elementos de estudio. El investigador debe precisar esos elementos para realizar la investigación” (p. 45).

Es decir, la población se refiere al grupo de estudio del cual se desea recolectar datos y sobre el cual se pretende analizar un fenómeno. Lo anterior tiene el fin de realizar una generalización de los resultados del estudio.

En el caso de estudio que abarca la investigación titulada *Sistema web para la gestión de pago de nómina en la empresa All Sport Nutrition*, la población está conformada por todas las personas colaboradoras de la compañía, quienes participan de forma directa o indirecta en el proceso mediante el cual se realiza el manejo de nómina. Este grupo de personas resulta esencial para el desarrollo del proyecto investigativo, ya que su experiencia en el contexto del que se obtiene información para el estudio, el conocimiento del proceso y el grado en que interactúan con

el sistema actual proporciona información clave para identificar requerimientos, evaluar el diseño y validar el nuevo sistema web.

Muestra

Dentro de toda investigación, la muestra es un extracto o subconjunto de la población, con el objetivo de recolectar datos y generalizar los resultados sobre el grupo total, en este caso, la población. Sin embargo, existen casos en los que la población es reducida o puede manejarse en su totalidad; en estos casos, se selecciona a todos los individuos que estén disponibles, eliminando la necesidad de realizar los cálculos matemáticos para determinar el muestreo. Al respecto de este término, Salazar Salazar (2021) menciona que: “Es la parte o subconjunto de la población, con la que se realiza el estudio y para seleccionarla se deben delimitar las características de la población” (p. 5).

En el caso del presente estudio, la población total está conformada por siete colaboradores que integran las diferentes áreas de la empresa. Puesto que este número es relativamente pequeño y manejable, se tomó la decisión de trabajar con la totalidad de la población, es decir, se aplica un censo poblacional.

Instrumentos de recolección de datos

Los instrumentos de recolección de datos son las herramientas que se utilizan para obtener la información requerida, analizar las variables definidas y cumplir con los objetivos establecidos del estudio. Es necesario seleccionar adecuadamente los instrumentos por emplear, ya que a través de ellos se garantiza la validez y pertinencia de los datos recolectados.

En la investigación: “Las técnicas de recolección de datos comprenden procedimientos y actividades que permiten al investigador obtener la información necesaria para responder a su pregunta de investigación” (Hernández y Duana, 2020, citados por Martínez, 2022, p. 38).

La observación. La observación es un instrumento que, al ejecutarlo, permite recopilar datos de forma empírica mediante la supervisión del entorno en el que se encuentra el objeto de estudio y los procesos que se desarrollan en este. Este mecanismo de recolección de datos resulta de gran ayuda cuando se trata de identificar una secuencia de actividades en un proceso o de reconocer patrones de comportamiento. Según Piza *et al.* (2019, citados por Martínez, 2022), la observación:

“Constituye un elemento esencial en un proyecto de investigación, ya que permite obtener el mayor número de datos de manera clara, definida y precisa” (p. 38).

La entrevista. En este método de recolección de datos se encuentran dos tipos de entrevistas: las estructuradas y las semiestructuradas. Las entrevistas estructuradas son aquellas en las que el o las personas investigadoras utilizan una cantidad fija de preguntas para recolectar datos, aplicándolas de la misma manera y en el mismo orden a quienes participen en dicha entrevista. Esto es importante en el análisis cualitativo de los resultados, ya que permite obtener información precisa y uniforme que, a la vez, puede compararse.

En palabras de Piza *et al.* (2019, citados por Martínez, 2022): “Las entrevistas implican a dos personas; un entrevistado y un entrevistador, el primero hace preguntas detalladas al segundo y esta actividad resulta ser una especie de filtro para una siguiente etapa en la investigación” (p. 38).

En otras instancias, las entrevistas semiestructuradas se diferencian de las explicadas por la razón de que ofrecen mayor flexibilidad para profundizar en ciertos aspectos o situaciones que surgen durante la plática, sin depender de mantener una estructura fija de preguntas que se seleccionaron.

La encuesta. A diferencia de las entrevistas, la encuesta utiliza un listado de preguntas escritas que se entregan directamente a quienes participan en brindar las respuestas. Estas personas deben adjuntar por escrito su respuesta a cada una. De esta forma, se garantiza que las respuestas sean objetivas y comparables y se evita la influencia que el investigador pueda ejercer sobre las respuestas de los sujetos.

Por medio de las palabras de Kendall (2017), se puede destacar lo siguiente sobre este instrumento de recolección de datos:

La Encuesta, en estudios con enfoque cuantitativos o cualitativos, es la más utilizada respecto al resto de técnicas, inclusive en el entorno virtual online y offline, apoyada siempre por un cuestionario debidamente estructurado y automatizado con el fin de asegurar el manejo transparente de un gran volumen de datos en tiempo casi real (p. 1166).

CAPÍTULO IV. ANÁLISIS DE RESULTADOS

El presente capítulo expone los hallazgos que se obtienen al aplicar tres instrumentos de recolección de datos en la empresa All Sport Nutrition: la guía de observación, las entrevistas semiestructuradas y la encuesta. Estos instrumentos permitieron diagnosticar la situación actual de la gestión de nómina en la compañía y fundamentar la necesidad de implementar un sistema web automatizado que optimice los procesos administrativos que se relacionan con el pago del personal.

Análisis de la observación

La guía de observación se aplicó con el objetivo de evaluar de manera directa las actividades y condiciones que se relacionan con la gestión del pago de nómina en All Sport Nutrition. Los aspectos observados abarcaron, desde el registro de asistencia del personal hasta el tiempo invertido en la elaboración de planillas mensuales, lo que permitió identificar de manera precisa las áreas que requieren intervención y mejora.

En cuanto al registro sistematizado de horas laborales y asistencia del personal, se determinó que este aspecto no cumple con las expectativas de un proceso eficiente. En la actualidad, la asistencia se registra de forma manual en hojas de Microsoft Excel y, en algunos casos, mediante mensajes enviados por WhatsApp. Esta situación denota la ausencia de un sistema centralizado que permita validar automáticamente las horas de entrada y salida de las personas colaboradoras. La falta de estandarización en el registro genera inconsistencias significativas y dificulta el control preciso de las horas trabajadas, lo que se traduce en posibles errores al calcular los salarios. La oportunidad de mejora identificada consiste en implementar un módulo de asistencia con registros automáticos que garantice la trazabilidad y exactitud de la información.

Respecto a la disponibilidad de un sistema automatizado para el cálculo de planillas, se observó que la empresa no cumple con este requisito fundamental. El cálculo de las planillas se realiza de forma manual cada mes, utilizando fórmulas básicas en Microsoft Excel, sin integración automática con los datos de horas trabajadas, deducciones legales u horas extra. Este proceso consume tiempo valioso del personal administrativo y resulta altamente propenso a errores humanos, en especial cuando existen variaciones en las jornadas laborales o deducciones

extraordinarias. La necesidad de automatizar el cálculo de planillas para reducir errores humanos y disminuir los tiempos de elaboración constituye una prioridad para la compañía.

En relación con la exactitud en los cálculos de deducciones legales, que incluyen las contribuciones a la Caja Costarricense de Seguro Social, el Instituto Nacional de Seguros y los impuestos correspondientes, se determinó que este aspecto cumple de forma parcial. Las deducciones se calculan manualmente siguiendo tablas consultadas en Internet y, aunque por lo general se realizan de manera correcta, existe un riesgo latente de error debido a que los porcentajes y montos legales suelen modificarse periódicamente. Cuando estos cambios ocurren es necesario ajustar manualmente las hojas de Microsoft Excel, lo cual incrementa el riesgo de errores y genera retrabajo. La oportunidad de mejora consiste en integrar las tablas y porcentajes legales en un sistema que actualice los cálculos de manera automática conforme a las disposiciones vigentes.

La seguridad en el manejo de datos de las personas colaboradoras constituye uno de los hallazgos más críticos de la observación. Este aspecto no cumple con los estándares mínimos de protección de información sensible. Los datos del personal, que incluyen salarios, información personal y deducciones, se almacenan en archivos de Microsoft Excel sin contraseñas de protección y con frecuencia se comparten por correo electrónico o WhatsApp. La ausencia de control de acceso y de mecanismos de seguridad pone en riesgo la confidencialidad de la información y expone a la empresa a posibles vulneraciones de privacidad. Por ende, es imperativo implementar un sistema con usuarios, contraseñas, perfiles diferenciados y encriptación básica para proteger adecuadamente los datos sensibles de los colaboradores.

El registro histórico de pagos y comprobantes de nómina tampoco cumple con las expectativas de un proceso organizado y auditable. Los comprobantes se generan de forma manual y, con frecuencia, se envían por correo individual a cada colaborador, sin que exista un archivo centralizado que facilite la consulta posterior. Localizar planillas de meses anteriores puede requerir la revisión de múltiples documentos dispersos en diferentes ubicaciones digitales, lo que dificulta la auditoría y el seguimiento histórico. La oportunidad de mejora consiste en centralizar y automatizar el almacenamiento de comprobantes y el historial de pagos, de modo que tanto las personas colaboradoras como el personal administrativo puedan acceder fácilmente a la información cuando sea necesario.

Por último, en lo que respecta al tiempo total invertido en la elaboración de planillas mensuales, se identificó que este aspecto no cumple con los criterios de eficiencia. En la actualidad, el proceso de creación de la planilla consume entre 4 y 6 horas al mes, según variables como las horas extra registradas, los permisos otorgados y las incapacidades del personal. Este tiempo se considera elevado, ya que todo debe verificarse manualmente y es necesario cruzar datos provenientes de varios archivos independientes. La implementación de un sistema que calcule de forma automática los montos, deducciones y validaciones permite reducir estos tiempos y liberar al personal administrativo para que se enfoque en actividades de mayor valor estratégico.

Análisis de las entrevistas

Las entrevistas semiestructuradas se aplicaron a dos actores clave en la organización: el gerente general, Juan Pablo Alvarado Barahona y el administrador, Minor Alfaro Rodríguez. Ambos entrevistados poseen una visión integral de los procesos administrativos y participan directamente en la gestión de nómina. Por lo tanto, sus percepciones resultan fundamentales para comprender el problema desde diferentes niveles jerárquicos.

Ambos entrevistados coincidieron en que los errores e inconsistencias en los cálculos de planilla se presentan con frecuencia. El gerente general indicó que al menos una vez al mes se detectan inconsistencias, en especial cuando intervienen variables como horas extra, incapacidades o permisos. El administrador reforzó esta percepción al señalar que casi todos los meses es necesario realizar correcciones debido a horas mal registradas, permisos que no se descontaron adecuadamente, incapacidades mal calculadas o deducciones que no se reflejaron de forma correcta. Esta situación evidencia que el método actual no solo resulta ineficiente, sino que también compromete la precisión de los pagos, lo cual puede afectar la confianza de las personas colaboradoras en el proceso.

En cuanto a las principales limitaciones del sistema o método actual, ambos entrevistados identificaron múltiples deficiencias. El gerente general destacó que todo el proceso se realiza de forma manual, lo que implica un alto riesgo de errores humanos. Además, señaló que no existe un registro automático de asistencia ni integración entre los diferentes módulos de información y que la dependencia de distintas hojas de cálculo, que no siempre están actualizadas o sincronizadas, genera confusión y retrasos.

Por otra parte, el administrador enfatizó el registro manual de asistencia, la falta de integración entre horas extra, permisos e incapacidades, la necesidad de reescribir la información en varias hojas, la dificultad para mantener actualizadas las tablas de deducciones y el riesgo de pérdida de información por archivos mal guardados o sobrescritos. Estas restricciones no solo afectan la calidad del proceso, sino que también incrementan la carga administrativa y generan atrasos en la entrega de las planillas.

Respecto a la información relacionada con la nómina que se considera más importante mantener actualizada y accesible, el gerente general mencionó las horas trabajadas y las horas extra, las deducciones legales actualizadas, el historial de incapacidades y permisos, los salarios y los ajustes salariales, así como los comprobantes y el historial de pagos. El administrador coincidió en gran medida y agregó que la asistencia diaria, los porcentajes de deducciones y los cambios salariales son elementos clave que, cuando no se mantienen al día, provocan que la planilla se atrase o genere errores. Ambos reconocieron que contar con estos datos actualizados evitaría errores y facilitaría las auditorías internas, lo que contribuiría a una gestión más transparente y confiable.

En cuanto al resguardo de la información confidencial de las personas colaboradoras, ambos entrevistados admitieron que las prácticas actuales no cumplen con estándares adecuados de seguridad. El gerente general reconoció que los datos se almacenan principalmente en archivos de Microsoft Excel con acceso manual, algunos protegidos y otros no y que en casos específicos se envían por correo o WhatsApp, lo cual no es seguro ni cumple con buenas prácticas de protección de datos. El administrador complementó esta visión al indicar que la información se guarda en carpetas de computadora, algunas con contraseña y otras sin ella y que compartir documentos por correo o WhatsApp expone datos sensibles de los colaboradores. Esta situación representa un riesgo significativo para la privacidad y puede derivar en consecuencias legales para la empresa.

El proceso de generar reportes de pago y deducciones legales para instituciones como la Caja Costarricense de Seguro Social, el Instituto Nacional de Seguros y el Ministerio de Hacienda fue calificado como complicado y lento por ambos entrevistados. El gerente general señaló que, al no contar con un sistema automatizado, los reportes deben generarse manualmente y cada dato debe verificarse uno por uno, lo que consume tiempo y requiere mucha revisión para evitar errores.

El administrador indicó que este proceso es bastante complicado porque se debe revisar cada cálculo de forma manual, verificar que los porcentajes sean correctos y luego trasladar la información a los formularios o plataformas correspondientes. Asimismo, agregó que si no se tiene todo el registro claro desde el inicio del mes, la tarea se vuelve aún más difícil. Esta complejidad afecta la puntualidad en el cumplimiento de obligaciones legales y aumenta el riesgo de sanciones.

Ambos entrevistados reconocieron que el sistema actual afecta significativamente el tiempo de trabajo y la productividad del Área Administrativa. El gerente general indicó que el Área Administrativa invierte muchas horas en tareas que pueden automatizarse, como cálculos, validaciones y consolidación de datos, lo que reduce la productividad e incluso retrasa otros procesos importantes de la empresa. El administrador fue enfático al señalar que preparar la planilla, revisar horas, confirmar permisos e incapacidades consume varias horas que pueden utilizarse en asignaciones más estratégicas. Además, destacó que, al depender de Microsoft Excel, es necesario revisar todo por duplicado para evitar errores, lo que disminuye considerablemente la productividad. Esta situación evidencia que la compañía desaprovecha recursos humanos valiosos en actividades operativas repetitivas.

En cuanto a los beneficios que aportaría implementar un sistema web automatizado para la gestión de nómina, ambos entrevistados identificaron múltiples ventajas. El gerente general mencionó la reducción de errores, una mayor rapidez en los cálculos y reportes, seguridad de la información, mejor organización del historial de planillas, ahorro de tiempo, optimización del trabajo administrativo y mayor transparencia para colaboradores y jefaturas. El administrador coincidió en gran medida y agregó que un sistema web proporcionaría mayor precisión en los cálculos, menos tiempo invertido en tareas repetitivas, un registro único y actualizado para todo el personal, reportes automáticos, transparencia con las personas colaboradoras y seguridad de la información. Ambos concluyeron que un sistema automatizado profesionalizaría el proceso de manejo de nómina en la empresa.

Respecto a las características o funcionalidades esenciales que debe incluir el sistema web, ambos entrevistados presentaron listas detalladas y complementarias. El gerente general consideró indispensables un acceso seguro con perfiles y permisos, un registro de asistencia y horas extra, el cálculo automático de planillas y deducciones, un historial de pagos con comprobantes descargables, reportes automáticos para la Caja Costarricense de Seguro Social, el Instituto

Nacional de Seguros y el Ministerio de Hacienda, una interfaz sencilla y usable, así como notificaciones internas.

El administrador añadió el control de permisos e incapacidades, perfiles con accesos diferenciados para recursos humanos, jefaturas y colaboradores, notificaciones internas sobre aprobaciones o rechazos y un respaldo seguro en la nube. Estas funcionalidades reflejan las necesidades reales de la organización y deben considerarse como requisitos fundamentales en el diseño del sistema.

En cuanto a la disposición del personal de la empresa ante el uso de nuevas tecnologías o sistemas digitales, ambos entrevistados mostraron una óptica positiva. El gerente general indicó que, en general, existe buena disposición, aunque algunos colaboradores pueden necesitar una breve curva de aprendizaje. Sin embargo, todos reconocen que el proceso actual resulta desgastante y están abiertos a un sistema que facilite el trabajo. El administrador coincidió al señalar que el personal está dispuesto a utilizar un sistema siempre y cuando sea sencillo y que, aunque algunos tengan menos experiencia con herramientas digitales, con capacitación se adaptan sin problema. Esta disposición favorable representa una ventaja importante para la implementación exitosa del proyecto.

Finalmente, ambos entrevistados ofrecieron recomendaciones valiosas para garantizar una implementación exitosa del nuevo sistema web. El gerente general sugirió capacitar al personal antes de lanzar el sistema, realizar pruebas piloto con casos reales, mantener soporte técnico durante las primeras semanas, documentar procesos y responsabilidades y escuchar la retroalimentación del personal para mejorar el sistema. El administrador recomendó capacitar a todas las personas colaboradoras y jefaturas, establecer un período de prueba en el que se comparen los resultados del sistema con la planilla manual, documentar cada proceso y cada módulo, mantener soporte técnico inicial y llevar a cabo una retroalimentación constante para optimizar el sistema. Estas sugerencias se consideran en la fase de puesta en funcionamiento para minimizar la resistencia al cambio y maximizar la adopción del sistema.

En síntesis, los resultados de la observación y las entrevistas convergen en un diagnóstico claro: el método actual de gestión de nómina en All Sport Nutrition presenta deficiencias significativas en términos de eficiencia, precisión, seguridad y trazabilidad. La implementación de un sistema web automatizado no solo resulta necesaria, sino también urgente, para profesionalizar

los procesos administrativos, reducir errores, optimizar el tiempo del personal y garantizar el resguardo de la información confidencial de las personas colaboradoras.

Requerimientos del sistema

A partir del análisis realizado mediante la observación y las entrevistas aplicadas en la empresa All Sport Nutrition, se identificaron los requerimientos que debe atender el sistema web para la gestión de pago de nómina, con el fin de cubrir las necesidades del Área Administrativa y de recursos humanos. Estos requisitos se clasifican en funcionales y no funcionales y constituyen la base para el diseño y desarrollo del prototipo funcional que permite optimizar los procesos actuales de la organización.

Requerimientos funcionales. En los requerimientos funcionales se indican las acciones o servicios específicos que el sistema debe atender para satisfacer las necesidades operativas de la empresa All Sport Nutrition. Estos requisitos se definieron a partir del análisis de los procesos actuales y de los datos recolectados mediante los instrumentos que se aplicaron, específicamente la guía de observación y las entrevistas semiestructuradas realizadas al gerente general y al administrador de la compañía.

Tabla 5
Requerimientos funcionales del sistema

Módulo	Código	Descripción del requerimiento funcional
Gestión de nóminas	REQ-NOM-01	Generar la planilla quincenal con salario bruto, deducciones y salario neto por pagar.
	REQ-NOM-02	Registrar rebajos legales como el de la CCSS y rebajos de cualquier otra índole.
Gestión de horas extra	REQ-HEX-01	Registrar la fecha, hora de inicio, hora de finalización y observaciones para las jornadas excepcionales de cada empleado.
	REQ-HEX-02	Mostrar un registro de horas extra entre un rango de fechas para cada colaborador.

Gestión de vacaciones	REQ-VAC-01	Registrar las solicitudes de vacaciones y su estado (pendiente, aprobado o rechazado).
	REQ-VAC-02	Registrar la cantidad de días disponibles que cada empleado tiene a disposición.
Cálculo aguinaldo	REQ-AGUI-01	Calcular automáticamente el aguinaldo con base en el promedio de salarios del período.
	REQ-AGUI-02	Generar el comprobante de pago correspondiente.
Gestión de incapacidades	REQ-INC-01	Registrar incapacidades otorgadas por el INS y la CCSS, registrando las fechas en que está inhabilitado el empleado.
	REQ-INC-02	Mostrar un historial de incapacidades de cada empleado filtrado por INS, CSS o todas.
Gestión de liquidaciones	REQ-LIQ-01	Calcular la liquidación según el tipo de finalización del contrato.
	REQ-LIQ-02	Cumplir con la legislación costarricense, completando el pago de cesantía y aguinaldo, vacaciones y preaviso.
Evaluación de rendimiento	REQ-EVAL-01	Permitir la definición de indicadores y criterios de evaluación personalizados para cada puesto.
	REQ-EVAL-02	Registrar las evaluaciones periódicas realizadas a cada colaborador.
Gestión de permisos	REQ-PER-01	Registrar permisos indicando el motivo, fechas y estado de cada empleado.
	REQ-PER-02	Mostrar un historial de permisos entre un rango de fechas por cada empleado.

Gestión de asistencia	REQ-ASÍ-01	Registrar la hora de entrada y salida de cada empleado en una fecha.
	REQ-ASÍ-02	Validar que no existan asistencias duplicadas
	REQ-ASÍ-03	Mostrar un historial de asistencia de cada empleado.
Seguridad	REQ-SEG-01	Controlar el acceso mediante usuario y contraseña.
	REQ-SEG-02	Asignar roles y permisos diferentes según el tipo de usuario.
	REQ-SEG-03	Registrar las acciones de los usuarios en el sistema para tener trazabilidad.
Reportes	REQ-REP-01	Generar reportes de empleado, asistencia, nómina, vacaciones, permisos y aguinaldo.

Requerimientos no funcionales. Los requerimientos no funcionales describen las características de calidad del sistema web para la gestión de pago de nómina, con el fin de asegurar su rendimiento, seguridad, confiabilidad y facilidad de uso. Estos aspectos establecen las condiciones bajo las cuales debe operar el sistema para garantizar su desempeño correcto en el entorno de la empresa All Sport Nutrition.

Tabla 6
Requerimientos no funcionales del sistema

Módulo	Código	Descripción del requerimiento funcional
Rendimiento	RNF-REN-01	El sistema debe permitir realizar operaciones comunes (registros, consulta, actualización y eliminación de datos) y cálculos de forma fluida, sin generar retrasos en condiciones normales de uso.
Seguridad	RNF-SEG-01	El acceso al sistema debe estar protegido mediante autenticación de usuario y contraseña, con

		encriptación de credenciales y control de roles.
Disponibilidad	RNF-DIS-01	El sistema debe estar disponible durante toda la jornada laboral, siempre que el equipo servidor esté en funcionamiento.
Usabilidad	RNF-USA-01	La interfaz debe ser clara, intuitiva y con colores neutros que faciliten la lectura y reduzcan la fatiga visual.
Mantenibilidad	RNF-MANT-01	El sistema debe permitir la actualización de los módulos sin afectar la estabilidad del resto del sistema.
Confiabilidad	RNF-CON-01	El sistema debe garantizar la integridad de los datos en todas las operaciones y cálculos que haga el sistema y restricciones en base de datos.

Estos lineamientos permiten orientar el diseño, desarrollo y validación del prototipo funcional, asegurando que la solución propuesta responda a las necesidades reales de la empresa All Sport Nutrition y opere de manera eficiente, segura y confiable en su entorno actual. La implementación de estos requerimientos funcionales y no funcionales contribuye significativamente a profesionalizar los procesos de gestión de nómina, reduce errores, optimiza tiempos y garantiza la seguridad de la información confidencial de las personas colaboradoras.

CAPÍTULO V. PROPUESTA

Casos de uso

Tabla 7

Caso de uso para el inicio de sesión en el sistema

Prototipo: Sistema web para gestión de pago de nómina en la empresa All Sport Nutrition	
Número del caso de uso: CU-MS-01	Nombre del caso de uso: iniciar sesión
Fecha de elaboración:	07/01/2026
Descripción del caso de uso	Le permite al usuario autenticarse, con la finalidad de tener acceso al sistema
Autor del caso de uso:	Kendall Salazar Soto
Actores que se relacionan:	Usuario del sistema Sistema
Precondiciones:	El usuario debe estar registrado en la base de datos El usuario debe poseer un rol y los permisos asignados El sistema debe estar funcionando
Flujo básico del caso de uso	
1. El usuario ingresa a la pantalla de inicio de sesión. 2. El sistema muestra el formulario de <i>login</i>. 3. El usuario introduce en el formulario su nombre de usuario y la contraseña. 4. El sistema valida los campos. 5. El sistema valida las credenciales en la base de datos. 6. El sistema autentica al usuario y genera un <i>token</i> de sesión. 7. El sistema registra el inicio de sesión en la tabla de auditoría. 8. El sistema redirige al usuario a la página principal del <i>dashboard</i> según su rol. 9. El caso de uso finaliza exitosamente.	
Flujos alternos	
Flujo alternativo n.º 1 Validación de campos vacíos	• El usuario deja campos en blanco. • Se muestra un mensaje de error y no se continúa

	con la autenticación. El flujo continúa en el punto 2 del flujo principal.
Flujo alternativo n.º 2 Validación de credenciales	- Las credenciales no son válidas y rechaza el inicio de sesión. - El sistema muestra un mensaje de error. - El flujo continúa en el punto 2 del flujo principal.
Requerimientos especiales	
Todas las acciones deben registrarse en la tabla auditoría. Las contraseñas deben viajar cifradas.	
Poscondiciones	
El usuario queda autenticado y con una sesión activa. El sistema asegura que se respeten los permisos según el rol asignado.	

Tabla 8

Caso de uso para la recuperación y cambio de contraseña

Prototipo: Sistema web para gestión de pago de nómina en la empresa All Sport Nutrition	
Número del caso de uso: CU-MS-02	Nombre del caso de uso: recuperación y cambio de contraseña
Fecha de elaboración:	07/01/2026
Descripción del caso de uso	Le permite al usuario recuperar su contraseña olvidada y cambiar su contraseña en caso de que se requiera
Autor del caso de uso:	Kendall Salazar Soto
Actores que se relacionan:	Usuario empleado Usuario administrador Sistema
Precondiciones:	El usuario administrador debe encontrarse registrado en la base de datos El usuario empleado debe existir El usuario administrador debe gestionar el proceso para suministrar una contraseña temporal al usuario

	El sistema debe estar funcionando
Flujo básico del caso de uso	
1. El usuario accede a la pantalla de inicio de sesión. 2. El sistema muestra el formulario de <i>login</i>. 3. El usuario ingresa en el formulario su nombre de usuario y contraseña temporal. 4. El sistema valida los campos. 5. El sistema valida las credenciales contra la base de datos 6. El sistema detecta la acción de cambiar contraseña 7. El sistema redirige automáticamente al usuario a la vista de cambio de contraseña. 8. El usuario ingresa la nueva contraseña. 9. El sistema valida la nueva contraseña. 10. El sistema actualiza la nueva contraseña en la base de datos. 11. El sistema registra la acción en la tabla auditoría. 12. El sistema redirige al usuario nuevamente al <i>login</i>. 13. El usuario ingresa las credenciales con su contraseña nueva. 14. El sistema valida los campos. 15. El usuario se redirige a la vista principal del <i>dashboard</i>. 16. Se registra la acción en la tabla auditoría. 17. El caso de uso finaliza exitosamente.	
Flujos alternos	
Flujo alternativo n.º 1 Validación de campos vacíos	• El usuario deja campos en blanco. • Se muestra un mensaje de error • El flujo continúa en el punto 3 del flujo principal.
Flujo alternativo n.º 2 Validación de requisitos de contraseña	• La contraseña no cumple con las reglas para gestionar el registro de este dato. • El sistema muestra un mensaje de error. • El flujo continúa en el punto 3 del flujo principal.
Flujo alternativo n.º 3 Validación de credenciales	• Las credenciales no son válidas y rechaza el inicio de sesión. • El sistema muestra un mensaje de error. • El flujo regresa al punto 3 del flujo principal.
Flujo alternativo n.º 4 Error del sistema	• El sistema no puede registrar o validar debido a un error técnico. • Se muestra un mensaje de error. • El flujo finaliza sin acceso al sistema.

Requerimientos especiales
Todas las acciones deben registrarse en la tabla auditoría. Las contraseñas deben viajar cifradas.
Poscondiciones
El usuario queda autenticado y con una sesión activa. Si el cambio fue exitoso, la contraseña temporal se elimina. El usuario puede iniciar sesión con una nueva contraseña. Se registran en la tabla auditoría los cambios realizados.

Tabla 9
Caso de uso para la actualización de los datos del usuario

Prototipo: Sistema web para gestión de pago de nómina en la empresa All Sport Nutrition	
Número del caso de uso: CU-MS-03	Nombre del caso de uso: actualizar datos de un usuario
Fecha de elaboración:	07/01/2026
Descripción del caso de uso	Le permite al usuario administrador actualizar los datos y la contraseña de un usuario para gestionar la recuperación de su cuenta
Autor del caso de uso:	Kendall Salazar Soto
Actores que se relacionan:	Usuario administrador Sistema
Precondiciones:	El usuario administrador debe encontrarse registrado en la base de datos El usuario administrador debe tener asignado el permiso para actualizar datos El sistema debe estar funcionando
Flujo básico del caso de uso	
1. El usuario administrador accede al <i>dashboard</i>. 2. Selecciona la opción <i>Usuarios del sistema</i>. 3. El sistema valida si la persona usuaria tiene los permisos correspondientes para ingresar.	

4. El sistema despliega la vista con todos los empleados disponibles. 5. El usuario selecciona el botón <i>Actualizar</i> del empleado que necesita recuperar su cuenta. 6. El sistema despliega el formulario Actualizar Cuenta. 7. El usuario ingresa los datos que se le solicitan en el formulario. 8. El sistema valida que los campos cumplan con los requerimientos. 9. El usuario da clic en el botón de actualizar. 10. El sistema regresa a la vista con todos los empleados. 11. El caso de uso finaliza exitosamente.	
Flujos alternos	
Flujo alternativo n.º 1 Validación de campos vacíos	• El usuario deja campos en blanco. • Se muestra un mensaje de error • El flujo continúa en el punto 7 del flujo principal.
Flujo alternativo n.º 2 Validación de requisitos de contraseña	• La contraseña no cumple con las reglas para gestionar el registro de este dato. • El sistema muestra un mensaje de error. • El flujo continúa en el punto 7 del flujo principal.
Flujo alternativo n.º 3 Validación de coincidencia de contraseñas	• El usuario digita los campos de nueva contraseña y confirma la contraseña. • El sistema muestra un mensaje de error si ambos campos no coinciden. • El flujo regresa al punto 7 del flujo principal.
Flujo alternativo n.º 4 Error del sistema	• El sistema no puede registrar o validar debido a un error técnico. • Se muestra un mensaje de error. • El flujo finaliza sin acceso al sistema.
Requerimientos especiales	
Cumplir con la autenticación del sistema. Todas las acciones deben ser registradas en la tabla auditoría. Solo los usuarios con el rol de administrador pueden modificar los datos de los demás usuarios. El sistema debe mantener control de trazabilidad sobre cada modificación de datos sensibles.	
Poscondiciones	
El usuario afectado recupera su cuenta El usuario afectado recibe una contraseña temporal para acceder a su cuenta.	

Se registran en la tabla auditoría los cambios realizados.

Tabla 10

Caso de uso para la creación de un usuario en el sistema

Prototipo: Sistema web para gestión de pago de nómina en la empresa All Sport Nutrition	
Número del caso de uso: CU-MS-04	Nombre del caso de uso: crear usuario
Fecha de elaboración:	07/01/2026
Descripción del caso de uso	Le permite al usuario administrador crear un registro de usuario en el sistema
Autor del caso de uso:	Kendall Salazar Soto
Actores que se relacionan:	Usuario administrador Sistema
Precondiciones:	El usuario administrador debe encontrarse registrado en la base de datos El usuario administrador debe tener asignado el permiso para gestionar el proceso. El sistema debe estar funcionando
Flujo básico del caso de uso	
1. El usuario administrador accede al <i>dashboard</i>. 2. Selecciona la opción <i>Usuarios del sistema</i>. 3. El sistema valida si la persona usuaria tiene los permisos correspondientes para ingresar. 4. El sistema despliega la vista con todos los empleados disponibles. 5. El usuario da clic en el botón <i>Crear Usuario</i>. 6. El sistema despliega el formulario Nuevo Usuario. 7. El usuario ingresa los datos que se le solicitan en el formulario. 8. El sistema valida que los campos cumplan con los requerimientos. 9. El usuario da clic en el botón <i>Crear</i>. 10. El sistema regresa a la vista de todos los usuarios. 11. El caso de uso finaliza exitosamente.	
Flujos alternos	
Flujo alternativo n.º 1	• El usuario deja campos en blanco.

Validación de campos vacíos	- Se muestra un mensaje de error - El flujo continúa en el punto 6 del flujo principal.
Flujo alternativo n.º 2 Validación de requisitos de contraseña	- La contraseña no cumple con las reglas para gestionar el registro de este dato. - El sistema muestra un mensaje de error. - El flujo continúa en el punto 6 del flujo principal.
Flujo alternativo n.º 3 Validación de coincidencia de contraseñas	- El usuario digita los campos de nueva contraseña y confirma la contraseña. - El sistema muestra un mensaje de error si ambos campos no coinciden. - El flujo regresa al punto 6 del flujo principal.
Flujo alternativo n.º 4 Error del sistema	- El sistema no puede registrar o validar debido a un error técnico. - Se muestra un mensaje de error. - El flujo finaliza sin acceso al sistema.
Requerimientos especiales	
Cumplir con la autenticación del sistema. Todas las acciones deben ser registradas en la tabla auditoría. Solo los usuarios con el rol de administrador pueden crear cuentas de usuario. Las contraseñas deben viajar encriptadas.	
Poscondiciones	
Se crea un registro del usuario nuevo en la base de datos. El usuario afectado recibe una contraseña temporal para acceder a su cuenta. Se registran en la tabla auditoría los cambios realizados.	

Tabla 11

Caso de uso para la eliminación de un usuario en el sistema

Prototipo: Sistema web para gestión de pago de nómina en la empresa All Sport Nutrition	
Número del caso de uso: CU-MS-05	Nombre del caso de uso: Eliminar un usuario
Fecha de elaboración:	07/01/2026
Descripción del caso de uso	Le permite al usuario administrador eliminar un registro

	de usuario en el sistema
Autor del caso de uso:	Kendall Salazar Soto
Actores que se relacionan:	Usuario administrador Sistema
Precondiciones:	El usuario administrador debe encontrarse registrado en la base de datos El usuario administrador debe tener asignado el permiso para gestionar el proceso. El sistema debe estar funcionando
Flujo básico del caso de uso	
1. El usuario administrador accede al <i>dashboard</i>. 2. Selecciona la opción <i>Usuarios del sistema</i>. 3. El sistema valida si la persona usuaria tiene los permisos correspondientes para ingresar. 4. El sistema despliega la vista con todos los usuarios disponibles. 5. El usuario da clic en el botón <i>Eliminar</i> correspondiente al usuario que se debe eliminar. 6. El sistema despliega una ventana de confirmación y pregunta si desea eliminar este usuario. 7. El usuario da clic en el botón <i>Sí</i> del mensaje de confirmación. 8. El sistema cierra la ventana de confirmación. 9. El sistema regresa a la vista de todos los usuarios. 10. El caso de uso finaliza exitosamente.	
Flujos alternos	
Flujo alternativo n.º 1 Cancelar proceso de eliminar usuario	• El usuario da clic en el botón <i>No</i> de la ventana de confirmación. • El sistema cancela la operación. • El flujo regresa al punto 4.
Flujo alternativo n.º 2 Error del sistema	• El sistema no puede registrar o validar debido a un error técnico. • Se muestra un mensaje de error. • El flujo finaliza sin acceso al sistema.
Requerimientos especiales	
Cumplir con la autenticación del sistema. Todas las acciones deben ser registradas en la tabla auditoría.	

Solo los usuarios con el rol de administrador pueden eliminar cuentas de usuario.
Poscondiciones
Se elimina el registro del usuario en la base de datos. El usuario queda inhabilitado para iniciar sesión en el sistema. Se registran en la tabla auditoría los cambios realizados.

Tabla 12

Caso de uso para el acceso a los usuarios activos en el sistema

Prototipo: Sistema web para gestión de pago de nómina en la empresa All Sport Nutrition	
Número del caso de uso: CU-MS-06	Nombre del caso de uso: acceder a todos los usuarios activos en el sistema
Fecha de elaboración:	07/01/2026
Descripción del caso de uso	Le permite al usuario administrador ver todos los usuarios activos para realizar acciones sobre los registros.
Autor del caso de uso:	Kendall Salazar Soto
Actores que se relacionan:	Usuario administrador Sistema
Precondiciones:	El usuario administrador debe encontrarse registrado en la base de datos El usuario administrador debe tener asignado el permiso para gestionar el proceso. El sistema debe estar funcionando
Flujo básico del caso de uso	
1. El usuario administrador accede al <i>dashboard</i>. 2. Selecciona la opción <i>Usuarios del sistema</i>. 3. El sistema valida si la persona usuaria tiene los permisos correspondientes para ingresar. 4. El sistema despliega la vista con todos los empleados disponibles. 5. El caso de uso finaliza exitosamente.	
Flujos alternos	

Flujo alternativo n.º 1 Error del sistema	- El sistema no puede registrar o validar debido a un error técnico. - Se muestra un mensaje de error. - El flujo finaliza sin acceso al sistema.
Requerimientos especiales	
Cumplir con la autenticación del sistema. Todas las acciones deben ser registradas en la tabla auditoría.	
Poscondiciones	
Se registran en la tabla auditoría los cambios realizados.	

Tabla 13

Caso de uso para listar la asistencial del día

Prototipo: Sistema web para gestión de pago de nómina en la empresa All Sport Nutrition	
Número del caso de uso: CU-MA-01	Nombre del caso de uso: listar los registros de la asistencia del día actual
Fecha de elaboración:	07/01/2026
Descripción del caso de uso	Le permite al usuario administrador observar los datos activos en el sistema
Autor del caso de uso:	Kendall Salazar Soto
Actores que se relacionan:	Usuario administrador Sistema
Precondiciones:	El usuario administrador debe encontrarse registrado en la base de datos El usuario administrador debe tener asignado el permiso para gestionar el proceso. El sistema debe estar funcionando
Flujo básico del caso de uso	
1. El usuario accede a la opción <i>Asistencia</i> del módulo de asistencia. 2. El sistema muestra la vista con todos los empleados activos en el sistema.	

3. El sistema muestra datos del día actual, como nombre del empleado, fecha, hora de entrada, hora de salida, estado; si está completada o no y descripción. 4. La vista cuenta con los botones Editar, Eliminar, Agregar y Regresar. 5. Finaliza el caso de uso exitosamente.	
Flujos alternos	
Flujo alternativo n.º 1 Validación de empleado activo	- El sistema consulta si el empleado está activo en el sistema. - Si existe muestra la información del empleado.
Flujo alternativo n.º 2 Accionar el botón de editar	- El usuario selecciona el botón <i>Editar</i>. - El sistema redirecciona a la vista de Actualizar datos.
Flujo alternativo n.º 3 Accionar el botón de eliminar	- El usuario selecciona el botón <i>Eliminar</i>. - El sistema muestra una ventana de confirmación.
Flujo alternativo n.º 4 Agregar asistencia	- El usuario selecciona el botón <i>Agregar</i>. - El sistema redirecciona a la vista Crear un registro de asistencia.
Flujo alternativo n.º 5 Accionar el botón de regresar	- El usuario selecciona el botón <i>Regresar</i>. - El sistema redirecciona a la vista principal del sistema.
Flujo alternativo n.º 6 Error del sistema	- El sistema no puede registrar o validar debido a un error técnico. - Se muestra un mensaje de error. - El flujo finaliza sin acceso al sistema.
Requerimientos especiales	
Validación de empleado activo. Todas las acciones deben ser registradas en la tabla auditoría.	
Poscondiciones	
Se observa el registro de los datos de asistencia de los empleados en el día actual de trabajo.	

Tabla 14*Caso de uso para agregar un registro de asistencia*

Prototipo: Sistema web para gestión de pago de nómina en la empresa All Sport Nutrition	
Número de Caso de Uso: CU-MA-02	Nombre del caso de uso: agregar un registro de asistencia de un empleado
Fecha elaboración:	07/01/2026
Descripción caso de uso	Le permite al usuario administrador realizar un registro de asistencia de un empleado.
Autor caso de uso:	Kendall Salazar Soto
Actores que se relacionan:	Usuario administrador Sistema
Precondiciones:	El usuario administrador debe encontrarse registrado en la base de datos El usuario administrador debe tener asignado el permiso para gestionar el proceso. Los datos del empleado deben estar registrados. El sistema debe estar funcionando
Flujo básico del caso de uso	
1. El usuario accede a la opción <i>Asistencia</i> de la barra lateral. 2. El sistema muestra la vista con todos los empleados activos en el sistema. 3. El sistema muestra datos del día actual como nombre del empleado, fecha, hora de entrada, hora de salida, estado, si está completada o no, descripción. 4. La vista cuenta con los botones Editar, Eliminar, Agregar y Regresar. 5. El usuario da clic al botón <i>Agregar</i> en la persona correspondiente. 6. El sistema redirecciona a la vista de <i>Agregar asistencia</i> y despliega el formulario. 7. El usuario completa los datos requeridos según el formulario en pantalla. 8. El sistema valida todos los campos. 9. El usuario selecciona la opción <i>Guardar</i>. 10. El sistema almacena la información en la base de datos. 11. El sistema confirma el registro exitoso. 12. El sistema sale del formulario y regresa a la vista principal del módulo de asistencia. 13. Finaliza el caso de uso exitosamente.	
Flujos alternos	

Flujo alternativo n.º 1 Validación de fecha no futura	• Se verifica que la fecha no sea futura al día presente. • El sistema bloquea los días futuros.
Flujo alternativo n.º 2 Validación de hora	• El sistema verifica que la hora esté dentro del rango de horas trabajadas. • Si no cumple se muestra un mensaje de advertencia. • Se bloquea el envío del formulario.
Flujo alternativo n.º 3 Validación de campos vacíos	• Se valida que se ingrese la información en los campos requeridos obligatorios. • Si no cumple se muestra un mensaje de advertencia. • Si no cumple se rechaza el envío.
Flujo alternativo n.º 4 Validación máximo de horas	• Se valida que las horas de entrada y de salida no sean mayores a 8 horas. • Si no cumple se muestra un mensaje de advertencia. • Si no cumple se rechaza el envío.
Flujo alternativo n.º 5 Cancelación de proceso	• Si el usuario decide cancelar la operación antes de guardar, el sistema no registra la información. • Regresa a la pantalla principal de asistencia.
Flujo alternativo n.º 6 Error del sistema	• El sistema no puede registrar o validar debido a un error técnico. • Se muestra un mensaje de error. • El flujo finaliza sin acceso al sistema.
Requerimientos especiales	
Validación de fecha futura, máximo de horas diarias, de horario y de campos requeridos. Todas las acciones deben ser registradas en la tabla auditoría.	
Poscondiciones	
La condición de la asistencia del empleado queda registrada.	

