

**UNIVERSIDAD INTERNACIONAL  
DE LAS AMÉRICAS**

**Escuela de Ingeniería en Sistemas de la Información**

**SISTEMA WEB PARA LA GESTIÓN DEL RECURSO  
HUMANO EN LA EMPRESA TENDENZA &  
PUERTALUM UBICADA EN ASERRÍ, SAN JOSÉ,  
COSTA RICA**

Proyecto de graduación  
Para optar por el grado de Bachillerato en Ingeniería en Sistemas de la  
Información

**MIGUEL AGUILAR RAMÍREZ  
AUTOR**

**FABIAN RODRIGUEZ SIBAJA  
TUTOR**

**San José, Costa Rica**

**ABRIL 2026**

## DEDICATORIA

El esfuerzo, el sacrificio y cada hora invertida en este proceso son el reflejo de un sueño que nació de repente y que hoy, con mucho orgullo, se convierte en realidad. Hubo momentos difíciles, noches largas y días agotadores, pero cada uno de ellos valió la pena. Por eso, dedico este trabajo de grado con todo mi cariño a las personas que hicieron posible este logro y que estuvieron a mi lado en cada etapa de este camino.

A mi amada pareja Kimberly, el pilar más importante de este proceso. Gracias por su apoyo incondicional, su inmensa paciencia y por nunca dejar de creer en mí, incluso en los momentos en que yo mismo dudé. Gracias por el tiempo que cediste, por las responsabilidades que asumiste sola para que yo pudiera avanzar, y por estar presente con una sonrisa cuando más lo necesitaba. Este logro es tan tuyo como mío.

A mis dos hijos, mi mayor motivación y la razón más poderosa para seguir adelante. A mi hijo mayor, que ya recorre su propio camino de superación persiguiendo sus metas con dedicación y esfuerzo, espero que este logro le sirva de ejemplo de que los sueños, por difíciles que parezcan, siempre son alcanzables con trabajo y constancia. Y a mi hija, que llegó a nuestras vidas en medio de este proceso llenando cada día difícil de alegría, de luz y de una nueva fuerza que no sabía que tenía. Su llegada fue el impulso más grande que pude recibir.

A mi padre, quien estuvo presente en los momentos en que más lo necesité. Su compañía y respaldo en los instantes clave de este proceso fueron un sostén invaluable, y le agradezco profundamente por nunca haberme dejado solo cuando las cosas se pusieron difíciles. Y a mi madre, quien con sus palabras siempre supo darme exactamente el consejo que necesitaba en el momento indicado, cambiando mi perspectiva cuando sentía que no encontraba el camino. Su sabiduría y apoyo en tantas ocasiones me ayudaron a retomar el rumbo y a seguir adelante con más claridad y convicción.

A mi abuela, que, aunque ya no está físicamente con nosotros, siempre estuvo presente en mi corazón a lo largo de este camino. Ella soñó con verme terminar una carrera y convertirme en un profesional exitoso, y hoy puedo decirle que lo logré.

## AGRADECIMIENTOS

En primera instancia, mi profundo agradecimiento va para Dios, por ser quien me ha dado conocimiento a lo largo de los años y por permitirme plasmar en este proyecto todo lo aprendido, dándome la fortaleza necesaria para no rendirme y la claridad para dar lo mejor de mí en cada etapa de este proceso.

Quiero dar gracias a todos mis profesores y profesoras que formaron parte de mi educación en sus diferentes niveles, por su dedicación y perseverancia en una labor tan hermosa, convirtiéndose en pilares fundamentales que trascienden de generación en generación.

Un agradecimiento muy especial a la directora de Carrera, quien en el momento más retador de este proceso nos reunió y nos regaló palabras que marcaron un antes y un después. Con honestidad y calidez nos advirtió que habría momentos de mucha presión, instantes en los que sentiríamos el impulso de dejarlo todo, pero nos pidió con el corazón que no claudicáramos, que no nos rindiéramos. Nos habló de carácter, de actitud y de emociones, recordándonos que este proceso era una serie de retos donde íbamos a validar quiénes somos como profesionales y como personas. Nos aseguró que estaría siempre disponible para apoyarnos, dejándonos claro que no seríamos ni los primeros ni los últimos en pasar por dificultades. Lo que me quedó grabado fue su convicción genuina en cada uno de nosotros, su invitación a dar la milla extra y a esforzarnos más allá de lo cómodo. Gracias a esas palabras, en los momentos más difíciles de este proyecto recordé que no podía rendirme, y hoy este trabajo es también el resultado de este aliento que llegó justo cuando más se necesitaba.

A mi esposa, a mis hijos, a mi padre y a mi suegra, quienes desde sus diferentes roles fueron un apoyo fundamental a lo largo de todo este camino. Gracias por cada palabra de aliento, por cada consejo que cambió mi perspectiva y por estar presentes cuando más lo necesité. Un agradecimiento especial a mi suegra, quien de una u otra manera contribuyó con el cuidado de mi hija, permitiéndome dedicar el tiempo necesario para sacar adelante este proyecto.

## CONTENIDO

<b>DEDICATORIA</b> .....	2
<b>AGRADECIMIENTOS</b> .....	3
<b>CARTA DEL TRIBUNAL EXAMINADOR</b> .....	4
<b>CARTA DE APROBACIÓN DEL TUTOR</b> .....	6
<b>DECLARACION JURADA DEL ESTUDIANTE</b> .....	14
<b>CARTA DE SOLICITUD DE DEFENSA</b> .....	15
<b>CONTENIDO</b> .....	16
<b>TABLAS</b> .....	22
<b>FIGURAS</b> .....	26
<b>CAPÍTULO I: INTRODUCCIÓN</b> .....	30
Planeamiento del Problema .....	30
Problemas por resolver en la empresa.....	30
<b>Objetivos</b> .....	31
Objetivo General .....	31
Objetivos Específicos.....	31
<b>Justificación</b> .....	32
<b>Viabilidad</b> .....	32
Viabilidad técnica .....	33
Viabilidad operativa .....	34
Viabilidad económica .....	35
Viabilidad legal .....	38
<b>Proyecciones</b> .....	39
Alcance funcional .....	40
Alcance metodológico.....	42
Alcance tecnológico .....	43
Back End.....	43
Front End.....	44
Arquitectura .....	44

	17
Framework .....	44
Base de datos .....	44
Autenticación .....	45
<b>CAPÍTULO II: MARCO REFERENCIAL .....</b>	<b>47</b>
Prototipado de Software en la Optimización Empresarial .....	47
Función en Procesos Internos. ....	47
Prototipado rápido como estrategia de validación. ....	47
Etapas iterativas del desarrollo de prototipos.....	47
Evaluación de usabilidad en prototipos.....	48
Aplicaciones Web como Herramienta Corporativa .....	48
Aplicaciones web en sectores empresariales. ....	48
Beneficios operativos en áreas administrativas. ....	48
Limitaciones técnicas y desafíos de implementación. ....	49
Relación entre software y aplicaciones web. ....	49
Tecnologías Utilizadas .....	49
Backend: Laravel como base Estructural.....	49
Front End: Blade, jQuery, Bootstrap y Velzon. ....	50
Interacción entre Front End y Back End. ....	50
Arquitectura del Sistema y Control de Accesos.....	50
Patrón MVC como modelo de desarrollo. ....	50
Gestión de datos con MySQL y Workbench. ....	51
Control de roles y permisos con Laravel Spatie.....	51
Frameworks en el Desarrollo Web Empresarial .....	52
Función estructural de los frameworks. ....	52
Ventajas en seguridad, escalabilidad y mantenimiento. ....	52
Aplicación en sistemas de gestión de recursos humanos.....	52
Laravel como Framework Estratégico .....	53
Estructura basada en Modelo-Vista-Controlador.....	53
Herramientas integradas para desarrollo ágil. ....	53
Seguridad y control de roles con librerías externas.....	53
Ecosistema y comunidad de Laravel.....	54