Tabla 15*Caso de uso para actualizar un registro de asistencia*

Prototipo: Sistema web para gestión de pago de nómina en la empresa All Sport Nutrition	
Número de caso de uso: CU-MA-03	Nombre del caso de uso: actualizar los datos del registro de asistencia.
Fecha elaboración:	07/01/2026
Descripción caso de uso	Le permite al usuario administrador actualizar los datos del registro de asistencia del empleado en el sistema con toda su información correspondiente y sus validaciones.
Autor caso de uso:	Kendall Salazar Soto
Actores que se relacionan:	Usuario administrador Sistema
Precondiciones:	El usuario administrador debe encontrarse registrado en la base de datos El usuario administrador debe tener asignado el permiso para gestionar el proceso. Los datos del empleado deben estar registrados. El sistema debe estar funcionando
Flujo básico del caso de uso	
1. El usuario selecciona la opción de <i>Asistencia</i> en la barra lateral. 2. El sistema muestra la pantalla de asistencia con todos los empleados activos del sistema. 3. El sistema muestra datos del día actual, como nombre del empleado, fecha, hora de entrada, hora de salida, estado, etc. 4. La vista cuenta con los botones Editar, Eliminar, Agregar y Regresar. 5. El usuario selecciona el botón <i>Editar</i>. 6. El sistema redirecciona a la vista <i>Actualizar asistencia</i> y muestra el formulario con los datos previamente registrados cargados. 7. El usuario actualiza los registros correspondientes. 8. El sistema valida todos los campos. 9. El usuario selecciona la opción <i>Guardar</i>. 10. El sistema almacena la información en la base de datos. 11. El sistema confirma el registro exitoso. 12. El sistema sale del formulario y regresa a la vista principal del módulo de asistencia. 13. Finaliza el caso de uso exitosamente.	

Flujos alternos	
Flujo alternativo n.º 1 Validación de fecha no futura	• Se verifica que la fecha no sea futura al día presente. • El sistema bloquea los días futuros.
Flujo alternativo n.º 2 Validación de hora	• El sistema verifica que la hora esté dentro del rango de horas trabajadas. • Si no cumple se muestra un mensaje de advertencia. • Se bloquea el envío del formulario.
Flujo alternativo n.º 3 Validación de campos vacíos	• Se valida que se ingrese la información en los campos requeridos obligatorios. • Si no cumple se muestra un mensaje de advertencia. • Si no cumple se rechaza el envío.
Flujo alternativo n.º 4 Validación máximo de horas	• Se valida que las horas de entrada y de salida no sean mayores a 8 horas. • Si no cumple se muestra un mensaje de advertencia. • Si no cumple se rechaza el envío.
Flujo alternativo n.º 5 Cancelación de proceso	• Si el usuario decide cancelar la operación antes de guardar, el sistema no registra la información. • Regresa a la pantalla principal de asistencia.
Flujo alternativo n.º 6 Error del sistema	• El sistema no puede registrar o validar debido a un error técnico. • Se muestra un mensaje de error. • El flujo finaliza sin acceso al sistema.
Requerimientos especiales	
Validación de fecha futura, máximo de horas diarias, de horario y de campos requeridos. Todas las acciones deben ser registradas en la tabla auditoría.	
Poscondiciones	
Se actualizan los datos de la asistencia del empleado seleccionado. La condición de la asistencia del empleado queda registrada.	

Tabla 16*Caso de uso para eliminar un registro de asistencia*

Prototipo: Sistema web para gestión de pago de nómina en la empresa All Sport Nutrition	
Número de caso de uso: CU-MA-04	Nombre del caso de uso: eliminar un registro de asistencia.
Fecha elaboración:	07/01/2026
Descripción caso de uso	Le permite al usuario administrador eliminar los datos de un registro de asistencia dentro del sistema.
Autor caso de uso:	Kendall Salazar Soto
Actores que se relacionan:	Usuario administrador Sistema
Precondiciones:	El usuario administrador debe encontrarse registrado en la base de datos El usuario administrador debe tener asignado el permiso para gestionar el proceso. Los datos del empleado deben estar registrados. El sistema debe estar funcionando.
Flujo básico del caso de uso	
1. El usuario selecciona la opción de asistencia en el menú lateral. 2. El sistema muestra la pantalla de asistencia con todos los empleados activos del sistema. 3. El sistema muestra datos del día actual, como nombre del empleado, fecha, hora de entrada, hora de salida, etc. 4. La vista cuenta con los botones Editar, Eliminar, Agregar y Regresar. 5. El usuario selecciona el botón <i>Eliminar</i> del empleado que requiere eliminar el registro. 6. El usuario confirma la eliminación. 7. El sistema actualiza el estado del empleado en la base de datos a inactivo. 8. El sistema registra la acción en la tabla de auditoría. 9. El sistema regresa a la vista principal de Asistencia. 10. Finaliza el caso de uso exitosamente.	
Flujos alternos	
Flujo alternativo n.º 1 Validación de permisos	• El sistema valida si el usuario tiene los permisos correspondientes para realizar la acción.

	• Si el usuario no tiene los permisos correspondientes el botón de eliminar se encuentra desactivado.
Flujo alternativo n.º 2 Validar confirmación de eliminación	• El sistema muestra un mensaje de confirmación para eliminar al empleado. • Si se confirma la eliminación se realiza la acción de eliminación. • El sistema regresa a la vista principal de asistencia.
Flujo alternativo n.º 3 Cancelar la operación	• El usuario desea cancelar la acción. • En la ventana de confirmación selecciona la opción <i>No</i>. • El sistema cierra el cuadro de diálogo y no realiza ninguna acción relativa al empleado. • Retorna a la vista de asistencia.
Flujo alternativo n.º 4 Validación máximo de horas	• Se valida que las horas de entrada y de salida no sean mayores a 8 horas. • Si no cumple se muestra un mensaje de advertencia. • Si no cumple se rechaza el envío.
Flujo alternativo n.º 5 Error del sistema	• El sistema no puede registrar o validar debido a un error técnico. • Se muestra un mensaje de error. • El flujo finaliza sin acceso al sistema.
Requerimientos especiales	
El sistema debe garantizar que se realice un borrado lógico y no físico para preservar la trazabilidad histórica. Todas las acciones deben ser registradas en la tabla auditoría.	
Poscondiciones	
Se elimina el registro de asistencia de la persona y queda marcado como inactivo en el sistema. Se conserva toda la información histórica de la asistencia del empleado. El registro de la acción queda almacenado en la tabla auditoría. El sistema refleja los cambios en la lista de asistencia actual.	

Tabla 17*Caso de uso para listar registros de horas extra*

Prototipo: Sistema web para gestión de pago de nómina en la empresa All Sport Nutrition	
Número del caso de uso: CU-MHE-01	Nombre del caso de uso: listar los registros de horas extra de los empleados del día actual.
Fecha de elaboración:	07/01/2026
Descripción del caso de uso	Le permite al usuario administrador observar los datos activos en el sistema
Autor del caso de uso:	Kendall Salazar Soto
Actores que se relacionan:	Usuario administrador Sistema
Precondiciones:	El usuario administrador debe encontrarse registrado en la base de datos El usuario administrador debe tener asignado el permiso para gestionar el proceso. Los datos del empleado deben estar registrados. El sistema debe estar funcionando
Flujo básico del caso de uso	
1. El usuario selecciona la opción de <i>Asistencia</i> en el menú lateral. 2. El usuario accede a la opción <i>Horas extra</i> en el menú lateral. 3. El sistema muestra la vista con todos los empleados activos en el sistema. 4. El sistema muestra datos del día actual, como nombre del empleado, fecha, hora de entrada, hora de salida y demás. 5. La vista cuenta con los botones Editar, Eliminar y Agregar. 6. Finaliza el caso de uso exitosamente.	
Flujos alternos	
Flujo alternativo n.º 1 Validación de empleado activo	• El sistema consulta si el empleado está activo en el sistema. • Si existe, muestra la información del empleado.
Flujo alternativo n.º 2 Accionar botón de editar	• El usuario selecciona el botón <i>Editar</i>. • El sistema redirecciona a la vista de actualizar

	datos.
Flujo alternativo n.º 3 Accionar botón de eliminar	- El usuario selecciona el botón <i>Eliminar</i>. - El sistema muestra una ventana de confirmación.
Flujo alternativo n.º 4 Accionar botón de agregar asistencia	- El usuario selecciona el botón <i>Agregar</i>. - El sistema redirecciona a la vista Crear un registro de horas extra.
Flujo alternativo n.º 6 Error del sistema	- El sistema no puede registrar o validar debido a un error técnico. - Se muestra un mensaje de error. - El flujo finaliza sin acceso al sistema.
Requerimientos especiales	
Validación de empleado activo. Todas las acciones deben ser registradas en la tabla auditoría.	
Poscondiciones	
Se accede a la información de las horas extra de los empleados activos en el sistema.	

Tabla 18*Caso de uso para agregar un registro de horas extra*

Prototipo: Sistema web para gestión de pago de nómina en la empresa All Sport Nutrition	
Número de caso de uso: CU-MHE-02	Nombre del caso de uso: agregar un registro de horas extra de un empleado.
Fecha elaboración:	07/01/2026
descripción caso de uso:	Le permite al usuario administrador agregar un nuevo registro de asistencia de un empleado.
Autor caso de uso:	Kendall Salazar Soto
Actores que se relacionan:	Usuario administrador Sistema
Precondiciones:	El usuario administrador debe encontrarse registrado

	en la base de datos El usuario administrador debe tener asignado el permiso para gestionar el proceso. Los datos del empleado deben estar registrados. El sistema debe estar funcionando
Flujo básico del caso de uso	
1. El usuario accede a la opción <i>Horas extra</i> del módulo de horas extra. 2. El sistema muestra la vista con todos los empleados activos en el sistema. 3. El sistema muestra datos del día actual, como nombre del empleado, fecha, hora de entrada, hora de salida, etc. 4. La vista cuenta con los botones Editar, Eliminar y Agregar. 5. El usuario da clic al botón <i>Agregar</i> en la persona correspondiente. 6. El sistema redirecciona a la vista <i>Agregar horas extra</i> y despliega el formulario. 7. El usuario completa los datos requeridos del empleado, como fecha, hora de entrada, hora de salida, etc. 8. El sistema valida todos los campos. 9. El usuario selecciona la opción <i>Guardar</i>. 10. El sistema almacena la información en la base de datos. 11. El sistema confirma el registro exitoso. 12. El sistema sale del formulario y regresa a la vista principal del módulo de asistencia. 13. Finaliza el caso de uso exitosamente.	
Flujos alternos	
Flujo alternativo n.º 1 Validación de fecha no futura	• Se verifica que la fecha no sea futura al día presente. • El sistema bloquea los días futuros.
Flujo alternativo n.º 2 Validación de hora	• El sistema verifica que la hora esté dentro del rango de horas trabajadas. • Si no cumple, se muestra un mensaje de advertencia. • Se bloquea el envío.
Flujo alternativo n.º 3 Validación de campos vacíos	• Se valida que se ingrese la información en los campos requeridos obligatorios. • Si no cumple, se muestra un mensaje de advertencia. • Se rechaza el envío.
Flujo alternativo n.º 4 Validación máximo de horas	• Se valida que entre la hora de entrada y la de salida no transcurran más de 12 horas diarias al sumar las horas base.

	• Si no cumple, se muestra un mensaje de advertencia. • Se rechaza el envío.
Flujo alternativo n.º 5 Cancelación de proceso	• Si el usuario decide cancelar la operación antes de guardar, el sistema no registra la información. • Regresa a la pantalla principal de horas extra.
Flujo alternativo n.º 6 Error del sistema	• El sistema no puede registrar o validar debido a un error técnico. • Se muestra un mensaje de error. • El flujo finaliza sin acceso al sistema.
Requerimientos especiales	
Validación de fecha futura, máximo de horas diarias, de horario y de campos requeridos. Todas las acciones deben ser registradas en la tabla auditoría.	
Poscondiciones	
La condición de las horas extra del empleado queda registrada. El registro de la acción queda almacenado en la tabla auditoría.	

Tabla 19*Caso de uso para actualizar un registro de horas extra*

Prototipo: Sistema web para gestión de pago de nómina en la empresa All Sport Nutrition	
Número de caso de uso: CU-MHE-03	Nombre del caso de uso: actualizar los datos del registro de horas extra.
Fecha elaboración:	07/01/2026
Descripción caso de uso	Le permite al usuario administrador actualizar un registro de horas extra en el sistema con la información necesaria, respetando las validaciones de esta.
Autor caso de uso:	Kendall Salazar Soto
Actores que se relacionan:	Usuario administrador Sistema
Precondiciones:	El usuario administrador debe encontrarse registrado

	en la base de datos El usuario administrador debe tener asignado el permiso para gestionar el proceso. Los datos del empleado deben estar registrados. El sistema debe estar funcionando
Flujo básico del caso de uso	
1. El usuario selecciona <i>Horas extra</i> del módulo de horas extra. 2. El sistema muestra la pantalla de horas extra con todos los empleados activos del sistema. 3. El sistema muestra datos del día actual, como nombre del empleado, fecha, hora de entrada, hora de salida, etc. 4. La vista cuenta con los botones Editar, Eliminar y Agregar. 5. El usuario selecciona el botón <i>Editar</i>. 6. El sistema redirecciona a la vista Actualizar horas extra y muestra el formulario con los datos previamente registrados cargados. 7. El usuario actualiza los registros correspondientes. 8. El sistema valida todos los campos. 9. El usuario selecciona la opción <i>Guardar</i>. 10. El sistema almacena la información en la base de datos. 11. El sistema confirma el registro exitoso. 12. El sistema sale del formulario y regresa a la vista principal del módulo de horas extra. 13. Finaliza el caso de uso exitosamente.	
Flujos alternos	
Flujo alternativo n.º 1 Validación de fecha no futura	• Se verifica que la fecha no sea futura al día presente. • El sistema bloquea los días futuros.
Flujo alternativo n.º 2 Validación de hora	• El sistema verifica que la hora esté dentro del rango de horas trabajadas. • Si no cumple, se muestra un mensaje de advertencia. • Se bloquea el envío.
Flujo alternativo n.º 3 Validación de campos vacíos	• Se valida que se ingrese la información en los campos requeridos obligatorios. • Si no cumple, se muestra un mensaje de advertencia. • Se rechaza el envío.
Flujo alternativo n.º 4	• Se valida que entre la hora de entrada y la de salida no transcurran más de 8 horas.

Validación máximo de horas	• Si no cumple, se muestra un mensaje de advertencia. • Si no cumple, se rechaza el envío.
Flujo alternativo n.º 5 Cancelación de proceso	• Si el usuario decide cancelar la operación antes de guardar, el sistema no registra la información. • Regresa a la pantalla principal de horas extra.
Flujo alternativo n.º 6 Error del sistema	• El sistema no puede registrar o validar debido a un error técnico. • Se muestra un mensaje de error. • El flujo finaliza sin acceso al sistema.
Requerimientos especiales	
Validación de fecha futura, máximo de horas diarias, de horario y de campos requeridos. Todas las acciones deben ser registradas en la tabla auditoría.	
Poscondiciones	
El registro de horas extra del empleado es actualizado correctamente. El registro de la acción queda almacenado en la tabla auditoría.	

Tabla 20*Caso de uso para eliminar un registro de horas extra*

Prototipo: Sistema web para gestión de pago de nómina en la empresa All Sport Nutrition	
Número de caso de uso: CU-MHE-04	Nombre del caso de uso: eliminar los datos de un registro de horas extra.
Fecha elaboración:	07/01/2026
Descripción caso de uso	Le permite al usuario administrador eliminar un registro de horas extra en el sistema.
Autor caso de uso:	Kendall Salazar Soto
Actores que se relacionan:	Usuario administrador Sistema
Precondiciones:	El usuario administrador debe encontrarse registrado en la base de datos

	El usuario administrador debe tener asignado el permiso para gestionar el proceso. Los datos del empleado deben estar registrados. El sistema debe estar funcionando
Flujo básico del caso de uso	
1. El usuario selecciona <i>Horas extra</i> del módulo de horas extra. 2. El sistema muestra la pantalla de horas extra con todos los empleados activos del sistema. 3. El sistema muestra datos del día actual, como nombre del empleado, fecha, hora de entrada, hora de salida, etc. 4. La vista cuenta con los botones Editar, Eliminar y Agregar. 5. El usuario selecciona el botón <i>Editar</i>. 6. El sistema redirecciona a la vista Actualizar horas extra y muestra el formulario con los datos previamente registrados cargados. 7. El usuario actualiza los registros correspondientes. 8. El sistema valida todos los campos. 9. El usuario selecciona la opción <i>Guardar</i>. 10. El sistema almacena la información en la base de datos. 11. El sistema confirma el registro exitoso. 12. El sistema sale del formulario y regresa a la vista principal del módulo de horas extra. 13. Finaliza el caso de uso exitosamente.	
Flujos alternos	
Flujo alternativo n.º 1 Validación de fecha no futura	• Se verifica que la fecha no sea futura al día presente. • El sistema bloquea los días futuros.
Flujo alternativo n.º 2 Validación de hora	• El sistema verifica que la hora esté dentro del rango de horas trabajadas. • Si no cumple, se muestra un mensaje de advertencia. • Se bloquea el envío.
Flujo alternativo n.º 3 Validación de campos vacíos	• Se valida que se ingrese la información en los campos requeridos obligatorios. • Si no cumple, se muestra un mensaje de advertencia. • Se rechaza el envío.
Flujo alternativo n.º 4 Validación máximo de horas	• Se valida que entre la hora de entrada y la de salida no transcurran más de 8 horas. • Si no cumple, se muestra un mensaje de

	advertencia. Si no cumple, se rechaza el envío.
Flujo alternativo n.º 5 Cancelación de proceso	- Si el usuario decide cancelar la operación antes de guardar, el sistema no registra la información. - Regresa a la pantalla principal de horas extra.
Flujo alternativo n.º 6 Error del sistema	- El sistema no puede registrar o validar debido a un error técnico. - Se muestra un mensaje de error. - El flujo finaliza sin acceso al sistema.
Requerimientos especiales	
Validación de fecha futura, máximo de horas diarias, de horario y de campos requeridos. Todas las acciones deben ser registradas en la tabla auditoría.	
Poscondiciones	
El registro de horas extra del empleado es actualizado correctamente. El registro de la acción queda almacenado en la tabla auditoría.	

Tabla 21*Caso de uso para listar permisos de los empleados*

Prototipo: Sistema web para gestión de pago de nómina en la empresa All Sport Nutrition	
Número de caso de uso: CU-MP-01	Nombre del caso de uso: listar los registros de los permisos de los empleados
Fecha elaboración:	07/01/2026
Descripción caso de uso	Le permite al usuario administrador observar los datos activos del módulo de permisos dentro del sistema.
Autor caso de uso:	Kendall Salazar Soto
Actores que se relacionan:	Usuario administrador Sistema
Precondiciones:	El usuario administrador debe encontrarse registrado en la base de datos El usuario administrador debe tener asignado el

	permiso para gestionar el proceso. Los datos del empleado deben estar registrados. El sistema debe estar funcionando
Flujo básico del caso de uso	
1. El usuario accede a la opción <i>Permisos</i> del módulo de permisos. 2. El sistema muestra la vista con todos los empleados activos en el sistema. 3. El sistema muestra datos correspondientes a los permisos, como nombre del empleado, fecha de inicio del permiso, fecha final del permiso, el estado y el motivo. 4. La vista cuenta con los botones Editar, Eliminar y Agregar.. 5. Finaliza el caso de uso exitosamente.	
Flujos alternos	
Flujo alternativo n.º 1 Validación de empleado activo	• El sistema consulta si el empleado está activo en el sistema. • Si existe, muestra la información del empleado.
Flujo alternativo n.º 2 Botón Editar	• El usuario selecciona el botón <i>Editar</i>. • El sistema redirecciona a la vista Actualizar Datos.
Flujo alternativo n.º 3 Botón Eliminar	• El usuario selecciona el botón <i>Eliminar</i>. • El sistema muestra una ventana de confirmación.
Flujo alternativo n.º 4 Botón Agregar asistencia	• El usuario selecciona el botón <i>Agregar</i>. • El sistema redirecciona a la vista Crear un Registro de Permisos.
Flujo alternativo n.º 5 Error del sistema	• El sistema no puede registrar o validar debido a un error técnico. • Se muestra un mensaje de error. • El flujo finaliza sin acceso al sistema.
Requerimientos especiales	
Validación de empleado activo. Todas las acciones deben ser registradas en la tabla auditoría.	
Poscondiciones	
Acceso a los registros de permisos correspondiente a cada empleado.	

El registro de la acción queda almacenado en la tabla auditoría.

Tabla 22

Caso de uso para agregar un registro de permisos solicitados

Prototipo: Sistema web para gestión de pago de nómina en la empresa All Sport Nutrition	
Número del caso de uso: CU-MP-02	Nombre del caso de uso: agregar un registro de permiso de un empleado
Fecha de elaboración:	07/01/2026
Descripción del caso de uso	Le permite al usuario administrador agregar un nuevo registro de permiso de un empleado
Autor del caso de uso:	Kendall Salazar Soto
Actores que se relacionan:	Usuario administrador Sistema
Precondiciones:	El usuario administrador debe encontrarse registrado en la base de datos El usuario administrador debe tener asignado el permiso para gestionar el proceso. Los datos del empleado deben estar registrados. El sistema debe estar funcionando
Flujo básico del caso de uso	
1. El usuario accede a la opción <i>Permisos</i> en el menú lateral. 2. El sistema muestra la vista con todos los empleados activos en el sistema. 3. El sistema muestra datos referentes al empleado. 4. La vista cuenta con los botones Editar, Eliminar y Agregar. 5. El usuario da clic al botón <i>Agregar</i> en la persona correspondiente. 6. El sistema redirecciona a la vista de agregar permiso y despliega el formulario. 7. El usuario completa los datos requeridos del empleado, como fecha de inicio, fecha de finalización del permiso, tipo de permiso, estado y motivo. 8. El sistema valida todos los campos. 9. El usuario selecciona la opción <i>Guardar</i>. 10. El sistema almacena la información en la base de datos. 11. El sistema confirma el registro exitoso. 12. El sistema sale del formulario y regresa a la vista principal del módulo de permisos.	

13. Finaliza el caso de uso exitosamente.	
Flujos alternos	
Flujo alternativo n.º 1 Validación de fecha no pasada	- Se verifica que la fecha no sea pasada al día presente. - El sistema bloquea los días pasados.
Flujo alternativo n.º 2 Validación de campos vacíos	- Se valida que se ingrese la información en los campos requeridos obligatorios. - Si no cumple, se muestra un mensaje de advertencia y se rechaza el envío.
Flujo alternativo n.º 3 Cancelación de proceso	- Si la persona usuaria decide cancelar la operación antes de guardar, el sistema no registra la información. - Regresa a la pantalla principal de permisos.
Flujo alternativo n.º 4 Error del sistema	- El sistema no puede registrar o validar debido a un error técnico. - Se muestra un mensaje de error. - El flujo finaliza sin acceso al sistema.
Requerimientos especiales	
Validación de fecha futura, máximo de horas diarias, de horario y de campos requeridos. Todas las acciones deben ser registradas en la tabla auditoría.	
Poscondiciones	
Acceso a los registros de permisos correspondiente a cada empleado. El registro de la acción queda almacenado en la tabla auditoría.	

Tabla 23

Caso de uso para actualizar un registro de permisos solicitados

Prototipo: Sistema web para gestión de pago de nómina en la empresa All Sport Nutrition	
Número del caso de uso: CU-MP-03	Nombre del caso de uso: actualizar los datos de un permiso
Fecha de elaboración:	07/01/2026
Descripción del caso de uso	Le permite al usuario administrador actualizar un nuevo

	registro de permiso de un empleado con la información correspondiente y siguiendo las validaciones del sistema.
Autor del caso de uso:	Kendall Salazar Soto
Actores que se relacionan:	Usuario administrador Sistema
Precondiciones:	El usuario administrador debe encontrarse registrado en la base de datos El usuario administrador debe tener asignado el permiso para gestionar el proceso. Los datos del empleado deben estar registrados. El sistema debe estar funcionando
Flujo básico del caso de uso	
1. El usuario selecciona permisos en el menú lateral. 2. El sistema muestra la pantalla de permisos con todos los empleados activos del sistema. 3. El sistema muestra datos referentes al empleado y el permiso. 4. La vista cuenta con los botones Editar, Eliminar y Agregar. 5. El usuario selecciona el botón <i>Editar</i>. 6. El sistema redirecciona a la vista <i>Actualizar Permiso</i> y muestra el formulario con los datos registrados cargados. 7. El usuario actualiza los registros correspondientes. 8. El sistema valida todos los campos. 9. El usuario selecciona la opción <i>Guardar</i>. 10. El sistema almacena la información en la base de datos. 11. El sistema confirma el registro exitoso. 12. El sistema sale del formulario y regresa a la vista principal del módulo de horas extra. 13. Finaliza el caso de uso exitosamente.	
Flujos alternos	
Flujo alternativo n.º 1 Validación de fecha no pasada	• Se verifica que la fecha no sea pasada al día presente. • El sistema bloquea los días pasados.
Flujo alternativo n.º 2 Validación de campos vacíos	• Se valida que se ingrese la información en los campos requeridos obligatorios. • Si no cumple, se muestra un mensaje de advertencia y se rechaza el envío.

Flujo alternativo n.º 3 Cancelación de proceso	• Si la persona usuaria decide cancelar la operación antes de guardar, el sistema no registra la información. • Regresa a la pantalla principal de permisos.
Flujo alternativo n.º 4 Error del sistema	• El sistema no puede registrar o validar debido a un error técnico. • Se muestra un mensaje de error. • El flujo finaliza sin acceso al sistema.
Requerimientos especiales	
Validación de fecha pasada y de campos requeridos. Todas las acciones deben ser registradas en la tabla auditoría.	
Poscondiciones	
Se actualizan los datos del permiso de un empleado seleccionado. El registro de la acción queda almacenado en la tabla auditoría.	

Tabla 24

Caso de uso para eliminar un registro de permisos solicitados

Prototipo: Sistema web para gestión de pago de nómina en la empresa All Sport Nutrition	
Número del caso de uso: CU-MP-03	Nombre del caso de uso: eliminar un registro de permiso de un empleado
Fecha de elaboración:	07/01/2026
Descripción del caso de uso	Le permite al usuario administrador eliminar un nuevo registro de permiso de un empleado en el sistema.
Autor del caso de uso:	Kendall Salazar Soto
Actores que se relacionan:	Usuario administrador Sistema
Precondiciones:	El usuario administrador debe encontrarse registrado en la base de datos El usuario administrador debe tener asignado el permiso para gestionar el proceso. Los datos del empleado deben estar registrados.

	El sistema debe estar funcionando
Flujo básico del caso de uso	
1. El usuario selecciona permisos en el menú lateral. 2. El sistema muestra la pantalla de todos los empleados activos del sistema. 3. El sistema muestra datos referentes al empleado y los permisos vinculados a este. 4. La vista cuenta con los botones Editar, Eliminar y Agregar. 5. El usuario selecciona el botón <i>Eliminar</i> del empleado del que requiere eliminar el registro. 6. El usuario confirma la eliminación. 7. El sistema actualiza el estado del empleado en la base de datos, actualizado como inactivo. 8. El sistema registra la acción en la tabla de auditoría. 9. El sistema regresa a la vista principal de permisos. 10. Finaliza el caso de uso exitosamente.	
Flujos alternos	
Flujo alternativo n.º 1 Validación de permisos	• El sistema valida si la persona usuaria tiene los permisos correspondientes para realizar la acción. • Si la persona usuaria no tiene los permisos correspondientes el botón eliminar se encuentra desactivado.
Flujo alternativo n.º 2 Validación de confirmación de eliminación	• El sistema muestra un mensaje de confirmación para eliminar al empleado. • Si se confirma la eliminación se realiza esta. • El sistema regresa a la vista principal de permisos.
Flujo alternativo n.º 3 Cancelación de proceso	• El usuario desea cancelar la acción. • En la ventana de confirmación selecciona la opción <i>No</i>. • El sistema cierra el cuadro de diálogo y no realiza ninguna acción para el empleado. • Retorna a la vista de permisos.
Flujo alternativo n.º 4 Error del sistema	• El sistema no puede registrar o validar debido a un error técnico. • Se muestra un mensaje de error. • El flujo finaliza sin acceso al sistema.
Requerimientos especiales	
El sistema debe garantizar que se haga un borrado lógico y no físico para preservar la trazabilidad histórica.	

Todas las acciones deben ser registradas en la tabla auditoría.
Poscondiciones
El sistema refleja los cambios en la lista de permisos actual. El registro de la acción queda almacenado en la tabla auditoría.

Tabla 25

Caso de uso para listar el registro de incapacidades

Prototipo: Sistema web para gestión de pago de nómina en la empresa All Sport Nutrition	
Número del caso de uso: CU-MI-01	Nombre del caso de uso: listar los registros de incapacidades de los empleados.
Fecha de elaboración:	07/01/2026
Descripción del caso de uso	Le permite al usuario administrador observar todas las incapacidades activas por empleado en el sistema.
Autor del caso de uso:	Kendall Salazar Soto
Actores que se relacionan:	Usuario administrador Sistema
Precondiciones:	El usuario administrador debe encontrarse registrado en la base de datos El usuario administrador debe tener asignado el permiso para gestionar el proceso. El sistema debe estar funcionando
Flujo básico del caso de uso	
1. El usuario accede a la opción <i>Incapacidades</i> del módulo de incapacidades. 2. El sistema muestra la vista con todos los empleados activos. 3. El sistema muestra datos correspondientes a los permisos 4. La vista cuenta con los botones Editar, Eliminar y Agregar. 5. Finaliza el caso de uso exitosamente.	
Flujos alternos	
Flujo alternativo n.º 1	• El sistema consulta si el empleado está activo en el

Validación de empleado activo	sistema. • Si existe, muestra la información del empleado.
Flujo alternativo n.º 2 Botón de editar	• El usuario selecciona el botón <i>Editar</i>. • El sistema redirige a la vista de Actualizar Datos.
Flujo alternativo n.º 3 Botón de eliminar	• El usuario selecciona el botón <i>Eliminar</i>. • El sistema muestra una ventana de confirmación.
Flujo alternativo n.º 4 Botón de agregar asistencia	• El usuario selecciona el botón <i>Agregar</i>. • El sistema redirige a la vista Crear un registro de incapacidad.
Flujo alternativo n.º 5 Error del sistema	• El sistema no puede registrar o validar debido a un error técnico. • Se muestra un mensaje de error. • El flujo finaliza sin acceso al sistema.
Requerimientos especiales	
Validación de empleado activo. Todas las acciones deben ser registradas en la tabla auditoría.	
Poscondiciones	
El usuario tiene acceso a la información correspondiente de cada empleado con respecto a las incapacidades. El registro de la acción queda almacenado en la tabla auditoría.	

Tabla 26

Caso de uso para agregar un registro de incapacidades

Prototipo: Sistema web para gestión de pago de nómina en la empresa All Sport Nutrition	
Número del caso de uso: CU-MI-02	Nombre del caso de uso: agregar un registro de incapacidad de un empleado
Fecha de elaboración:	07/01/2026
Descripción del caso de uso	Le permite al usuario administrador agregar un nuevo registro de incapacidad a un colaborador.

Autor del caso de uso:	Kendall Salazar Soto
Actores que se relacionan:	Usuario administrador Sistema
Precondiciones:	El usuario administrador debe encontrarse registrado en la base de datos El usuario administrador debe tener asignado el permiso para gestionar el proceso. El sistema debe estar funcionando
Flujo básico del caso de uso	
1. El usuario accede a la opción <i>Incapacidades</i> del módulo de incapacidades. 2. El sistema muestra la vista con todos los empleados activos. 3. El sistema muestra datos como nombre del empleado, fecha de inicio de la incapacidad, fecha de finalización de la incapacidad, tipo de incapacidad y la cantidad de días de incapacidad. 4. La vista cuenta con los botones Editar, Eliminar y Agregar. 5. El usuario da clic al botón <i>Agregar</i> en la persona correspondiente. 6. El sistema redirecciona a la vista de Agregar incapacidad y despliega el formulario. 7. El usuario completa los datos requeridos del empleado, como fecha de inicio, fecha de finalización, tipo de incapacidad, estado y motivo. 8. El sistema valida todos los campos. 9. El usuario selecciona la opción <i>Guardar</i>. 10. El sistema almacena la información en la base de datos. 11. El sistema confirma el registro exitoso. 12. El sistema sale del formulario y regresa a la vista principal del módulo de incapacidades. 13. Finaliza el caso de uso exitosamente.	
Flujos alternos	
Flujo alternativo n.º 1 Validación de campos vacíos	• Se valida que se ingrese la información en los campos requeridos obligatorios. • Si no cumple, se muestra un mensaje de advertencia y se rechaza el envío.
Flujo alternativo n.º 2 Cancelación de proceso	• Si la persona usuaria decide cancelar la operación antes de guardar, el sistema no registra la información y regresa a la pantalla principal de permisos.
Flujo alternativo n.º 3 Error del sistema	• El sistema no puede registrar o validar debido a un error técnico. • Se muestra un mensaje de error.

	El flujo finaliza sin acceso al sistema.
Requerimientos especiales	
Validación de fecha futura, máximo de horas diarias, de horario y de campos requeridos. Todas las acciones deben ser registradas en la tabla auditoría.	
Poscondiciones	
La incapacidad del empleado queda registrada. El registro de la acción queda almacenado en la tabla auditoría.	

Tabla 27*Caso de uso para actualizar un registro de incapacidades*

Prototipo: Sistema web para gestión de pago de nómina en la empresa All Sport Nutrition	
Número del caso de uso: CU-MI-03	Nombre del caso de uso: actualizar un registro de incapacidad de un empleado
Fecha de elaboración:	07/01/2026
Descripción del caso de uso	Le permite al usuario administrador actualizar un registro de incapacidad a un colaborador.
Autor del caso de uso:	Kendall Salazar Soto
Actores que se relacionan:	Usuario administrador Sistema
Precondiciones:	El usuario administrador debe encontrarse registrado en la base de datos El usuario administrador debe tener asignado el permiso para gestionar el proceso. El sistema debe estar funcionando
Flujo básico del caso de uso	
- El usuario selecciona <i>Incapacidades</i> del módulo de incapacidad. - El sistema muestra la pantalla de incapacidades con todos los empleados activos del sistema. - El sistema muestra datos como nombre del empleado, fecha de inicio de incapacidad, fecha de finalización de la incapacidad, tipo de incapacidad y cantidad de días de incapacidad.	

4. La vista cuenta con los botones Editar, Eliminar y Agregar. 5. El usuario selecciona el botón <i>Editar</i>. 6. El sistema redirecciona a la vista Actualizar permiso y muestra el formulario con los datos registrados cargados. 7. El usuario actualiza los registros correspondientes. 8. El sistema valida todos los campos. 9. El usuario selecciona la opción <i>Guardar</i>. 10. El sistema almacena la información con la base de datos. 11. El sistema confirma el registro exitoso. 12. El sistema sale del formulario y regresa a la vista principal del módulo incapacidades. 13. Finaliza el caso de uso exitosamente.	
Flujos alternos	
Flujo alternativo n.º 1 Validación de campos vacíos	- Se valida que se ingrese la información en los campos requeridos obligatorios. - Si no cumple, se muestra un mensaje de advertencia y se rechaza el envío.
Flujo alternativo n.º 2 Cancelación de proceso	Si la persona usuaria decide cancelar la operación antes de guardar, el sistema no registra la información y regresa a la pantalla principal de permisos.
Flujo alternativo n.º 3 Error del sistema	- El sistema no puede registrar o validar debido a un error técnico. - Se muestra un mensaje de error. - El flujo finaliza sin acceso al sistema.
Requerimientos especiales	
Cancelación de proceso. Todas las acciones deben ser registradas en la tabla auditoría.	
Poscondiciones	
La incapacidad del empleado queda registrada. El registro de la acción queda almacenado en la tabla auditoría.	

Tabla 28*Caso de uso para eliminar un registro de incapacidades*

Prototipo: Sistema web para gestión de pago de nómina en la empresa All Sport Nutrition	
Número del caso de uso: CU-MI-04	Nombre del caso de uso: eliminar un registro de incapacidad de un empleado
Fecha de elaboración:	07/01/2026
Descripción del caso de uso	Le permite al usuario administrador actualizar un registro de incapacidad a un colaborador.
Autor del caso de uso:	Kendall Salazar Soto
Actores que se relacionan:	Usuario administrador Sistema
Precondiciones:	El usuario administrador debe encontrarse registrado en la base de datos El usuario administrador debe tener asignado el permiso para gestionar el proceso. El sistema debe estar funcionando
Flujo básico del caso de uso	
1. El usuario selecciona <i>Incapacidad</i> en el menú lateral. 2. El sistema muestra la pantalla de incapacidad con todos los empleados activos. 3. El sistema muestra datos como nombre del empleado, fecha de inicio, fecha de finalización, tipo de incapacidad, tipo y cantidad de días. 4. La vista cuenta con los botones Editar, Eliminar y Agregar. 5. El usuario selecciona el botón de <i>Eliminar</i> del empleado que requiere eliminar el registro. 6. El usuario confirma la eliminación. 7. El sistema actualiza el estado del empleado en la base de datos, actualizándolo como inactivo. 8. El sistema registra la acción en la tabla de auditoría. 9. El sistema regresa a la vista principal de incapacidades. 10. Finaliza el caso de uso exitosamente.	
Flujos alternos	
Flujo alternativo n.º 1 Validación de permisos	• El sistema valida si la persona usuaria tiene los permisos correspondientes para realizar la acción. • Si la persona usuaria no tiene los permisos

	correspondientes, el botón de <i>Eliminar</i> se encuentra desactivado.
Flujo alternativo n.º 2 Validación de confirmación de eliminación	- El sistema muestra un mensaje de confirmación para eliminar al empleado. - Si se confirma la eliminación, se realiza la acción de eliminación. - El sistema regresa a la vista principal de permisos.
Flujo alternativo n.º 3 Cancelar la operación	- El usuario desea cancelar la acción. - En la ventana de confirmación selecciona la opción <i>No</i>. - El sistema cierra el cuadro de diálogo y no realiza ninguna acción para el empleado. - Retorna a la vista de permisos.
Flujo alternativo n.º 4 Error del sistema	- El sistema no puede registrar o validar debido a un error técnico. - Se muestra un mensaje de error. - El flujo finaliza sin acceso al sistema.
Requerimientos especiales	
El sistema debe garantizar que se haga un borrado lógico y no físico para preservar la trazabilidad histórica. Todas las acciones deben ser registradas en la tabla auditoría.	
Poscondiciones	
Se elimina el registro de incapacidad de la persona y queda marcado como inactivo en el sistema. Se conserva toda la información histórica de incapacidades del empleado. El registro de la acción queda almacenado en la tabla auditoría. El sistema refleja los cambios.	

Tabla 29*Caso de uso para listar registros de liquidaciones*

Prototipo: Sistema web para gestión de pago de nómina en la empresa All Sport Nutrition	
Número del caso de uso: CU-ML-01	Nombre del caso de uso: listar los registros de liquidaciones de los empleados
Fecha de elaboración:	07/01/2026

Descripción del caso de uso	Le permite al usuario administrador visualizar los registros de liquidaciones de las personas colaboradoras de la organización.
Autor del caso de uso:	Kendall Salazar Soto
Actores que se relacionan:	Usuario administrador Sistema
Precondiciones:	El usuario administrador debe encontrarse registrado en la base de datos El usuario administrador debe tener asignado el permiso para gestionar el proceso. El sistema debe estar funcionando
Flujo básico del caso de uso	
1. El usuario accede a la opción <i>Liquidaciones</i> en el menú lateral. 2. El sistema muestra la vista con todos los empleados con liquidaciones en el sistema. 3. El sistema muestra datos correspondientes a las liquidaciones realizadas, como nombre del empleado, fecha de ingreso, fecha de finalización laboral; el tipo, el motivo y los detalles. 4. El usuario puede elegir acciones disponibles de editar, eliminar, agregar y regresar. 5. Finaliza el caso de uso exitosamente.	
Flujos alternos	
Flujo alternativo n.º 1 Validación de empleado activo	• El sistema consulta si el empleado está activo en el sistema. • Si existe, muestra la información del empleado.
Flujo alternativo n.º 2 Acción botón de editar	• El sistema consulta si el empleado está activo en el sistema. • Si existe, muestra la información del empleado.
Flujo alternativo n.º 3 Acción botón de eliminar	• El usuario selecciona el botón <i>Eliminar</i>. • El sistema muestra una ventana de confirmación.
Flujo alternativo n.º 4 Acción botón de agregar	• El usuario selecciona el botón <i>Agregar</i>. • El sistema redirecciona a la vista Crear un registro de liquidación.
Flujo alternativo n.º 5 Error del sistema	• El sistema no puede registrar o validar debido a un error técnico.

	• Se muestra un mensaje de error. • El flujo finaliza sin acceso al sistema.
Requerimientos especiales	
Todas las acciones deben ser registradas en la tabla auditoría.	
Poscondiciones	
El usuario visualiza la información histórica y puede acceder a otras acciones que se relacionan con ella.	

Tabla 30*Caso de uso para agregar un registro de liquidación*

Prototipo: Sistema web para gestión de pago de nómina en la empresa All Sport Nutrition	
Número del caso de uso: CU-ML-02	Nombre del caso de uso: agregar un registro de liquidación de un empleado
Fecha de elaboración:	07/01/2026
Descripción del caso de uso	Le permite al usuario administrador actualizar un registro de liquidación a un colaborador.
Autor del caso de uso:	Kendall Salazar Soto
Actores que se relacionan:	Usuario administrador Sistema
Precondiciones:	El usuario administrador debe encontrarse registrado en la base de datos El usuario administrador debe tener asignado el permiso para gestionar el proceso. El sistema debe estar funcionando
Flujo básico del caso de uso	
1. El usuario accede al módulo de liquidaciones. 2. Selecciona la opción <i>Agregar liquidación</i>. 3. El sistema despliega el formulario de nueva liquidación. 4. El usuario completa los datos requeridos del empleado, como fecha de inicio, fecha de finalización, tipo de liquidación, estado y motivo.	