Diseño Responsivo e Interacción de Usuario .....	54
Adaptabilidad de interfaces en múltiples dispositivos. ....	54
Bootstrap como herramienta de diseño responsivo. ....	55
Interactividad en tiempo real con Blade y jQuery. ....	55
Impacto en la experiencia del usuario.....	55
Bases de Datos y Gestión de Información .....	55
Rol de la base de datos en sistemas administrativos. ....	56
Uso de MySQL y Workbench en el proyecto. ....	56
Integridad de datos y validaciones estructurales. ....	56
Normalización y Eficiencia de Datos.....	57
Importancia de la normalización en bases de datos.....	57
Relación entre eficiencia y estructura de datos. ....	57
Validación de claves y compatibilidad referencial.....	57
Automatización Mediante Procedimientos Almacenados .....	58
Concepto y función de los procedimientos almacenados. ....	58
Aplicación en módulos de Recursos Humanos. ....	58
Ventajas frente a consultas tradicionales.....	59
Seguridad de la Información y Cifrado de Contraseñas.....	59
Importancia de la seguridad en sistemas administrativos.....	59
Cifrado de contraseñas con algoritmo con algoritmos hash. ....	59
Gestión de roles y accesos diferenciados. ....	60
Prevención de vulnerabilidades comunes.....	60
Tendencias en la Digitalización de Recursos Humanos. ....	60
Transformación digital en procesos laborales. ....	60
Plataformas integradas para gestión del talento. ....	61
Interoperabilidad y Escalabilidad en Sistemas Empresariales.....	61
Concepto de interoperabilidad en entornos digitales. ....	61
Escalabilidad como principio de sostenibilidad tecnológica.....	62
Impacto del Software en la Toma de Decisiones Administrativas .....	62
Digitalización como soporte para decisiones estratégicas. ....	62
Generación de reportes dinámicos y visuales. ....	63

	19
Evaluación de indicadores clave de desempeño (KPI).....	63
Cierre del Capítulo.....	63
<b>CAPITULO III: MARCO METODOLÓGICO .....</b>	<b>65</b>
<b>Enfoques de Investigación.....</b>	<b>65</b>
Enfoque Cuantitativo .....	65
Enfoque Cualitativo.....	65
Enfoque Mixto.....	65
Enfoque de Investigación Seleccionado .....	66
<b>Tipos de Investigación .....</b>	<b>66</b>
Investigación Exploratoria .....	67
Investigación Descriptiva.....	67
Investigación Explicativa .....	67
Tipo de Investigación Seleccionado.....	67
<b>Fuentes de Información .....</b>	<b>68</b>
Fuente de Información Primaria .....	68
Fuente de Información Secundaria .....	69
Fuente de Información Terciaria .....	69
<b>Variables .....</b>	<b>69</b>
Tipos de Variables Consideradas.....	70
Variables conceptuales.....	70
Variables operacionales. ....	70
Variables instrumentales.....	70
<b>Instrumentos de Recolección de Datos.....</b>	<b>73</b>
Entrevista .....	73
Observación .....	73
Documentos de Archivo.....	74
Procesos de Recolección y Análisis de Datos.....	74
<b>CAPITULO IV: ANÁLISIS DE RESULTADOS .....</b>	<b>75</b>

	20
Análisis Resultante del Proceso de Observación y Revisión Documental .....	80
<b>CAPITULO V: PROPUESTA</b> .....	<b>82</b>
<b>Análisis</b> .....	<b>82</b>
Análisis Detallado del Software .....	82
Módulo de Gestión de Horas Extra. ....	82
Módulo de Gestión de Vacaciones. ....	83
Módulo de Gestión de Asistencia. ....	83
Módulo de Gestión de Nómina. ....	83
Módulo de Gestión de Permisos. ....	83
Módulo de Evaluación de Desempeño. ....	84
Módulo de Gestión de Incapacidades. ....	84
Módulo de Gestión de Liquidaciones. ....	84
Módulo de Cálculo de Aguinaldo. ....	84
Módulo de Mantenimientos. ....	84
Módulo de Consultas. ....	85
Módulo de Reportes.....	85
Módulo de Seguridad.....	85
Análisis Detallado del Software .....	86
Análisis Detallado del Hardware .....	86
Análisis Detallado de Telecomunicaciones.....	87
Análisis Detallado de Herramientas y licencias con costos y capacidades. ....	88
Análisis Detallado del personal requerido para el uso del sistema. ....	88
Análisis del Conocimiento del personal requerido para el mantenimiento del sistema.....	89
Diagrama de casos de uso. ....	89
Casos de Uso .....	91
Módulo Gestión de Solicitudes RRHH.....	91
Módulo de Consultas. ....	102
Módulo de Reportes.....	110
Módulo de Seguridad.....	116
Módulo Gestión Financiera .....	122

Módulo de Mantenimientos .....	127
Arquitectura del Sistema.....	138
Arquitectura del Software.....	140
Capa de Modelo. ....	140
Capa de Controlador. ....	141
Capa de Vista.....	141
Diseño de Interfaz .....	142
Diseño de la base de datos.....	152
Diccionario de la base de datos.....	153
Diseño de procesos .....	165
Diseño de salidas.....	181
Diagrama UML.....	186
<b>Programación</b> .....	210
Gestión Colaboradores.....	210
Gestión Nomina .....	219
<b>Pruebas</b> .....	237
<b>CAPITULO VI: CONCLUSIONES Y RECOMENDACIONES</b> .....	246
Conclusiones .....	246
Recomendaciones .....	248
<b>Apéndice</b> .....	256
Apéndice A: Guía de Entrevista Aplicada al Área de Recursos Humanos .....	256
Apéndice B: Guía de Entrevista Aplicada al Área de Recursos Humanos .....	258

## TABLAS

Tabla 1 <i>Infraestructura para la implementación del prototipo.</i> .....	33
Tabla 2 <i>Herramientas de software para la elaboración del prototipo.</i> .....	34
Tabla 3 <i>Costo de Hardware.</i> .....	35
Tabla 4 <i>Hardware empresarial.</i> .....	36
Tabla 5 <i>Costo Software.</i> .....	36
Tabla 6 <i>Costo de Infraestructura digital.</i> .....	37
Tabla 7 <i>Costo de Personal</i> .....	37
Tabla 8 <i>Módulos Gestión Recursos Humanos Tendenzza &amp; puertAlum.</i> .....	40
Tabla 9 <i>Componentes de Desarrollo.</i> .....	45
Tabla 10 <i>Cuadro de variables.</i> .....	71
Tabla 11 <i>Caso de uso 1.</i> .....	91
Tabla 12 <i>Caso de uso 2.</i> .....	92
Tabla 13 <i>Caso de uso 3.</i> .....	93
Tabla 14 <i>Caso de uso 4.</i> .....	94
Tabla 15 <i>Caso de uso 5.</i> .....	96
Tabla 16 <i>Caso de uso 6.</i> .....	97
Tabla 17 <i>Caso de uso 7.</i> .....	98
Tabla 18 <i>Caso de uso 8.</i> .....	99
Tabla 19 <i>Caso de uso 9.</i> .....	101
Tabla 20 <i>Caso de uso 10.</i> .....	103
Tabla 21 <i>Caso de uso 11.</i> .....	104
Tabla 22 <i>Caso de uso 12.</i> .....	105
Tabla 23 <i>Caso de uso 13.</i> .....	106
Tabla 24 <i>Caso de uso 14.</i> .....	107
Tabla 25 <i>Caso de uso 15.</i> .....	108
Tabla 26 <i>Caso de uso 16.</i> .....	110
Tabla 27 <i>Caso de uso 17.</i> .....	111
Tabla 28 <i>Caso de uso 18.</i> .....	112
Tabla 29 <i>Caso de uso 19.</i> .....	113