5. El sistema valida todos los campos obligatorios y la coherencia de fechas. 6. El sistema calcula automáticamente los montos correspondientes (aguinaldo, vacaciones, cesantía, salarios promediados). 7. El usuario selecciona la opción <i>Guardar</i> . 8. El sistema almacena la información en la base de datos. 9. El sistema confirma el registro exitoso. 10. El sistema regresa a la vista principal del módulo de liquidaciones. 11. Finaliza el caso de uso exitosamente.	
Flujos alternos	
Flujo alternativo n.º 1 Validación de campos vacíos	- Se valida que se ingrese la información en los campos requeridos obligatorios. - Si no cumple, se muestra un mensaje de advertencia. - Se rechaza el envío.
Flujo alternativo n.º 2 Validación de fechas futuras	- Se valida que la fecha de salida y la fecha en la que se realiza la liquidación no sea futura. - Si no cumple, se muestra un mensaje de advertencia - Se rechaza el envío.
Flujo alternativo n.º 3 Validación de fecha de ingreso	- Se valida que la fecha de finalización sea después de la fecha de ingreso. - Si no cumple, se muestra un mensaje de error. - Se rechaza el envío.
Flujo alternativo n.º 4 Cancelación de proceso	- Si la persona usuaria decide cancelar la operación antes de guardar, el sistema no registra la información. - Se regresa a la pantalla principal de permisos.
Flujo alternativo n.º 5 Error del sistema	- El sistema no puede registrar o validar debido a un error técnico. - Se muestra un mensaje de error. - El flujo finaliza sin acceso al sistema.
Requerimientos especiales	
Validación de fecha futura y máximo de horas diarias, de horario y de campos requeridos. Todas las acciones deben ser registradas en la tabla auditoría.	
Poscondiciones	

El registro de liquidación queda guardado en el sistema.
 Queda almacenado el detalle de cálculos realizados (aguinaldo, vacaciones, cesantía, etc.).
 El registro de la acción queda almacenado en la tabla auditoría.
 El sistema refleja los cambios.

Tabla 31

Caso de uso para actualizar un registro de liquidación

Prototipo: Sistema web para gestión de pago de nómina en la empresa All Sport Nutrition	
Número del caso de uso: CU-ML-03	Nombre del caso de uso: actualizar un registro de liquidación de un colaborador
Fecha de elaboración:	07/01/2026
Descripción del caso de uso	Le permite al usuario administrador actualizar un registro de liquidación a un colaborador.
Autor del caso de uso:	Kendall Salazar Soto
Actores que se relacionan:	Usuario administrador Sistema
Precondiciones:	El usuario administrador debe encontrarse registrado en la base de datos El usuario administrador debe tener asignado el permiso para gestionar el proceso. El sistema debe estar funcionando
Flujo básico del caso de uso	
1. El usuario accede al módulo de liquidaciones. 2. El usuario selecciona un registro de liquidación y pulsa la opción <i>Editar</i>. 3. El sistema despliega el formulario con los datos de la liquidación existente. 4. El usuario modifica los campos necesarios como fecha de inicio, fecha de finalización, tipo de liquidación, estado y motivo. 5. El sistema valida todos los campos requeridos. 6. El sistema recalcula automáticamente los montos (aguinaldo, vacaciones, cesantía, salario promedio). 7. El usuario selecciona la opción <i>Guardar</i>. 8. El sistema almacena la información en la base de datos. 9. El sistema confirma el registro exitoso. 10. El sistema retorna a la vista principal del módulo de liquidaciones.	

11. Finaliza el caso de uso exitosamente.	
Flujos alternos	
Flujo alternativo n.º 1 Validación de campos vacíos	- Se valida que se ingrese la información en los campos requeridos obligatorios. - Si no cumple, se muestra un mensaje de advertencia. - Se rechaza el envío.
Flujo alternativo n.º 2 Validación de fechas futuras	- Se valida que la fecha de salida y la fecha en la que se realiza la liquidación no sea futura. - Si no cumple, se muestra un mensaje de advertencia - Se rechaza el envío.
Flujo alternativo n.º 3 Validación de fecha de ingreso	- Se valida que la fecha de finalización sea después de la fecha de ingreso. - Si no cumple, se muestra un mensaje de error. - Se rechaza el envío.
Flujo alternativo n.º 4 Error de cálculos automáticos	- Si al realizar un cálculo automático (cálculo de salarios promedio por mes y día de los últimos 6 meses, cálculo de vacaciones y cálculo de aguinaldos) falla. - El sistema muestra un mensaje de error. - No se ejecuta ninguna modificación de la información en la base de datos.
Flujo alternativo n.º 5 Cancelación de proceso	- Si la persona usuaria decide cancelar la operación antes de guardar, el sistema no registra la información. - Se regresa a la pantalla principal de permisos.
Flujo alternativo n.º 6 Error del sistema	- El sistema no puede registrar o validar debido a un error técnico. - Se muestra un mensaje de error. - El flujo finaliza sin acceso al sistema.
Requerimientos especiales	
Validación de fecha futura y máximo de horas diarias, de horario y de campos requeridos. Los cálculos de aguinaldo, vacaciones y cesantía se ajustan a la nueva información. Todas las acciones deben ser registradas en la tabla auditoría.	

Poscondiciones
El registro de liquidación queda actualizado. Los cálculos de aguinaldo, vacaciones y cesantía se ajustan a la nueva información. El registro de la acción queda almacenado en la tabla auditoría. El sistema refleja los cambios.

Tabla 32

Caso de uso para eliminar un registro de liquidación

Prototipo: Sistema web para gestión de pago de nómina en la empresa All Sport Nutrition	
Número del caso de uso: CU-ML-04	Nombre del caso de uso: eliminar un registro de liquidación de un colaborador
Fecha de elaboración:	07/01/2026
Descripción del caso de uso	Le permite al usuario administrador eliminar un registro de liquidación a un colaborador.
Autor del caso de uso:	Kendall Salazar Soto
Actores que se relacionan:	Usuario administrador Sistema
Precondiciones:	El usuario administrador debe encontrarse registrado en la base de datos El usuario administrador debe tener asignado el permiso para gestionar el proceso. El sistema debe estar funcionando
Flujo básico del caso de uso	
1. El usuario selecciona <i>Liquidaciones</i> del módulo de liquidaciones. 2. El sistema muestra la pantalla de liquidaciones con todos los empleados activos del sistema. 3. El sistema muestra datos como nombre del empleado, fecha de ingreso, fecha de salida de la empresa, fecha en la que se realiza la liquidación, tipo de liquidación, motivo, monto total de la liquidación y detalles. 4. La vista cuenta con los botones Editar, Eliminar y Agregar. 5. El usuario selecciona el botón <i>Eliminar</i> del empleado que requiere eliminar el registro. 6. El usuario confirma la eliminación. 7. El sistema actualiza el estado del registro de liquidación en la base de datos, marcado como inactivo. 8. El sistema registra la acción en la tabla de auditoría.	

9. El sistema regresa a la vista principal de Liquidaciones. 10. Finaliza el caso de uso exitosamente.	
Flujos alternos	
Flujo alternativo n.º 1 Validación de permisos	- El sistema valida si la persona usuaria tiene los permisos correspondientes para realizar la acción. - Si la persona usuaria no tiene los permisos correspondientes, el botón de eliminar se encuentra desactivado.
Flujo alternativo n.º 2 Validación de confirmación de eliminación	- El sistema muestra un mensaje de confirmación para eliminar al empleado. - Si se confirma la eliminación se realiza la acción de eliminación. - El sistema regresa a la vista principal de Liquidaciones.
Flujo alternativo n.º 3 Cancelar la operación	- El usuario manifiesta su intención de cancelar la acción. - En la ventana de confirmación, selecciona la opción <i>No</i>. - El sistema cierra el cuadro de diálogo sin ejecutar ninguna acción sobre el empleado. - Finalmente, el sistema retorna a la vista de permisos.
Flujo alternativo n.º 4 Error del sistema	- El sistema no puede registrar o validar debido a un error técnico. - Se muestra un mensaje de error. - El flujo finaliza sin acceso al sistema.
Requerimientos especiales	
El sistema debe garantizar que se haga un borrado lógico y no físico para preservar la trazabilidad histórica. Todas las acciones deben ser registradas en la tabla auditoría.	
Poscondiciones	
El registro de liquidación queda marcado como inactivo en el sistema. El registro de la acción queda almacenado en la tabla auditoría. El sistema refleja los cambios.	

Tabla 33*Caso de uso para listar la cantidad de días disponibles de vacaciones*

Prototipo: Sistema web para gestión de pago de nómina en la empresa All Sport Nutrition	
Número del caso de uso: CU-MV-01	Nombre del caso de uso: listar los registros de los días disponibles de las personas colaboradoras
Fecha de elaboración:	07/01/2026
Descripción del caso de uso	Le permite al usuario administrador observar los días de vacaciones disponibles de cada colaborador
Autor del caso de uso:	Kendall Salazar Soto
Actores que se relacionan:	Usuario administrador Sistema
Precondiciones:	El usuario administrador debe encontrarse registrado en la base de datos El usuario administrador debe tener asignado el permiso para gestionar el proceso. El sistema debe estar funcionando
Flujo básico del caso de uso	
1. El usuario entra a la opción <i>Vacaciones</i> en el menú lateral. 2. El sistema muestra la vista con todos los empleados del sistema y con sus días de vacaciones disponibles. 3. El sistema muestra el nombre de los empleados con su cantidad de días disponibles de disfrute de vacaciones. 4. El usuario puede elegir acciones disponibles de solicitudes de vacaciones y regresar. 5. Finaliza el caso de uso exitosamente.	
Flujos alternos	
Flujo alternativo n.º 1 Validación de colaborador activo	• El sistema consulta si el empleado está activo en el sistema. • Si existe, muestra la información del empleado.
Flujo alternativo n.º 2 Acción botón de solicitud de vacaciones	• El usuario selecciona el botón <i>Solicitud de vacaciones</i> en el registro de la persona que solicita los días. • El sistema redirecciona a la vista con un formulario para ingresar los datos necesarios para la solicitud.

Flujo alternativo n.º 3 Error del sistema	- El sistema no puede registrar o validar debido a un error técnico. - Se muestra un mensaje de error. - El flujo finaliza sin acceso al sistema.
Requerimientos especiales	
Todas las acciones deben ser registradas en la tabla auditoría.	
Poscondiciones	
El usuario visualiza la información actualizada de los días de vacaciones disponibles y puede acceder a otras acciones que se relacionan con ella.	

Tabla 34

Caso de uso para listar solicitudes de vacaciones

Prototipo: Sistema web para gestión de pago de nómina en la empresa All Sport Nutrition	
Número del caso de uso: CU-MV-02	Nombre del caso de uso: listar los registros de solicitudes de vacaciones de las personas colaboradoras
Fecha de elaboración:	07/01/2026
Descripción del caso de uso	Le permite al usuario administrador observar los días de vacaciones disponibles de cada colaborador
Autor del caso de uso:	Kendall Salazar Soto
Actores que se relacionan:	Usuario administrador Sistema
Precondiciones:	El usuario administrador debe encontrarse registrado en la base de datos El usuario administrador debe tener asignado el permiso para gestionar el proceso. El sistema debe estar funcionando
Flujo básico del caso de uso	
- El usuario accede a la opción <i>Solicitud de vacaciones</i>. - El sistema muestra la vista del empleado seleccionado.	

3. El sistema muestra una lista de los registros de solicitudes de vacaciones y su estado. 4. El usuario puede elegir acciones disponibles de editar, eliminar o agregar. 5. Finaliza el caso de uso exitosamente.	
Flujos alternos	
Flujo alternativo n.º 1 Validación de colaborador activo	- El sistema consulta si el empleado está activo en el sistema. - Si existe, muestra la información del empleado.
Flujo alternativo n.º 2 Acción botón de editar	- El usuario selecciona el botón <i>Editar</i>. - El sistema redirecciona a la vista de Actualizar datos.
Flujo alternativo n.º 3 Acción botón de eliminar	- El usuario selecciona el botón <i>Eliminar</i>. - El sistema muestra una ventana de confirmación.
Flujo alternativo n.º 4 Acción botón de agregar solicitud de vacaciones	- El usuario selecciona el botón <i>Agregar</i>. - El sistema redirecciona a la vista Crear solicitud de vacaciones.
Flujo alternativo n.º 5 Error del sistema	- El sistema no puede registrar o validar debido a un error técnico. - Se muestra un mensaje de error. - El flujo finaliza sin acceso al sistema.
Requerimientos especiales	
Todas las acciones deben ser registradas en la tabla auditoría.	
Poscondiciones	
El usuario visualiza la información del historial de solicitudes de vacaciones y puede acceder a otras acciones que se relacionan con ella.	

Tabla 35*Caso de uso para agregar un registro de vacaciones*

Prototipo: Sistema web para gestión de pago de nómina en la empresa All Sport Nutrition	
Número del caso de uso: CU-MV-03	Nombre del caso de uso: agregar un registro de solicitud de vacaciones de un colaborador

Fecha de elaboración:	07/01/2026
Descripción del caso de uso	Le permite al usuario administrador observar los días de vacaciones disponibles de cada colaborador
Autor del caso de uso:	Kendall Salazar Soto
Actores que se relacionan:	Usuario administrador Sistema
Precondiciones:	El usuario administrador debe encontrarse registrado en la base de datos El usuario administrador debe tener asignado el permiso para gestionar el proceso. El sistema debe estar funcionando
Flujo básico del caso de uso	
1. Selecciona la opción <i>Agregar solicitud</i>. 2. El sistema despliega el formulario de agregar solicitud de vacaciones. 3. El usuario completa los datos requeridos del empleado, la fecha de inicio de vacaciones, la fecha en la que finalizan las vacaciones, la cantidad de días solicitados, el estado (aprobado, pendiente o rechazado) y las observaciones. 4. El sistema valida todos los campos obligatorios y la coherencia de fechas. 5. El usuario selecciona la opción <i>Guardar</i>. 6. El sistema almacena la información en la base de datos. 7. El sistema confirma el registro exitoso. 8. El sistema regresa a la vista principal del historial de solicitudes de vacaciones del empleado. 9. Finaliza el caso de uso exitosamente.	
Flujos alternos	
Flujo alternativo n.º 1 Validación de campos vacíos	• Se valida que se ingrese la información en los campos requeridos obligatorios. • Si no cumple, se muestra un mensaje de advertencia. • Se rechaza el envío.
Flujo alternativo n.º 2 Validación de fecha de inicio	• Se valida que la fecha de finalización sea después de la fecha de inicio de vacaciones. • Si no cumple, se muestra un mensaje de error. • Se rechaza el envío.

Flujo alternativo n.º 3 Validación de días disponibles	- El sistema valida que la solicitud de días solicitados no sea mayor que la cantidad de días disponibles. - Si no cumple, el sistema muestra una advertencia. - El sistema bloqueó el envío.
Flujo alternativo n.º 4 Cancelación de proceso	- Si la persona usuaria decide cancelar la operación antes de guardar, el sistema no registra la información. - Regresa a la pantalla principal del módulo.
Flujo alternativo n.º 5 Error del sistema	- El sistema no puede registrar o validar debido a un error técnico. - Se muestra un mensaje de error. - El flujo finaliza sin acceso al sistema.
Requerimientos especiales	
Validación de fecha futura, máximo de días disponibles y de campos requeridos. Todas las acciones deben ser registradas en la tabla auditoría.	
Poscondiciones	
El registro de solicitud de vacaciones queda guardado en el sistema. La acción queda registrada en la tabla de auditoría.	

Tabla 36

Caso de uso para actualizar un registro de solicitud de vacaciones

Prototipo: Sistema web para gestión de pago de nómina en la empresa All Sport Nutrition	
Número del caso de uso: CU-MV-04	Nombre del caso de uso: actualizar un registro de solicitud de vacaciones de un colaborador
Fecha de elaboración:	07/01/2026
Descripción del caso de uso	Le permite al usuario administrador observar los días de vacaciones disponibles de cada colaborador
Autor del caso de uso:	Kendall Salazar Soto
Actores que se relacionan:	Usuario administrador Sistema

Precondiciones:	El usuario administrador debe encontrarse registrado en la base de datos El usuario administrador debe tener asignado el permiso para gestionar el proceso. El sistema debe estar funcionando
Flujo básico del caso de uso	
1. Selecciona la opción <i>Editar solicitud</i>. 2. El sistema despliega el formulario de editar la solicitud de vacaciones. 3. El usuario completa los datos requeridos del empleado, la fecha de inicio de vacaciones, la fecha en la que finaliza las vacaciones, la cantidad de días solicitados, el estado (aprobado, pendiente o rechazado) y las observaciones. 4. El sistema valida todos los campos obligatorios y la coherencia de fechas. 5. El sistema valida todos los campos requeridos. 6. El usuario selecciona la opción <i>Actualizar</i>. 7. El sistema almacena la información en la base de datos. 8. El sistema confirma el registro exitoso. 9. El sistema retorna a la vista principal del historial de solicitudes de vacaciones del empleado. 10. Finaliza el caso de uso exitosamente.	
Flujos alternos	
Flujo alternativo n.º 1 Validación de campos vacíos	• Se valida que se ingrese la información en los campos requeridos obligatorios. • Si no cumple, se muestra un mensaje de advertencia. • Se rechaza el envío.
Flujo alternativo n.º 2 Validación de fecha de inicio	• Se valida que la fecha de finalización sea después de la fecha de inicio de vacaciones. • Si no cumple, se muestra un mensaje de error. • Se rechaza el envío.
Flujo alternativo n.º 3 Validación de días disponibles	• El sistema valida que la solicitud de días solicitados no sea mayor que la cantidad de días disponibles. • Si no cumple, el sistema muestra una advertencia. • El sistema bloqueó el envío.
Flujo alternativo n.º 4 Cancelación de proceso	• Si la persona usuaria decide cancelar la operación antes de guardar, el sistema no registra la información. • Regresa a la pantalla principal del módulo.

Flujo alternativo n.º 5 Error del sistema	- El sistema no puede registrar o validar debido a un error técnico. - Se muestra un mensaje de error. - El flujo finaliza sin acceso al sistema.
Requerimientos especiales	
Validación de fecha futura, máximo de días disponibles y de campos requeridos. Todas las acciones deben ser registradas en la tabla auditoría.	
Poscondiciones	
El registro de solicitud de vacaciones queda guardado en el sistema. La acción queda registrada en la tabla de auditoría.	

Tabla 37

Caso de uso para actualizar un registro de solicitud de vacaciones

Prototipo: Sistema web para gestión de pago de nómina en la empresa All Sport Nutrition	
Número del caso de uso: CU-MV-05	Nombre del caso de uso: eliminar un registro de solicitud de vacaciones de un colaborador
Fecha de elaboración:	07/01/2026
Descripción del caso de uso	Le permite al usuario administrador observar los días de vacaciones disponibles de cada colaborador
Autor del caso de uso:	Kendall Salazar Soto
Actores que se relacionan:	Usuario administrador Sistema
Precondiciones:	El usuario administrador debe encontrarse registrado en la base de datos El usuario administrador debe tener asignado el permiso para gestionar el proceso. El sistema debe estar funcionando
Flujo básico del caso de uso	
1. El usuario selecciona el botón <i>Eliminar</i> del empleado que requiere eliminar del registro.	

2. El usuario confirma la eliminación. 3. El sistema actualiza el estado del registro de liquidación en la base de datos y lo marca como inactivo. 4. El sistema registra la acción en la tabla de auditoría. 5. El sistema regresa a la vista principal del historial de solicitudes de vacaciones del empleado. 6. Finaliza el caso de uso exitosamente.	
Flujos alternos	
Flujo alternativo n.º 1 Validación de permisos	- El sistema valida si la persona usuaria tiene los permisos correspondientes para realizar la acción. - Si la persona usuaria no tiene los permisos correspondientes el botón de eliminar se encuentra desactivado.
Flujo alternativo n.º 2 Validación de confirmación de eliminación	- El sistema muestra un mensaje de confirmación para eliminar al empleado. - Si se confirma la eliminación se realiza la acción de eliminación. - El sistema regresa a la vista principal de liquidaciones.
Flujo alternativo n.º 3 Validación de días disponibles	- El sistema valida que la solicitud de días solicitados no sea mayor que la cantidad de días disponibles. - Si no cumple, el sistema muestra una advertencia. - El sistema bloqueó el envío.
Flujo alternativo n.º 4 Cancelar la operación	- El usuario desea cancelar la acción. - En la ventana de confirmación selecciona la opción <i>No</i>. - El sistema cierra el cuadro de diálogo y no realiza ninguna acción sobre el empleado. - Retorna a la vista de permisos.
Flujo alternativo n.º 5 Error del sistema	- El sistema no puede registrar o validar debido a un error técnico. - Se muestra un mensaje de error. - El flujo finaliza sin acceso al sistema.
Requerimientos especiales	
El sistema debe garantizar que se haga un borrado lógico y no físico para preservar la trazabilidad histórica. Todas las acciones deben ser registradas en la tabla auditoría.	

Poscondiciones
El registro de solicitud de vacaciones queda marcado como inactivo en el sistema. Se conserva toda la información histórica de solicitud de vacaciones de un empleado. Se conserva toda la información histórica de vacaciones del empleado. La acción queda registrada en la tabla de auditoría.

Tabla 38*Caso de uso del cálculo de aguinaldo*

Prototipo: Sistema web para gestión de pago de nómina en la empresa All Sport Nutrition	
Número del caso de uso: CU-MAG-01	Nombre del caso de uso: realizar el cálculo del aguinaldo de los empleados
Fecha de elaboración:	07/01/2026
Descripción del caso de uso	Le permite al usuario administrador observar los días de vacaciones disponibles de cada colaborador
Autor del caso de uso:	Kendall Salazar Soto
Actores que se relacionan:	Usuario administrador Sistema
Precondiciones:	El usuario administrador debe encontrarse registrado en la base de datos El usuario administrador debe tener asignado el permiso para gestionar el proceso. El sistema debe estar funcionando
Flujo básico del caso de uso	
1. El usuario accede al módulo de aguinaldos. 2. El usuario selecciona la opción <i>Calcular aguinaldo</i>. 3. El sistema despliega la vista con todos los empleados activos. 4. El usuario selecciona al empleado para realizar el cálculo. 5. El sistema despliega la vista con los campos necesarios para realizar el cálculo. 6. El usuario completa los datos requeridos del empleado, como fecha de inicio del período que calcula y la fecha de finalización del período por calcular. 7. El sistema valida todos los campos obligatorios y la coherencia de fechas. 8. El usuario selecciona la opción <i>Calcular</i>.	

9. El sistema calcula automáticamente el monto del aguinaldo usando la fórmula: $\text{aguinaldo} = \text{suma de salarios brutos} / 12$. 10. Se muestran el desglose de los salarios y el monto calculado. 11. El usuario selecciona la opción <i>Guardar</i>. 12. El sistema almacena la información en la base de datos. 13. El sistema confirma el registro exitoso. 14. El sistema visualiza los datos del registro con el nombre y el monto por pagar. 15. Finaliza el caso de uso exitosamente.	
Flujos alternos	
Flujo alternativo n.º 1 Validación de campos vacíos	• Se valida que se ingrese la información en los campos requeridos obligatorios. • Si no cumple, se muestra un mensaje de advertencia. • Se rechaza el envío.
Flujo alternativo n.º 2 Validación de fechas futuras	• Se valida que la fecha en la que se realiza el cálculo del aguinaldo no sea futura. • Si no cumple, se muestra un mensaje de error. • Se rechaza el envío.
Flujo alternativo n.º 3 Validación de fecha de ingreso	• Se valida que la fecha de finalización sea después de la fecha de ingreso. • Si no cumple, se muestra un mensaje de error. • Se rechaza el envío.
Flujo alternativo n.º 4 Fecha de inicio del período	• El sistema valida que la fecha de inicio del período de cálculo no sea anterior al 01 de diciembre del año anterior del año por pagar. • Si no se cumple, el sistema muestra un mensaje de advertencia. • El sistema bloquea el cálculo del aguinaldo.
Flujo alternativo n.º 5 Cancelación del proceso	• Si la persona usuaria decide cancelar la operación antes de guardar, el sistema no registra la información. • Se regresa a la pantalla principal de permisos.
Flujo alternativo n.º 6 Error del sistema	• El sistema no puede registrar o validar debido a un error técnico. • Se muestra un mensaje de error. • El flujo finaliza sin acceso al sistema.
Requerimientos especiales	

Debe existir al menos una línea de nómina. El cálculo debe considerar únicamente los salarios brutos devengados. Todas las acciones deben ser registradas en la tabla auditoría.
Poscondiciones
El registro del cálculo del aguinaldo queda guardado en el sistema. El registro del monto del aguinaldo queda visible con el nombre del empleado. La acción queda registrada en la tabla de auditoría.

Tabla 39*Caso de uso del cálculo de salario bruto*

Prototipo: Sistema web para gestión de pago de nómina en la empresa All Sport Nutrition	
Número del caso de uso: CU-CN-01	Nombre del caso de uso: realizar el cálculo del salario bruto de un colaborador
Fecha de elaboración:	07/01/2026
Descripción del caso de uso	Le permite al sistema calcular automáticamente el salario bruto de cada empleado
Autor del caso de uso:	Kendall Salazar Soto
Actores que se relacionan:	Usuario administrador Sistema
Precondiciones:	El usuario administrador debe encontrarse registrado en la base de datos El usuario administrador debe tener asignado el permiso para gestionar el proceso. El sistema debe estar funcionando
Flujo básico del caso de uso	
1. El sistema accede a la información del empleado en el período activo. 2. El sistema obtiene la cantidad de horas base trabajadas. 3. El sistema obtiene la cantidad de horas extra registradas. 4. El sistema calcula el salario bruto según la fórmula: $\text{salario_bruto} = (\text{cantidad_horas_base} * \text{salario_hora}) + (\text{cantidad_horas_extras} * (\text{salario_hora} * (\text{si es extra a tiempo y medio y se multiplica por 1.5 y si es doble se multiplica por 2})))$.	

5. El sistema calcula deducciones de forma automática: ○ CCSS = salario_bruto x aporte vigente. ○ Impuesto de la renta = según la tabla progresiva vigente. 6. El sistema guarda el salario bruto calculado en la tabla de nómina. 7. El sistema genera un registro de trazabilidad 8. Finaliza el caso de uso exitosamente.	
Flujos alternos	
Flujo alternativo n.º 1 Inexistencia de horas registradas en el período	• El sistema valida si existen horas registradas en el sistema. • Si no cumple, se muestra un mensaje de advertencia. • El salario bruto queda en 0.
Flujo alternativo n.º 2 Error del sistema	• El sistema no puede registrar o validar debido a un error técnico. • Se muestra un mensaje de error. • El flujo finaliza sin acceso al sistema.
Requerimientos especiales	
Todas las acciones deben ser registradas en la tabla auditoría.	
Poscondiciones	
El salario bruto del empleado queda calculado y registrado en la nómina. Los cálculos quedan disponibles para la consolidación automática de la nómina. La acción queda registrada en la tabla de auditoría.	

Tabla 40*Caso de uso del cálculo de deducciones salariales*

Prototipo: Sistema web para gestión de pago de nómina en la empresa All Sport Nutrition	
Número del caso de uso: CU-CN-02	Nombre del caso de uso: realizar el cálculo de deducciones salariales
Fecha de elaboración:	07/01/2026
Descripción del caso de uso	Le permite al usuario calcular las deducciones aplicables a la nómina de las personas colaboradoras

Autor del caso de uso:	Kendall Salazar Soto
Actores que se relacionan:	Usuario administrador Sistema
Precondiciones:	El usuario administrador debe encontrarse registrado en la base de datos El usuario administrador debe tener asignado el permiso para gestionar el proceso. El sistema debe estar funcionando
Flujo básico del caso de uso	
1. El usuario accede al módulo de nómina. 2. El usuario selecciona la opción <i>Calcular deducciones</i>. 3. El sistema despliega la lista de empleados activos con salario bruto en la nómina que se seleccionó. 4. El usuario selecciona al empleado para realizar el cálculo. 5. El sistema calcula deducciones de forma automática: - o CCSS = salario_bruto x aporte vigente. - o Impuesto de la renta = según la tabla progresiva vigente. 6. El sistema despliega el desglose de deducciones. 7. El usuario selecciona la opción <i>Guardar</i>. 8. El sistema almacena la información en la base de datos. 9. El sistema confirma el registro exitoso. 10. El sistema visualiza los datos del registro con el nombre y el monto por pagar. 11. Finaliza el caso de uso exitosamente.	
Flujos alternos	
Flujo alternativo n.º 1 Validación de campos vacíos	• Se valida que se ingrese la información en los campos requeridos obligatorios. • Si no cumple, se muestra un mensaje de advertencia. • Se rechaza el envío.
Flujo alternativo n.º 2 Cancelación de proceso	• Se valida que se ingrese la información en los campos requeridos obligatorios. • Si no cumple, se muestra un mensaje de advertencia. • Se rechaza el envío.
Flujo alternativo n.º 3	• El sistema no puede registrar o validar debido a un error técnico.

Error del sistema	- Se muestra un mensaje de error. - El flujo finaliza y el sistema no ejecuta los cambios.
Requerimientos especiales	
Uso de tablas de porcentajes y rangos tributarios. Todas las acciones deben ser registradas en la tabla auditoría.	
Poscondiciones	
Las deducciones quedan registradas en la base de datos. La acción queda registrada en la tabla de auditoría.	

Tabla 41*Caso de uso del registro y aplicación de rebajos*

Prototipo: Sistema web para gestión de pago de nómina en la empresa All Sport Nutrition	
Número del caso de uso: CU-CN-03	Nombre del caso de uso: registro y aplicación de rebajos
Fecha de elaboración:	07/01/2026
Descripción del caso de uso	Le permite al sistema calcular automáticamente el salario bruto de cada empleado
Autor del caso de uso:	Kendall Salazar Soto
Actores que se relacionan:	Usuario administrador Sistema
Precondiciones:	El usuario administrador debe encontrarse registrado en la base de datos El usuario administrador debe tener asignado el permiso para gestionar el proceso. El sistema debe estar funcionando
Flujo básico del caso de uso	
1. El usuario accede al módulo de nómina. 2. El usuario selecciona la opción <i>Rebajos</i> . 3. El sistema despliega una vista con el formulario para ingresar los rebajos.	

4. El usuario selecciona <i>Agregar rebajo</i>. 5. El sistema despliega un formulario con los campos requeridos, como tipo de rebajo, monto por rebajar, porcentaje aplicado (si se aplica) y descripción. 6. El usuario selecciona la opción <i>Guardar</i>. 7. El sistema valida la información. 8. El sistema almacena la información en la base de datos. 9. El sistema confirma el registro exitoso. 10. El rebajo queda referido a un empleado y marcado como activo para aplicarse en la siguiente nómina. 11. Finaliza el caso de uso exitosamente.	
Flujos alternos	
Flujo alternativo n.º 1 Validación de campos vacíos	• Se valida que se ingrese la información en los campos requeridos obligatorios. • Si no cumple, se muestra un mensaje de advertencia. • Se rechaza el envío.
Flujo alternativo n.º 2 Duplicidad de rebajo	• Si el sistema detecta dos rebajos del mismo tipo en el mismo período. • El sistema muestra un mensaje de error. • El sistema bloquea el envío. • El flujo vuelve al punto 5 del flujo básico.
Flujo alternativo n.º 3 Error del sistema	• El sistema no puede registrar o validar debido a un error técnico. • Se muestra un mensaje de error. • El flujo finaliza y el sistema no ejecuta los cambios.
Requerimientos especiales	
Todas las acciones deben ser registradas en la tabla auditoría.	
Poscondiciones	
Las deducciones quedan registradas en la base de datos. El rebajo activo se aplica automáticamente en el cálculo de la siguiente nómina. La acción queda registrada en la tabla de auditoría.	

Tabla 42*Caso de uso de realizar el cálculo de la nómina*

Prototipo: Sistema web para gestión de pago de nómina en la empresa All Sport Nutrition	
Número del caso de uso: CU-CN-04	Nombre del caso de uso: realizar el cálculo completo de la nómina
Fecha de elaboración:	07/01/2026
Descripción del caso de uso	Le permite al usuario hacer el registro de los cálculos de la nómina completa de un período específico
Autor del caso de uso:	Kendall Salazar Soto
Actores que se relacionan:	Usuario administrador Sistema
Precondiciones:	El usuario administrador debe encontrarse registrado en la base de datos El usuario administrador debe tener asignado el permiso para gestionar el proceso. El sistema debe estar funcionando
Flujo básico del caso de uso	
1. El usuario accede al módulo nómina. 2. El usuario selecciona la opción <i>Calcular nómina</i>. 3. El sistema despliega la lista de todas las nóminas en línea activas del período. 4. El usuario selecciona el empleado al que realiza el cálculo. 5. El sistema ejecuta cálculos automáticos como salario bruto, deducciones, rebajos y salario neto. 6. El usuario revisa los cálculos por empleado. 7. El usuario selecciona la opción <i>Guardar</i>. 8. El sistema almacena la información en la base de datos. 9. El sistema confirma el registro exitoso. 10. Finaliza el caso de uso exitosamente.	
Flujos alternos	
Flujo alternativo n.º 1 Validación de campos vacíos	• Se valida que se ingrese la información en los campos requeridos obligatorios. • Si no cumple, se muestra un mensaje de advertencia.

	Se rechaza el envío.
Flujo alternativo n.º 2 Validación de nómina cerrada	- El sistema valida que la nómina creada al período correspondiente esté activa. - Si no, muestra un mensaje de error. - No se realizan los cálculos. - No se modifican los datos en la base de datos.
Flujo alternativo n.º 3 Cancelación de proceso	- Si la persona usuaria decide cancelar la operación antes de guardar, el sistema no registra la información. - Regresa a la pantalla principal de permisos.
Flujo alternativo n.º 4 Error de sistema	- El sistema no puede registrar o validar debido a un error técnico. - Se muestra un mensaje de error. - El flujo finaliza y el sistema no ejecuta los cambios.
Requerimientos especiales	
Automatización de cálculo según fórmulas predefinidas. Todas las acciones deben ser registradas en la tabla auditoría.	
Poscondiciones	
La nómina queda calculada en el sistema. La acción queda registrada en la tabla de auditoría.	

Tabla 43*Caso de uso de generar reporte de empleados*

Prototipo: Sistema web para gestión de pago de nómina en la empresa All Sport Nutrition	
Número del caso de uso: CU-RPT-01	Nombre del caso de uso: generar reporte de empleados
Fecha de elaboración:	07/01/2026
Descripción del caso de uso	Le permite al usuario generar un reporte de las personas colaboradoras registradas en el sistema.
Autor del caso de uso:	Kendall Salazar Soto

Actores que se relacionan:	Usuario administrador Sistema
Precondiciones:	El usuario administrador debe encontrarse registrado en la base de datos El usuario administrador debe tener asignado el permiso para gestionar el proceso. El sistema debe estar funcionando
Flujo básico del caso de uso	
1. El usuario accede al módulo Reportes. 2. El usuario selecciona la opción <i>Reporte de empleados</i>. 3. El sistema despliega los filtros disponibles. 4. El usuario selecciona los filtros deseados o deja la selección <i>Todos</i>. 5. El sistema consulta en la base de datos los registros de empleados activos. 6. El sistema genera el reporte con la siguiente información: datos personales, puesto actual y jornada laboral actual. 7. El reporte se muestra en la pantalla de forma tabular. 8. El usuario puede optar por exportar el reporte en PDF o Microsoft Excel. 9. El sistema confirma el registro exitoso. 10. Finaliza el caso de uso exitosamente.	
Flujos alternos	
Flujo alternativo n.º 1 No existen empleados registrados	• Se valida si existen empleados en el sistema. • Si no cumple, se muestra un mensaje de advertencia. • El reporte no se genera.
Flujo alternativo n.º 2 Cancelación de proceso	• Si la persona usuaria decide cancelar la operación antes de generar el reporte se cancela la generación. • El flujo regresa al flujo básico n.º 3.
Flujo alternativo n.º 3 Error de sistema	• El sistema no puede registrar o validar debido a un error técnico. • Se muestra un mensaje de error. • El flujo finaliza y el sistema no ejecuta los cambios.
Requerimientos especiales	
Debe permitir la exportación de reportes a PDF y Microsoft Excel.	

Todas las acciones deben ser registradas en la tabla auditoría.
Poscondiciones
El reporte queda disponible en pantalla para la persona usuaria. Si se exporta el archivo queda generado en el formato seleccionado. La acción queda registrada en la tabla de auditoría.

Tabla 44*Caso de uso de generar un reporte de historial laboral*

Número de caso de uso: CU-RPT-02	Nombre del caso de uso: generar reporte de historial laboral
Fecha elaboración:	07/01/2026
Descripción caso de uso	Permite generar un reporte de la asistencia, horas extra, permisos e incapacidades de los empleados.
Autor caso de uso:	Kendall Salazar Soto
Actores relacionados:	Usuario administrador Sistema
Precondiciones:	• El usuario debe estar debidamente registrado en la base de datos del sistema. • El usuario administrador debe contar con los permisos correspondientes para la gestión del proceso. • El sistema debe disponer de la información necesaria para llevar a cabo la gestión del proceso. • El sistema debe encontrarse operativo y en correcto funcionamiento.
Flujo básico del caso de uso	
1. El usuario accede al módulo de reportes. 2. El usuario selecciona la opción <i>Reporte de tiempo laboral</i>. 3. El sistema despliega los filtros disponibles. 4. El usuario selecciona los filtros deseados o deja la selección <i>Todos</i>. 5. El sistema consulta en la base de datos los registros de empleados activos. 6. El sistema genera el reporte con la siguiente información: asistencia por rango de fecha, horas extra por empleado y totales por mes e incapacidades por tipo y por empleado. 7. El reporte se muestra en la pantalla de forma tabular.	

8. El usuario puede optar por exportar el reporte en PDF o Excel. 9. El sistema confirma el registro exitoso. 10. Finaliza el caso de uso exitosamente.	
Flujos alternos	
Flujo alternativo n.º 1 No existen empleados registrados	- Se valida si existen empleados en el sistema. - Si no cumple, se muestra un mensaje de advertencia. - El reporte no se genera.
Flujo alternativo n.º 2 Cancelación de proceso	- Si el usuario decide cancelar la operación antes de generar el reporte, se cancela la generación. - El flujo regresa al flujo básico n.º 3.
Flujo alternativo n.º 3 Error de sistema	- El sistema no puede registrar o validar debido a un error técnico. - Se muestra un mensaje de error. - El flujo finaliza y el sistema no ejecuta los cambios.
Requerimientos especiales	
Debe permitir la exportación de reportes a PDF y Excel. Todas las acciones deben ser registradas en la tabla auditoría.	
Poscondiciones	
El reporte queda disponible en pantalla para el usuario. Si se exporta el archivo queda generado en el formato seleccionado. La acción queda registrada en la tabla de auditoría.	

Tabla 45

Caso de uso de generar un reporte de beneficios del colaborador

Prototipo: Sistema web para gestión de pago de nómina en la empresa All Sport Nutrition	
Número del caso de uso: CU-RPT-03	Nombre del caso de uso: generar reporte de beneficios del colaborador
Fecha de elaboración:	07/01/2026
Descripción del caso de uso	Permite generar un reporte de la asistencia, horas

	extra, permisos e incapacidades de los empleados.
Autor del caso de uso:	Kendall Salazar Soto
Actores que se relacionan:	Usuario administrador Sistema
Precondiciones:	El usuario administrador debe encontrarse registrado en la base de datos El usuario administrador debe tener asignado el permiso para gestionar el proceso. El sistema debe estar funcionando
Flujo básico del caso de uso	
1. El usuario accede al módulo Reportes del sistema. 2. El usuario selecciona la opción <i>Reporte de beneficios</i>. 3. El sistema recupera y procesa la información desde las tablas correspondientes, lo que incluye vacaciones tomadas y disponibles por empleado, así como los aguinaldos calculados y pagados. 4. El usuario selecciona la opción <i>Generar</i>. 5. El sistema presenta el reporte en pantalla en formato tabular. 6. El usuario tiene la opción de exportar el reporte en formato PDF o Microsoft Excel. 7. El sistema confirma la generación del reporte de manera exitosa. 8. El caso de uso finaliza correctamente.	
Flujos alternos	
Flujo alternativo n.º 1 No existen empleados registrados	• El sistema valida la existencia de empleados registrados. • En caso de no encontrarse registros, el sistema muestra un mensaje de advertencia al usuario. • El reporte no se genera y el flujo se interrumpe.
Flujo alternativo n.º 2 Cancelación del proceso	• Si la persona usuaria decide cancelar la operación antes de generar el reporte, el sistema cancela su generación. • El flujo retorna al paso 3 del flujo principal.
Flujo alternativo n.º 3 Error de sistema	• El sistema no puede procesar la solicitud debido a un error técnico. • El sistema muestra un mensaje de error al usuario. • El flujo finaliza sin que se ejecuten cambios en el sistema.

Requerimientos especiales
El sistema debe permitir la exportación de reportes en formatos PDF y Microsoft Excel. Todas las acciones deben ser registradas en la tabla auditoría.
Poscondiciones
El reporte queda disponible para su visualización en pantalla por parte del usuario. En caso de exportación, el archivo se genera correctamente en el formato seleccionado. La acción realizada queda registrada en el sistema de auditoría y trazabilidad.

Tabla 46

Caso de uso de generar un reporte de nómina del colaborador

Prototipo: Sistema web para gestión de pago de nómina en la empresa All Sport Nutrition	
Número de caso de uso: CU-RPT-04	Nombre del caso de uso: generar reporte de nómina
Fecha elaboración:	07/01/2026
Descripción caso de uso	Este caso de uso permite al encargado del área de Recursos Humanos generar un reporte detallado de los cálculos de nómina de los empleados registrados en el sistema, incluyendo información salarial, deducciones y liquidaciones.
Autor caso de uso:	Kendall Salazar Soto
Actores que se relacionan:	Usuario administrador Sistema
Precondiciones:	- El usuario debe haber iniciado sesión en el sistema con un rol autorizado. - El usuario debe contar con los permisos necesarios para la generación de reportes de nómina. - Deben existir registros de nómina almacenados en el sistema. - Los empleados asociados a la nómina deben encontrarse registrados en el sistema.
Flujo básico del caso de uso	
1. El usuario accede al módulo Reportes.	

2. El usuario selecciona la opción <i>Reporte de Nómina</i> . 3. El sistema recupera la información desde las tablas correspondientes, incluyendo salario bruto y neto, deducciones, rebajos y liquidaciones por salida de empleados. 4. El usuario selecciona la opción <i>Generar</i> . 5. El sistema muestra el reporte en pantalla en formato tabular. 6. El usuario puede optar por exportar el reporte en formato PDF o Excel. 7. El sistema confirma la generación exitosa del reporte. 8. El caso de uso finaliza correctamente.	
Flujos alternos	
Flujo alternativo n.º 1 No existen empleados registrados	- El sistema valida la existencia de empleados registrados. - En caso de no encontrarse registros, el sistema muestra un mensaje de advertencia al usuario. - El reporte no es generado y el flujo se interrumpe.
Flujo alternativo n.º 2 Cancelación del proceso	- Si el usuario decide cancelar la operación antes de generar el reporte, el sistema cancela la generación de este. - El flujo retorna al paso 3 del flujo principal.
Flujo alternativo n.º 3 Error de sistema	- El sistema no puede procesar la solicitud debido a un error técnico. - El sistema muestra un mensaje de error al usuario. - El flujo finaliza sin que se ejecuten cambios en el sistema.
Requerimientos especiales	
El sistema debe contar con auditoría extendida que permita la trazabilidad de las acciones realizadas. El sistema debe permitir la exportación de reportes en formatos PDF y Excel.	
Poscondiciones	
El reporte queda disponible para su visualización en pantalla por parte del usuario. En caso de exportación, el archivo se genera correctamente en el formato seleccionado. La acción realizada queda registrada en el sistema de auditoría y trazabilidad.	