Tabla 30 <i>Caso de uso 20.</i> .....	115
Tabla 31 <i>Caso de uso 21.</i> .....	116
Tabla 32 <i>Caso de uso 22.</i> .....	117
Tabla 33 <i>Caso de uso 23.</i> .....	118
Tabla 34 <i>Caso de uso 24.</i> .....	119
Tabla 35 <i>Caso de uso 25.</i> .....	121
Tabla 36 <i>Caso de uso 26.</i> .....	122
Tabla 37 <i>Caso de uso 27.</i> .....	124
Tabla 38 <i>Caso de uso 28.</i> .....	126
Tabla 39 <i>Caso de uso 29.</i> .....	127
Tabla 40 <i>Caso de uso 30.</i> .....	129
Tabla 41 <i>Caso de uso 31.</i> .....	131
Tabla 42 <i>Caso de uso 32.</i> .....	133
Tabla 43 <i>Caso de uso 33.</i> .....	134
Tabla 44 <i>Caso de uso 34.</i> .....	135
Tabla 45 <i>Caso de uso 35.</i> .....	137
Tabla 46 <i>Diccionario de datos - Adelanto Salario</i> .....	153
Tabla 47 <i>Diccionario de datos - Aguinaldo</i> .....	153
Tabla 48 <i>Diccionario de datos - Asistencia</i> .....	153
Tabla 49 <i>Diccionario de datos - Catalogo_canton</i> .....	154
Tabla 50 <i>Diccionario de datos - Catalogo_categoria</i> .....	154
Tabla 51 <b><i>Diccionario de datos - Catalogo_Credito_Fiscal_Hijo_Conyugue</i></b> .....	154
Tabla 52 <i>Diccionario de datos - Catalogo_distrito</i> .....	154
Tabla 53 <i>Diccionario de datos - Catalogo_estado_asistencia</i> .....	155
Tabla 54 <i>Diccionario de datos - Catalogo_estado_evaluacion</i> .....	155
Tabla 55 <i>Diccionario de datos - Catalogo_estado_liquidacion</i> .....	155
Tabla 56 <i>Diccionario de datos - Catalogo_estado_solicitud</i> .....	155
Tabla 57 <i>Diccionario de datos – Catalogo_genero</i> .....	155
Tabla 58 <i>Diccionario de datos - Catalogo_motivo_salida</i> .....	156
Tabla 59 <i>Diccionario de datos - Catalogo_nivel</i> .....	156
Tabla 60 <i>Diccionario de datos - Catalogo_ocupacion</i> .....	156

Tabla 61	<i>Diccionario de datos - Catalogo_pais</i>	156
Tabla 62	<i>Diccionario de datos - Catalogo_provincia</i>	156
Tabla 63	<i>Diccionario de datos - Catalogo_tasa_ccss</i>	157
Tabla 64	<i>Diccionario de datos - Catalogo_tipo_concepto</i>	157
Tabla 65	<i>Diccionario de datos - Catalogo_tipo_correo</i>	157
Tabla 66	<i>Diccionario de datos - Catalogo_tipo_direccion</i>	157
Tabla 67	<i>Diccionario de datos - Catalogo_tipo_identificacion</i>	157
Tabla 68	<i>Diccionario de datos - Catalogo_tipo_incapacidad</i>	158
Tabla 69	<i>Diccionario de datos - Catalogo_tipo_permiso</i>	158
Tabla 70	<i>Diccionario de datos - Catalogo_tipo_telefono</i>	158
Tabla 71	<i>Diccionario de datos - Catalogo_tramo_renta</i>	158
Tabla 72	<i>Diccionario de datos - Cliente</i>	159
Tabla 73	<i>Diccionario de datos - Colaborador</i>	159
Tabla 74	<i>Diccionario de datos - Colaborador_Credito_Fiscal_Hijo_Conyugue</i>	159
Tabla 75	<i>Diccionario de datos - Colaborador_Información_Familiar</i>	159
Tabla 76	<i>Diccionario de datos - Correo</i>	160
Tabla 77	<i>Diccionario de datos - Direccion</i>	160
Tabla 78	<i>Diccionario de datos - Evaluacion_desempeño</i>	160
Tabla 79	<i>Diccionario de datos - Horas_extra</i>	161
Tabla 80	<i>Diccionario de datos - Incapacidades</i>	161
Tabla 81	<i>Diccionario de datos - Liquidación</i>	161
Tabla 82	<i>Diccionario de datos - Nomina</i>	162
Tabla 83	<i>Diccionario de datos - Nomina_detalle</i>	162
Tabla 84	<i>Diccionario de datos - Permisos_por_usuario</i>	162
Tabla 85	<i>Diccionario de datos - Permisos</i>	163
Tabla 86	<i>Diccionario de datos - Permisos_disponibles</i>	163
Tabla 87	<i>Diccionario de datos - Persona</i>	163
Tabla 88	<i>Diccionario de datos - Saldo_vacaciones</i>	164
Tabla 89	<i>Diccionario de datos - Telefono</i>	164
Tabla 90	<i>Diccionario de datos - Ubicación_Autorizada</i>	164
Tabla 91	<i>Diccionario de datos - Usuario</i>	164