Tabla 47*Caso de uso de generar un reporte de nómina del colaborador*

Prototipo: Sistema web para gestión de pago de nómina en la empresa All Sport Nutrition	
Número de caso de uso: CU-RPT-04	Nombre del caso de uso: generar reporte de nómina
Fecha elaboración:	07/01/2026
Descripción caso de uso	Este caso de uso le permite al encargado del área de Recursos Humanos generar un reporte detallado de los cálculos de nómina de los empleados registrados en el sistema, incluyendo información salarial, deducciones y liquidaciones.
Autor caso de uso:	Kendall Salazar Soto
Actores que se relacionan:	Usuario administrador Sistema
Precondiciones:	- El usuario debe haber iniciado sesión en el sistema con un rol autorizado. - El usuario debe contar con los permisos necesarios para la generación de reportes de nómina. - Deben existir registros de nómina almacenados en el sistema. - Los empleados asociados a la nómina deben encontrarse registrados en el sistema.
Flujo básico del caso de uso	
1. El usuario accede al módulo Reportes. 2. El usuario selecciona la opción <i>Reporte de Nómina</i>. 3. El sistema recupera la información desde las tablas correspondientes, incluyendo salario bruto y neto, deducciones, rebajos y liquidaciones por salida de empleados. 4. El usuario selecciona la opción <i>Generar</i>. 5. El sistema muestra el reporte en pantalla en formato tabular. 6. El usuario puede optar por exportar el reporte en formato PDF o Excel. 7. El sistema confirma la generación exitosa del reporte. 8. El caso de uso finaliza correctamente.	
Flujos alternos	
Flujo alternativo n.º 1	El sistema valida la existencia de empleados

No existen empleados registrados	registrados. • En caso de no encontrarse registros, el sistema muestra un mensaje de advertencia al usuario. • El reporte no es generado y el flujo se interrumpe.
Flujo alternativo n.º 2 Cancelación del proceso	• Si el usuario decide cancelar la operación antes de generar el reporte, el sistema cancela la generación de este. • El flujo retorna al paso 3 del flujo principal.
Flujo alternativo n.º 3 Error de sistema	• El sistema no puede procesar la solicitud debido a un error técnico. • El sistema muestra un mensaje de error al usuario. • El flujo finaliza sin que se ejecuten cambios en el sistema.
Requerimientos especiales	
El sistema debe contar con auditoría extendida que permita la trazabilidad de las acciones realizadas. El sistema debe permitir la exportación de reportes en formatos PDF y Excel.	
Poscondiciones	
El reporte queda disponible para su visualización en pantalla por parte del usuario. En caso de exportación, el archivo se genera correctamente en el formato seleccionado. La acción realizada queda registrada en el sistema de auditoría y trazabilidad.	

Tabla 48

Caso de uso para generar un reporte de evaluaciones de rendimiento

Prototipo: Sistema web para gestión de pago de nómina en la empresa All Sport Nutrition	
Número del caso de uso: CU-RPT-05	Nombre del caso de uso: generar reporte de evaluaciones de rendimiento
Fecha de elaboración:	07/01/2026
Descripción del caso de uso	Este caso de uso permite al administrador del sistema generar un reporte consolidado de las evaluaciones de rendimiento de los empleados registrados en el sistema, lo que incluye información sobre períodos que se evaluaron, calificaciones, observaciones y estado de cada evaluación.

Autor del caso de uso:	Kendall Salazar Soto
Actores que se relacionan:	Usuario administrador Sistema
Precondiciones:	• El usuario debe haber iniciado sesión en el sistema con un rol autorizado. • El usuario debe contar con los permisos necesarios para generar reportes de evaluación de rendimiento. • Deben existir evaluaciones de rendimiento registradas en el sistema. • Los empleados que se evaluaron deben encontrarse registrados y activos en el sistema.
Flujo básico del caso de uso	
1. El usuario accede al módulo Reportes. 2. El usuario selecciona la opción <i>Reporte de evaluaciones de rendimiento</i>. 3. El sistema recupera la información de las evaluaciones almacenadas, lo que incluye período evaluado, calificación, observaciones y estado de cada evaluación. 4. El usuario selecciona la opción <i>Generar</i>. 5. El sistema muestra el reporte en pantalla en formato tabular. 6. El usuario puede optar por exportar el reporte en PDF o Microsoft Excel. 7. El sistema confirma la generación exitosa del reporte. 8. El caso de uso finaliza correctamente.	
Flujos alternos	
Flujo alternativo n.º 1 No existen evaluaciones registradas	• El sistema valida si existen evaluaciones de rendimiento registradas. • En caso de no encontrarse registros, se muestra un mensaje de advertencia al usuario. • El reporte no se genera y el flujo se interrumpe.
Flujo alternativo n.º 2 Cancelación del proceso	• Si la persona usuaria decide cancelar la operación antes de generar el reporte, el sistema cancela la acción. • El flujo retorna al paso 3 del flujo básico.
Flujo alternativo n.º 3 Error de sistema	• El sistema no puede procesar la solicitud debido a un error técnico. • Se muestra un mensaje de error al usuario. • El flujo finaliza sin que se realicen cambios en el

	sistema.
Requerimientos especiales	
El sistema debe contar con auditoría extendida para garantizar la trazabilidad de las acciones realizadas. El sistema debe permitir la exportación de reportes en formatos PDF y Microsoft Excel.	
Poscondiciones	
El reporte queda disponible para su visualización en pantalla por parte del usuario. En caso de exportación, el archivo se genera correctamente en el formato seleccionado. La acción realizada queda registrada en el sistema de auditoría y trazabilidad.	

Tabla 49

Caso de uso para listar las evaluaciones de rendimiento

Prototipo: Sistema web para gestión de pago de nómina en la empresa All Sport Nutrition	
Número del caso de uso: CU-MER-01	Nombre del caso de uso: listar las evaluaciones de rendimiento de los empleados
Fecha de elaboración:	07/01/2026
Descripción del caso de uso	Este caso de uso le permite al usuario visualizar el listado de las evaluaciones de rendimiento asociadas a los empleados activos del sistema, con el fin de consultar el desempeño laboral registrado para cada colaborador.
Autor del caso de uso:	Kendall Salazar Soto
Actores que se relacionan:	Usuario administrador Sistema
Precondiciones:	• El usuario debe haber iniciado sesión en el sistema con un rol autorizado. • El usuario debe contar con los permisos necesarios para consultar evaluaciones de rendimiento. • Deben existir empleados registrados en el sistema. • Deben existir evaluaciones de rendimiento asociadas a los empleados. • El sistema debe contar con acceso a la base de datos.

Flujo básico del caso de uso	
9. El usuario accede a la opción <i>Evaluar rendimiento</i> desde el módulo correspondiente. 10. El sistema muestra la vista con el listado de empleados activos registrados. 11. El sistema despliega la información asociada a las evaluaciones de rendimiento, lo que incluye nombre del empleado, período de evaluación, calificación y estado. 12. La vista presenta las opciones Editar, Eliminar y Agregar. 13. El caso de uso finaliza de manera exitosa.	
Flujos alternos	
Flujo alternativo n.º 1 Validación de empleado activo	- El sistema valida que el empleado se encuentre activo. - Si el empleado está activo, se muestra la información correspondiente a su evaluación de rendimiento.
Flujo alternativo n.º 2 Error del sistema	- El sistema no puede procesar la solicitud debido a un error técnico. - Se muestra un mensaje de error al usuario. - El flujo finaliza sin que se ejecuten cambios en el sistema.
Requerimientos especiales	
Validación de empleados activos. Auditoría extendida con trazabilidad de la consulta realizada. Visualización eficiente del listado de evaluaciones.	
Poscondiciones	
El usuario tiene acceso a la información de las evaluaciones de rendimiento de los empleados. La acción de consulta queda registrada en el sistema de auditoría y trazabilidad.	

Tabla 50

Caso de uso para agregar una evaluación de rendimiento

Prototipo: Sistema web para gestión de pago de nómina en la empresa All Sport Nutrition	
Número del caso de uso: CU-MER-02	Nombre del caso de uso: listar las evaluaciones de rendimiento de los empleados
Fecha de elaboración:	07/01/2026

Descripción del caso de uso	Este caso de uso le permite al administrador registrar una nueva evaluación de rendimiento para un empleado, con el fin de documentar su desempeño laboral durante un período determinado.
Autor del caso de uso:	Kendall Salazar Soto
Actores que se relacionan:	Usuario administrador Sistema
Precondiciones:	• El usuario debe haber iniciado sesión en el sistema con un rol autorizado. • El usuario debe contar con los permisos necesarios para registrar evaluaciones de rendimiento. • El empleado debe encontrarse registrado y activo en el sistema.
Flujo básico del caso de uso	
1. El usuario accede a la opción <i>Evaluar rendimiento</i> desde el módulo correspondiente. 2. El sistema muestra la vista con el listado de empleados activos registrados. 3. La vista presenta información básica del empleado, así como las opciones Editar, Eliminar y Agregar. 4. El usuario selecciona la opción <i>Agregar</i> en el empleado correspondiente. 5. El sistema redirecciona a la vista Agregar evaluación de rendimiento y despliega el formulario. 6. El usuario completa los datos requeridos tales como período de evaluación, calificación obtenida, observaciones y estado. 7. El sistema valida que todos los campos obligatorios hayan sido completados correctamente. 8. El usuario selecciona la opción <i>Guardar</i>. 9. El sistema almacena la información en la base de datos. 10. El sistema confirma el registro exitoso de la evaluación. 11. El sistema retorna a la vista principal del módulo Evaluar Rendimiento. 12. El caso de uso finaliza de manera exitosa.	
Flujos alternos	
Flujo alternativo n.º 1 Validación de campos vacíos	• El sistema verifica que los campos obligatorios contengan información válida. • Si la validación no se cumple, se muestra un mensaje de advertencia. • El registro no se almacena hasta que la información sea corregida.
Flujo alternativo n.º 2 Cancelación del proceso	• Si la persona usuaria decide cancelar la operación antes de guardar, el sistema no registra la información.

	El sistema regresa a la vista principal del módulo Evaluar Rendimiento.
Flujo alternativo n.º 3 Error del sistema	- El sistema no puede procesar la solicitud debido a un error técnico. - Se muestra un mensaje de error al usuario. - El flujo finaliza sin que se realicen cambios en el sistema.
Requerimientos especiales	
Validación de campos obligatorios y formatos de datos. Validación del período de evaluación. Auditoría extendida con trazabilidad de los eventos que se generan.	
Poscondiciones	
La evaluación de rendimiento del empleado queda registrada en el sistema. El evento queda almacenado en el sistema de auditoría y trazabilidad.	

Tabla 51

Caso de uso para actualizar una evaluación de rendimiento

Prototipo: Sistema web para gestión de pago de nómina en la empresa All Sport Nutrition	
Número del caso de uso: CU-MER-03	Nombre del caso de uso: actualizar los datos de una evaluación de rendimiento
Fecha de elaboración:	07/01/2026
Descripción del caso de uso	El usuario debe haber iniciado sesión en el sistema. El usuario debe contar con los permisos necesarios para actualizar evaluaciones de rendimiento. El empleado debe encontrarse registrado y activo en el sistema. La evaluación de rendimiento debe haber sido creada previamente. El sistema debe contar con acceso a los datos del empleado en la base de datos.
Autor del caso de uso:	Kendall Salazar Soto
Actores que se relacionan:	Usuario administrador

	Sistema
Precondiciones:	• El usuario debe haber iniciado sesión en el sistema con un rol autorizado. • El usuario debe contar con los permisos necesarios para registrar evaluaciones de rendimiento. • El empleado debe encontrarse registrado y activo en el sistema.
Flujo básico del caso de uso	
1. El usuario accede a la opción <i>Evaluar rendimiento</i> desde el módulo correspondiente. 2. El sistema muestra la pantalla con el listado de empleados activos y sus evaluaciones de rendimiento. 3. El sistema despliega información como nombre del empleado, período de evaluación, calificación y estado de la evaluación. 4. La vista presenta las opciones Editar, Eliminar y Agregar. 5. El usuario selecciona la opción <i>Editar</i>. 6. El sistema redirecciona a la vista Actualizar evaluación de rendimiento y carga el formulario con los datos registrados. 7. El usuario modifica los datos correspondientes de la evaluación. 8. El sistema valida todos los campos ingresados. 9. El usuario selecciona la opción <i>Guardar</i>. 10. El sistema actualiza la información en la base de datos. 11. El sistema confirma la actualización exitosa del registro. 12. El sistema retorna a la vista principal del módulo Evaluar Rendimiento. 13. El caso de uso finaliza de manera exitosa.	
Flujos alternos	
Flujo alternativo n.º 1 Validación de campos vacíos	• El sistema verifica que los campos obligatorios contengan información válida. • Si la validación no se cumple, se muestra un mensaje de advertencia. • El registro no se almacena hasta que la información sea corregida.
Flujo alternativo n.º 2 Cancelación del proceso	• Si la persona usuaria decide cancelar la operación antes de guardar, el sistema no registra la información. • El sistema regresa a la vista principal del módulo Evaluar Rendimiento.
Flujo alternativo n.º 3 Error del sistema	• El sistema no puede procesar la solicitud debido a un error técnico. • Se muestra un mensaje de error al usuario. • El flujo finaliza sin que se realicen cambios en el

	sistema.
Requerimientos especiales	
Validación de campos obligatorios y formatos de datos. Validación del período de evaluación. Auditoría extendida con trazabilidad de los eventos que se generan.	
Poscondiciones	
La evaluación de rendimiento del empleado queda registrada en el sistema. El evento queda almacenado en el sistema de auditoría y trazabilidad.	

Tabla 52

Caso de uso para eliminar una evaluación de rendimiento

Prototipo: Sistema web para gestión de pago de nómina en la empresa All Sport Nutrition	
Número del caso de uso: CU-MER-03	Nombre del caso de uso: eliminar los datos de una evaluación de rendimiento
Fecha de elaboración:	07/01/2026
Descripción del caso de uso	El usuario debe haber iniciado sesión en el sistema. El usuario debe contar con los permisos necesarios para actualizar evaluaciones de rendimiento. El empleado debe encontrarse registrado y activo en el sistema. La evaluación de rendimiento debe haber sido creada previamente. El sistema debe contar con acceso a los datos del empleado en la base de datos.
Autor del caso de uso:	Kendall Salazar Soto
Actores que se relacionan:	Usuario administrador Sistema
Precondiciones:	• El usuario debe haber iniciado sesión en el sistema con un rol autorizado. • El usuario debe contar con los permisos necesarios para registrar evaluaciones de rendimiento. • El empleado debe encontrarse registrado y activo

	en el sistema.
Flujo básico del caso de uso	
1. El usuario accede a la opción <i>Evaluar rendimiento</i> desde el módulo correspondiente. 2. El sistema muestra la pantalla con el listado de empleados activos y sus evaluaciones de rendimiento. 3. El sistema despliega información como nombre del empleado, período evaluado, calificación y estado de la evaluación. 4. La vista presenta las opciones Editar, Eliminar y Agregar. 5. El usuario selecciona la opción <i>Eliminar</i> de la evaluación que desea suprimir. 6. El sistema solicita confirmación de la eliminación. 7. Una vez confirmada la acción, el sistema actualiza el estado de la evaluación en la base de datos, marcándose como inactiva. 8. El sistema registra la acción en la tabla de auditoría. 9. El sistema regresa a la vista principal del módulo Evaluar rendimiento. 10. El caso de uso finaliza de manera exitosa.	
Flujos alternos	
Flujo alternativo n.º 1 Validación de permisos	• El sistema verifica que la persona usuaria cuente con los permisos necesarios para realizar la eliminación. • Si la persona usuaria no posee los permisos correspondientes, la opción <i>Eliminar</i> se mantiene deshabilitada.
Flujo alternativo n.º 2 Confirmación de eliminación	• El sistema muestra un mensaje de confirmación antes de la eliminación. • Si la persona usuaria confirma la acción, el sistema procede con la eliminación lógica. • El sistema retorna a la vista principal del módulo.
Flujo alternativo n.º 3 Cancelación de la operación	• El usuario decide cancelar la acción de eliminación. • En la ventana de confirmación selecciona la opción <i>No</i>. • El sistema cierra el cuadro de diálogo sin realizar cambios. • El sistema retorna a la vista principal del módulo.
Flujo alternativo n.º 4 Error del sistema	• El sistema no puede procesar la solicitud debido a un error técnico. • Se muestra un mensaje de error al usuario. • El flujo finaliza sin que se ejecuten cambios en el sistema.

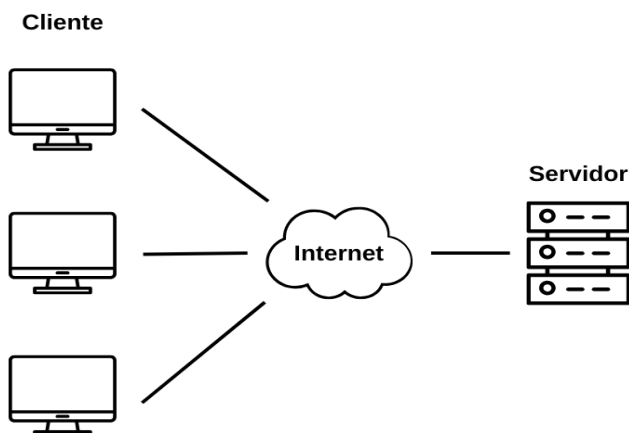
Requerimientos especiales
Registro de la acción en la tabla auditoría asociado al usuario que ejecuta la operación. Auditoría extendida con trazabilidad de eventos. El sistema debe garantizar la eliminación lógica de las evaluaciones de rendimiento, lo que evita el borrado físico para preservar la información histórica.
Poscondiciones
La evaluación de rendimiento queda marcada como inactiva en el sistema. Se conserva la información histórica asociada a la evaluación del empleado. La acción queda registrada en la tabla auditoría. El sistema refleja los cambios en el listado de evaluaciones de rendimiento.

Diseño del sistema

Arquitectura del sistema. La arquitectura del sistema se basa en un modelo cliente-servidor en la nube diseñado para operar mediante un servidor privado virtual que garantiza la accesibilidad, la escalabilidad y la disponibilidad continua para la empresa All Sport Nutrition. El sistema se encuentra desplegado en un VPS que actúa como servidor central, encargado de ejecutar la aplicación web desarrollada en Spring Boot con servidor Tomcat embebido y de gestionar la base de datos MySQL instalada en el mismo entorno virtualizado.

Figura 1

Arquitectura cliente-servidor



El sistema se ejecuta en la nube y es accesible mediante cualquier navegador web desde cualquier dispositivo con conexión a Internet, a través de una dirección web específica asignada

al VPS. Esta configuración permite que el personal administrativo, las jefaturas y las personas colaboradoras accedan al sistema de gestión de nómina desde sus equipos de trabajo, dispositivos móviles o cualquier ubicación con acceso a la red, sin necesidad de instalar *software* adicional en sus dispositivos.

La arquitectura propuesta es moderna, escalable y profesional, ya que se adapta a las necesidades actuales de la empresa y permite un crecimiento futuro sin que se requieran cambios significativos en la infraestructura. Al estar alojado en un VPS, el sistema garantiza la disponibilidad durante toda la jornada laboral, facilita el acceso remoto y centraliza la totalidad de la información en un único punto seguro y controlado. Además, esta arquitectura permite validar el prototipo funcional del sistema de gestión de nómina en un entorno real accesible desde múltiples ubicaciones, respalda su operatividad en las condiciones reales del negocio y facilita la colaboración entre los diferentes departamentos de la empresa.

El modelo cliente-servidor en la nube ofrece ventajas significativas, como la eliminación de la dependencia de equipos físicos específicos, la posibilidad de realizar respaldos automáticos de la información, la facilidad para actualizar y mantener el sistema sin afectar a los usuarios y la capacidad de escalar recursos según las necesidades futuras de la empresa. Esta arquitectura asegura que el sistema web para la gestión de pago de nómina sea una solución robusta, confiable y preparada para soportar el crecimiento de All Sport Nutrition.

Arquitectura del software. El sistema web de gestión de nómina está construido sobre un conjunto de tecnologías modernas e integradas que conforman la arquitectura de *software*. En el *front-end* se utiliza React como biblioteca principal para la construcción de la interfaz de usuario, lo que permite crear componentes reutilizables y ofrecer una experiencia interactiva y dinámica. React se complementa con Next.js, un *framework* que facilita el renderizado del lado del servidor, la optimización del rendimiento y el enrutamiento de la aplicación. Estas tecnologías, junto con HTML5, CSS3 y librerías modernas de elementos, permiten desarrollar una interfaz profesional, responsiva y visualmente organizada que se adapta a diferentes dispositivos y tamaños de pantalla.

El *front-end* se comunica con el *back-end* mediante una arquitectura de API RESTful, desarrollada en C# bajo el *framework* .NET. El *back-end* utiliza ASP.NET Core para la construcción de servicios web robustos y escalables e incorpora características como la inyección de dependencias, *middleware* personalizado para la gestión de peticiones HTTP y mecanismos de

autenticación y autorización con base en JWT (JSON Web Tokens) para garantizar la seguridad de acceso. La lógica de negocio está organizada siguiendo principios de arquitectura limpia y patrones de diseño que facilitan el mantenimiento y la evolución del sistema.

Para la gestión de datos, el sistema utiliza Entity Framework Core como ORM (Object-Relational Mapping), lo que permite mapear las entidades del dominio con las tablas de la base de datos de manera eficiente y segura. Entity Framework Core facilita las operaciones de consulta, inserción, actualización y eliminación de datos mediante LINQ (Language Integrated Query), lo que garantiza la integridad de la información y reduce la complejidad del código relacionado con el acceso a datos. El sistema almacena toda la información en una base de datos MySQL, elegida por su estabilidad, rendimiento, compatibilidad multiplataforma y amplia adopción en la industria.

El conjunto conforma una arquitectura moderna de tipo cliente-servidor basada en servicios REST, en la que la persona usuaria accede a la interfaz web a través de un navegador desde cualquier dispositivo con conexión a Internet. El *front-end*, desarrollado en React y Next.js, consume los servicios expuestos por el *back-end* en C# y .NET que, a la vez, interactúa con la base de datos MySQL mediante Entity Framework Core. Esta separación entre la capa de presentación y la capa de servicios permite una mayor flexibilidad y escalabilidad y facilita implementar mejoras o cambios en cualquiera de las capas sin afectar al resto del sistema.

La arquitectura propuesta garantiza un sistema robusto, seguro y de alto rendimiento, capaz de manejar las operaciones de gestión de nómina de manera eficiente. El uso de tecnologías modernas y ampliamente adoptadas asegura la disponibilidad de recursos, documentación y soporte técnico para el desarrollo y el mantenimiento continuo del sistema. Además, esta arquitectura facilita la integración futura con otros sistemas o servicios externos que la empresa pueda requerir a medida que crece y evoluciona.

Diseño de entradas. Este apartado presenta las interfaces principales que componen el prototipo del sistema de administración de personal. Cada pantalla se desarrolló considerando principios de sencillez y usabilidad, con el objetivo de que se asegure una experiencia fluida entre la persona usuaria y la plataforma. Las siguientes imágenes exhiben las vistas más significativas del proyecto, esto muestra la organización de la navegación, los formularios para la captura de información y las funciones clave incorporadas en el sistema.

Figura 2
Diseño de la pantalla de inicio de sesión

Sistema de RRHH
Inicie sesión para acceder al sistema

Iniciar Sesión
Ingrese sus credenciales para continuar

Usuario
Ingrese su nombre de usuario

Contraseña
Ingrese su contraseña

[¿Olvidó su contraseña?](#)

Iniciar Sesión

[¿Problemas para iniciar sesión?](#)
Contacte al [Departamento de Sistemas](#)

© 2026 Sistema RRHH - Todos los derechos reservados
v1.0.0

Figura 3
Diseño de la pantalla principal de entrada del sistema

¡Bienvenido, Laura García Fernández!
Código: EMP00000001 • EMPLEADO 22/01/2026

Total Empleados
124
+12% vs mes anterior

Asistencias Hoy
98
+3% vs mes anterior

Horas Extra
45h
-3% vs mes anterior

Departamentos
8
0% vs mes anterior

Gestión de Empleados
Ver, crear y administrar empleados de la empresa

Control de Asistencia
Registrar y revisar asistencias diarias

Puestos
Administrar puestos y roles laborales

Departamentos
Gestionar áreas y estructura organizacional

Actividad Reciente

- Juan Pérez registró su entrada Hace 5 min
- Maria García solicitó permiso Hace 15 min
- Carlos López registró horas extra Hace 1 hora

La interfaz de inicio del sistema actúa como el portal central a través del cual los usuarios pueden ingresar a los distintos componentes de la plataforma. Su estructura permite una navegación fluida y bien organizada, lo que facilita el acceso a las funciones principales del sistema, tales como la administración de personal, el control de asistencias, el registro de tiempo adicional, las solicitudes de permisos, la gestión vacacional, el manejo de incapacidades, el procesamiento de salarios, el cálculo de finiquitos y el pago de prestaciones anuales, entre otras opciones. Por medio de esta pantalla, la persona usuaria obtiene una perspectiva completa del sistema y puede dirigirse sin complicaciones a los módulos que necesita consultar o utilizar.

Figura 4
Diseño de entrada para el ingreso de un empleado

Crear Nuevo Empleado

×

Código	Email
Nombre	Primer apellido
Segundo apellido	Teléfono
Fecha de contratación	Departamento
Selecciona una fecha	Selecciona un departamento
Puesto	Rol
Selecciona un puesto	Selecciona un rol
Salario base	Usuario
0	
Contraseña	

La interfaz del formulario de personal está destinada a ingresar los datos de cada uno de los trabajadores de la empresa. A través de esta pantalla se capturan datos fundamentales, tales como nombre y apellidos, documento de identidad, dirección de correo electrónico y demás información pertinente que la compañía necesita para sus procesos de gestión en el Área de Recursos Humanos.

Su estructura permite el ingreso sistemático y exacto de los datos, lo que asegura que los expedientes permanezcan vigentes y completos.

Figura 5

Diseño de entrada para registrar una asistencia

Registrar Asistencia ✕

Empleado *

Selecciona un empleado ▼

Fecha *

Selecciona una fecha ▼

Hora de Entrada **Hora de Salida**

⌚ --:-- -- ⌚ ⌚ --:-- -- ⌚

Estado *

PRESENTE ▼

Observaciones

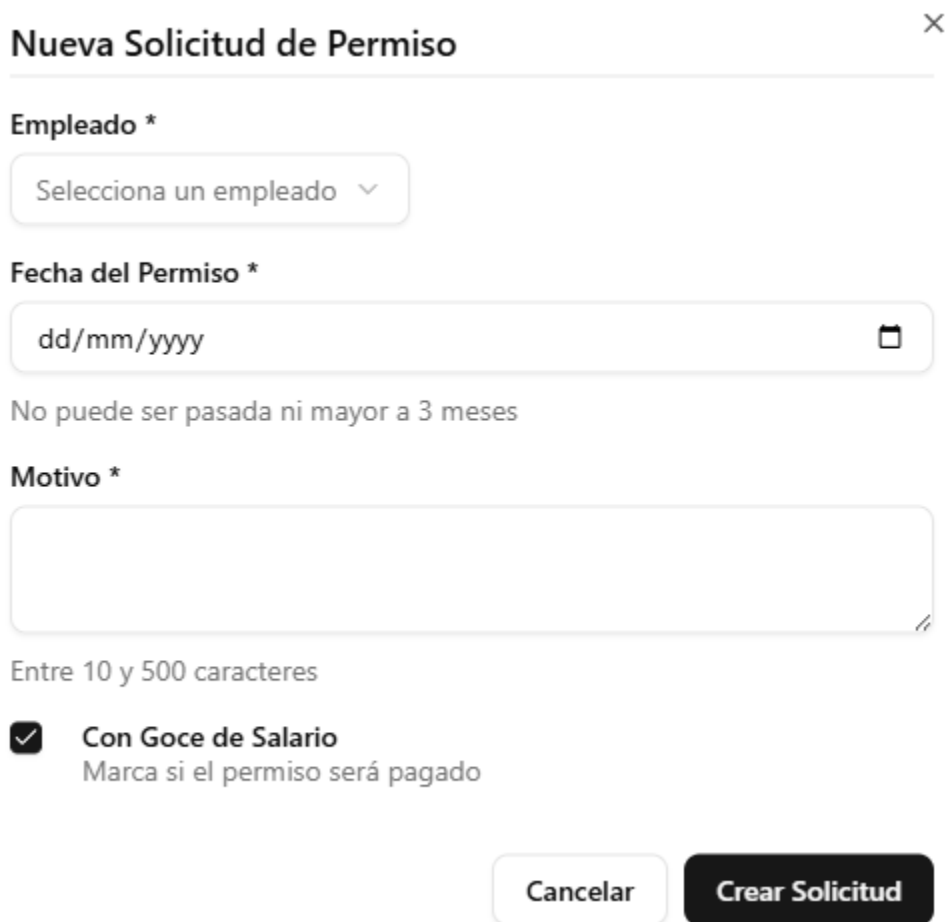
Agregar comentarios adicionales...

Cancelar **Registrar Asistencia**

La interfaz del formulario de control de asistencias posibilita documentar la presencia del personal en días específicos, mediante el registro de los horarios de ingreso y egreso respectivos. Esta estructura permite mantener un seguimiento organizado y exacto de la asistencia de las personas colaboradoras, con el propósito de apoyar la administración del recurso humano y el monitoreo detallado del cumplimiento de los horarios de trabajo establecidos.

Figura 6

Diseño de entrada para realizar la solicitud de un permiso



Nueva Solicitud de Permiso ✕

Empleado *

Selecciona un empleado ▾

Fecha del Permiso *

dd/mm/yyyy 

No puede ser pasada ni mayor a 3 meses

Motivo *

Entre 10 y 500 caracteres

☒ **Con Goce de Salario**
Marca si el permiso será pagado

Cancelar Crear Solicitud

La interfaz del formulario de autorizaciones permite documentar las peticiones de ausencia realizadas por los empleados de la empresa. A través de esta pantalla se capturan datos como las fechas de inicio y término de la autorización, la condición que se otorga y la justificación asociada. Su estructura facilita la administración y el monitoreo de las solicitudes, con el objetivo de garantizar un manejo adecuado de las inasistencias en el Área de Recursos Humanos.

Diseño físico de la base de datos

Figura 7

Modelo entidad-relación de la base de datos

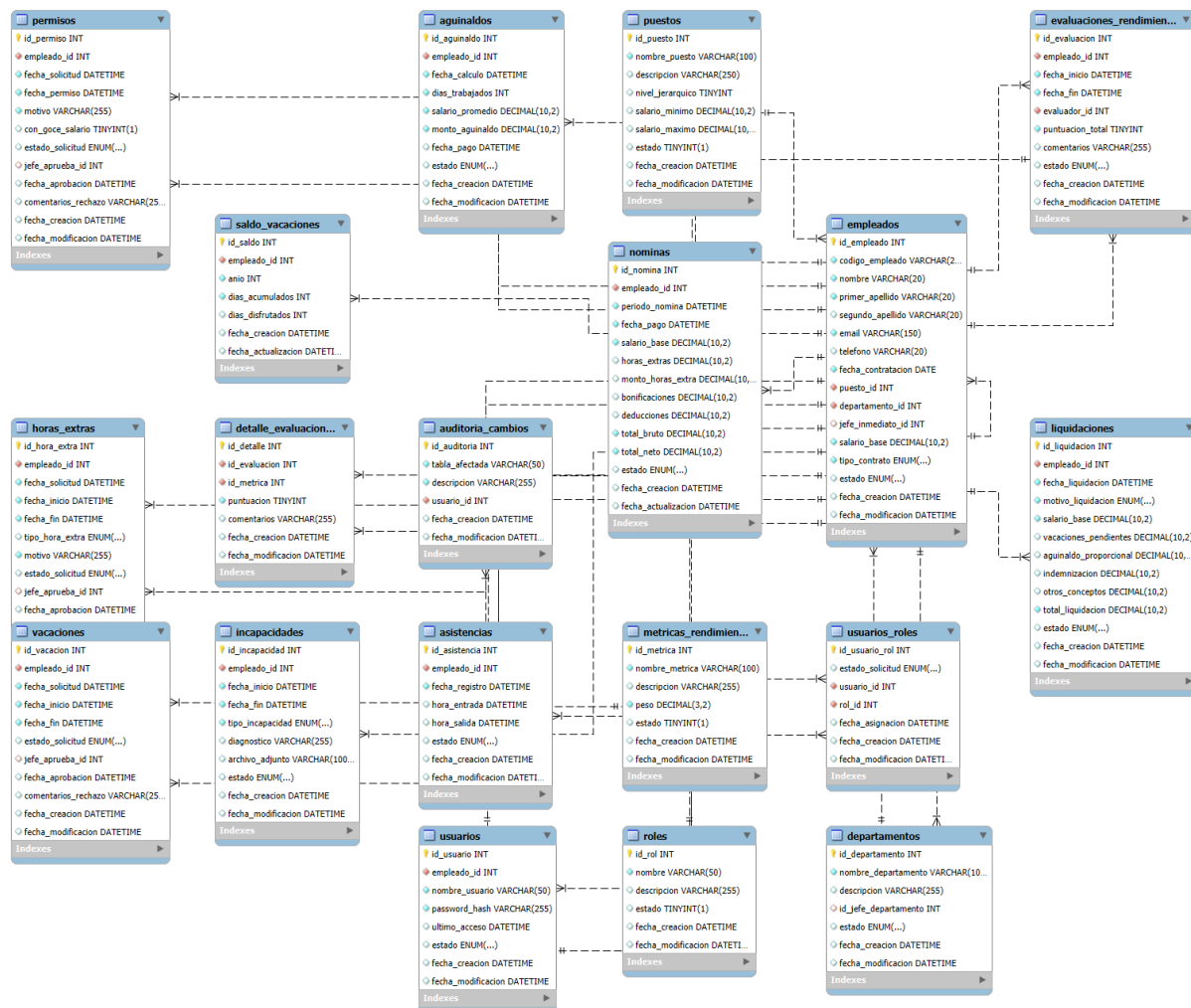


Tabla 53

Diccionario de datos de la tabla aguinaldo

Columna		Tipo de dato	Descripción
PK	id_aguinaldo	INT	Código de aguinaldo
FK	empleado_id	INT	Código de empleado
	fecha_calculo	DATETIME	Fecha en la que se realizó el cálculo
	dias_trabajados	INT	Días totales trabajados según el colaborador

	salario_promedio	DECIMAL(10,2)	Salario promedio del colaborador
	monto_aguinaldo	DECIMAL(10,2)	Monto total de aguinaldo
	fecha_pago	DATETIME	Fecha en la que se realizará el pago del aguinaldo
	estado	ENUM('CALCULADO', 'PAGADO')	Estado actual del proceso de pago del aguinaldo
	fecha_creacion	DATETIME	Fecha de creación de la fila en la tabla
	fecha_modificacion	DATETIME	Fecha de modificación de la fila en la tabla

Tabla 54*Diccionario de datos de la tabla asistencia*

Columna		Tipo de dato	Descripción
PK	id_asistencia	INT	Código de asistencia
FK	empleado_id	INT	Código de empleado
	fecha_registro	DATETIME	Fecha de registro de la asistencia
	hora_entrada	DATETIME	Hora de entrada al trabajo
	hora_salida	DATETIME	Hora de salida del trabajo
	estado	ENUM('PRESENTE', 'AUSENTE', 'TARDANZA', 'PERMISO')	Estado de asistencia del colaborador según la fecha
	fecha_creacion	DATETIME	Fecha de creación de la fila en la tabla
	fecha_modificacion	DATETIME	Fecha de modificación de la fila en la tabla

Tabla 55*Diccionario de datos de la tabla auditoría*

Columna		Tipo de dato	Descripción
PK	id_auditoria	INT	Código de auditoria
	tabla_afectada	VARCHAR(50)	Tabla a la cual se le realizaron cambios
	descripcion	VARCHAR(255)	Descripción de los cambios o

			acciones tomadas
FK	usuario_id	INT	Código de usuario que realizó los cambios
	fecha_creacion	DATETIME	Fecha de creación de la fila en la tabla
	fecha_modificacion	DATETIME	Fecha de modificación de la fila en la tabla

Tabla 56*Diccionario de datos de la tabla departamento*

Columna		Tipo de dato	Descripción
PK	id_departamento	INT	Código del departamento
	nombre_departamento	VARCHAR(100)	Nombre del departamento
	descripcion	VARCHAR(255)	Descripción del departamento
FK	id_jefe_departamento	INT	Código del empleado que hace de jefe en el departamento
	estado	ENUM('ACTIVO', 'INACTIVO')	Estado del registro del departamento
	fecha_creacion	DATETIME	Fecha de creación de la fila en la tabla
	fecha_modificacion	DATETIME	Fecha de modificación de la fila en la tabla

Tabla 57*Diccionario de datos de la tabla detalles de evaluación*

Columna		Tipo de dato	Descripción
PK	id_detalle	INT	Código de detalle de evaluación
FK	id_evaluacion	INT	Código de evaluación
FK	id_metrica	INT	Código de métrica
	puntuacion	TINYINT	Puntuación de la métrica en conjunto con la evaluación
	comentarios	VARCHAR(255)	Retroalimentación o notas sobre el detalle de la evaluación
	fecha_creacion	DATETIME	Fecha de creación de la fila en la tabla

	fecha_modificacion	DATETIME	Fecha de modificación de la fila en la tabla
--	--------------------	----------	--

Tabla 58*Diccionario de datos de la tabla empleados*

Columna		Tipo de dato	Descripción
PK	id_empleado	INT	Código del registro de empleado
	codigo_empleado	VARCHAR(20)	Código del empleado
	nombre	VARCHAR(20)	Nombre del empleado
	primer_apellido	VARCHAR(20)	Primer apellido del empleado
	segundo_apellido	VARCHAR(20)	Segundo apellido del empleado
	email	VARCHAR(150)	Correo electrónico del empleado
	telefono	VARCHAR(20)	Número de teléfono del empleado
	fecha_contratacion	DATETIME	Fecha en la que se contrató al empleado
FK	puesto_id	INT	Código de referencia al puesto del empleado
FK	departamento_id	INT	Código de referencia al departamento que pertenece el empleado
FK	jefe_inmediato_id	INT	Código de referencia al jefe al que el empleado rinde cuentas
	salario_base	DECIMAL(10,2)	Salario base del empleado
	tipo_contrato	ENUM('FIJO', 'TEMPORAL', 'PRUEBA') NOT NULL	Tipo de contrato que mantiene el empleado
	estado	ENUM('ACTIVO', 'INACTIVO', 'LICENCIA')	Estado del empleado en la organización
	fecha_creacion	DATETIME	Fecha de creación de la fila en la tabla
	fecha_modificacion	DATETIME	Fecha de modificación de la fila en la tabla

Tabla 59*Diccionario de datos de la tabla evaluaciones de rendimiento*

Columna		Tipo de dato	Descripción
PK	id_evaluacion	INT	Código de referencia a la evaluación
FK	empleado_id	INT	Código del empleado al que se le realiza la evaluación
	fecha_inicio	DATETIME	Fecha de inicio de la evaluación
	fecha_fin	DATETIME	Fecha de finalización de la evaluación
FK	evaluador_id	INT	Código del empleado que actúa como evaluador
	puntuacion_total	TINYINT	Puntuación total de la evaluación
	comentarios	VARCHAR(255)	Retroalimentación de la evaluación efectuada
	estado	ENUM('BORRADOR', 'COMPLETADA')	Estado del registro en el que se encuentra
	fecha_creacion	DATETIME	Fecha de creación de la fila en la tabla
	fecha_modificacion	DATETIME	Fecha de modificación de la fila en la tabla

Tabla 60*Diccionario de datos de la tabla horas extra*

Columna		Tipo de dato	Descripción
PK	id_hora_extra	INT	Código de referencia al registro de la hora extra
FK	empleado_id	INT	Código de referencia al empleado
	fecha_solicitud	DATETIME	Fecha en la que se hace la solicitud de la hora extra
	fecha_inicio	DATETIME	Fecha en la que empiezan a correr las horas extra
	fecha_fin	DATETIME	Fecha en la que se dan por finalizadas las horas extras
	tipo_hora_extra	ENUM('PENDIENTE', 'APROBADA')	Tipo de hora extra

		'RECHAZADA')	
	motivo	VARCHAR(255)	Explicación de la hora extra
	estado_solicitud	ENUM('PENDIENTE', 'APROBADA', 'RECHAZADA')	Estado en el que se encuentra la solicitud de las horas extra
FK	jefe_aprueba_id	INT	Colaborar que aprueba las horas extra o que las rige
	fecha_aprobacion	DATETIME	Fecha que se aprueba las solicitud de las horas extra
	fecha_creacion	DATETIME	Fecha de creación de la fila en la tabla
	fecha_modificacion	DATETIME	Fecha de modificación de la fila en la tabla

Tabla 61*Diccionario de datos de la tabla incapacidades*

Columna		Tipo de dato	Descripción
PK	id_incapacidad	INT	Código de referencia al registro de la incapacidad
FK	empleado_id	INT	Código de referencia al empleado
	fecha_inicio	DATETIME	Fecha de inicio de la incapacidad
	fecha_fin	DATETIME	Fecha de finalización de la incapacidad
	tipo_incapacidad	ENUM('ENFERMEDAD', 'ACCIDENTE', 'MATERNIDAD', 'PATERNIDAD')	Tipo de incapacidad
	diagnostico	VARCHAR(255)	Diagnóstico de la incapacidad
	archivo_adjunto	VARCHAR(1000)	Estado ENUM('ACTIVA', 'FINALIZADA')
	estado	ENUM('ACTIVA', 'FINALIZADA')	Estado del proceso de incapacidad
	fecha_creacion	DATETIME	Fecha de creación de la fila en la tabla
	fecha_modificacion	DATETIME	Fecha de modificación de la fila en la tabla

Tabla 62*Diccionario de datos de la tabla liquidaciones*

Columna		Tipo de dato	Descripción
PK	id_liquidacion	INT	Código de referencia de la liquidación
FK	empleado_id	INT	Empleado al que se le aplica la liquidación
	fecha_liquidacion	DATETIME	Fecha en la que se realiza la liquidación
	motivo_liquidacion	ENUM('RENUNCIA', 'DESPIDO', 'DESPIDO_JUSTIFICADO', 'JUBILACION')	Motivo por el cual se realiza el proceso de liquidación
	salario_base	DECIMAL(10,2)	Salario base del colaborador
	vacaciones_pendientes	DECIMAL(10,2)	Vacaciones pendientes del colaborador
	aguinaldo_proporcional	DECIMAL(10,2)	Aguinaldo proporcional
	indemnizacion	DECIMAL(10,2)	Monto de indemnización
	otros_conceptos	DECIMAL(10,2)	Otros conceptos
	total_liquidacion	DECIMAL(10,2)	Monto total de la liquidación
	estado	ENUM('CALCULADA', 'PAGADA')	Estado de la liquidación
	fecha_creacion	DATETIME	Fecha de creación de la fila en la tabla
	fecha_modificacion	DATETIME	Fecha de modificación de la fila en la tabla

Tabla 63*Diccionario de datos de la tabla métricas de rendimiento*

Columna		Tipo de dato	Descripción
PK	id_metrica	INT	Código de la métrica
	nombre_metrica	VARCHAR(100)	Nombre de la métrica
	descripcion	VARCHAR(255)	Descripción sobre la métrica
	peso	DECIMAL(3,2)	Grado de importancia de la métrica

	estado	BOOLEAN	Estado de la métrica
	fecha_creacion	DATETIME	Fecha de creación de la fila en la tabla
	fecha_modificacion	DATETIME	Fecha de modificación de la fila en la tabla

Tabla 64*Diccionario de datos de la tabla nóminas*

Columna		Tipo de dato	Descripción
PK	id_nomina	INT	Identificador único de la nómina
FK	empleado_id	INT	ID del empleado al que pertenece la nómina
	periodo_nomina	DATETIME	Periodo que comprende la nómina
	fecha_pago	DATETIME	Fecha en la que se efectúa el pago de la nómina
	salario_base	DECIMAL(10,2)	Salario fijo asignado al empleado.
	horas_extras	DECIMAL(10,2)	Cantidad de horas extras trabajadas.
	monto_horas_extra	DECIMAL(10,2)	Monto total correspondiente a las horas extras.
	bonificaciones	DECIMAL(10,2)	Pagos adicionales (bonos, incentivos).
	deducciones	DECIMAL(10,2)	Descuentos aplicados a la nómina.
	total_bruto	DECIMAL(10,2)	Total antes de deducciones.
	total_netto	DECIMAL(10,2)	Total final después de deducciones.
	estado	ENUM('PENDIENTE','PAGADA','ANULADA')	Estado actual de la nómina.
	fecha_creacion	DATETIME	Fecha y hora en que se creó el registro.
	fecha_actualizacion	DATETIME	Última fecha de actualización del registro.

Tabla 65*Diccionario de datos de la tabla roles*

Columna		Tipo de dato	Descripción
PK	id_rol	INT	Identificador único del rol.
	nombre	VARCHAR(50)	Nombre del rol; debe ser único.
	descripcion	VARCHAR(255)	Descripción del rol.
	estado	BOOLEAN	Indica si el rol está activo (TRUE) o inactivo (FALSE).
	fecha_creacion	DATETIME	Fecha y hora en que se registró el rol.
	fecha_modificacion	DATETIME	Fecha y hora de la última modificación.

Tabla 66*Diccionario de datos de la tabla permisos*

Columna		Tipo de dato	Descripción
PK	id_permiso	INT	Identificador único del permiso solicitado.
FK	empleado_id	INT	ID del empleado que solicita el permiso.
	fecha_solicitud	DATETIME	Fecha en que se realizó la solicitud del permiso.
	fecha_permiso	DATETIME	Fecha para la cual se solicita el permiso.
	motivo	VARCHAR(255)	Razón o justificación del permiso.
	con_goce_salario	BOOLEAN	Indica si el permiso es con goce de salario.
	estado_solicitud	ENUM('PENDIENTE', 'APROBADA', 'RECHAZADA')	Estado actual de la solicitud del permiso.
FK	jefe_aprueba_id	INT	ID del jefe o supervisor que aprueba o rechaza el permiso.
	fecha_aprobacion	DATETIME	Fecha en la que se aprobó o rechazó la solicitud.
	comentarios_rechazo	VARCHAR(255)	Comentarios del rechazo, si aplica.

	fecha_creacion	DATETIME	Fecha de creación del registro.
	fecha_modificacion	DATETIME	Última fecha de modificación del registro.

Tabla 67*Diccionario de datos de la tabla puestos*

Columna		Tipo de dato	Descripción
PK	id_puesto	INT	Identificador único del puesto.
	nombre_puesto	VARCHAR(100)	Nombre del puesto dentro de la organización.
	descripcion	VARCHAR(250)	Descripción general del puesto.
	nivel_jerarquico	TINYINT	Nivel o rango jerárquico del puesto.
	salario_minimo	DECIMAL(10,2)	Salario mínimo asignado al puesto.
	salario_maximo	DECIMAL(10,2)	Salario máximo permitido para el puesto.
	estado	BOOLEAN	Indica si el puesto está activo (TRUE) o inactivo (FALSE).
	fecha_creacion	DATETIME	Fecha de creación del registro.
	fecha_modificacion	DATETIME	Última fecha de modificación del registro.

Tabla 68*Diccionario de datos de la tabla saldo de vacaciones*

Columna		Tipo de dato	Descripción
PK	id_saldo	INT	Identificador único del saldo de vacaciones.
FK	empleado_id	INT	ID del empleado al que pertenece el saldo.
	anio	INT	Año al que corresponde el saldo vacacional.
	dias_acumulados	INT	Total de días de vacaciones acumulados en el año.
	dias_disfrutados	INT	Cantidad de días de vacaciones ya utilizados.

	fecha_creacion	DATETIME	Fecha de creación del registro.
	fecha_actualizacion	DATETIME	Fecha de la última actualización del saldo.

Tabla 69*Diccionario de datos de la tabla usuarios*

Columna		Tipo de dato	Descripción
PK	id_usuario	INT	Identificador único del usuario del sistema.
FK	empleado_id	INT	ID del empleado asociado al usuario; relación 1:1.
	nombre_usuario	VARCHAR(50)	Nombre de inicio de sesión; debe ser único.
	password_hash	VARCHAR(255)	Contraseña en formato <i>hash</i> para seguridad.
	ultimo_acceso	DATETIME	Última fecha y hora en que el usuario ingresó.
	estado	ENUM('ACTIVO','INACTIVO','BLOQUEADO')	Estado actual del usuario en el sistema.
	fecha_creacion	DATETIME	Fecha de creación del registro.
	fecha_modificacion	DATETIME	Última fecha de modificación del registro.

Tabla 70*Diccionario de datos de la tabla usuarios roles*

Columna		Tipo de dato	Descripción
PK	id_usuario_rol	INT	Identificador único de la asignación usuario-rol.
	estado_solicitud	ENUM('PENDIENTE', 'APROBADA', 'RECHAZADA')	Estado del proceso de asignación del rol.
FK	usuario_id	INT	ID del usuario al que se asigna el rol.
FK	rol_id	INT	ID del rol asignado al usuario.
	fecha_asignacion	DATETIME	Fecha en que se registró la asignación del rol.

	fecha_creacion	DATETIME	Fecha de creación del registro.
	fecha_modificacion	DATETIME	Última fecha de modificación del registro.

Tabla 71*Diccionario de datos de la tabla vacaciones*

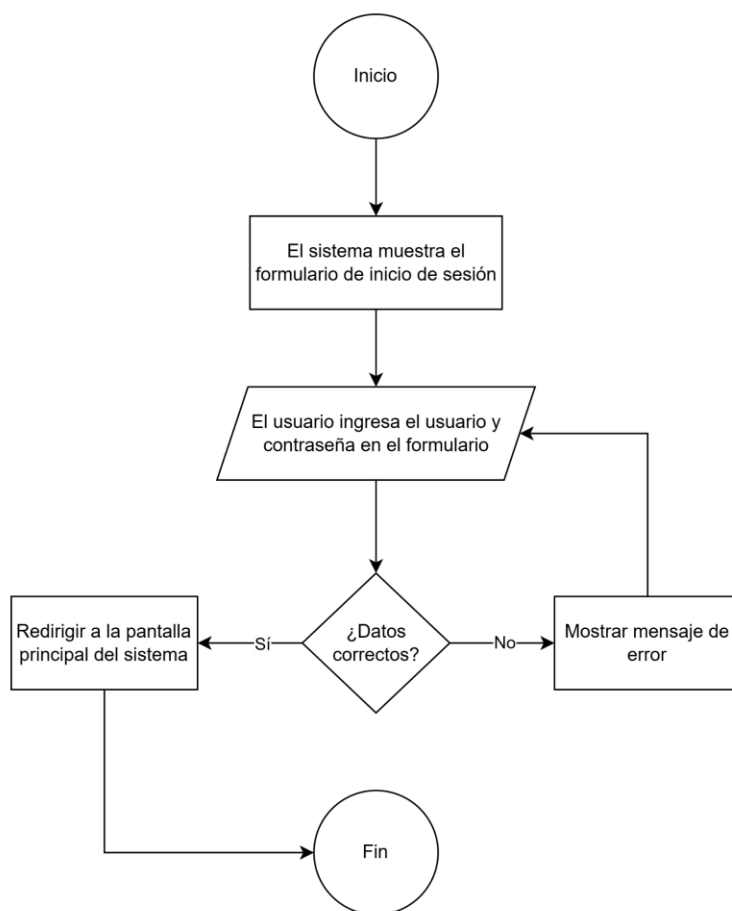
Columna		Tipo de dato	Descripción
PK	id_vacacion	INT	Identificador único de la solicitud de vacaciones.
FK	empleado_id	INT	ID del empleado que solicita las vacaciones.
	fecha_solicitud	DATETIME	Fecha en que se realizó la solicitud.
	fecha_inicio	DATETIME	Fecha de inicio del período de vacaciones.
	fecha_fin	DATETIME	Fecha de finalización del período de vacaciones.
	estado_solicitud	ENUM('PENDIENTE','APROBADA','RECHAZADA','DISFRUTANDO','COMPLETADA')	Estado actual de la solicitud.
FK	jefe_aprueba_id	INT	ID del jefe o supervisor que aprueba o rechaza la solicitud.
	fecha_aprobacion	DATETIME	Fecha en la que se aprobó o rechazó la solicitud.
	comentarios_rechazo	VARCHAR(255)	Comentarios en caso de rechazo.
	fecha_creacion	DATETIME	Fecha de creación del registro.
	fecha_modificacion	DATETIME	Fecha de última modificación del registro.

Diseño de procesos. Esta sección incluye los diagramas de flujo que detallan los procesos fundamentales del prototipo funcional del sistema de recursos humanos. Estos diagramas ofrecen una representación visual organizada y comprensible de la serie de acciones que realiza la persona usuaria en cada módulo. Además, de mostrar las decisiones automatizadas que ejecuta el sistema a lo largo de las distintas operaciones.

El propósito de estos diagramas es facilitar la comprensión del funcionamiento de los procesos del sistema, al ilustrar de manera secuencial y estructurada la interacción entre las acciones del usuario y las respuestas que se generan por el programa. Estos procedimientos sirven como herramienta de apoyo para el análisis y la documentación del proyecto, además de contribuir a la identificación de los datos de entrada, los procesos intermedios y los resultados.

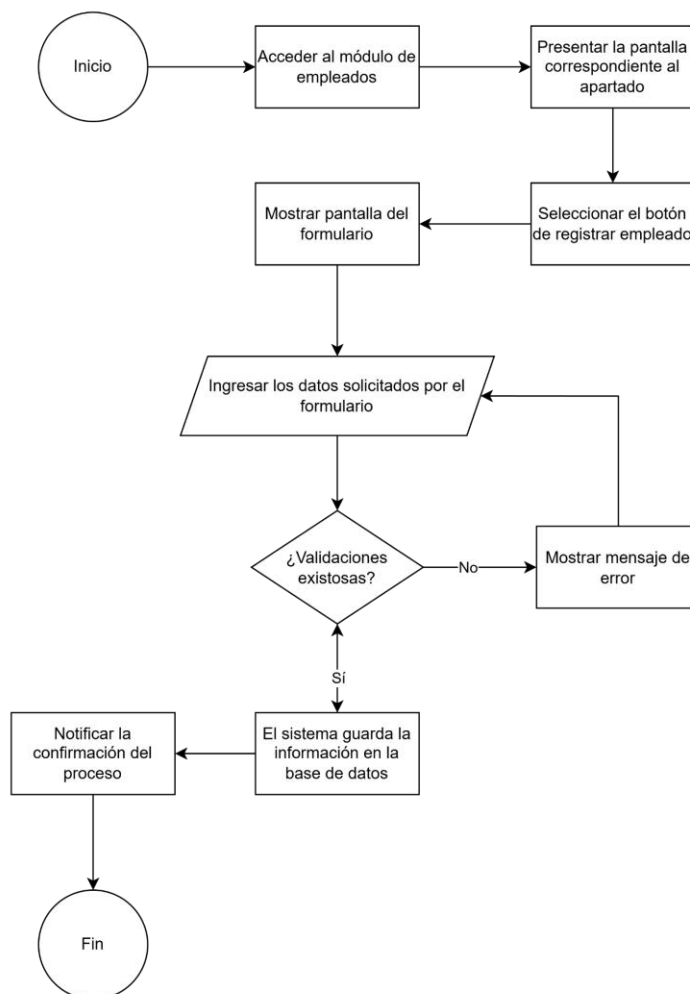
Figura 8

Diagrama de flujo del proceso de inicio de sesión del sistema

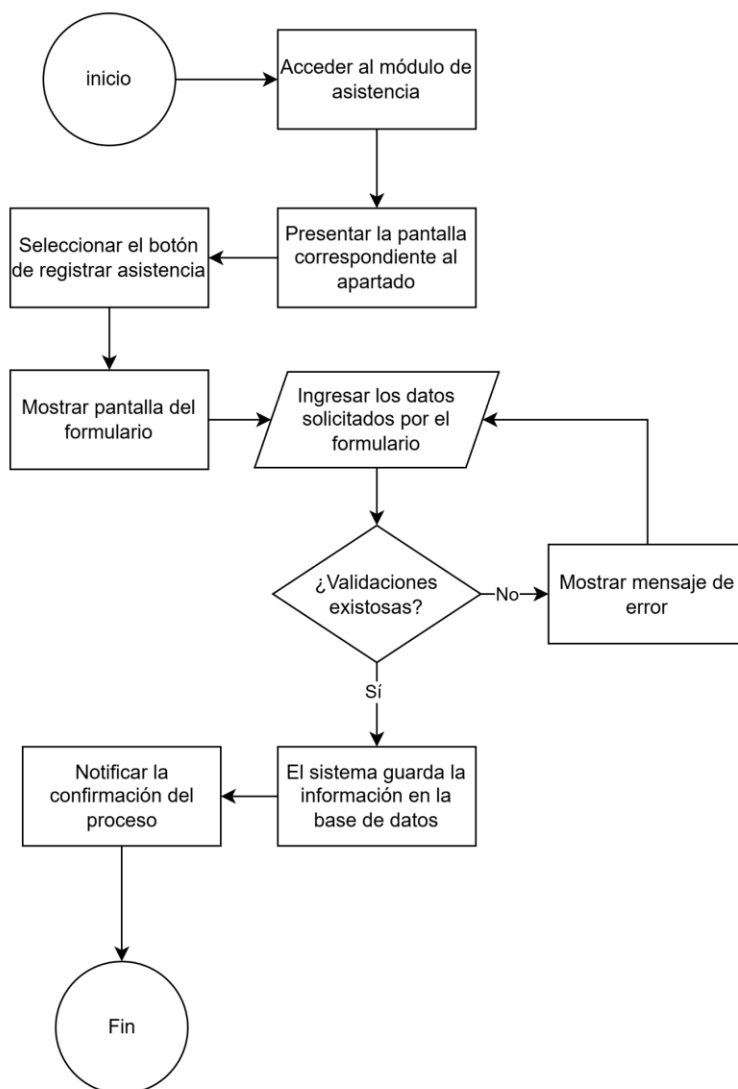


El diagrama presentado en la Figura 8 describe el proceso mediante el cual la persona usuaria obtiene acceso al sistema. El flujo comienza con la visualización del formulario de inicio de sesión, donde la persona usuaria introduce su nombre de usuario y contraseña. Si las credenciales ingresadas son válidas, el sistema concede el acceso a la vista principal; de lo contrario, muestra un mensaje de error que indica que los datos proporcionados no son correctos.

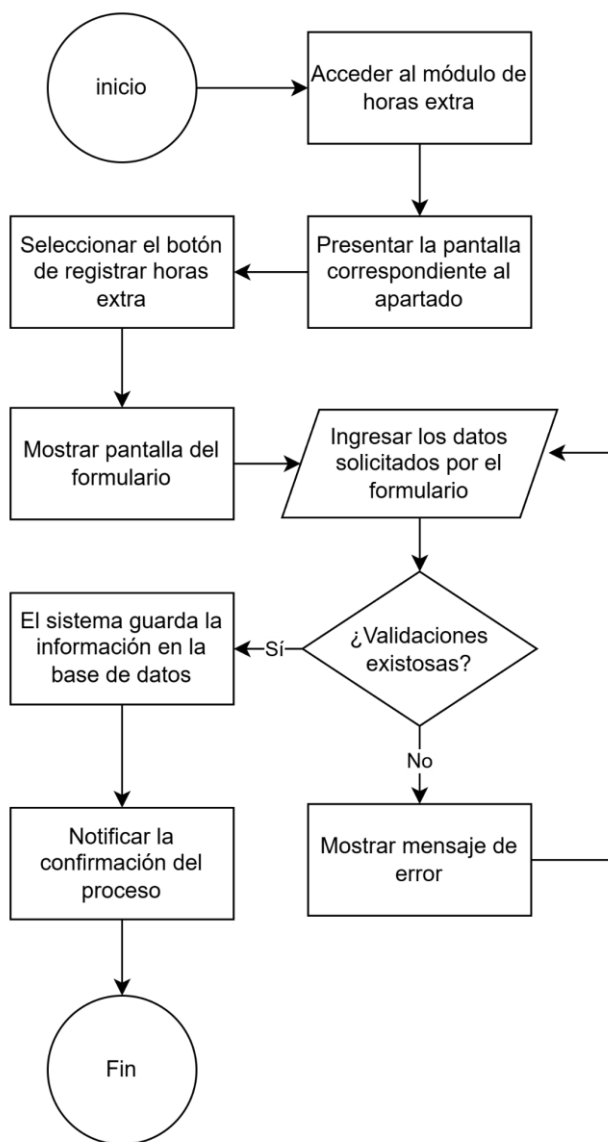
Figura 9
Diagrama de flujo del registro de un empleado



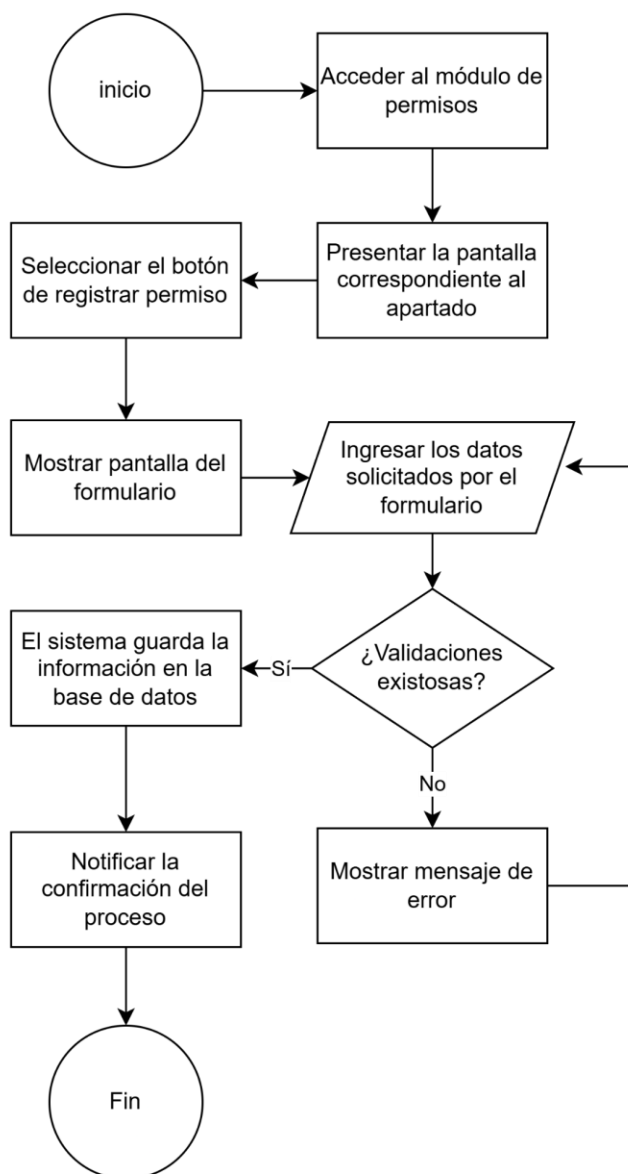
El diagrama ilustra el procedimiento que ejecuta el sistema para el registro de un nuevo empleado. El proceso comienza cuando la persona usuaria ingresa al módulo de empleados y selecciona la opción *registrar empleado*. Posteriormente, se despliega el formulario correspondiente para el ingreso de la información requerida. Una vez que los datos se han completado, el sistema valida la información proporcionada. Si la validación es correcta, el sistema almacena el registro en la base de datos y muestra un mensaje de confirmación; de lo contrario, presenta un mensaje de error solicitando la corrección de los datos antes de concluir el proceso.

Figura 10*Diagrama de flujo del registro de asistencia*

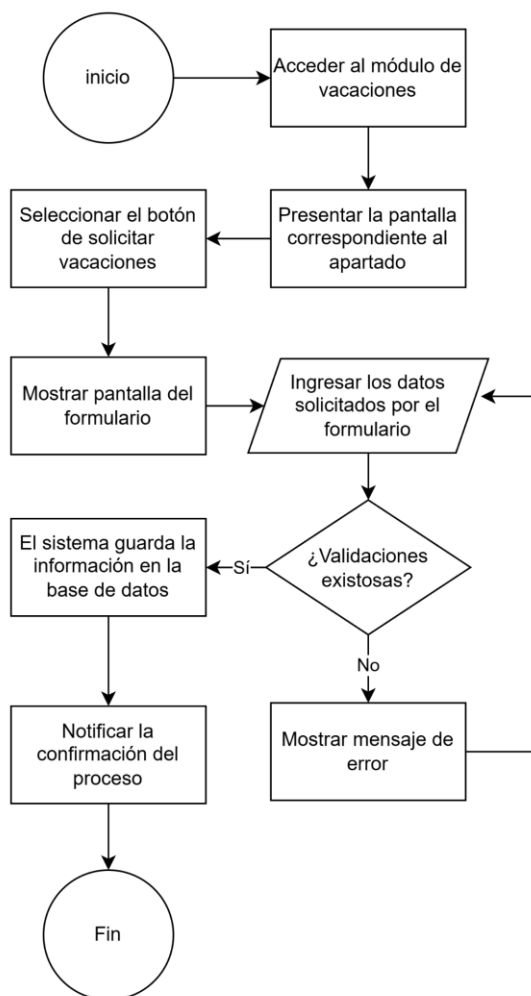
El diagrama ilustra el proceso mediante el cual el personal de Recursos Humanos registra la asistencia diaria de los empleados. El flujo comienza con el acceso al módulo de asistencia, donde se muestra la vista principal de este. Enseguida, se selecciona la opción *Registrar* del empleado correspondiente y se despliega el formulario para ingresar la información requerida. Una vez que se completan los datos, el sistema valida la información. Si los datos son correctos, el registro se guarda en la base de datos y se presenta un mensaje de éxito; de lo contrario, se notifica al usuario por medio de un mensaje de error, solicitando la corrección de los datos antes de concluir el proceso.

Figura 11*Diagrama de flujo del registro de horas extra*

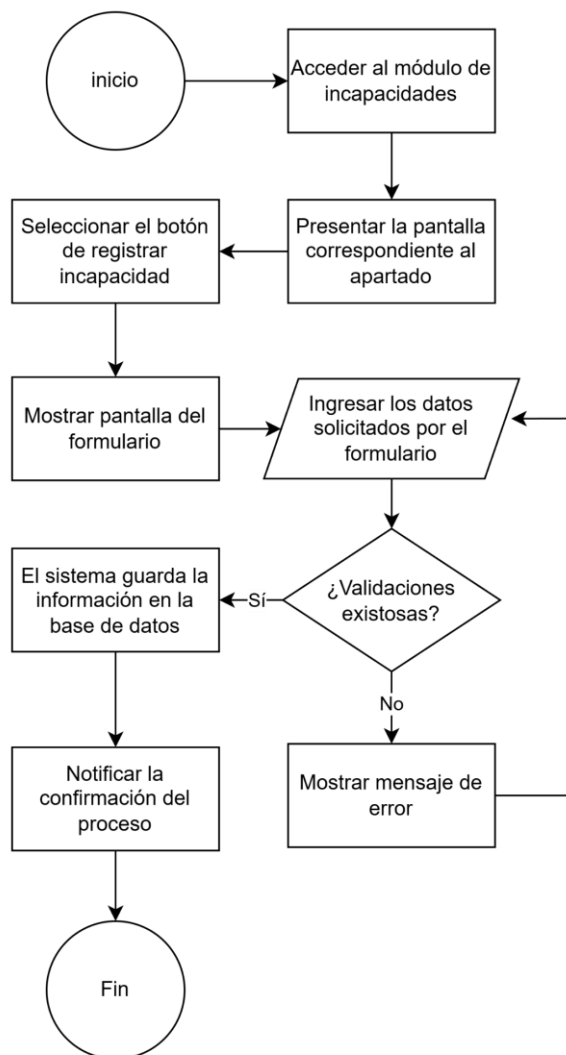
El diagrama muestra el proceso mediante el cual se registran las horas extra trabajadas por un empleado. El flujo comienza con el acceso al módulo de horas extra, donde se presenta la vista principal de este. Enseguida, la persona usuaria selecciona la opción *Registrar* y se despliega el formulario para ingresar la información requerida. Una vez que se completan todos los campos, el sistema valida los datos ingresados. Si la información es correcta, el registro se guarda en la base de datos y se muestra un mensaje de éxito; en caso contrario, se despliega un mensaje de error que le permite al usuario corregir los datos antes de finalizar el proceso.

Figura 12*Diagrama de flujo del registro de permisos*

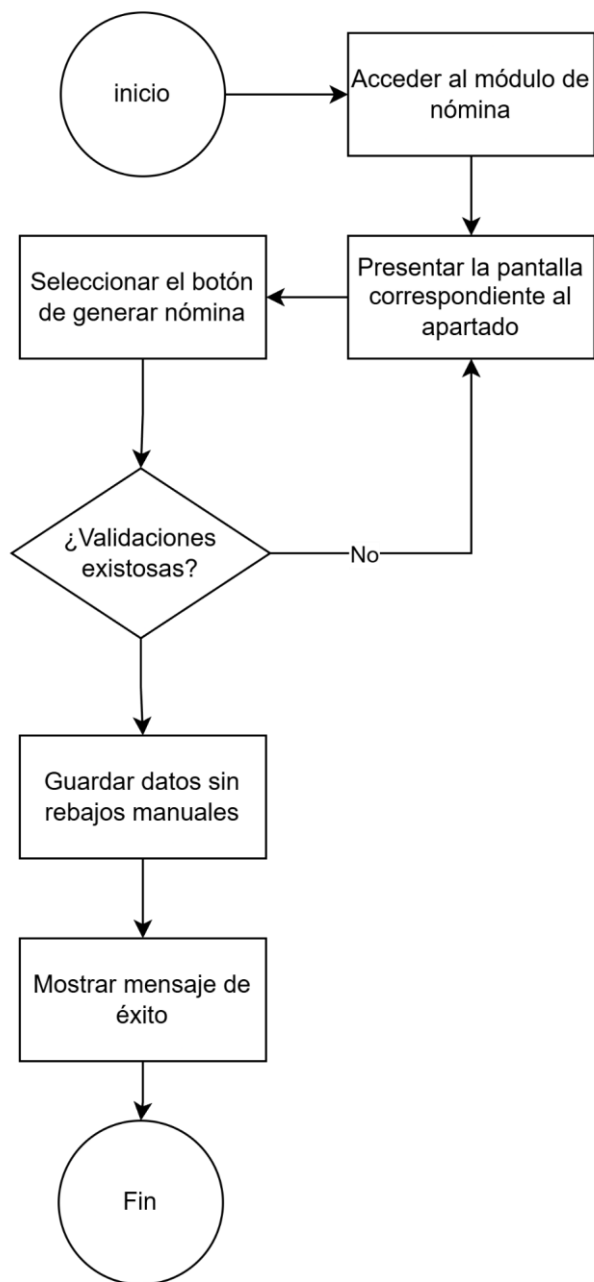
El diagrama ilustra el proceso que realiza el sistema para registrar un permiso de un empleado. El flujo comienza con el ingreso al módulo de permisos, donde se selecciona la opción *Registrar*. Posteriormente, la persona usuaria completa el formulario con la información requerida y el sistema valida los datos ingresados. Si la información es correcta, el registro se almacena en la base de datos y se muestra un mensaje de éxito; de lo contrario, se presenta un mensaje de error que indica que los datos deben corregirse.

Figura 13*Diagrama de flujo del registro de solicitud de vacaciones*

El diagrama describe el procedimiento que sigue la persona usuaria para registrar una solicitud de vacaciones. El proceso comienza cuando se accede al módulo de vacaciones y se selecciona la opción *Solicitar* del empleado. A continuación, el sistema muestra el formulario correspondiente para el ingreso de la información requerida. Si los datos ingresados son correctos, el sistema almacena la petición en la base de datos y presenta un mensaje de confirmación; en caso contrario, muestra un mensaje de error.

Figura 14*Diagrama de flujo del registro de incapacidad*

El diagrama ilustra el proceso mediante el cual se registra una incapacidad de un empleado en el sistema. El procedimiento comienza cuando se accede al módulo de incapacidades, donde la persona usuaria selecciona la opción *Agregar* correspondiente al empleado. Posteriormente, el sistema despliega el formulario para ingresar los datos requeridos y, si la información resulta válida, almacena el registro en la base de datos y muestra un mensaje de confirmación; en caso contrario, presenta un mensaje de error.

Figura 15*Diagrama de flujo del cálculo de nómina*

El proceso inicia cuando la persona usuaria ingresa al módulo de nómina, selecciona el período correspondiente y elige la opción *Generar*. El sistema ejecuta el cálculo masivo de las nóminas para los empleados y, una vez finalizado, estas quedan generadas y disponibles para su validación.

Diseño de salidas. En esta sección se describen los principales formatos de salida del

sistema, los cuales permiten mostrar la información procesada y los resultados obtenidos a partir de los datos almacenados. Estos formatos ayudan al usuario a visualizar de forma clara y organizada los registros y reportes que se generan, lo que facilita la toma de decisiones dentro del Área de Recursos Humanos. Además, cada uno se diseñó considerando las necesidades específicas de la empresa, asegurando que la información presentada sea correcta, relevante y útil para una gestión administrativa adecuada.

Figura 16

Diseño de la salida de datos personales del empleado

The screenshot shows a web application interface with a modal window titled "Detalle del Empleado". The modal contains the following information:

ID	Estado
1	ACTIVO
Nombre completo	
Minor Alfaro Rodriguez	
Código	Email
EMP001	minor@asncr.com
Teléfono	Departamento
88767234	Administración
Puesto	Fecha de contratación
Administrador	30/9/2025
Salario base	Tipo de contrato
650000	FIJO

Esta vista presenta de forma clara, ordenada y accesible la información personal del empleado seleccionado en el Sistema de Gestión de Recursos Humanos. De esta manera, permite consultar datos clave como nombre, correo electrónico, teléfono, departamento, puesto, salario base y tipo de contrato. Asimismo, ofrece opciones para editar o eliminar la información, lo que facilita una gestión eficiente y actualizada de los datos del colaborador.

Figura 17
Diseño de salida del registro de asistencia semanal

Lista de Asistencias							Agregar Asistencia
Buscar empleado...	Todos los estados	Hoy	Esta Semana	Este Mes	Todos		
Mostrando 2 de 2 registros							
Fecha	Empleado	Hora Entrada	Hora Salida	Horas Trabajadas	Estado	Acciones	
04/02/2026	Romario Hidalgo	07:58	17:00	9h 2m	PRESENTE	...	
03/02/2026	Romario Hidalgo	08:00	17:00	9h 0m	PRESENTE	...	

Esta pantalla muestra el registro semanal de asistencia del empleado, donde se detallan las horas de entrada y salida correspondientes a cada día en el período seleccionado. Además, permite visualizar claramente el estado de cada jornada laboral, así como ofrecer opciones para añadir nuevas asistencias o modificar las registradas e incluir notas importantes que se relacionan con las jornadas. Su diseño está pensado para facilitar el monitoreo y control del historial de asistencia, lo que proporciona al usuario una herramienta eficiente para la gestión del control de asistencia en el sistema.

Figura 18
Diseño de salida del historial de permisos

Lista de Permisos									Agregar Solicitud
Buscar empleado...	Todos los estados	Con/Sin Goce	Hoy	Esta Semana	Este Mes	Todos			
Mostrando 2 de 2 permisos									
ID	Empleado	Fecha Solicitud	Fecha Permiso	Motivo	Con Goce	Estado	Aprobado Por	Acciones	
1	Romario Hidalgo Garro	04/02/2026	06/02/2026	No voy a estar	No	RECHAZADA	Jefe #1 04/02/2026	...	
2	Romario Hidalgo Garro	05/02/2026	09/02/2026	Debo acudir a una cita medica	Si	APROBADA	Jefe #1 05/02/2026	...	

Esta pantalla presenta el listado de permisos solicitados por los empleados, donde se muestra información detallada sobre cada solicitud, incluidas las fechas, los motivos y si el permiso es con o sin goce de sueldo. Permite consultar el estado de cada permiso —ya sea aprobado o rechazado—, así como identificar quién autorizó la decisión y cuándo lo hizo. Además, ofrece la opción de agregar nuevas solicitudes. Su diseño facilita la gestión y el seguimiento eficiente de los permisos en el sistema, lo que le proporciona al usuario una herramienta clara y práctica para el control administrativo.

Figura 19
Diseño de salida de incapacidades registradas

Lista de Incapacidades

Buscar empleado...

Todos los tipos

Todos los estados

Hoy

Esta Semana

Este Mes

Todos

Agregar Incapacidad

Mostrando 2 de 2 incapacidades

Empleado	Tipo	Estado	Fecha Inicio	Fecha Fin	Días	Diagnóstico	Adjunto	Acciones
Romario Hidalgo Garro	Enfermedad	ACTIVA	01/02/2026	10/02/2026	11 días	Lumbalgia aguda por sobreesfuerzo fi...		...
Romario Hidalgo Garro	Accidente	ACTIVA	06/02/2026	06/02/2026	1 día	Tuve un choque	-	...

Esta sección muestra la lista de incapacidades registradas para los empleados, detallando el tipo de incapacidad, el estado, las fechas de inicio y fin, la duración en días y el diagnóstico asociado. Permite visualizar de manera clara y organizada las incapacidades activas o pasadas, lo que facilita el seguimiento de cada caso. Además, brinda la opción de agregar nuevas incapacidades y consultar documentos adjuntos que se relacionan. Su diseño ofrece una herramienta práctica para la gestión eficiente del control de incapacidades en el sistema.

Figura 20
Diseño de la salida de los comprobantes de pago de nómina

Nómina Quincenal

4 de 4 nóminas

Generar Quincena

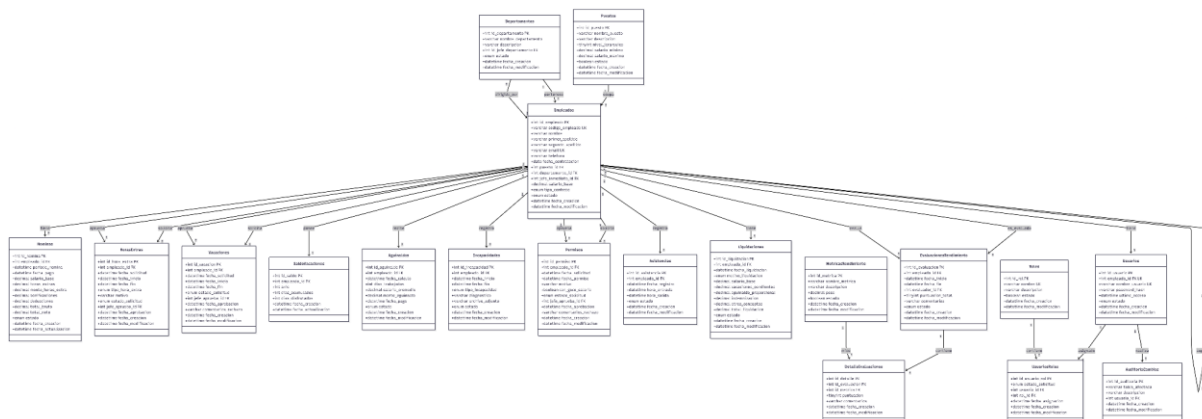
Período	Empleado	Puesto	Departamento	Salario Base	Total Bruto	Deducciones	Total Neto	Acciones
2ª Quincena enero 2026	Minor Alfaro EMP001	Administrador	Administración	₡325 000	₡325 000	-₡54 178	₡270 823	...
2ª Quincena enero 2026	Romario Hidalgo EMP002	Administrador	Administración	₡350 000	₡350 000	-₡58 345	₡291 655	...
1ª Quincena enero 2026	Minor Alfaro EMP001	Administrador	Administración	₡325 000	₡325 000	-₡54 178	₡270 823	...
1ª Quincena enero 2026	Romario Hidalgo EMP002	Administrador	Administración	₡350 000	₡350 000	-₡58 345	₡291 655	...

Esta sección muestra la lista de comprobantes de nómina quincenal que se generan para los empleados, detallando el período, puesto, departamento, salario base, total bruto, deducciones y total neto. Permite visualizar la información de manera clara y organizada, lo que facilita la revisión y el control de los pagos realizados. Además, ofrece la opción de generar nuevas quincenas y gestionar cada comprobante desde el apartado de acciones. Su diseño proporciona una herramienta práctica para la administración eficiente del proceso de nómina en el sistema.

Diagramas UML

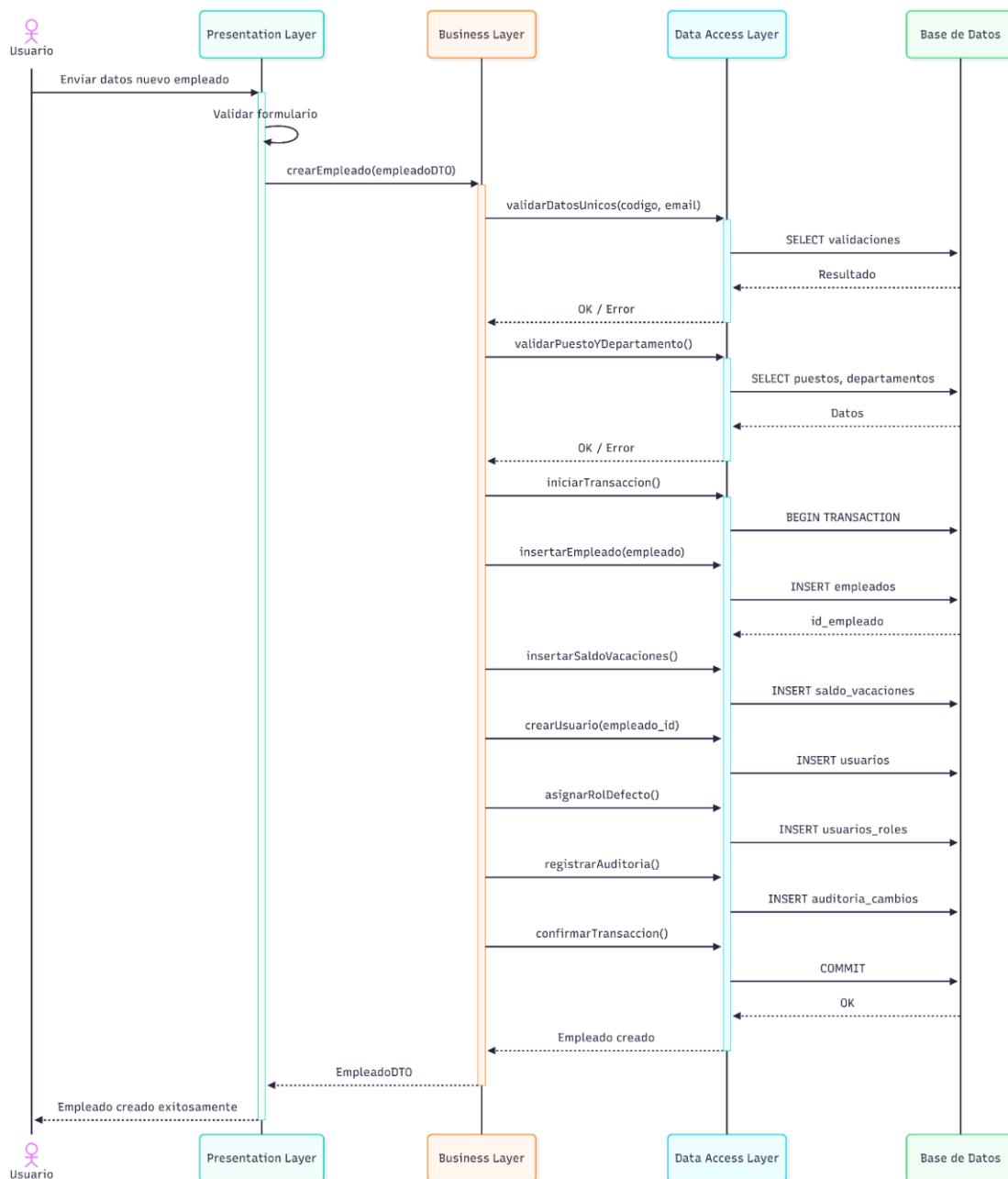
Figura 21

Diagrama de clases del prototipo de recursos humanos



Esta figura muestra el diagrama de clases del prototipo de recursos humanos, donde se detallan las principales entidades del sistema y sus relaciones, como empleados, departamentos, puestos, nóminas, permisos y evaluaciones. De esta manera, permite visualizar de forma clara la estructura de la base de datos y la interacción entre los distintos módulos. Su diseño facilita la comprensión del funcionamiento interno del sistema y sirve como base para su desarrollo e implementación.