Tabla 92 <i>Diccionario de datos - Vacaciones</i> .....	165
Tabla 93 <i>Caso de prueba - Inicio de sesión</i> .....	238
Tabla 94 <i>Caso de prueba - Registro de vacaciones</i> .....	239
Tabla 95 <i>Caso de prueba - Registro de permisos</i> .....	240
Tabla 96 <i>Caso de prueba - Registro de marca</i> .....	241
Tabla 97 <i>Caso de prueba - Cálculo de aguinaldo</i> .....	242
Tabla 98 <i>Caso de prueba - Procesamiento de nómina</i> .....	243
Tabla 99 <i>Caso de prueba - Liquidación</i> .....	244

## FIGURAS

Figura 1 <i>Diagrama Casos de Uso</i> .....	90
Figura 2 <i>Arquitectura del Sistema</i> .....	140
Figura 3 <i>Arquitectura del Software</i> .....	142
Figura 4 <i>Inicio de Sesión</i> .....	143
Figura 5 <i>Vista principal del gerente general</i> .....	144
Figura 6 <i>Vista principal de jefe de RH</i> .....	145
Figura 7 <i>Vista principal del empleado</i> .....	146
Figura 8 <i>Agregar Colaborador</i> .....	146
Figura 9 <i>Agregar datos laborales</i> .....	147
Figura 10 <i>Agregar usuario</i> .....	148
Figura 11 <i>Lista de colaboradores</i> .....	149
Figura 12 <i>Evaluación de desempeño</i> .....	150
Figura 13 <i>Realizar evaluación</i> .....	150
Figura 14 <i>Solicitar vacaciones</i> .....	151
Figura 15 <i>Diseño Base de Datos</i> .....	152
Figura 16 <i>Diseño de procesos - Autenticación</i> .....	166
Figura 17 <i>Diseño de procesos - Agregar Colaborador</i> .....	167
Figura 18 <i>Diseño de procesos – Editar Colaborador</i> .....	168
Figura 19 <i>Diseño de procesos – Vacaciones</i> .....	169
Figura 20 <i>Diseño de procesos – Horas Extra</i> .....	170
Figura 21 <i>Diseño de procesos - Permisos</i> .....	171
Figura 22 <i>Diseño de procesos - Incapacidades</i> .....	172
Figura 23 <i>Diseño de procesos - Asistencia</i> .....	173
Figura 24 <i>Diseño de procesos – Procesar Nómina</i> .....	174
Figura 25 <i>Diseño de procesos – Gestión de Nómina</i> .....	175
Figura 26 <i>Diseño de procesos – Evaluación Desempeño</i> .....	176
Figura 27 <i>Diseño de procesos – Ubicaciones Autorizadas</i> .....	177
Figura 28.....	178
Figura 29 <i>Diseño de procesos – Aguinaldos</i> .....	179