Figura 22
Diagrama de secuencia para crear un empleado

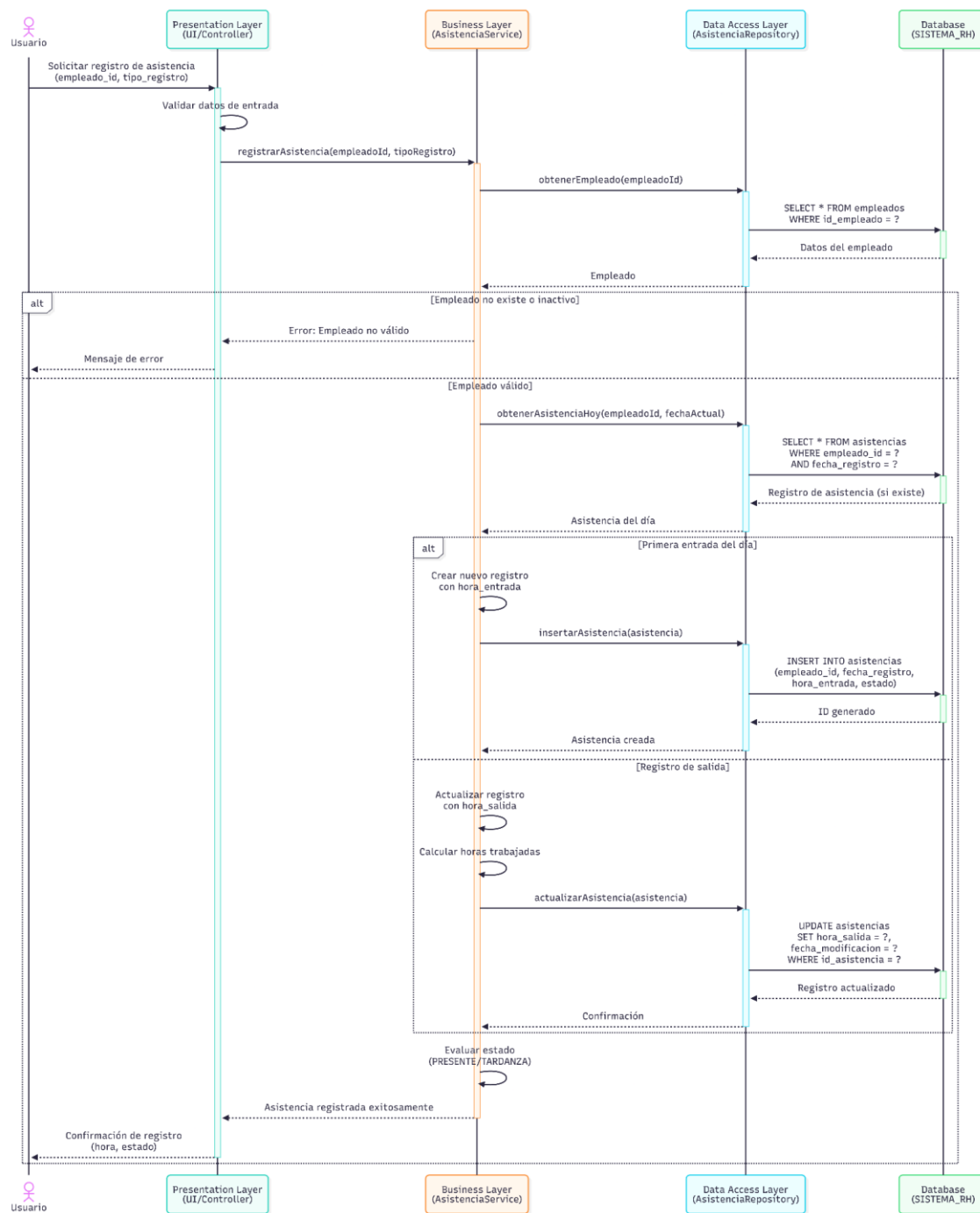


El diagrama de secuencia ilustra el proceso completo de creación de un nuevo empleado en el sistema. En este flujo, la persona usuaria ingresa la información requerida y el sistema valida los datos, verifica su unicidad y comprueba la relación con el puesto y el departamento. Posteriormente, se ejecuta una transacción que registra al empleado, crea su usuario, asigna un rol

por defecto y guarda la auditoría correspondiente. Por último, el sistema confirma la operación, lo que garantiza la integridad y consistencia de la información almacenada.

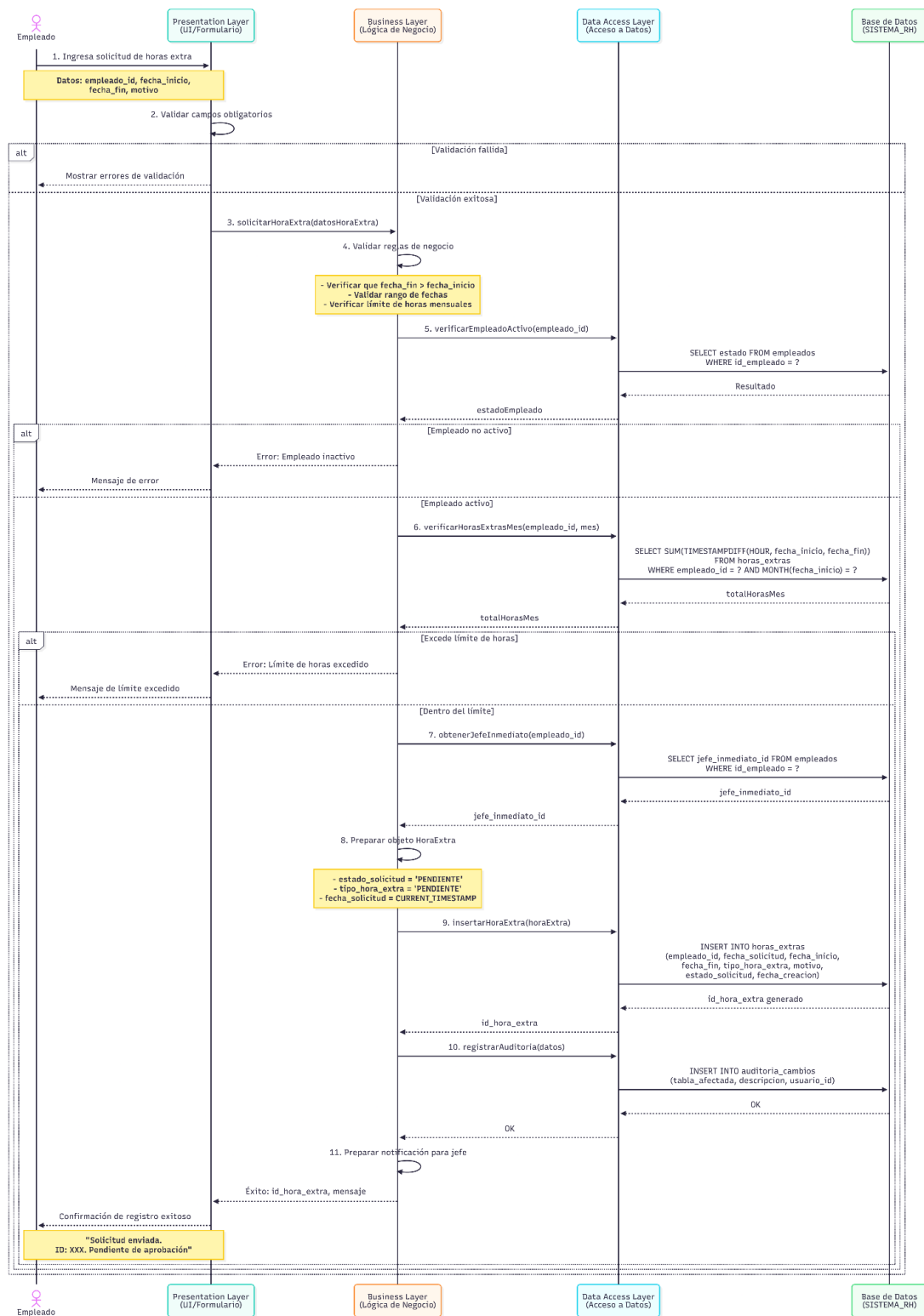
Figura 23

Diagrama de secuencia para registrar la asistencia



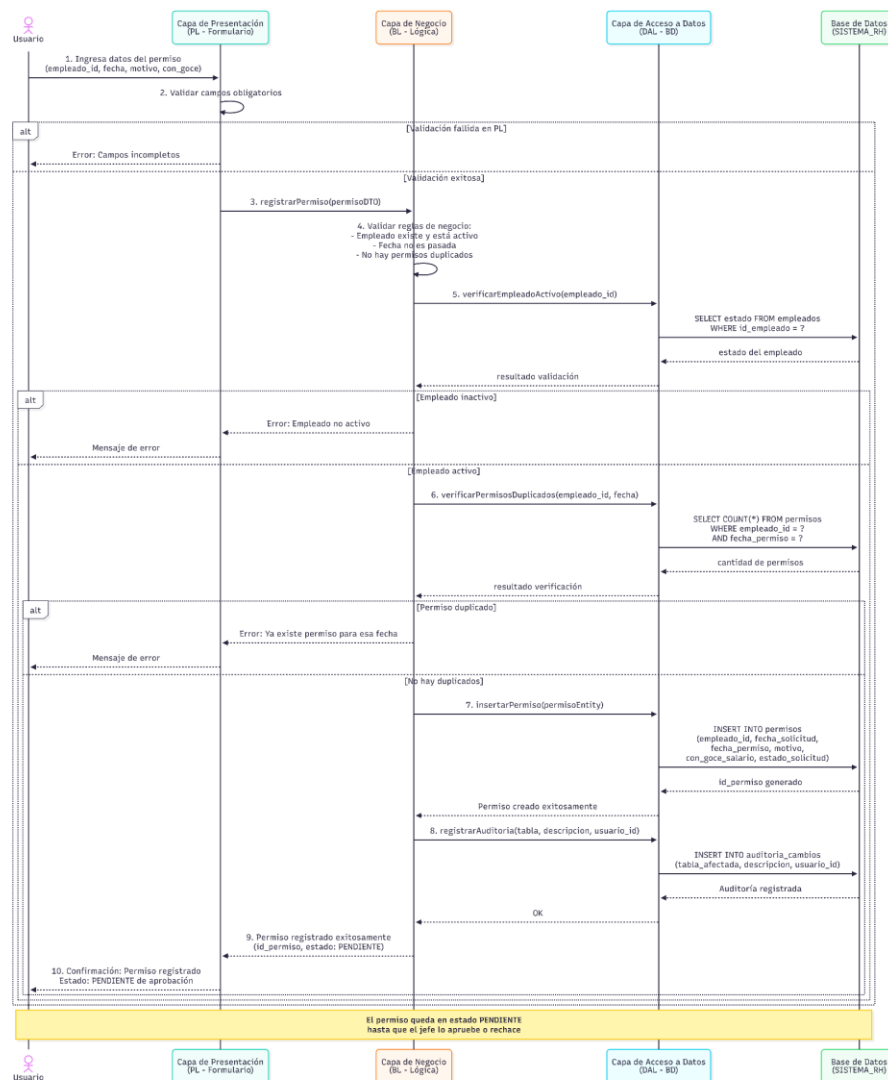
El diagrama de secuencia ilustra el proceso completo de registro de asistencia de un empleado en el sistema de recursos humanos. En este flujo, la persona usuaria (ya sea un empleado o supervisor) solicita un registro, indicando su identificación y tipo de marcaje (entrada o salida). La capa de presentación valida los datos y envía la solicitud a la capa de negocio, que verifica la existencia y el estado activo del empleado en la base de datos. Si es válido, el sistema consulta si ya existe un registro de entrada para ese día: si no existe, crea un nuevo registro con la hora de entrada; si existe, lo actualiza con la hora de salida y calcula las horas trabajadas. Finalmente, el sistema evalúa y asigna un estado (PRESENTE o TARDANZA), guarda la transacción y envía una confirmación al usuario, garantizando la integridad y trazabilidad del proceso.

Figura 24
Diagrama de secuencia para registrar horas extra



El diagrama de secuencia ilustra el proceso completo de solicitud de horas extra por parte de un empleado en el sistema. En este flujo, el empleado ingresa los datos de la petición (fecha, duración y motivo), el sistema valida los campos, verifica que la persona colaboradora esté activa y que no haya excedido el límite mensual de horas. Si todo es correcto, se crea un registro con estado *PENDIENTE*, se identifica al jefe inmediato para su futura aprobación, se registra una traza de auditoría y se envía al empleado una confirmación con el número de referencia. Finalmente, la solicitud queda a la espera del proceso de autorización, lo que garantiza el control y la trazabilidad del procedimiento.

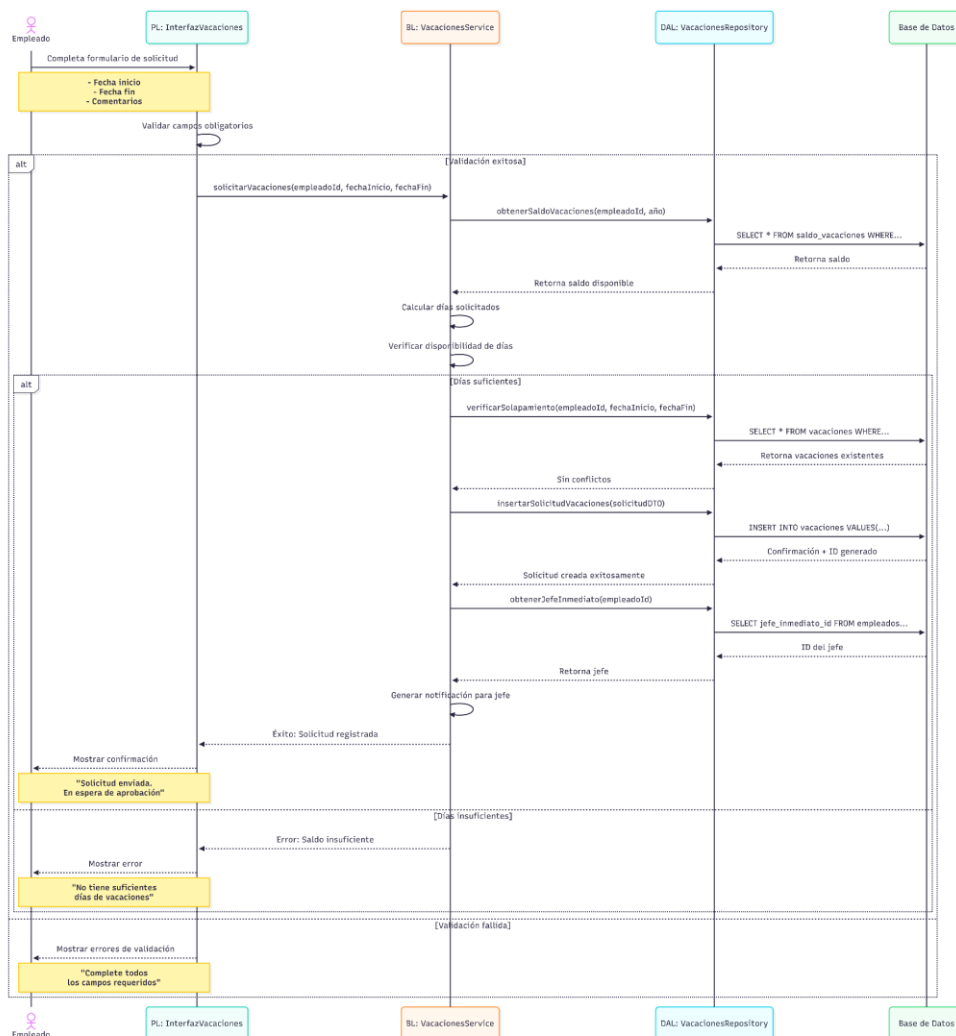
Figura 25
Diagrama de secuencia para registrar un permiso



El diagrama de secuencia ilustra el proceso completo de solicitud de un permiso por parte de un empleado en el sistema. En este flujo, la persona usuaria ingresa los datos del permiso (empleado, fecha, motivo y tipo de goce); la capa de presentación valida los campos obligatorios y la capa de negocio verifica que el empleado esté activo, que la fecha no sea pasada y que no exista un permiso duplicado para ese día. Si todas las validaciones resultan exitosas, el sistema inserta el registro en la base de datos con estado *PENDIENTE*, registra la acción en la auditoría para su trazabilidad y notifica al usuario sobre la creación exitosa de la petición. Finalmente, el permiso queda a la espera del proceso de aprobación o rechazo por parte del jefe inmediato, lo que garantiza el control y la integridad del procedimiento administrativo.

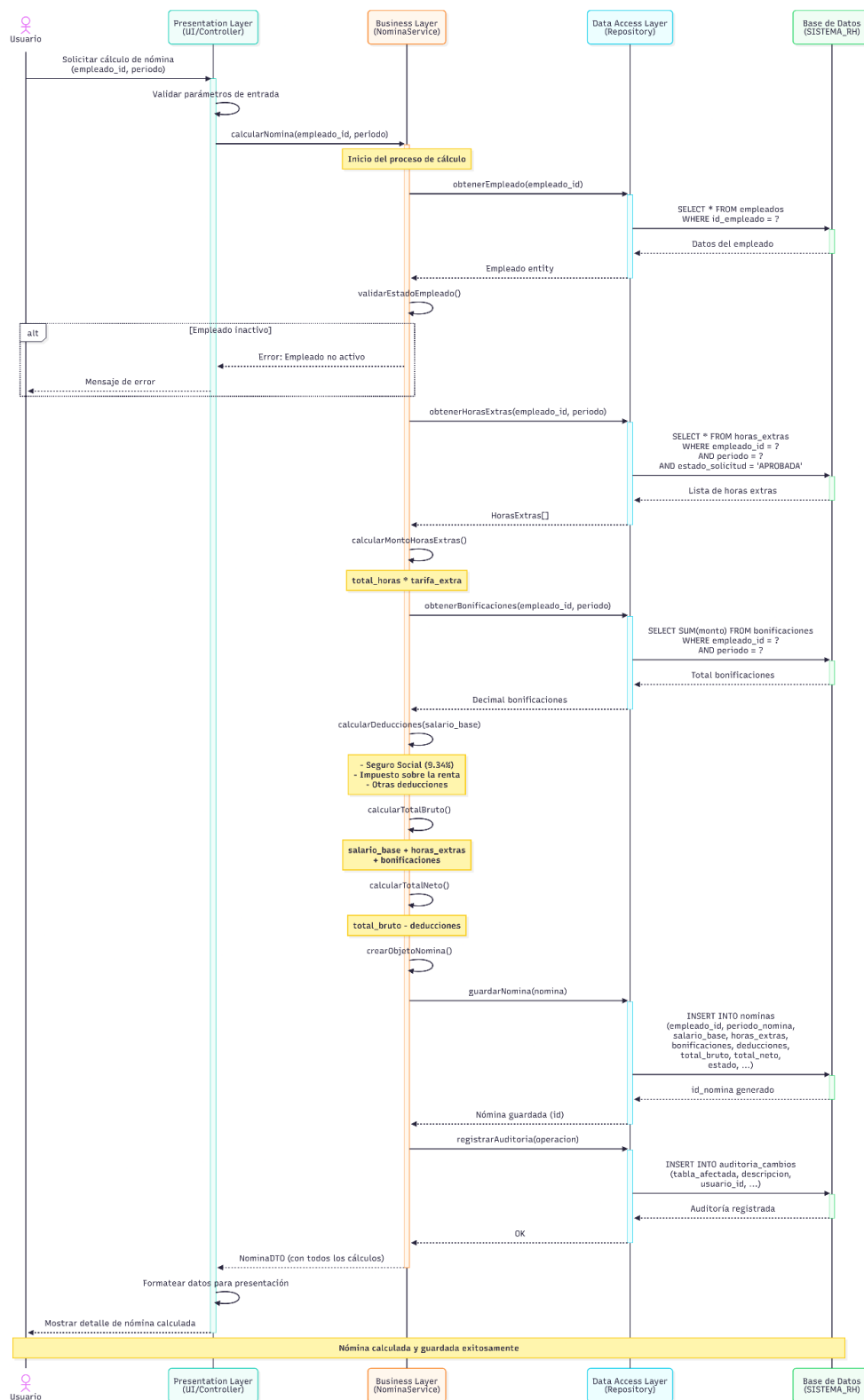
Figura 26

Diagrama de secuencia para registrar una solicitud de vacaciones



El diagrama de secuencia ilustra el proceso completo de solicitud de vacaciones por parte de un empleado en el sistema. En este flujo, el empleado completa y envía el formulario con las fechas de inicio y fin; el sistema valida los datos y consulta el saldo de días de vacaciones disponibles del colaborador para el año en curso. Si el saldo es suficiente, verifica que no existan solicitudes de vacaciones que se solapen en el mismo período, crea el registro en la base de datos con estado pendiente, identifica al jefe inmediato para su aprobación, genera una notificación y confirma al empleado el envío exitoso de la petición. Finalmente, la petición queda a la espera de la autorización del supervisor, lo que garantiza el control sobre los días disponibles y evita conflictos en el calendario laboral.

Figura 27
Diagrama de secuencia para el cálculo de nómina



El diagrama de secuencia ilustra el proceso completo de cálculo de nómina para un empleado en el sistema. En este flujo, la persona usuaria solicita el cálculo para un empleado y un período específicos; el sistema valida los parámetros, obtiene y verifica los datos del empleado y consulta en la base de datos las horas extra aprobadas y las bonificaciones correspondientes al período. Luego, realiza el cálculo de los montos de horas extra y de las deducciones obligatorias (seguro social e impuestos), suma los componentes para obtener el total bruto y resta las deducciones para determinar el total neto. Enseguida, crea y guarda el registro completo de la nómina en la base de datos, registra la operación en la auditoría para su trazabilidad; por último, presenta al usuario el detalle calculado con todos los elementos desglosados, garantizando la precisión y el cumplimiento de las normativas de pago.

Programación

A través de esta sección se presentan diferentes representaciones del código fuente que forma parte del aplicativo resultante como solución al problema descrito. Cada uno de los extractos de código adjuntos corresponde a distintas características que el sistema posee, tales como la captura de datos, la generación de reportes y otros aspectos, como la verificación de datos, parte de las validaciones necesarias y las estructuras que componen los módulos definidos en el alcance del presente proyecto.

Gracias a estas secciones, se evidencia la ejecución correcta de buenas prácticas de escritura y arquitectura de *software*. Gran parte de esto se debe a la implementación de la *suite* de .NET y EF Core, sin dejar de lado las tecnologías que acompañan el proceso para garantizar la entrega de un producto funcional, seguro, mantenible y escalable a largo plazo.

Entradas. Para empezar, se presentan diferentes contextos en los que el sistema realiza la captura de datos y el procesamiento en el ámbito de servidor, así como los formularios en pantalla que se utilizan para el ingreso de datos en los distintos módulos del aplicativo. Cada uno de los siguientes fragmentos de código muestra cómo se gestionan las entradas mediante formularios diseñados en React, el *framework* que se utiliza para el *front-end* y las validaciones aplicadas en el ámbito de servidor con C# y .NET.

Figura 28

Extracto de código que se utiliza para registrar una asistencia desde la capa de lógica de negocio

```

1  public async Task<MarcarAsistenciaResponse> MarcarAsistenciaAsync(int empleadoId)
2  {
3      var empleado = await _empleadosRepo.GetByIdAsync(empleadoId);
4      if (empleado == null)
5      {
6          throw new BusinessException("Empleado no encontrado", "EMPLEADO_NO_ENCONTRADO");
7      }
8
9      var hoy = DateTime.Today;
10     var ahora = DateTime.Now;
11     var registro = await _asistenciasRepo.GetByEmpleadoYFechaAsync(empleadoId, hoy);
12
13     if (registro == null)
14     {
15         var nuevoRegistro = new Asistencias
16         {
17             EmpleadoId = empleadoId,
18             FechaRegistro = hoy,
19             HoraEntrada = ahora,
20             Estado = DeterminarEstado(ahora, null),
21             FechaCreacion = DateTime.UtcNow
22         };
23
24         await _asistenciasRepo.CreateAsync(nuevoRegistro);
25
26         return new MarcarAsistenciaResponse
27         {
28             Accion = "ENTRADA",
29             Hora = ahora,
30             HoraEntrada = ahora,
31             Estado = nuevoRegistro.Estado,
32             Mensaje = "Entrada registrada correctamente",
33             Exito = true
34         };
35     }
36     else if (registro.HoraSalida == null)
37     {
38         registro.HoraSalida = ahora;
39         registro.FechaModificacion = DateTime.UtcNow;
40         await _asistenciasRepo.UpdateAsync(registro);
41
42         return new MarcarAsistenciaResponse
43         {
44             Accion = "SALIDA",
45             Hora = ahora,
46             HoraEntrada = registro.HoraEntrada,
47             HoraSalida = ahora,
48             Estado = registro.Estado,
49             Mensaje = "Salida registrada correctamente",
50             Exito = true
51         };
52     }
53     else
54     {
55         return new MarcarAsistenciaResponse
56         {
57             Accion = "NINGUNA",
58             Hora = ahora,
59             HoraEntrada = registro.HoraEntrada,
60             HoraSalida = registro.HoraSalida,
61             Estado = registro.Estado,
62             Mensaje = "Ya has registrado entrada y salida para hoy",
63             Exito = false
64         };
65     }
66 }

```

El código ilustrado en este fragmento hace referencia al método dentro de la capa lógica de negocio encargado de recibir los datos enviados desde el controlador del API. En él se gestionan las reglas de negocio y las validaciones sobre los datos recibidos. Lo anterior tiene el fin de identificar los posibles errores antes de escribir la información en la base de datos.

Figura 29

Extracto de código que se utiliza para generar un período de nómina

```

1 public async Task<List<DetalleNominaDTO>> GenerarNominaQuincenalAsync(GenerarNominaQuincenalDTO dto)
2 {
3     if (dto.Quincena != 1 && dto.Quincena != 2)
4         throw new ArgumentException("La quincena debe ser 1 o 2");
5
6     if (dto.Mes < 1 || dto.Mes > 12)
7         throw new ArgumentException("El mes debe estar entre 1 y 12");
8
9     List<Empleado> empleados;
10    if (dto.EmpleadosIds != null && dto.EmpleadosIds.Any())
11    {
12        empleados = new List<Empleado>();
13        foreach (var id in dto.EmpleadosIds)
14        {
15            var emp = await _empleadosRepo.GetByIdAsync(id);
16            if (emp != null && emp.Estado == "ACTIVO")
17                empleados.Add(emp);
18        }
19    }
20    else
21    {
22        var todosEmpleados = await _empleadosRepo.GetAllAsync();
23        empleados = todosEmpleados.Where(e => e.Estado == "ACTIVO").ToList();
24    }
25
26    var detalles = new List<DetalleNominaDTO>();
27
28    foreach (var empleado in empleados)
29    {
30        var existe = await _nominaRepo.ExisteNominaQuincenalAsync(
31            empleado.IdEmpleado, dto.Quincena, dto.Mes, dto.Año);
32
33        if (existe)
34            continue;
35
36        var horasExtrasEnumerable = await _horasExtraRepo.GetByEmpleadoAsync(empleado.IdEmpleado);
37        var horasExtras = horasExtrasEnumerable.ToList();
38
39        var incapacidades = await _incapacidadesRepo.ListarIncapacidadesAsync();
40        var permisos = await _permisosRepo.GetAllPermisosAsync();
41
42        var incapacidadesEmpleado = incapacidades
43            .Where(i => i.EmpleadoId == empleado.IdEmpleado).ToList();
44        var permisosEmpleado = permisos
45            .Where(p => p.EmpleadoId == empleado.IdEmpleado).ToList();
46
47        var detalle = _calculador.CalcularNominaQuincenal(
48            empleado,
49            dto.Quincena,
50            dto.Mes,
51            dto.Año,
52            horasExtras,
53            incapacidadesEmpleado,
54            permisosEmpleado
55        );
56
57        var nomina = new Nomina
58        {
59            EmpleadoId = empleado.IdEmpleado,
60            PeriodoNomina = new DateTime(dto.Año, dto.Mes, dto.Quincena == 1 ? 1 : 16),
61            FechaPago = dto.FechaPago,
62            SalarioBase = detalle.SalarioBaseQuincenal,
63            HorasExtras = detalle.TotalHorasExtra > 0
64                ? (decimal?)(detalle.HorasExtraDiurnas + detalle.HorasExtraNocturnas + detalle.HorasExtraFeriados) / (empleado.SalarioBase / 240)
65                : null,
66            MontoHorasExtra = detalle.TotalHorasExtra > 0 ? (decimal?)detalle.TotalHorasExtra : null,
67            Bonificaciones = detalle.Bonificaciones > 0 ? (decimal?)detalle.Bonificaciones : null,
68            Deducciones = detalle.TotalDeducciones,
69            TotalBruto = detalle.TotalBruto,
70            TotalNeto = detalle.TotalNeto,
71            Estado = "PENDIENTE"
72        };
73
74        await _nominaRepo.CrearNominaAsync(nomina);
75        detalles.Add(detalle);
76    }
77
78    return detalles;
79 }

```

Por medio del método de la Figura 29, se logra generar un nuevo período de nómina a partir de los datos entrantes del controlador, que se encuentran en la capa de presentación hasta la capa de negocio. Esta logra validar los datos y gestionar errores para asegurar el flujo correcto de información y la creación de datos con base en estos. Finalmente, envía los datos procesados a la capa de acceso a datos, donde se registran en la base de datos.

Figura 30*Extracto de código que se utiliza para actualizar la información de un empleado*

```

1 public async Task UpdateAsync(int id, ActualizarEmpleadoDTO dto)
2 {
3     if (dto == null)
4         throw new ArgumentNullException(nameof(dto));
5
6     if (id <= 0)
7         throw new ArgumentException("El ID debe ser mayor a 0.", nameof(id));
8
9     var empleado = await _repoEmpleados.GetByIdAsync(id);
10    if (empleado == null)
11        throw new KeyNotFoundException($"Empleado con ID {id} no encontrado.");
12
13    if (dto.CodigoEmpleado != null)
14        empleado.CodigoEmpleado = dto.CodigoEmpleado;
15
16    if (dto.Nombre != null)
17        empleado.Nombre = dto.Nombre;
18
19    if (dto.PrimerApellido != null)
20        empleado.PrimerApellido = dto.PrimerApellido;
21
22    if (dto.SegundoApellido != null)
23        empleado.SegundoApellido = dto.SegundoApellido;
24
25    if (dto.Email != null)
26        empleado.Email = dto.Email;
27
28    if (dto.Telefono != null)
29        empleado.Telefono = dto.Telefono;
30
31    if (dto.FechaContratacion.HasValue)
32        empleado.FechaContratacion = dto.FechaContratacion.Value;
33
34    if (dto.PuestoId.HasValue)
35        empleado.PuestoId = dto.PuestoId.Value;
36
37    if (dto.DepartamentoId.HasValue)
38        empleado.DepartamentoId = dto.DepartamentoId.Value;
39
40    if (dto.JefeInmediatoId.HasValue)
41        empleado.JefeInmediatoId = dto.JefeInmediatoId.Value;
42
43    if (dto.SalarioBase.HasValue)
44        empleado.SalarioBase = dto.SalarioBase.Value;
45
46    if (dto.TipoContrato.HasValue)
47        empleado.TipoContrato = dto.TipoContrato.Value.ToString();
48
49    if (dto.Estado.HasValue)
50        empleado.Estado = dto.Estado.Value.ToString();
51
52    empleado.FechaModificacion = DateTime.UtcNow;
53
54    await _repoEmpleados.UpdateAsync(empleado);
55
56    if (dto.RolId.HasValue && dto.RolId.Value > 0 && empleado.Usuarios != null)
57    {
58        var usuario = empleado.Usuarios;
59
60        usuario.UsuariosRoles.Clear();
61        usuario.UsuariosRoles.Add(new UsuariosRoles
62        {
63            RolId = dto.RolId.Value
64        });
65
66        usuario.FechaModificacion = DateTime.UtcNow;
67
68        await _repoUsuarios.UpdateAsync(usuario);
69    }
70 }

```

Este método brinda al aplicativo la función de actualizar los datos de un empleado o colaborador de la empresa, con la finalidad de realizar algún cambio en la información existente. Como se puede observar, se valida que exista un registro correspondiente al empleado cuya información se desea poner al día y se aplican las validaciones necesarias para procesar estos datos, de manera que la actualización de los datos existente se haga de forma exitosa y se refleje en la base de datos.

Salidas. A lo largo de este apartado, se documentan de forma gráfica extractos de código que se relacionan con generar y presentar información dentro del aplicativo programado. Cada una de las figuras muestra cómo se construyen las salidas de datos a partir del procesamiento interno, mediante vistas dinámicas y reportes que permiten al usuario consultar resultados, listas de datos y comprobantes generados por el sistema.

Figura 31

Extracto de código que se utiliza para la salida de datos respecto al historial de incapacidades



```

1  public async Task<IEnumerable<IncapacidadDto>> ListarIncapacidadesAsync()
2  {
3      var incapacidades = await _repoIncapacidades.ListarIncapacidadesAsync();
4
5      var listaDtos = new List<IncapacidadDto>();
6
7      foreach (var inc in incapacidades)
8      {
9          listaDtos.Add(new IncapacidadDto
10             {
11                 IdIncapacidad = inc.IdIncapacidad,
12                 EmpleadoId = inc.EmpleadoId,
13                 FechaInicio = inc.FechaInicio,
14                 FechaFin = inc.FechaFin,
15                 TipoIncapacidad = inc.TipoIncapacidad,
16                 Diagnostico = inc.Diagnostico,
17                 ArchivoAdjunto = inc.ArchivoAdjunto,
18                 Estado = inc.Estado ?? EstadoIncapacidad.ACTIVA.ToString(),
19                 FechaCreacion = inc.FechaCreacion ?? DateTime.UtcNow,
20                 FechaModificacion = inc.FechaModificacion
21             });
22      }
23
24      return listaDtos;
25  }

```

En el extracto de código anterior, se muestra cómo el sistema consulta el repositorio referente a la entidad de incapacidades, transforma los datos en un DTO y mapea los valores relevantes que

se deben mostrar para enviarse al controlador del API, devolviendo una lista estructurada de las incapacidades existentes en la base de datos.

Figura 32

Extracto de código que se utiliza para la salida de las listas de nóminas

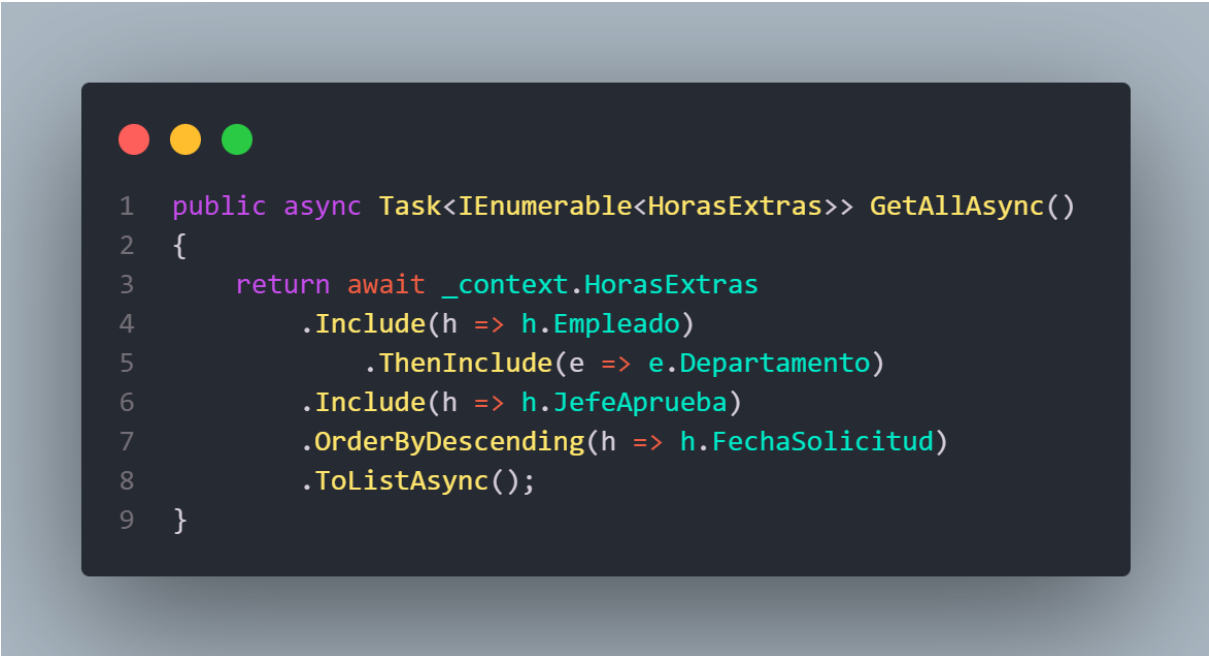
A screenshot of a code editor with a dark background and light-colored text. The code is a C# method named `ListarNominasAsync()` which is asynchronous and returns a `Task<List<Nominas>>`. The method body starts with `return await _context.Nominas` followed by a chain of LINQ queries: `.Include(n => n.Empleado)`, `.ThenInclude(e => e.Puesto)`, `.Include(n => n.Empleado)`, `.ThenInclude(e => e.Departamento)`, `.OrderByDescending(n => n.PeriodoNomina)`, and finally `.ToListAsync();`. The code is numbered from 1 to 10 on the left side of the editor.

```
1 public async Task<List<Nominas>> ListarNominasAsync()  
2 {  
3     return await _context.Nominas  
4         .Include(n => n.Empleado)  
5         .ThenInclude(e => e.Puesto)  
6         .Include(n => n.Empleado)  
7         .ThenInclude(e => e.Departamento)  
8         .OrderByDescending(n => n.PeriodoNomina)  
9         .ToListAsync();  
10 }
```

Este método asincrónico se encarga de tomar los valores correspondientes a la nómina desde la base de datos, lo que incluye la información relacionada con el empleado, como su puesto y el departamento en el que trabaja. Por otra parte, organiza cada uno de los registros de forma descendente según el período de nómina y devuelve una lista de los registros de nómina.

Figura 33

Extracto de código que se utiliza para la salida de horas extra de la semana



```
1 public async Task<IEnumerable<HorasExtras>> GetAllAsync()
2 {
3     return await _context.HorasExtras
4         .Include(h => h.Empleado)
5         .ThenInclude(e => e.Departamento)
6         .Include(h => h.JefeAprueba)
7         .OrderByDescending(h => h.FechaSolicitud)
8         .ToListAsync();
9 }
```

Este método asincrónico se encarga de obtener los valores correspondientes a las horas extra desde la base de datos, lo que incluye la información relacionada con el empleado, como el departamento en el que trabaja, así como los datos del colaborador encargado de aprobar la o las horas extra, con la finalidad de mostrar una lista estructurada de los registros o el historial de horas extra.

Procesos. En este apartado se pueden apreciar ejemplos de código que presentan la función lógica del sistema para realizar cálculos, aplicar validaciones y ejecutar operaciones cruciales para las características que constituyen el pilar en la orientación del aplicativo como solución a un problema empresarial. Cada uno de estos ejemplos permite que el sistema cumpla con el funcionamiento eficiente y correcto de los módulos que componen este proyecto y con la gestión efectiva de la información.

Figura 34

Extracto de código que se utiliza para el proceso de cálculo del impuesto sobre la renta

```

1  private ImpuestoRentaDTO CalcularImpuestoRenta(decimal baseImponibleQuincenal)
2  {
3      var proyeccionMensual = baseImponibleQuincenal * 2;
4      var tramo = _tramosImpuesto.FirstOrDefault(t =>
5          proyeccionMensual >= t.Desde && proyeccionMensual <= t.Hasta
6      ) ?? _tramosImpuesto.Last();
7
8      decimal impuestoMensual = 0;
9      if (tramo.Tarifa > 0)
10     {
11         var excedente = proyeccionMensual - tramo.Desde + 1;
12         impuestoMensual = tramo.ImpuestoBase + (excedente * tramo.Tarifa);
13     }
14
15     var impuestoQuincenal = Math.Round(impuestoMensual / 2, 2);
16
17     return new ImpuestoRentaDTO
18     {
19         BaseImponible = baseImponibleQuincenal,
20         ProyeccionMensual = proyeccionMensual,
21         ImpuestoMensual = Math.Round(impuestoMensual, 2),
22         ImpuestoQuincenal = impuestoQuincenal,
23         TramoAplicado = $"€{tramo.Desde:N0} - €{tramo.Hasta:N0} ({tramo.Tarifa * 100}%)"
24     };
25 }

```

Este método tiene como objetivo calcular el impuesto de renta sobre una base quincenal de nómina. Primero, proyecta el ingreso mensual y determina el impuesto con base en los datos correspondientes a la ley. Si el tramo aplica una tarifa, calcula el excedente sobre el límite inferior y suma el impuesto base junto con el porcentaje correspondiente. Luego, obtiene el monto quincenal al dividir el impuesto mensual entre dos y redondear los valores a dos decimales. Finalmente, retorna un objeto DTO con el detalle del cálculo, que incluye la base que se utiliza, la proyección mensual, el impuesto mensual y quincenal, así como el tramo aplicado.

Figura 35

Extracto de código que se utiliza para el proceso de cálculo de las horas extra respecto a la nómina

```

1 private (decimal total, decimal diurnas, decimal nocturnas, decimal feriados) CalcularHorasExtra(
2     decimal salarioMensual, List<HorasExtras>? horasExtras, int quincena, int mes, int anio)
3 {
4     if (horasExtras == null || !horasExtras.Any())
5         return (0, 0, 0, 0);
6
7     var salarioPorHora = salarioMensual / HORAS_MENSUALES;
8     var fechaInicio = new DateTime(anio, mes, quincena == 1 ? 1 : 16);
9     var fechaFin = quincena == 1
10         ? new DateTime(anio, mes, 15)
11         : new DateTime(anio, mes, DateTime.DaysInMonth(anio, mes));
12
13     var horasQuincena = horasExtras.Where(he =>
14         he.EstadoSolicitud == "APROBADA" &&
15         he.FechaInicio.Date >= fechaInicio &&
16         he.FechaFin.Date <= fechaFin
17     ).ToList();
18
19     decimal montoDiurnas = 0, montoNocturnas = 0, montoFeriados = 0;
20
21     foreach (var he in horasQuincena)
22     {
23         var horas = (decimal)(he.FechaFin - he.FechaInicio).TotalHours;
24         switch (he.TipoHoraExtra?.ToUpper())
25         {
26             case "DIURNA":
27                 montoDiurnas += horas * salarioPorHora * RECARGO_HORA_EXTRA_NORMAL;
28                 break;
29
30             case "NOCTURNA":
31                 montoNocturnas += horas * salarioPorHora * RECARGO_HORA_EXTRA_NORMAL;
32                 break;
33
34             case "FERIADO":
35             case "DOMINGO":
36                 montoFeriados += horas * salarioPorHora * RECARGO_HORA_EXTRA_FERIADO;
37                 break;
38
39             default:
40                 montoDiurnas += horas * salarioPorHora * RECARGO_HORA_EXTRA_NORMAL;
41                 break;
42         }
43     }
44
45     return (
46         Math.Round(montoDiurnas + montoNocturnas + montoFeriados, 2),
47         Math.Round(montoDiurnas, 2),
48         Math.Round(montoNocturnas, 2),
49         Math.Round(montoFeriados, 2)
50     );
51 }

```

Este método calcula el pago de horas extra para una quincena específica, valida primero si existen registros y determina el rango de fechas correspondiente. Además, filtra únicamente las solicitudes aprobadas en el período, calcula el valor por hora a partir del salario mensual y aplica el recargo según el tipo de hora extra (diurna, nocturna, feriado o domingo). Por último, retorna el total acumulado y el desglose por tipo, redondeados a dos decimales para su integración en la nómina.

Figura 36

Extracto de código que se utiliza para el proceso de cálculo del ajuste salarial por ausencias

```

1 private AjustesAusenciasDTO CalcularAjustesAusencias(
2     decimal salarioMensual, int quincena, int mes, int anio,
3     List<Incapacidades>? incapacidades, List<Permisos>? permisos)
4 {
5     var ajustes = new AjustesAusenciasDTO();
6     var salarioDiario = salarioMensual / 30;
7
8     var fechaInicio = new DateTime(anio, mes, quincena == 1 ? 1 : 16);
9     var fechaFin = quincena == 1
10         ? new DateTime(anio, mes, 15)
11         : new DateTime(anio, mes, DateTime.DaysInMonth(anio, mes));
12
13     if (incapacidades != null)
14     {
15         foreach (var inc in incapacidades.Where(i => i.Estado == "ACTIVA"))
16         {
17             var inicio = inc.FechaInicio > fechaInicio ? inc.FechaInicio : fechaInicio;
18             var fin = inc.FechaFin < fechaFin ? inc.FechaFin : fechaFin;
19
20             if (inicio <= fechaFin && fin >= fechaInicio)
21             {
22                 var dias = (fin - inicio).Days + 1;
23                 ajustes.DiasIncapacidad += dias;
24                 ajustes.MontoIncapacidad += dias * salarioDiario * 0.5m;
25             }
26         }
27     }
28
29     if (permisos != null)
30     {
31         foreach (var per in permisos.Where(p =>
32             p.EstadoSolicitud == "APROBADA" && p.ConGoceSalario == false))
33         {
34             if (per.FechaPermiso >= fechaInicio && per.FechaPermiso <= fechaFin)
35             {
36                 ajustes.DiasPermisoSinGoce++;
37                 ajustes.MontoPermisoSinGoce += salarioDiario;
38             }
39         }
40     }
41
42     ajustes.TotalAjustes = ajustes.MontoIncapacidad + ajustes.MontoPermisoSinGoce;
43     return ajustes;
44 }

```

Este método calcula los ajustes salariales por ausencias dentro de una quincena específica, lo que determina primero el rango de fechas y el salario diario con base en el salario mensual. Luego, evalúa las incapacidades activas que coinciden con el período, calcula los días aplicables y el monto correspondiente al 50 % del salario diario. Después, procesa los permisos aprobados sin goce de salario dentro del rango, acumula los días y el descuento total. Finalmente, consolida ambos conceptos en el total de ajustes y retorna un objeto con el desglose de días y montos aplicados.

Figura 37

Extracto de código que se utiliza para el proceso de construcción de un token JWT como medio de autenticación

```

1  private AuthResponseDTO GenerateJwt(User user)
2  {
3      var jwtSettings = _config.GetSection("Jwt");
4
5      var claims = new List<Claim>()
6      {
7          new Claim(ClaimTypes.NameIdentifier, user.IdUsuario.ToString()),
8          new Claim(ClaimTypes.Name, user.NombreUsuario),
9          new Claim("UserId", user.IdUsuario.ToString()),
10         new Claim("Username", user.NombreUsuario)
11     };
12
13     UsuarioInfoDTO usuarioInfo = null;
14     if (user.Empleado != null)
15     {
16         var nombreCompleto = $"{user.Empleado.Nombre} {user.Empleado.PrimerApellido} {user.Empleado.SegundoApellido}.Trim();
17
18         claims.Add(new Claim("EmployeeId", user.Empleado.IdEmpleado.ToString()));
19         claims.Add(new Claim("EmployeeCode", user.Empleado.CodigoEmpleado.ToString()));
20         claims.Add(new Claim("Fullname", nombreCompleto));
21         claims.Add(new Claim("Email", user.Empleado.Email));
22         claims.Add(new Claim("Department", user.Empleado.DependenciaId.ToString()));
23         claims.Add(new Claim("PositionId", user.Empleado.PuestoId.ToString()));
24
25         if (user.Empleado.JefeInmediatoId.HasValue)
26         {
27             claims.Add(new Claim("ManagerId", user.Empleado.JefeInmediatoId.Value.ToString()));
28         }
29
30         usuarioInfo = new UsuarioInfoDTO
31         {
32             IdUsuario = user.IdUsuario,
33             NombreUsuario = user.NombreUsuario,
34             NombreCompleto = nombreCompleto,
35             Email = user.Empleado.Email,
36             EmpleadoId = user.Empleado.IdEmpleado,
37             CodigoEmpleado = user.Empleado.CodigoEmpleado,
38             Departamento = user.Empleado.DependenciaId,
39             NombreDepartamento = user.Empleado.DependenciaId,
40             PuestoId = user.Empleado.PuestoId,
41             NombrePuesto = user.Empleado.PuestoId,
42         };
43     }
44
45     var rolesList = new List<string>();
46
47     if (user.UsuarioRoles != null && user.UsuarioRoles.Any())
48     {
49         foreach (var usuarioRol in user.UsuarioRoles)
50         {
51             if (usuarioRol.Rol != null && !string.IsNullOrEmpty(usuarioRol.Rol.Nombre))
52             {
53                 claims.Add(new Claim(ClaimTypes.Role, usuarioRol.Rol.Nombre));
54                 claims.Add(new Claim("RoleId", usuarioRol.RolId.ToString()));
55                 rolesList.Add(usuarioRol.Rol.Nombre);
56             }
57         }
58     }
59
60     if (rolesList.Any())
61     {
62         claims.Add(new Claim("Roles", string.Join(",", rolesList)));
63     }
64
65     if (usuarioInfo != null)
66     {
67         usuarioInfo.Roles = rolesList;
68     }
69
70     var key = new SymmetricSecurityKey(
71         Encoding.UTF8.GetBytes(jwtSettings["Key"]));
72
73     var creds = new SigningCredentials(
74         key,
75         SecurityAlgorithms.HmacSha256);
76
77     var expiration = DateTime.UtcNow.AddMinutes(
78         double.Parse(jwtSettings["ExpirationMinutes"]));
79
80     var token = new JwtSecurityToken(
81         issuer: jwtSettings["Issuer"],
82         audience: jwtSettings["Audience"],
83         claims: claims,
84         expires: expiration,
85         signingCredentials: creds);
86
87     return new AuthResponseDTO
88     {
89         Token = new JwtSecurityTokenHandler().WriteToken(token),
90         Expiration = expiration,
91         UsuarioInfo = usuarioInfo
92     };
93 }

```

Este método genera un *token* JWT para un usuario autenticado, obtiene la configuración desde el archivo de ajustes y construye una lista de *claims* con la información básica del usuario. Si el usuario está vinculado a un empleado, agrega datos adicionales como identificador, código, nombre completo, correo, departamento, puesto y jefe inmediato, además de construir un objeto con la información detallada del usuario. Asimismo, incorpora los roles asignados como *claims* individuales y consolidados. Posteriormente, crea la clave de seguridad y las credenciales de firma, define la expiración del *token* y construye el JWT firmado. Por último, retorna un objeto de respuesta que incluye el *token* generado, su fecha de expiración y la información del usuario autenticado.

Validaciones. En esta sección se presentan fragmentos de código orientados a asegurar la integridad de la información y el funcionamiento adecuado del sistema. Estas validaciones permiten identificar y evitar inconsistencias en los datos ingresados o procesados y garantizan que los cálculos, registros y operaciones se realicen conforme a las reglas de negocio establecidas.

Figura 38
Validaciones en el ámbito de DTO

```

1  public class CrearEmpleadoYUsuarioDTO
2  {
3      [Required]
4      [StringLength(20, MinimumLength = 3)]
5      public string CodigoEmpleado { get; set; } = null!;
6
7      [Required]
8      [StringLength(50)]
9      public string Nombre { get; set; } = null!;
10
11     [Required]
12     [StringLength(50)]
13     public string PrimerApellido { get; set; } = null!;
14
15     [StringLength(50)]
16     public string? SegundoApellido { get; set; }
17
18     [Required]
19     [EmailAddress]
20     [StringLength(150)]
21     public string Email { get; set; } = null!;
22
23     [Phone]
24     [StringLength(8)]
25     public string? Telefono { get; set; }
26
27     [Required]
28     [DataType(DataType.Date)]
29     public DateOnly FechaContratacion { get; set; }
30
31     [Required]
32     [Range(1, int.MaxValue, ErrorMessage = "PuestoId debe ser mayor a 0.")]
33     public int PuestoId { get; set; }
34
35     [Required]
36     [Range(1, int.MaxValue, ErrorMessage = "DepartamentoId debe ser mayor a 0.")]
37     public int DepartamentoId { get; set; }
38
39     [Range(1, int.MaxValue, ErrorMessage = "JefeInmediatoId debe ser mayor a 0.")]
40     public int? JefeInmediatoId { get; set; }
41
42     [Range(0, double.MaxValue, ErrorMessage = "El salario no puede ser negativo.")]
43     public decimal SalarioBase { get; set; }
44
45     [Required]
46     [EnumDataType(typeof(TipoContrato))]
47     public TipoContrato TipoContrato { get; set; }
48
49     [Required]
50     public string NombreUsuario { get; set; } = null!;
51
52     [Required]
53     public string Password { get; set; } = null!;
54
55     [Required]
56     public int RolId { get; set; }
57 }

```

Este DTO incorpora un conjunto de validaciones orientadas a garantizar la correcta creación de un empleado y su usuario en el sistema. Se establecen campos obligatorios para datos esenciales como código, nombre, apellidos, correo electrónico, fecha de contratación, puesto, departamento, tipo de contrato, nombre de usuario, contraseña y rol. Además, se controlan las longitudes mínimas y máximas en las cadenas de texto, se valida el formato del correo electrónico y del teléfono y se restringen los valores numéricos mediante rangos que evitan identificadores inválidos o salarios negativos. Asimismo, se asegura que la clase de contrato corresponda a un valor definido en el enumerador. En conjunto, estas validaciones previenen inconsistencias, datos incompletos o formatos incorrectos y aseguran que la información cumpla con las reglas de negocio antes de ser procesada o almacenada.

Figura 39

Validaciones a nivel de controlador de la API



```

1  [HttpGet("{id}")]
2  public async Task<IActionResult> ObtenerPorId(int id)
3  {
4      try
5      {
6          var aguinaldo = await _aguinaldoManager.ObtenerPorIdAsync(id);
7
8          if (aguinaldo == null)
9              return NotFound(new { mensaje = $"Aguinaldo {id} no encontrado" });
10
11         return Ok(new
12             {
13                 mensaje = "Aguinaldo encontrado",
14                 datos = aguinaldo
15             });
16     }
17     catch (Exception ex)
18     {
19         return StatusCode(500, new { mensaje = "Error al obtener aguinaldo", error = ex.Message });
20     }
21 }

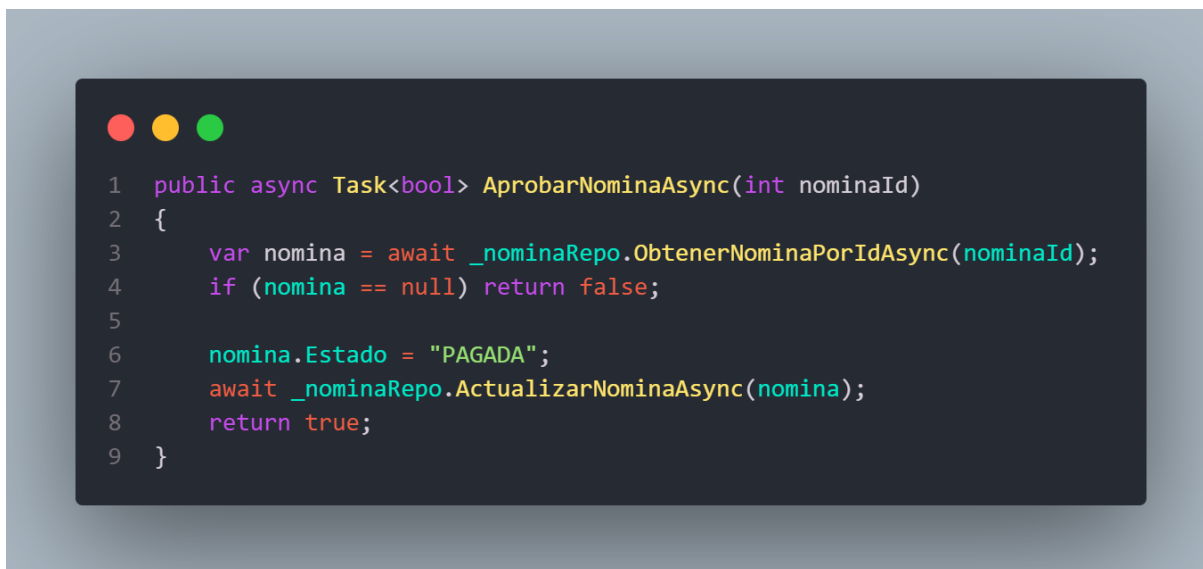
```

Este método implementa validaciones para obtener un registro de aguinaldo por su identificador. Primero, verifica si el registro existe y retorna un mensaje claro de *no encontrado* cuando el ID no corresponde a ningún registro. Además, maneja posibles excepciones durante la operación y devuelve un estado de error con información sobre el problema en caso de fallos.

inesperados. De esta manera, garantiza que la respuesta sea consistente, informativa y segura frente a datos inexistentes o errores en el proceso.

Figura 40

Validaciones a nivel de capa de negocio

A screenshot of a code editor with a dark background and light-colored text. The code is a C# method named `AprobarNominaAsync` that takes an `int nominaId` as a parameter. It is an asynchronous method returning a `Task<bool>`. The method body is enclosed in curly braces. Inside, it first declares a `nomina` variable by awaiting `_nominaRepo.ObtenerNominaPorIdAsync(nominaId)`. It then checks if `nomina` is null; if so, it returns `false`. If not null, it sets `nomina.Estado` to `"PAGADA"` and awaits `_nominaRepo.ActualizarNominaAsync(nomina)` before returning `true`. The code is numbered from 1 to 9 on the left side of the editor.

```
1 public async Task<bool> AprobarNominaAsync(int nominaId)
2 {
3     var nomina = await _nominaRepo.ObtenerNominaPorIdAsync(nominaId);
4     if (nomina == null) return false;
5
6     nomina.Estado = "PAGADA";
7     await _nominaRepo.ActualizarNominaAsync(nomina);
8     return true;
9 }
```

Esta función valida, en el ámbito de negocio, la aprobación de una nómina. Primero verifica si la nómina existe y devuelve *false* si el identificador no corresponde a ningún registro. Si el registro es válido, actualiza su estado a *PAGADA* y guarda el cambio en el repositorio. Esta validación asegura que solo las nóminas existentes puedan ser aprobadas, evita inconsistencias en el sistema y garantiza la integridad de los estados de pago.

Módulos señalados en el alcance. En esta sección se describen los módulos principales que se implementan en el sistema, los cuales se establecieron en el alcance del proyecto. Cada módulo cumple una tarea específica en la gestión de recursos humanos y contribuye al cumplimiento de los objetivos definidos.

Figura 41
Código para marcar asistencia

```

1 public async Task<MarcarAsistenciaResponse> MarcarAsistenciaAsync(int empleadoId)
2 {
3     var empleado = await _empleadosRepo.GetByIdAsync(empleadoId);
4     if (empleado == null)
5     {
6         throw new BusinessException("Empleado no encontrado", "EMPLEADO_NO_ENCONTRADO");
7     }
8
9     var hoy = DateTime.Today;
10    var ahora = DateTime.Now;
11    var registro = await _asistenciasRepo.GetByEmpleadoYFechaAsync(empleadoId, hoy);
12
13    if (registro == null)
14    {
15        var nuevoRegistro = new Asistencias
16        {
17            EmpleadoId = empleadoId,
18            FechaRegistro = hoy,
19            HoraEntrada = ahora,
20            Estado = DeterminarEstado(ahora, null),
21            FechaCreacion = DateTime.UtcNow
22        };
23
24        await _asistenciasRepo.CreateAsync(nuevoRegistro);
25
26        return new MarcarAsistenciaResponse
27        {
28            Accion = "ENTRADA",
29            Hora = ahora,
30            HoraEntrada = ahora,
31            Estado = nuevoRegistro.Estado,
32            Mensaje = "Entrada registrada correctamente",
33            Exito = true
34        };
35    }
36    else if (registro.HoraSalida == null)
37    {
38        registro.HoraSalida = ahora;
39        registro.FechaModificacion = DateTime.UtcNow;
40        await _asistenciasRepo.UpdateAsync(registro);
41
42        return new MarcarAsistenciaResponse
43        {
44            Accion = "SALIDA",
45            Hora = ahora,
46            HoraEntrada = registro.HoraEntrada,
47            HoraSalida = ahora,
48            Estado = registro.Estado,
49            Mensaje = "Salida registrada correctamente",
50            Exito = true
51        };
52    }
53    else
54    {
55        return new MarcarAsistenciaResponse
56        {
57            Accion = "NINGUNA",
58            Hora = ahora,
59            HoraEntrada = registro.HoraEntrada,
60            HoraSalida = registro.HoraSalida,
61            Estado = registro.Estado,
62            Mensaje = "Ya has registrado entrada y salida para hoy",
63            Exito = false
64        };
65    }
66 }

```

Este método pertenece al módulo de gestión de asistencias y se encarga de registrar la entrada y la salida diaria de los empleados en el sistema. Primero, valida que el empleado exista y luego verifica si ya tiene un registro para el día actual. Según el caso, crea un nuevo registro de entrada, actualiza la hora de salida o notifica que ambas acciones ya se registraron. Toda la información se almacena y actualiza en la base de datos, lo que asegura el control correcto de la asistencia diaria.

Figura 42

Código para registrar una horas extra

```

1  [HttpPost]
2  public async Task<ActionResult<HoraExtraDTO>> Create([FromBody] CrearHoraExtraDTO dto)
3  {
4      try
5      {
6          _logger.LogInformation("📩 Recibiendo solicitud: {@DTO}", dto);
7
8          if (!ModelState.IsValid)
9          {
10             var errors = ModelState
11                 .Where(x => x.Value?.Errors.Count > 0)
12                 .ToDictionary(
13                     kvp => kvp.Key,
14                     kvp => kvp.Value?.Errors.Select(e => e.ErrorMessage).ToArray()
15                 );
16
17             _logger.LogWarning("❌ ModelState inválido: {@Errors}", errors);
18
19             return BadRequest(new
20             {
21                 message = "Datos de entrada inválidos",
22                 errors = errors
23             });
24         }
25
26         var horaExtra = await _horasExtrasManager.CreateAsync(dto);
27
28         _logger.LogInformation("✅ Hora extra creada: {Id}", horaExtra.IdHoraExtra);
29
30         return CreatedAtAction(
31             nameof(GetById),
32             new { id = horaExtra.IdHoraExtra },
33             horaExtra);
34     }
35     catch (BusinessException ex)
36     {
37         _logger.LogWarning(ex, "⚠️ Error de negocio al crear hora extra");
38         return BadRequest(new { message = ex.Message, code = ex.Code });
39     }
40     catch (Exception ex)
41     {
42         _logger.LogError(ex, "❗ Error inesperado al crear hora extra");
43         return StatusCode(500, new
44         {
45             message = "Error interno del servidor al crear hora extra",
46             detail = ex.Message
47         });
48     }
49 }

```

Este método pertenece al módulo de gestión de horas extra y se encarga de recibir y procesar solicitudes de creación desde la capa de presentación. Primero, valida que los datos de entrada

cumplan con las reglas definidas en el modelo y devuelve errores claros en caso de inconsistencias. Luego, invoca la lógica de negocio para crear la hora extra y, si el proceso resulta exitoso, retorna el registro creado con su identificador. Además, maneja de manera diferenciada los errores de negocio y las excepciones inesperadas, registrando toda la información relevante en los *logs* para garantizar la trazabilidad y el control de las validaciones.

Figura 43
Código para registrar un permiso

```

1  public async Task<CrearPermisoDto> CrearPermisoAsync(CrearPermisoDto dto)
2  {
3      var permiso = new Permisos
4      {
5          ConGoceSalario = dto.ConGoceSalario,
6          EmpleadoId = dto.EmpleadoId,
7          EstadoSolicitud = EstadoSolicitud.PENDIENTE.ToString(),
8          FechaPermiso = dto.FechaPermiso,
9          FechaSolicitud = DateTime.Now,
10         Motivo = dto.Motivo,
11         FechaCreacion = DateTime.Now
12     };
13
14     await _repo.CreatePermisoAsync(permiso);
15
16     var registroCreado = await _repo.GetPermisoByIdAsync(permiso.IdPermiso);
17
18     if (registroCreado == null)
19         throw new BusinessException("Error al crear el permiso", "PERMISO_NO_CREADO");
20
21     var detalles = $"
22     <p><strong>Fecha del Permiso:</strong> {dto.FechaPermiso:dd/MM/yyyy}</p>
23     <p><strong>Motivo:</strong> {dto.Motivo}</p>
24     <p><strong>Con Goce de Salario:</strong> {(dto.ConGoceSalario ?? false ? "Sí" : "No")}</p>
25     <p><strong>Estado:</strong> PENDIENTE</p>
26     ";
27
28     await _notificacionesManager.NotificarSolicitudCreadaAsync(
29         dto.EmpleadoId,
30         "Permiso",
31         detalles
32     );
33
34     return new CrearPermisoDto
35     {
36         ConGoceSalario = registroCreado.ConGoceSalario,
37         EmpleadoId = registroCreado.EmpleadoId,
38         FechaPermiso = registroCreado.FechaPermiso,
39         Motivo = registroCreado.Motivo,
40     };
41 }

```

Este método pertenece al módulo de gestión de permisos y se encarga de validar y registrar nuevas solicitudes en el sistema. De esta manera, crea un registro de permiso con la información proporcionada, asigna automáticamente el estado *PENDIENTE* y las fechas de solicitud y creación. Después de almacenar el permiso en la base de datos, verifica que se haya creado de forma correcta y genera una notificación detallada para el empleado correspondiente. Por lo tanto, asegura que los permisos se registren de forma consistente y que los interesados sean informados oportunamente.

Figura 44
Código para actualizar los datos de un usuario empleado

```

1 public async Task UpdateAsync(int id, ActualizarEmpleadoDTO dto)
2 {
3     if (dto == null)
4         throw new ArgumentNullException(nameof(dto));
5
6     if (id <= 0)
7         throw new ArgumentException("El ID debe ser mayor a 0.", nameof(id));
8
9     var empleado = await _repoEmpleados.GetByIdAsync(id);
10    if (empleado == null)
11        throw new KeyNotFoundException($"Empleado con ID {id} no encontrado.");
12
13    if (dto.CodigoEmpleado != null)
14        empleado.CodigoEmpleado = dto.CodigoEmpleado;
15
16    if (dto.Nombre != null)
17        empleado.Nombre = dto.Nombre;
18
19    if (dto.PrimerApellido != null)
20        empleado.PrimerApellido = dto.PrimerApellido;
21
22    if (dto.SegundoApellido != null)
23        empleado.SegundoApellido = dto.SegundoApellido;
24
25    if (dto.Email != null)
26        empleado.Email = dto.Email;
27
28    if (dto.Telefono != null)
29        empleado.Telefono = dto.Telefono;
30
31    if (dto.FechaContratacion.HasValue)
32        empleado.FechaContratacion = dto.FechaContratacion.Value;
33
34    if (dto.PuestoId.HasValue)
35        empleado.PuestoId = dto.PuestoId.Value;
36
37    if (dto.DepartamentoId.HasValue)
38        empleado.DepartamentoId = dto.DepartamentoId.Value;
39
40    if (dto.JefeInmediatoId.HasValue)
41        empleado.JefeInmediatoId = dto.JefeInmediatoId.Value;
42
43    if (dto.SalarioBase.HasValue)
44        empleado.SalarioBase = dto.SalarioBase.Value;
45
46    if (dto.TipoContrato.HasValue)
47        empleado.TipoContrato = dto.TipoContrato.Value.ToString();
48
49    if (dto.Estado.HasValue)
50        empleado.Estado = dto.Estado.Value.ToString();
51
52    empleado.FechaModificacion = DateTime.UtcNow;
53
54    await _repoEmpleados.UpdateAsync(empleado);
55
56    if (dto.RolId.HasValue && dto.RolId.Value > 0 && empleado.Usuarios != null)
57    {
58        var usuario = empleado.Usuarios;
59
60        usuario.UsuariosRoles.Clear();
61        usuario.UsuariosRoles.Add(new UsuariosRoles
62        {
63            RolId = dto.RolId.Value
64        });
65
66        usuario.FechaModificacion = DateTime.UtcNow;
67
68        await _repoUsuarios.UpdateAsync(usuario);
69    }
70 }

```

Este método pertenece al módulo de gestión de empleados y se encarga de validar y actualizar la información de un empleado existente en el sistema. Primero, verifica que el DTO y el identificador sean válidos y que el empleado exista; de lo contrario, lanza excepciones claras. Luego, actualiza únicamente los campos proporcionados en el DTO, incluidos los datos personales, el puesto, el departamento, el salario, el tipo de contrato y el estado y registra la fecha de modificación. Además, si se especifica un nuevo rol, actualiza los roles asociados al usuario vinculado al empleado. Esta validación asegura que solo se modifiquen registros existentes y que los cambios cumplan con las reglas de negocio, manteniendo la integridad de la información.

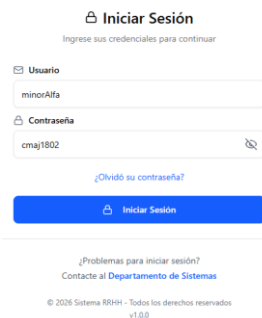
Pruebas del sistema

En esta sección se detallan los casos de prueba realizados sobre el Sistema de Gestión de Recursos Humanos, con el objetivo de garantizar su funcionamiento correcto, confiabilidad y cumplimiento de los requisitos definidos durante el desarrollo. Cada caso de prueba documenta el propósito, los datos de entrada, los pasos ejecutados, el resultado que se espera y el resultado obtenido, lo que permite comparar ambos y evaluar si el comportamiento del sistema es adecuado. Asimismo, se incluyen figuras que respaldan la ejecución de las pruebas en los distintos módulos que se implementan.

Tabla 72

Caso de prueba para inicio de sesión exitoso

Prototipo funcional para la gestión de pago de nómina en la empresa All Sport Nutrition, ubicada en Grecia	
ID del caso de prueba	CT-001
Nombre del caso de prueba	Autenticación con credenciales válidas
Módulo / funcionalidad	Seguridad / Login
Tipo de prueba	Sistema / Aceptación
Prioridad	Alta
Precondiciones	El usuario administrador está registrado en el sistema con un rol válido y está activo. Se conocen el usuario y la contraseña correctos.

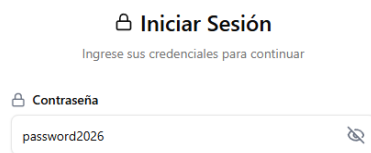
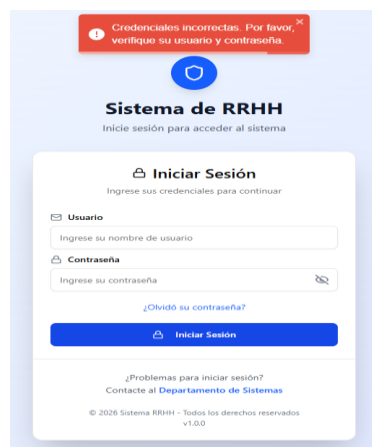
Datos de entrada	usuario: minorAlfa contraseña: cmja1802		
Detalles de pruebas			
Pasos por seguir	Resultado esperado	Resultado obtenido	Estado
1. Ingresar el nombre de usuario en el campo “Usuario”	Permite escribir en el campo del formulario denominado como “Usuario”		Aprobado
2. Ingresar la contraseña en el campo de “Contraseña”.	Permite escribir en el campo del formulario denominado como “Contraseña”		Aprobado
3. Presionar el botón denominado como “Iniciar Sesión”.	Se validan las credenciales y los roles		Aprobado

La prueba se ejecutó de manera satisfactoria y permitió el acceso al sistema con las credenciales del usuario autorizado. El comportamiento registrado coincidió con el resultado que se espera, ya que el sistema validó correctamente las credenciales y dirigió al usuario al panel principal.

Tabla 73

Caso de prueba para inicio de sesión fallido

Prototipo funcional para la gestión de pago de nómina en la empresa All Sport Nutrition, ubicada en Grecia	
ID del caso de prueba	CT-002
Nombre del caso de prueba	Autenticación con credenciales incorrectas

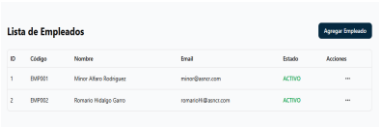
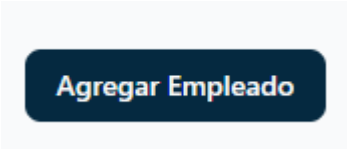
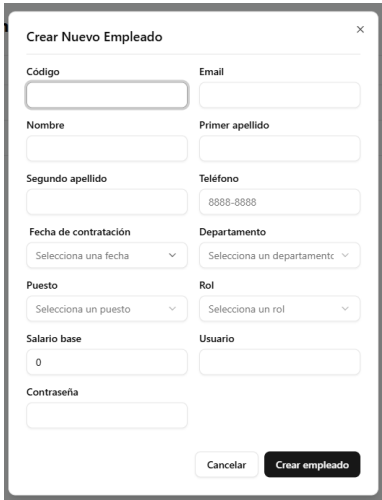
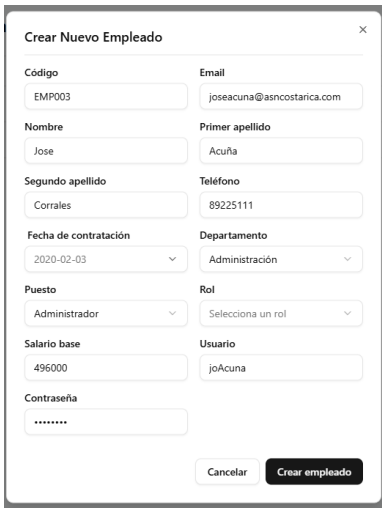
Módulo / funcionalidad	Seguridad / Login		
Tipo de prueba	Sistema / Aceptación negativa		
Prioridad	Alta		
Precondiciones	El usuario administrador está registrado en el sistema con un rol válido y está activo. El usuario intenta ingresar con credenciales inválidas		
Datos de entrada	usuario: minorAlfa contraseña: password2026		
Detalles de pruebas			
Pasos por seguir	Resultado esperado	Resultado obtenido	Estado
1. Ingresar el nombre de usuario en el campo "Usuario"	Permite escribir en el campo del formulario denominado como "Usuario"		Aprobado
2. Ingresar la contraseña en el campo de "Contraseña".	Permite escribir en el campo del formulario denominado como "Contraseña"		Aprobado
3. Presionar el botón denominado como "Iniciar Sesión".	Se validan las credenciales y los roles		Aprobado

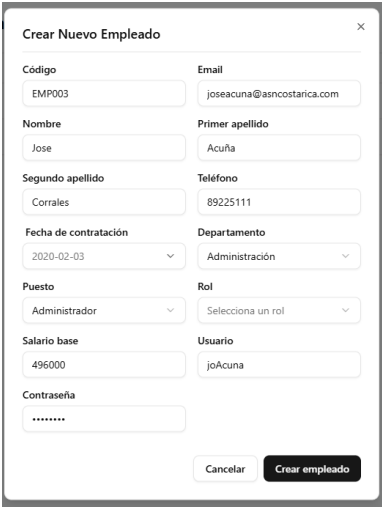
El sistema actuó de manera adecuada cuando se ingresaron credenciales incorrectas, ya que mostró un mensaje de error de autenticación e impidió el acceso. Esto demuestra que el mecanismo de validación y seguridad del módulo de inicio de sesión funciona correctamente.

Tabla 74

Caso de prueba para registrar a una persona exitosamente

Prototipo funcional para la gestión de pago de nómina en la empresa All Sport Nutrition, ubicada en Grecia			
ID del caso de prueba	CT-003		
Nombre del caso de prueba	Registro de datos de un empleado nuevo		
Módulo/funcionalidad	Gestión de empleado/persona		
Tipo de prueba	Integración/Sistema		
Prioridad	Alta		
Precondiciones	El usuario administrador tiene sesión iniciada. No existe previamente una persona con los mismos datos por ingresar.		
Datos de entrada	Código: EMP003 Email: joseacuna@asncostarica.com Nombre: José Primer Apellido: Acuña Segundo Apellido: Corrales Teléfono: 8922 5111 Fecha de contratación: 03/02/2020 Departamento: Administración Puesto: administrador Rol: empleado Salario base: 496,000 Usuario: joAcuna Contraseña: cmja1802		
Detalles de pruebas			
Pasos por seguir	Resultado esperado	Resultado obtenido	Estado

1. Ingresar al módulo de empleados	Permitir ingresar a la vista principal de empleados en el sistema		Aprobado
2. Presionar el botón de "Agregar empleado"	Botón visible para ingresar al formulario de registro		Aprobado
3. Formulario carga los catálogos definidos	El sistema carga los catálogos para que sean seleccionados		Aprobado
4. Se completa el formulario con los datos solicitados	Debe permitir ingresar los datos en sus respectivos campos		Aprobado

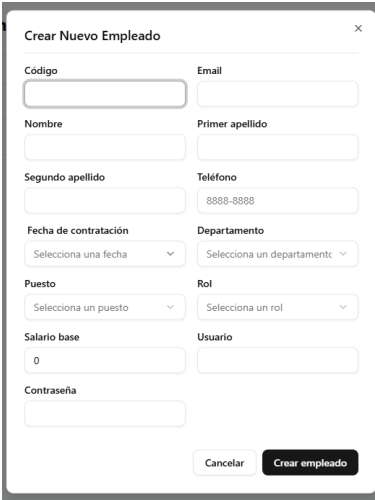
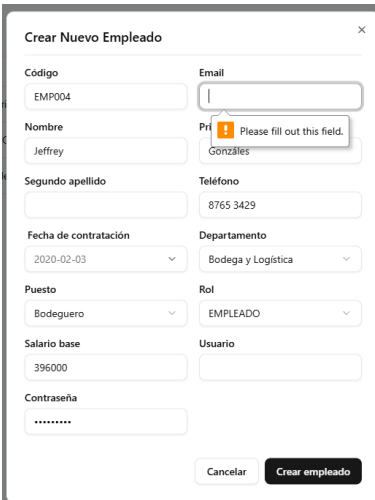
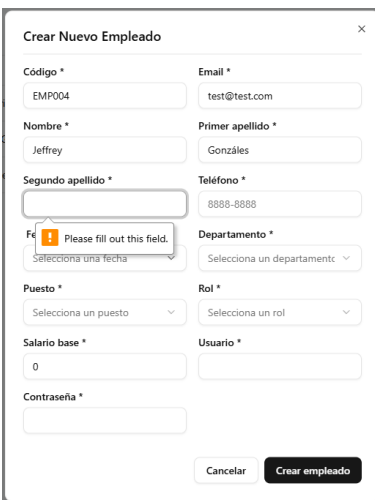
5. Seleccionar el botón "Crear empleado"	Se validan los datos y de ser correctos se crea la persona y se almacena en la base de datos		Aprobado
6. Persona registrada correctamente. Aparece en la lista de empleados	Si la persona se agregó correctamente se debe mostrar en la lista de empleados		Aprobado

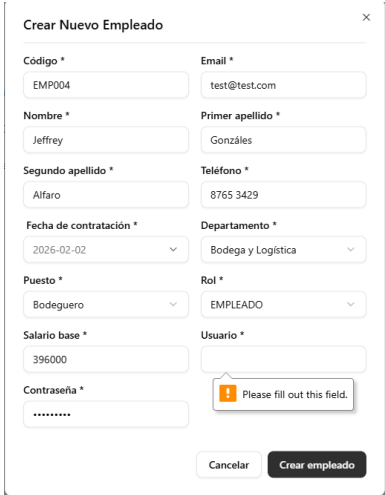
La prueba evidenció el funcionamiento correcto del proceso de registro de personas, ya que se validaron adecuadamente los campos obligatorios y la información se almacenó de forma correcta en la base de datos. Asimismo, el nuevo empleado apareció en la lista de empleados activos, lo que confirma la correcta persistencia de los datos y el cumplimiento del flujo establecido.

Tabla 75
Caso de prueba para registrar a una persona de manera fallida

Prototipo funcional para la gestión de pago de nómina en la empresa All Sport Nutrition, ubicada en Grecia	
ID del Caso de prueba	CT-004
Nombre del Caso de prueba	Registro fallido de datos de un empleado nuevo
Módulo / funcionalidad	Gestión de empleado / persona
Tipo de prueba	Integración / Sistema
Prioridad	Alta

Precondiciones	El usuario administrador tiene sesión iniciada. No existe previamente una persona con los mismos datos por ingresar. Se realiza el intento de registrar un nuevo empleado con datos incompletos que no son válidos.																				
Datos de entrada	Código: EMP004 Email: vacío Nombre: Jeffrey Primer apellido: Gonzáles Segundo apellido: vacío Teléfono: 8765 3429 Fecha de contratación: 03/02/2020 Departamento: Bodega y Logística Puesto: bodeguero Rol: empleado Salario base: 396,000 Usuario: vacío Contraseña: jffeY7890																				
Detalles de pruebas																					
Pasos por seguir	Resultado esperado	Resultado obtenido	Estado																		
1. Ingresar al módulo de empleados	Permitir ingresar a la vista principal de empleados en el sistema	Lista de Empleados Agregar Empleado <table><thead><tr><th>ID</th><th>Código</th><th>Nombre</th><th>Email</th><th>Estado</th><th>Acciones</th></tr></thead><tbody><tr><td>1</td><td>EMP001</td><td>Wilver Allen Rodriguez</td><td>wilver@banco.com</td><td>ACTIVO</td><td>...</td></tr><tr><td>2</td><td>EMP002</td><td>Romario Hidalgo Carrero</td><td>romarioH@banco.com</td><td>ACTIVO</td><td>...</td></tr></tbody></table>	ID	Código	Nombre	Email	Estado	Acciones	1	EMP001	Wilver Allen Rodriguez	wilver@banco.com	ACTIVO	...	2	EMP002	Romario Hidalgo Carrero	romarioH@banco.com	ACTIVO	...	Aprobado
ID	Código	Nombre	Email	Estado	Acciones																
1	EMP001	Wilver Allen Rodriguez	wilver@banco.com	ACTIVO	...																
2	EMP002	Romario Hidalgo Carrero	romarioH@banco.com	ACTIVO	...																
2. Presionar el botón de “Agregar empleado”	Botón visible para ingresar al formulario de registro	Agregar Empleado	Aprobado																		

3. Formulario carga los catálogos previamente definidos	El sistema carga los catálogos para que sean seleccionados	 The screenshot shows the 'Crear Nuevo Empleado' form with the following populated fields: Código (EMP004), Email (test@test.com), Nombre (Jeffrey), Primer apellido (González), Segundo apellido (), Teléfono (8888-8888), Fecha de contratación (2020-02-03), Departamento (Bodega y Logística), Puesto (Bodeguero), Rol (EMPLEADO), Salario base (396000), Usuario (), and Contraseña (*****). The 'Cancelar' and 'Crear empleado' buttons are at the bottom.	Aprobado
4. Se completa el formulario con los datos de forma errónea	No se ingresa ningún dato en el campo de email. Como resultado se obtiene un error sobre que el campo no puede estar vacío	 The screenshot shows the 'Crear Nuevo Empleado' form with an error message 'Please fill out this field.' in the Email field. The other fields are populated: Código (EMP004), Nombre (Jeffrey), Segundo apellido (), Teléfono (8765 3429), Fecha de contratación (2020-02-03), Departamento (Bodega y Logística), Puesto (Bodeguero), Rol (EMPLEADO), Salario base (396000), and Usuario ().	Aprobado
5. Se completa el formulario con los datos de forma errónea	No se ingresa ningún dato en el campo de segundo apellido. Como resultado se obtiene un error sobre que el campo no puede estar vacío.	 The screenshot shows the 'Crear Nuevo Empleado' form with an error message 'Please fill out this field.' in the Segundo apellido field. The other fields are populated: Código (EMP004), Email (test@test.com), Nombre (Jeffrey), Primer apellido (González), Teléfono (8888-8888), Fecha de contratación (2020-02-03), Departamento (Bodega y Logística), Puesto (Bodeguero), Rol (EMPLEADO), Salario base (396000), and Usuario ().	Aprobado

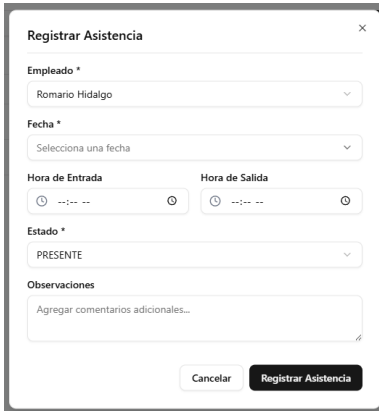
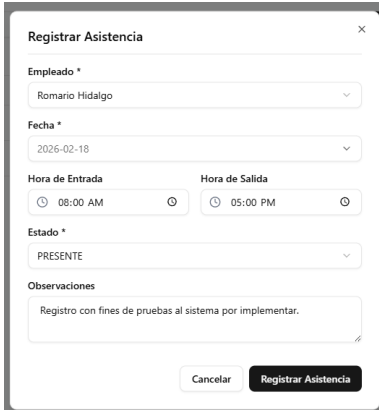
6. Se completa el formulario con los datos de forma errónea	No se ingresa ningún dato en el campo de usuario. Como resultado se obtiene un error sobre que se debe rellenar dicho campo.		Aprobado
---	--	--	----------

Durante la ejecución de la prueba, el sistema respondió de manera adecuada ante la presencia de campos incompletos o con información inválida, mostrando los mensajes de advertencia correspondientes e impidiendo el guardado del registro. El resultado obtenido confirma que las validaciones implementadas, tanto a nivel de formulario como de servicio, funcionan correctamente, asegurando así la integridad y consistencia de los datos almacenados.

Tabla 76

Caso de prueba para registrar la asistencia de forma exitosa

Prototipo funcional para la gestión de pago de nómina en la empresa All Sport Nutrition, ubicada en Grecia	
ID del Caso de prueba	CT-005
Nombre del Caso de prueba	Registro de asistencia de un empleado
Módulo / funcionalidad	Gestión de empleado / persona
Tipo de prueba	Integración / Sistema
Prioridad	Alta
Precondiciones	El usuario administrador tiene sesión iniciada. No existe otra asistencia registrada para ese mismo día. Se cuenta con la fecha y horas de la jornada laboral

Datos de entrada	Empleado: Romario Hidalgo Fecha: 18/02/2026 Hora de entrada: 8:00 a.m. Hora de salida: 5:00 p.m. Estado: presente Observaciones: registro con fines de pruebas al sistema por implementar.		
Detalles de pruebas			
Pasos por seguir	Resultado esperado	Resultado obtenido	Estado
1. Ingresar al módulo de asistencia.	Permite navegar a la ruta de asistencia en el sistema		Aprobado
2. Seleccionar el botón "Registrar Asistencia".	Se abre el pop con el formulario para ejecutar un nuevo registro de asistencia		Aprobado
3. Rellenar el formulario con los datos solicitados	El formulario debe permitirle al usuario rellenar sus campos		Aprobado
4. Seleccionar el botón de "Registrar Asistencia".	El sistema realiza la validación de cada uno de los campos rellenos y si la validación es exitosa, se realiza la inserción de		Aprobado

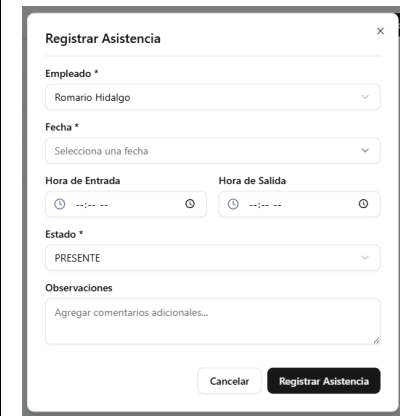
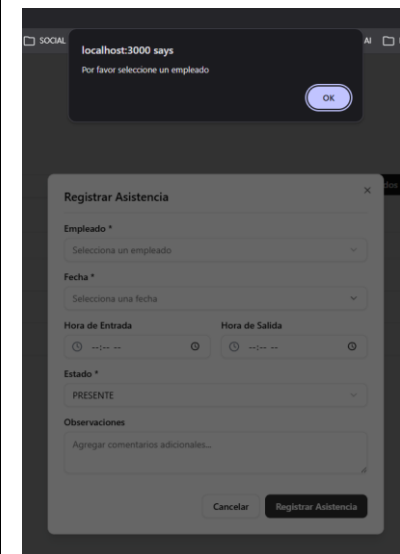
	la información en la base de datos. El usuario puede ver reflejado el registro agregado de la nueva asistencia.		
--	--	--	--

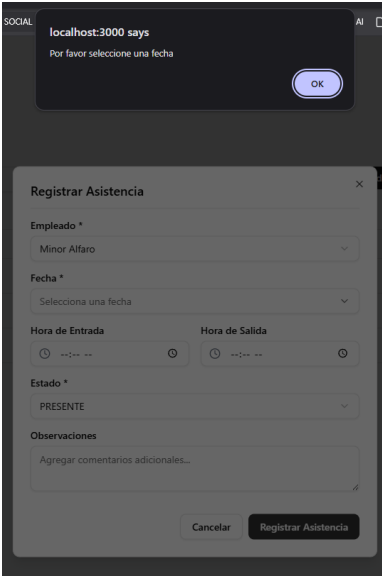
La prueba se ejecutó de manera satisfactoria, permitiendo registrar correctamente la asistencia del empleado y visualizarla en la lista correspondiente. Este resultado confirma el funcionamiento adecuado del proceso de control de asistencia y la correcta persistencia de la información en la base de datos.

Tabla 77

Caso de prueba para registro fallido de la asistencia

Prototipo funcional para la gestión de pago de nómina en la empresa All Sport Nutrition, ubicada en Grecia	
ID del caso de prueba	CT-006
Nombre del caso de prueba	Registro fallido de asistencia de un empleado
Módulo/funcionalidad	Gestión de empleado/persona
Tipo de prueba	Integración/Sistema
Prioridad	Alta
Precondiciones	El usuario administrador tiene sesión iniciada. No existe otra asistencia registrada para ese mismo día a ese mismo empleado. Se cuenta con la fecha y horas de la jornada laboral
Datos de entrada	Empleado: Fecha: Hora de entrada: 8:00 a. m. Hora de salida: 5:00 p. m. Estado: presente Observaciones: registro con fines de pruebas al sistema por implementar.
Detalles de pruebas	

Pasos por seguir	Resultado esperado	Resultado obtenido	Estado
1. Ingresar al módulo de asistencia.	Permite navegar a la ruta de asistencia en el sistema		Aprobado
2. Seleccionar el botón "Registrar Asistencia".	Se abre el <i>pop</i> con el formulario para ejecutar un nuevo registro de asistencia		Aprobado
3. Rellenar el formulario con los datos de forma errónea.	El formulario muestra un mensaje advirtiéndole de que el campo empleado no fue llenado.		Aprobado

4. Rellenar el formulario con los datos de forma errónea.	El formulario muestra un mensaje advirtiéndolo de que el campo fecha no fue llenado.	 The screenshot shows a mobile application interface. At the top, a dark toast message box says "localhost:3000 says" and "Por favor seleccione una fecha" (Please select a date) with an "OK" button. Below this is a "Registrar Asistencia" (Register Attendance) form. The form has fields for "Empleado *" (Employee) with "Minor Alfaro" selected, "Fecha *" (Date) with "Selecciona una fecha" (Select a date) as a placeholder, "Hora de Entrada" (Entry Time) and "Hora de Salida" (Exit Time) with clock icons, "Estado *" (Status) with "PRESENTE" selected, and an "Observaciones" (Observations) text area. At the bottom are "Cancelar" and "Registrar Asistencia" buttons.	Aprobado
--	---	--	-----------------

El sistema respondió adecuadamente ante un intento de registro duplicado, mostrando un mensaje de advertencia e impidiendo la inserción del nuevo registro. Esto confirma que la validación de solapamiento por empleado y fecha funciona correctamente, preserva la integridad de la información y evita duplicidades en el módulo de asistencia.

La ejecución de los casos de prueba permitió comprobar el desempeño correcto de los distintos módulos del Sistema de Gestión de Recursos Humanos, tanto en escenarios exitosos como en situaciones de error controlado. Los resultados evidencian que el sistema cumple con los requerimientos funcionales establecidos, valida apropiadamente los datos ingresados y garantiza la integridad de la información almacenada. En consecuencia, se confirma la estabilidad, confiabilidad y consistencia del prototipo funcional desarrollado para la empresa All Sport Nutrition.

CAPÍTULO VI. CONCLUSIONES Y RECOMENDACIONES

En este capítulo se presentan las conclusiones y recomendaciones derivadas del desarrollo del prototipo funcional del Sistema de Gestión de Recursos Humanos para la empresa All Sport Nutrition. A través de las conclusiones, se destacan los principales logros alcanzados a lo largo del proyecto, lo que evidencia el esfuerzo y la dedicación invertidos en cada etapa del proceso. Por otra parte, las recomendaciones proponen acciones orientadas a la mejora continua y al establecimiento de buenas prácticas, con el objetivo de garantizar un uso adecuado del sistema y maximizar su aprovechamiento en beneficio de la organización y de su equipo humano.

Conclusiones

En este apartado se resumen los hallazgos más relevantes que se obtienen a lo largo del proyecto, el grado de cumplimiento de los objetivos que se plantearon desde su inicio y el impacto concreto que representa el prototipo desarrollado para All Sport Nutrition. Asimismo, se reconoce la importancia de esta iniciativa como un aporte, tanto académico como práctico en el ámbito de la gestión de nómina y recursos humanos, un área que, aunque fundamental para cualquier organización, con frecuencia opera con procesos manuales susceptibles a errores e ineficiencias.

El desarrollo del sistema web de gestión de pago de nómina para All Sport Nutrition permitió alcanzar de manera satisfactoria cada uno de los objetivos que se plantearon al inicio del proyecto. En primer lugar, se realizó un análisis exhaustivo de los requerimientos funcionales y no funcionales de la empresa, con el propósito de comprender en profundidad el problema existente y diseñar una solución verdaderamente adaptada a sus necesidades operativas y administrativas. Este proceso de levantamiento de requisitos resultó esencial para garantizar que el sistema no solo respondiera a las expectativas técnicas, sino también a las realidades cotidianas del equipo que lo utilizaría.

A partir de dicho análisis, se construyó una arquitectura de *software* ajustada a la infraestructura tecnológica disponible en la empresa y se estableció una base de datos robusta en MySQL, capaz de gestionar de forma segura y eficiente la información sensible vinculada a las personas colaboradoras y sus pagos. Sobre esta base, se implementó un prototipo funcional utilizando Java y Spring Boot, tecnologías que permitieron automatizar los procesos centrales del área de nómina, lo que incluye el cálculo del salario bruto y neto, las deducciones legales

correspondientes a la Caja Costarricense de Seguro Social, el aguinaldo, las vacaciones y las liquidaciones, entre otros componentes críticos de la gestión salarial. Finalmente, se realizaron pruebas exhaustivas orientadas a verificar la integridad de los datos, la seguridad del sistema y el cumplimiento estricto de la legislación laboral costarricense vigente, asegurando que la herramienta opere dentro del marco legal que protege, tanto a la compañía como a sus trabajadores.

Uno de los hallazgos más significativos del proyecto fue la demostración clara de automatizar procesos como el cálculo de salarios, deducciones, aguinaldo, vacaciones y liquidaciones reduce de manera considerable los errores humanos que inevitablemente se producen cuando estas tareas se realizan de forma manual. En el contexto de All Sport Nutrition, donde el volumen de colaboradores y la dinámica comercial del sector nutricional y deportivo exigen precisión y agilidad, contar con un sistema que ejecuta estos cálculos de manera automática, consistente y auditada representa un avance significativo. Además de reducir los errores, el sistema elimina la duplicidad de asignaciones, optimiza el tiempo que el personal administrativo dedica a procedimientos repetitivos y eleva la confiabilidad de la información que respalda las decisiones gerenciales.

La elección de una arquitectura monolítica con servidor embebido Tomcat, combinada con tecnologías de código abierto como Spring Boot, Thymeleaf, Bootstrap y MySQL, respondió a una decisión técnica y estratégica bien fundamentada. Esta combinación garantizó la compatibilidad con la infraestructura tecnológica de la empresa y, al mismo tiempo, permitió reducir los costos de licenciamiento y simplificar, tanto la implementación inicial como el mantenimiento futuro del sistema. Lejos de ser una limitante, esta resolución convirtió al proyecto en una solución viable, técnica y económicamente accesible para una compañía en crecimiento como All Sport Nutrition, que requiere herramientas poderosas sin incurrir en inversiones desproporcionadas.

Más allá de los beneficios operativos inmediatos, el sistema desarrollado representa una herramienta estratégica que impulsa la transformación digital de All Sport Nutrition. Al modernizar uno de los procesos administrativos más sensibles de cualquier empresa, el proyecto no solo fortalece la gestión interna, sino que también promueve una cultura organizacional más orientada a la eficiencia, la transparencia y el uso inteligente de la tecnología. En una industria tan

dinámica y competitiva como la de la nutrición deportiva, esta capacidad de adaptación y modernización constituye un diferenciador real.

Por último, este proyecto constituyó una oportunidad invaluable para aplicar de manera integral los conocimientos adquiridos a lo largo de la carrera de Ingeniería Informática. Cada etapa del desarrollo, desde el análisis de requerimientos hasta las pruebas finales, representó un espacio para consolidar competencias técnicas y metodológicas, así como para adquirir nuevas habilidades surgidas del contacto directo con un entorno empresarial real. Esta experiencia práctica enriqueció la formación profesional del estudiante y reafirmó la importancia de vincular el ámbito académico con las necesidades concretas del sector productivo, lo que genera valor, tanto para la institución educativa como para la empresa que confió en el proyecto.

Recomendaciones

En este apartado se presentan las recomendaciones derivadas del proceso de desarrollo del prototipo funcional, con el propósito de orientar las acciones futuras que favorezcan su implementación correcta y aprovechamiento dentro de All Sport Nutrition. Estas sugerencias se fundamentan en los resultados, en las observaciones realizadas durante cada etapa del proyecto y en las particularidades operativas de la empresa. Lo anterior tiene el fin de fortalecer el sistema y garantizar su uso eficiente, seguro y sostenible a lo largo del tiempo.

Implementación en un servidor local dedicado y el plan de respaldo de información. Se recomienda que, antes de la puesta en producción del sistema, All Sport Nutrition disponga de un servidor local dedicado que garantice la estabilidad, disponibilidad y continuidad operativa del sistema de gestión de pago de nómina. Contar con una infraestructura propia y bien configurada constituye una condición fundamental para que el sistema funcione de manera confiable en sus actividades diarias, sin depender de equipos compartidos o conexiones externas que puedan comprometer su rendimiento.

Paralelamente, es indispensable establecer un esquema de copias de respaldo automáticas que incluya, tanto la base de datos como el código fuente y las configuraciones del sistema. Se recomienda realizar respaldos diarios y mantener al menos una copia adicional almacenada fuera de las instalaciones físicas de la empresa, ya sea en un servicio de almacenamiento en la nube o en un dispositivo externo resguardado en un lugar seguro. Esta práctica garantiza el cumplimiento de

los principios de integridad, disponibilidad y confidencialidad de la información, pilares esenciales en la gestión de datos sensibles como los vinculados a la nómina y a las personas colaboradoras. De este modo, ante cualquier incidente técnico, falla eléctrica o eventualidad imprevista, All Sport Nutrition dispondrá de los mecanismos necesarios para recuperar su información y reanudar sus operaciones con la menor afectación posible.

Se recomienda que esta acción se ejecute en la fase previa a la puesta en producción del sistema, con una duración estimada de entre dos y cuatro semanas. Este periodo permite configurar adecuadamente la infraestructura tecnológica y establecer los mecanismos de respaldo necesarios para garantizar la disponibilidad, integridad y continuidad de la información desde el inicio de la operación.

Ejecución de un plan piloto en entorno simulado de producción. Antes de realizar la adopción oficial del sistema, se recomienda implementar un plan piloto en un entorno que reproduzca con la mayor fidelidad posible las condiciones reales de operación de All Sport Nutrition. Este entorno simulado debe considerar el uso de datos de ejemplo representativos y la participación de usuarios de prueba que realicen las tareas cotidianas del área de nómina y recursos humanos, tales como el registro de colaboradores, el procesamiento de planillas, el cálculo de deducciones y la generación de reportes.

Durante esta fase se evalúan de forma sistemática la estabilidad del sistema en condiciones de uso normal, su comportamiento ante situaciones excepcionales, la seguridad de la información gestionada y la usabilidad de la interfaz desde la perspectiva del usuario final. Todos los hallazgos, incidencias y observaciones deben documentarse con detalle, de manera que el desarrollador pueda aplicar los ajustes y correcciones necesarios antes del lanzamiento oficial. Esta recomendación resulta especialmente relevante en el contexto de All Sport Nutrition, donde la precisión en el pago de nómina impacta de forma directa en la confianza y el bienestar de las personas colaboradoras. Por ende, cualquier error no detectado a tiempo puede tener consecuencias, tanto económicas como humanas.

Esta recomendación debe desarrollarse durante la fase de validación del sistema, posterior a la preparación de la infraestructura, con una duración aproximada de dos a tres semanas. Su finalidad es evaluar el comportamiento del sistema en condiciones similares a las reales, permitiendo identificar y corregir posibles errores antes de su adopción oficial.

Desarrollo e implementación de un plan de capacitación para el personal. El éxito de cualquier sistema tecnológico no depende únicamente de su calidad técnica, sino también de la capacidad de las personas que lo utilizan para aprovecharlo de manera efectiva. Por este motivo, se recomienda diseñar e implementar un plan de capacitación estructurado dirigido al personal de All Sport Nutrition que tiene contacto directo con el sistema, antes de que este entre en operación formal.

Este plan debe desarrollarse mediante sesiones prácticas realizadas en un entorno controlado, en las que las personas participantes puedan interactuar con el sistema utilizando ejemplos reales extraídos del contexto propio de la empresa, como el procesamiento de una planilla quincenal o el cálculo del aguinaldo de un colaborador específico. Este enfoque práctico facilita un aprendizaje más significativo y reduce la incertidumbre que surge de manera natural ante una herramienta nueva. Además de instruir sobre las funcionalidades técnicas del sistema, es fundamental que la capacitación incluya una dimensión orientada a la actitud y la motivación, en la que se expliquen con claridad los beneficios concretos que el sistema aporta al trabajo diario de cada persona, desde la reducción de errores hasta el ahorro de tiempo en tareas repetitivas.

Complementariamente, se recomienda ofrecer sesiones de acompañamiento durante las primeras semanas de uso real del sistema, en las que un referente técnico esté disponible para atender dudas, resolver inconvenientes y reforzar el aprendizaje en contexto. Esta etapa de acompañamiento resulta clave para consolidar la confianza del personal en la herramienta y asegurar una transición fluida hacia el nuevo modelo de gestión.

La capacitación debe implementarse en la etapa previa a la salida a producción y extenderse durante las primeras semanas de uso real del sistema, con una duración estimada de una a dos semanas más un periodo adicional de acompañamiento. Este proceso asegura que los usuarios adquieran las competencias necesarias para utilizar el sistema de forma eficiente y segura.

Promoción de una cultura organizacional orientada a la transformación digital. Toda implementación tecnológica exitosa requiere no solo una buena solución técnica y una capacitación adecuada, sino también un entorno organizacional que favorezca el cambio. Por esto, se recomienda que All Sport Nutrition trabaje de forma paralela al proceso de puesta en funcionamiento en fortalecer una cultura interna que promueva la confianza en la tecnología, la apertura al cambio y el uso responsable de las herramientas digitales.

El objetivo es que ningún colaborador perciba la llegada del sistema como una amenaza a su rol o a su estabilidad laboral, sino como un aliado que simplifica y enriquece su trabajo cotidiano. Para lograrlo, se sugiere generar espacios de diálogo abierto en los que los empleados puedan expresar sus inquietudes, hacer preguntas y conocer de primera mano cómo el sistema impacta positivamente en sus funciones. Las demostraciones prácticas, en las que se evidencian casos concretos de mejora, como la reducción del tiempo de cálculo de planillas o la eliminación de errores recurrentes, resultan efectivas para construir confianza y sentido de pertenencia hacia la nueva herramienta.

Esta recomendación debe iniciarse antes de implementar formalmente el sistema o, como mínimo, de forma paralela a la fase de capacitación, de manera que, cuando el sistema entre en operación, el equipo humano de All Sport Nutrition ya lo haya integrado como parte natural de su cultura de trabajo y no como algo ajeno o impuesto. Una empresa que adopta la tecnología con convicción y comprensión está mucho mejor posicionada para aprovechar al máximo sus herramientas digitales y para continuar creciendo de manera sostenible en un mercado cada vez más exigente y competitivo.

Se recomienda iniciar esta acción de forma paralela a la fase preparatoria del sistema y mantenerla de manera continua durante todo el proceso de implementación y operación. Al tratarse de un proceso progresivo, no posee una duración limitada, sino que se consolida a lo largo del tiempo mediante estrategias de comunicación, sensibilización y participación del personal.

REFERENCIAS

- Ames, N. G.; Gutiérrez, G. Q. y Espinoza, B. S. (2023). La digitalización empresarial: un fenómeno de la era postpandemia. *Revista Lidera*, 18, 62-69.
<https://revistas.pucp.edu.pe/index.php/revistalidera/article/download/28552/26318>
- Araujo-Inastrilla, C. R.; Roche-Madrigal, M. D. C. y García-Savón, Y. (2023). Diseño de base de datos para el departamento de Sistemas de Información en Salud, La Habana 2021. *Revista Información Científica*, 102. <http://scielo.sld.cu/pdf/ric/v102/1028-9933-ric-102-4099.pdf>
- Asamblea Legislativa de la República de Costa Rica. (1982). *Ley n.º 6683: Ley de Derechos de Autor y Derechos Conexos*. La Gaceta n.º 102.
- Asamblea Legislativa de la República de Costa Rica. (2001a). *Ley n.º 8148: Reforma al Código Penal en materia de delitos informáticos*. La Gaceta n.º 109.
- Asamblea Legislativa de la República de Costa Rica. (2001b). *Ley n.º 4573: Ley para reprimir y sancionar los delitos informáticos*. La Gaceta n.º 174.
- Asamblea Legislativa de la República de Costa Rica. (2011a). *Ley n.º 8968: Ley de Protección de la Persona frente al Tratamiento de sus Datos Personales*. La Gaceta n.º 170.
<https://www.poder-judicial.go.cr/observatoriojuridico/index.php/legislacion/normas-nacionales/206-ley-8968>
- Asamblea Legislativa de la República de Costa Rica. (2011b). *Ley n.º 8968: Ley de protección de la persona frente al tratamiento de sus datos personales*. La Gaceta n.º 170.
- Barrera, J. R. A.; Torres, Y. E. S. y Alarcón, L. A. V. (2024). La Tecnología y su Impacto en la Gestión de Procesos y Estrategias de Automatización. *Ciencia Latina Revista Científica Multidisciplinar*, 8(4), 6198-6221.
<https://ciencialatina.org/index.php/cienciala/article/view/12822/18518>
- Capuñay Uceda, O. E. y Antón Pérez, J. M. (2021). Influencia de SCRUM en los plazos de entrega y rendimiento en los proyectos de las asignaturas de Desarrollo de Software. *Revista Iberoamericana de Tecnología en Educación y Educación en tecnología*, 29, 36-

42. https://www.scielo.org.ar/scielo.php?pid=S1850-99592021000200005&script=sci_abstract&tlng=en
- Chero-Pacheco, V. (2024). Población y muestra. *International Journal Of Interdisciplinary Dentistry*, 17(2), 66-66. https://www.scielo.cl/scielo.php?pid=S2452-55882024000200066&script=sci_arttext&tlng=en
- Cisneros Caicedo, A. J.; Guevara García, A. F.; Urdánigo Cedeño, J. J. y Garcés Bravo, J. E. (2022). Técnicas e Instrumentos para la Recolección de Datos que apoyan a la Investigación Científica en tiempo de Pandemia. *Dominio de las Ciencias*, 8(1), 1165-1185. <https://educaciontributaria.com.py/revista/index.php/rcetca/article/view/70/104>
- Colombero, V. T.; Estelles, J. P.; Gallegos, L. y López, P. (2024). Monolitos vs. Microservicios en Arquitectura de Software: Perspectivas para un Desarrollo Eficiente. *JAIIO, Jornadas Argentinas de Informática*, 10(5), 42-54. https://sedici.unlp.edu.ar/bitstream/handle/10915/177339/Documento_completo.pdf?sequence=1&isAllowed=y
- Espinosa-Hurtado, R. (2021). Análisis comparativo para la evaluación de frameworks usados en el desarrollo de aplicaciones web. *Cedamaz*, 11(2), 133-141. <https://revistas.unl.edu.ec/index.php/cedamaz/article/view/1182/853>
- Esquivel, W. C. y Sevilla, G. L. (2021). Paralelismos entre bases de datos relacionales y no relacionales (un enfoque en seguridad). *ReCIBE. Revista electrónica de Computación, Informática, Biomédica y Electrónica*, 10(2), 1-16. <https://www.redalyc.org/journal/5122/512269058002/512269058002.pdf>
- Faneite, S. F. A. (2023). Los enfoques de investigación en las Ciencias Sociales. *Revista Latinoamericana Ogmios*, 3(8), 82-95. <https://idicap.com/ojs/index.php/ogmios/article/view/226/237>
- Guerra, F. J. (2021). *Una propuesta para facilitar la trazabilidad de los requerimientos en un desarrollo ágil* (Doctoral dissertation, Universidad Nacional de La Plata). https://sedici.unlp.edu.ar/bitstream/handle/10915/122955/Documento_completo.pdf-PDFA.pdf?sequence=1&isAllowed=y.

- Hernández Sampieri, R.; Méndez Valencia, S.; Mendoza Torres, C. P. y Cuevas Romo, A. (2017). *Fundamentos de investigación* (1.^a ed.). McGraw-Hill Education.
- Hurtado Díaz, A. A. y Samboni Bolaños, M. J. (2021). *La nómina electrónica como nuevo requerimiento de información a reportar ante la DIAN*.
<https://repositorio.unicomfauca.edu.co/bitstream/handle/3000/34/La%20n%c3%b3min%20electr%c3%b3nica%20como%20nuevo%20requerimiento%20de%20informaci%c3%b3n%20a%20reportar%20ante%20la%20DIAN.pdf?sequence=1&isAllowed=y>.
- Larrea, N. P. L.; Valencia, M. V. R.; Cazco, S. A. S. y Hermida, B. A. B. (2022). Análisis de los lenguajes de programación más utilizados en el desarrollo de aplicaciones web y móviles. *Dominio de las Ciencias*, 8(3), 132.
<https://dialnet.unirioja.es/servlet/articulo?codigo=8635309>
- Mamani, C. A. L. (2023). Pruebas de software para microservicios. *Innovation and Software*, 4(1), 151-160. <https://revistas.ulasalle.edu.pe/innosoft/article/view/86>
- Maranto Rivera, M. y González Fernández, M. E. (2015). *Fuentes de información*. Universidad Autónoma del Estado de Hidalgo.
<https://dspace.uaeh.edu.mx/server/api/core/bitstreams/624c644f-fe81-42be-9a5d-2ccde73a78e6/content>
- Martínez, D. V. S. (2022). Técnicas e instrumentos de recolección de datos en investigación. *TEPEXI boletín científico de la escuela superior Tepeji del Río*, 9(17), 38-39.
<https://repository.uaeh.edu.mx/revistas/index.php/tepexi/article/view/7928/8457>
- Ministerio de Ciencia, Innovación, Tecnología y Telecomunicaciones (Micitt). (2022). *Plan Nacional Ciencia Tecnología Innovación 2022-2027*.
https://www.micitt.go.cr/sites/default/files/planes_estrategias/Plan_Nacional_Ciencia_Tecnologia_Innovacion_2022-2027.pdf
- Ministerio de Trabajo y Seguridad Social (MTSS). (1943). *Código de Trabajo (Ley n.º 2)*. *La Gaceta*. https://www.mtss.go.cr/elministerio/marco-legal/documentos/Codigo_Trabajo_RPL.pdf

- Ministerio de Trabajo y Seguridad Social de Costa Rica. (1943). *Código de Trabajo (Ley n.º 2)*. Publicado en *La Gaceta* n.º 262 del 26 de noviembre de 1943.
https://www.mtss.go.cr/legislacion/codigo_trabajo.html
- Neninger, J. C. B.; Quintero, Y. C. y Márquez, N. D. C. Á. (2024). La seguridad de la información a través de la creación y restauración de las copias. *Journal TechInnovation*, 3(1), 67-77. <https://revistas.unesum.edu.ec/JTI/index.php/JTI/article/view/74/121>
- Palomeque-Zambrano, E. E. y Campoverde-Molina, M. (2023). Propuesta de una arquitectura de software para el desarrollo de sitios web accesibles. *MQRInvestigar*, 7(3), 1458-1474.
<https://www.investigarmqr.com/ojs/index.php/mqr/article/view/523/2104>
- Prats, G. M.; Oliva, F. H.; Hernández, F. S. y Franco, F. M. (2021). Impacto del diseño de un manual de procedimiento en la nómina para mejorar su control en las empresas. *Revista de Investigación Académica Sin Frontera: Facultad Interdisciplinaria de Ciencias Económicas Administrativas-Departamento de Ciencias Económico Administrativas-Campus Navojoa*, 36.
<https://revistainvestigacionacademicasinfrontera.unison.mx/index.php/RDIASF/article/view/423/396>
- Procuraduría General de la República. (s. f.). *Ley Constitutiva de la Caja Costarricense de Seguro Social CCSS*.
https://pgrweb.go.cr/scij/Busqueda/Normativa/Normas/nrm_texto_completo.aspx?nValor1=1&nValor2=2340
- Salazar Salazar, C. (2021). *Población, muestra y muestreo*.
<https://repositorio.usam.ac.cr/xmlui/bitstream/handle/11506/2634/LEC%20MET%200020%202021.pdf?sequence=1&isAllowed=y>
- Santander-Salmon, E. S.; Herrera-Sánchez, M. J. y Bravo-Bravo, I. F. (2023). La importancia de la digitalización en la administración empresarial mediante un análisis bibliográfico actualizado. *Multidisciplinary Collaborative Journal*, 1(2), 39-51.
<https://mcjournal.editorialdoso.com/index.php/home/article/view/15/40>

- Segura, A. S. (2025). *Transformación digital y gestión de recursos humanos: impacto de la IA, la automatización y las tecnologías emergentes*. Gestión.
<https://www.dostecnologiaynegocios.com/2025/08/transformacion-digital-y-gestion-de.html>
- Valle, A.; Manrique, L. y Revilla, D. (2022). *La investigación descriptiva con enfoque cualitativo en educación*.
<https://repositorio.pucp.edu.pe/server/api/core/bitstreams/20d5ac9a-57bf-42e9-966e-73c5aabbbf95/content>
- Velasco, M. V. E.; Villacis, J. A. N.; Chávez, P. R. S. y Cuchipec, W. C. C. (2021). Revisión sistemática de la metodología SCRUM para el desarrollo de Software. *Dominio de las Ciencias*, 7(4), 54. <https://dialnet.unirioja.es/servlet/articulo?codigo=8384028>
- Velasco, M. V. E.; Villacis, J. A. N.; Chávez, P. R. S. y Cuchipec, W. C. C. (2021). Revisión sistemática de la metodología SCRUM para el desarrollo de Software. *Dominio de las Ciencias*, 7(4), 54. <https://dialnet.unirioja.es/servlet/articulo?codigo=8384028>
- World Wide Web Consortium (W3C). (2023). *Web Content Accessibility Guidelines (WCAG) 2.2*. <https://www.w3.org/TR/WCAG22/>
- Zapana, G. I. H.; Castro, N. H.; Tico, M. A. S.; Choquehuanca, E. D. C. y Bejarano, A. D. T. (2021). Aplicación del método Design Thinking en el área de requerimientos de software. *Innovation and Software*, 2(1), 43-52.
<https://revistas.ulasalle.edu.pe/innosoft/article/view/37/33>
- Zapana, G. I. H.; Castro, N. H.; Tico, M. A. S.; Choquehuanca, E. D. C. y Bejarano, A. D. T. (2021). Aplicación del método Design Thinking en el área de requerimientos de software. *Innovation and Software*, 2(1), 43-52.
<https://revistas.ulasalle.edu.pe/innosoft/article/view/37/33>

APÉNDICES

Apéndice A. Guía de observación

Nombre de la empresa: All Sport Nutrition.

Actividad de la empresa: comercialización y distribución de suplementos nutricionales y productos deportivos.

Objetivo: Observar y evaluar las actividades que se relacionan con el proyecto de sistema web para la gestión del pago de nómina.

No	Aspectos por observar	Cumple	No cumple	Oportunidad de mejora	Detalle de observación
1	Existencia de un registro sistematizado de horas laborales y asistencia del personal.				
2	Disponibilidad de un sistema automatizado para el cálculo de planillas.				
3	Exactitud en los cálculos de deducciones legales (CCSS, INS e impuestos).				
4	Seguridad en el manejo de datos de las personas colaboradoras.				
5	Registro histórico de pagos y comprobantes de nómina.				
6	Tiempo total invertido en la elaboración de planillas mensuales.				

Apéndice B. Guía de observación aplicada a la empresa All Sport Nutrition

Nombre de la empresa: All Sport Nutrition.

Actividad de la empresa: comercialización y distribución de suplementos nutricionales y productos deportivos.

Objetivo: Observar y evaluar las actividades que se relacionan con el proyecto del sistema web para la gestión del pago de nómina.

No	Aspectos por observar	Cumple	No cumple	Oportunidad de mejora	Detalle de observación
1	Existencia de un registro sistematizado de horas laborales y asistencia del personal.		X	Implementar un módulo de asistencia con registros automáticos.	Actualmente, la asistencia se registra de forma manual en hojas de Microsoft Excel y en algunos casos mediante mensajes enviados por WhatsApp. No existe un sistema centralizado ni validación automática de horas de entrada y salida, lo que genera inconsistencias y dificulta el control de horas trabajadas.
2	Disponibilidad de un sistema automatizado para el cálculo de planillas.		X	Automatizar el cálculo de planillas para reducir errores humanos y tiempos de elaboración.	El cálculo de planillas se realiza manualmente cada mes, utilizando fórmulas básicas en Microsoft Excel. No existe integración automática con datos de horas, deducciones u horas extra. El proceso consume tiempo y es propenso a

					errores.
3	Exactitud en los cálculos de deducciones legales (CCSS, INS e impuestos).	Parcialmente cumple con las deducciones		Integrar las tablas y porcentajes legales en un sistema que actualice los cálculos automáticamente .	Las deducciones se calculan manualmente siguiendo tablas consultadas en Internet. Aunque generalmente se hacen bien, es común que cambien los porcentajes y se deban ajustar los Microsoft Excel, lo cual aumenta el riesgo de errores y retrabajo.
4	Seguridad en el manejo de datos de las personas colaboradoras.		X	Implementar un sistema con usuarios, contraseñas, perfiles y encriptación básica para proteger datos sensibles.	Los datos del personal (salarios, información personal y deducciones) se almacenan en archivos de Microsoft Excel sin contraseñas y muchas veces se comparten por correo o WhatsApp. No existe control de acceso, lo que pone en riesgo la confidencialidad de la información.
5	Registro histórico de pagos y comprobantes de nómina.		X	Centralizar y automatizar el almacenamiento de comprobantes y el historial de pagos.	Los comprobantes se generan manualmente y a veces se envían por correo individual. No hay un archivo centralizado y localizar planillas antiguas puede requerir revisar múltiples documentos. El historial no es consistente ni fácilmente

					auditable.
6	Tiempo total invertido en la elaboración de planillas mensuales.		X	Reducir tiempos implementando un sistema que calcule automáticamente montos, deducciones y validaciones.	Actualmente, el proceso de elaborar la planilla toma entre 4 y 6 horas al mes, según el mes, las horas extra y los permisos del personal. El tiempo es elevado debido a que todo se debe verificar manualmente y cruzar datos desde varios archivos.

Apéndice C. Entrevista semiestructurada

Organización: All Sport Nutrition.

Nombre del entrevistado:

Cargo:

Preguntas:

1. ¿Con qué frecuencia se presentan errores o inconsistencias en los cálculos de planilla?
2. ¿Cuáles considera que son las principales limitaciones del sistema o método actual para gestionar la nómina?
3. ¿Qué tipo de información relacionada con la nómina considera más importante mantener actualizada y accesible?
4. ¿De qué forma se resguarda actualmente la información confidencial de las personas colaboradoras (salarios, deducciones, datos personales)?
5. ¿Cuán sencillo o complicado resulta el proceso de generar los reportes de pago y deducciones legales (CCSS, INS, Hacienda)?
6. Desde su experiencia ¿cómo afecta el sistema actual al tiempo de trabajo y a la productividad del Área Administrativa?
7. ¿Cuáles beneficios considera que aportaría la implementación de un sistema web automatizado para la gestión de nómina?
8. ¿Cuáles características o funcionalidades considera esenciales que debe incluir dicho sistema web (por ejemplo, acceso seguro, historial de pagos, reportes automáticos, interfaz sencilla, etc.)?
9. ¿Cómo percibe la disposición del personal de la empresa ante el uso de nuevas tecnologías o sistemas digitales para este tipo de procesos?
10. ¿Cuáles recomendaciones haría para garantizar una implementación exitosa del nuevo sistema web en la empresa?

Apéndice D. Entrevista semiestructurada aplicada al gerente general de la empresa

Organización: All Sport Nutrition.

Nombre del entrevistado: Juan Pablo Alvarado Barahona.

Cargo: gerente general.

Preguntas:

1. ¿Con qué frecuencia se presentan errores o inconsistencias en los cálculos de la planilla?

Los errores son relativamente frecuentes, diría que al menos una vez al mes encontramos inconsistencias, especialmente cuando hay horas extra, incapacidades o permisos. Como la información se maneja manualmente en Excel, cualquier ajuste puede generar cálculos incorrectos o duplicados.

2. ¿Cuáles considera que son las principales limitaciones del sistema o método actual para gestionar la nómina?

La principal limitación es que todo se hace de manera manual, lo que implica un alto riesgo de errores humanos. No contamos con un registro automático de asistencia, no existe integración entre módulos y dependemos de diferentes hojas de cálculo que no siempre están actualizadas o sincronizadas. Además, el proceso consume demasiado tiempo administrativo.

3. ¿Qué tipo de información relacionada con la nómina considera más importante mantener actualizada y accesible?

Necesitamos acceso rápido y preciso a información como: horas trabajadas y horas extra, deducciones legales actualizadas, historial de incapacidades y permisos, salarios y ajustes salariales y comprobantes e historial de pagos. Tener estos datos al día evitaría errores y facilitaría auditorías internas.

4. ¿De qué forma se resguarda actualmente la información confidencial de los colaboradores (salarios, deducciones, datos personales)?

Actualmente, los datos se resguardan principalmente en archivos de Excel con acceso manual, algunos protegidos y otros no. Además, se envían por correo o WhatsApp en casos específicos. Reconozco que esto no es seguro ni cumple buenas prácticas de protección de datos.

5. ¿Cuán sencillo o complicado resulta el proceso de generar los reportes de pago y deducciones legales (CCSS, INS, Hacienda)?

El proceso es complicado y lento. Al no tener un sistema automatizado, los reportes deben generarse manualmente y verificar cada dato uno por uno. Esto toma tiempo y requiere mucha revisión para evitar errores.

6. Desde su experiencia ¿cómo afecta el sistema actual al tiempo de trabajo y a la productividad del Área Administrativa?

Afecta significativamente. El área administrativa invierte muchas horas en tareas que podrían automatizarse, como cálculos, validaciones y consolidación de información. Esto reduce la productividad e incluso retrasa otros procesos importantes de la empresa.

7. ¿Cuáles beneficios considera que aportaría la implementación de un sistema web automatizado para la gestión de nómina?

Un sistema automatizado aportaría: reducción de errores, mayor rapidez en cálculos y reportes, seguridad de la información, mejor organización del historial de planillas, ahorro de tiempo, optimización del trabajo administrativo y mayor transparencia para colaboradores y jefaturas.

8. ¿Cuáles características o funcionalidades considera esenciales que debe incluir dicho sistema web (por ejemplo, acceso seguro, historial de pagos, reportes automáticos, interfaz sencilla, etc.)?

Considero indispensables un acceso seguro con perfiles y permisos, un registro de asistencia y horas extra, el cálculo automático de planillas y deducciones, un historial de pagos con comprobantes descargables, reportes automáticos para CCSS, INS y Hacienda, una interfaz sencilla y usable, así como notificaciones internas.

9. ¿Cómo percibe la disposición del personal de la empresa ante el uso de nuevas tecnologías o sistemas digitales para este tipo de procesos?

En general, hay buena disposición, aunque algunos colaboradores pueden necesitar una pequeña curva de aprendizaje. Sin embargo, todos reconocen que el proceso actual es desgastante y están abiertos a un sistema que facilite el trabajo.

10. ¿Cuáles recomendaciones haría para garantizar una implementación exitosa del nuevo sistema web en la empresa?

Recomiendo capacitar al personal antes de lanzarlo, realizar pruebas piloto con casos reales, mantener soporte técnico durante las primeras semanas, documentar procesos y responsabilidades y escuchar la retroalimentación del personal para mejorar el sistema.

Apéndice E. Entrevista semiestructurada aplicada al administrador de la empresa

Organización: All Sport Nutrition.

Nombre del entrevistado: Minor Alfaro Rodríguez.

Cargo: administrador.

Preguntas:

1. ¿Con qué frecuencia se presentan errores o inconsistencias en los cálculos de la planilla?

De forma frecuente. Casi todos los meses tenemos que corregir algo: horas mal registradas, permisos que no se descontaron bien, incapacidades mal calculadas o deducciones que no se reflejaron correctamente. El uso de múltiples Excel nos obliga a revisar todo varias veces.

2. ¿Cuáles considera que son las principales limitaciones del sistema o método actual para gestionar la nómina?

Las principales limitaciones son el registro manual de asistencia, la falta de integración entre horas extra, permisos e incapacidades, la necesidad de reescribir la información en varias hojas, la dificultad para mantener actualizadas las tablas de deducciones y el riesgo de pérdida de información por archivos mal guardados o sobrescritos; todo esto genera atrasos y una alta carga administrativa.

3. ¿Qué tipo de información relacionada con la nómina considera más importante mantener actualizada y accesible?

Para mí es clave tener actualizados la asistencia diaria, las horas extra, los permisos e incapacidades, los porcentajes de deducciones, los cambios salariales y el historial de planillas y pagos; cuando esta información no está al día, la planilla termina atrasándose o generando errores.

4. ¿De qué forma se resguarda actualmente la información confidencial de las personas colaboradoras (salarios, deducciones, datos personales)?

En este momento, la información se guarda principalmente en Excel dentro de carpetas de la computadora. Algunas tienen contraseña, pero otras no. En ocasiones también se comparten documentos por correo o WhatsApp, lo cual no es seguro y deja expuestos datos sensibles de los colaboradores.

5. ¿Cuán sencillo o complicado resulta el proceso de generar los reportes de pago y deducciones legales (CCSS, INS, Hacienda)?

Es un proceso bastante complicado porque se debe revisar cada cálculo manualmente, verificar que los porcentajes estén correctos y luego trasladar la información a los formularios o plataformas correspondientes. Si no se tiene todo el registro claro desde el inicio del mes, se vuelve todavía más difícil.

6. Desde su experiencia ¿cómo afecta el sistema actual al tiempo de trabajo y a la productividad del Área Administrativa?

Nos afecta muchísimo. Preparar la planilla, revisar horas, confirmar permisos e incapacidades consume varias horas que podrían utilizarse en tareas más estratégicas. Al depender de Excel, se requiere revisar todo por duplicado para evitar errores, lo que baja la productividad.

7. ¿Cuáles beneficios considera que aportaría la implementación de un sistema web automatizado para la gestión de nómina?

Un sistema web nos aportaría mayor precisión en los cálculos, menos tiempo invertido en tareas repetitivas, un registro único, claro y actualizado para todo el personal, menor carga administrativa, reportes automáticos, transparencia con los colaboradores y seguridad de la información; en pocas palabras, profesionalizaría el proceso.

8. ¿Cuáles características o funcionalidades considera esenciales que debe incluir dicho sistema web (por ejemplo, acceso seguro, historial de pagos, reportes automáticos, interfaz sencilla, etc.)?

Para que realmente nos ayude, debería contar con un registro de asistencia y horas extra, control de permisos e incapacidades, cálculo automático de deducciones y montos salariales, perfiles con accesos diferenciados para R.R. H.H., jefaturas y colaboradores, un historial completo de pagos, reportes automáticos para CCSS, INS y Hacienda, notificaciones internas sobre aprobaciones o rechazos y un respaldo seguro en la nube.

9. ¿Cómo percibe la disposición del personal de la empresa ante el uso de nuevas tecnologías o sistemas digitales para este tipo de procesos?

En general, el personal está abierto a usar un sistema siempre y cuando sea sencillo. Algunos pueden tener menos experiencia con herramientas digitales, pero si se les capacita, estoy seguro de que se adaptarán sin problema.

10. ¿Cuáles recomendaciones haría para garantizar una implementación exitosa del nuevo sistema web en la empresa?

Recomiendo capacitar a todos los colaboradores y jefaturas, establecer un periodo de prueba en el que se comparen los resultados del sistema con la planilla manual, documentar cada proceso y cada módulo, mantener soporte técnico inicial y realizar una retroalimentación constante para mejorar el sistema.