Figura 30 <i>Diseño de procesos – Liquidación</i> .....	180
Figura 31 <i>Diseño de Salidas - Colaboradores</i> .....	181
Figura 32 <i>Diseño de Salidas - Evaluaciones</i> .....	182
Figura 33 <i>Diseño de salidas - Vacaciones</i> .....	183
Figura 34 <i>Diseño de salidas - Horas Extra</i> .....	184
Figura 35 <i>Diseño de salidas - Solicitud de Permisos</i> .....	185
En la siguiente figura, se muestra el diagrama de clases en al que se trabaja para esta aplicación. Figura 36 .....	186
Figura 37 <b>Diagrama de Clases</b> .....	187
Figura 38.....	188
Figura 39 <i>Diagrama de Secuencia - Autenticación</i> .....	189
Figura 40 <i>Diagrama de Secuencia - Crear Colaborador</i> .....	190
Figura 41 <b>Diagrama de Secuencia - Editar Colaborador</b> .....	191
Figura 42 <i>Diagrama de Secuencia - Eliminar Colaborador</i> .....	192
Figura 43 <i>Diagrama de Secuencia - Solicitar Vacaciones</i> .....	193
Figura 44 <i>Diagrama de Secuencia - Aprobar / Rechazar Vacaciones</i> .....	194
Figura 45 <i>Diagrama de Secuencia - Solicitar Horas Extra</i> .....	195
Figura 46 <i>Diagrama de Secuencia - Aprobar / Rechazar Horas Extra</i> .....	196
Figura 47 <i>Diagrama de Secuencia - Solicitar Permisos</i> .....	197
Figura 48 <i>Diagrama de Secuencia - Aprobar / Rechazar Permisos</i> .....	198
Figura 49 <i>Diagrama de Secuencia - Incapacidades</i> .....	199
Figura 50 <i>Diagrama de Secuencia - Asistencia</i> .....	200
Figura 51 <i>Diagrama de Secuencia - Procesar Nomina</i> .....	201
Figura 52 <i>Diagrama de Secuencia - Correo Nomina</i> .....	202
Figura 53 <i>Diagrama de Secuencia - Pagar Nomina</i> .....	203
Figura 54 <i>Diagrama de Secuencia - Evaluación Desempeño</i> .....	204
Figura 55 <i>Diagrama de Secuencia - Gestión de Usuarios</i> .....	205
Figura 56 <i>Diagrama de Secuencia - Ubicaciones Autorizadas</i> .....	206
Figura 57 <i>Diagrama de Secuencia - Procesar Liquidación</i> .....	207
Figura 58 <i>Diagrama de Secuencia - Procesar Liquidación</i> .....	207
Figura 59 <i>Diagrama de Secuencia - Procesar Liquidación</i> Fuente: Elaboración Propia (2026) .....	208

Figura 60 <i>Diagrama de Secuencia - Procesar Aguinaldo</i> .....	209
Figura 61 <i>Diagrama de Secuencia - Procesar Aguinaldo</i> .....	209
Figura 62 <i>Lista de Colaboradores</i> .....	210
Figura 63 <i>Formulario de Registro</i> .....	211
Figura 64 <i>Guardar Nuevo Colaborador</i> .....	212
Figura 65 <i>Guardar Nuevo Colaborador</i> .....	213
Figura 66 <i>Detalle Colaborador</i> .....	214
Figura 67 <i>Actualizar Colaborador</i> .....	215
Figura 68 <i>Actualizar Colaborador</i> .....	216
Figura 69 <i>Actualizar Colaborador</i> .....	216
Figura 70 <i>Actualizar Colaborador</i> .....	217
Figura 71 <i>Borrado Lógico y AJAX</i> .....	218
Figura 72 <i>Lista Nómina</i> .....	219
Figura 73 <i>Boleta de Pago</i> .....	220
Figura 74 <i>Generar Nómina del Mes</i> .....	221
Figura 75 <i>Generar Nómina del Mes</i> .....	222
Figura 76 <i>Generar Nómina del Mes</i> .....	223
Figura 77 <i>Generar Nómina del Mes</i> .....	224
Figura 78 <i>Generar Nomina del Mes</i> .....	225
Figura 79 <i>Control de Pago - Marcar Pagada</i> .....	225
Figura 80 <i>Control de Pago - Marcar Todas Pagadas</i> .....	226
Figura 81 <i>Distribución de Boletas - Enviar Correo</i> .....	227
Figura 82 <i>Distribución de Boletas - Enviar Todos los Correos</i> .....	228
Figura 83.....	229
Figura 84 <i>Exportar a Excel</i> .....	230
Figura 85 <i>Exportar a Excel</i> .....	231
Figura 86 <i>Exportar a Excel</i> .....	232
Figura 87 <i>Exportar a Excel</i> .....	233
Figura 88 <i>Exportar a Excel</i> .....	234
Figura 89.....	234
Figura 90 <i>Wizard - Vista Previa</i> .....	235

Figura 91 <i>Wizard - Vista Previa</i> .....	236
Figura 92 <i>Wizard - Vista Previa</i> .....	237

## **CAPÍTULO I: INTRODUCCIÓN**

### **Planeamiento del Problema**

Tendenzza & puertAlum. Es una empresa de referencia en el sector del inmobiliario y productos de vidrio de alta gama, ofreciendo soluciones que combinan diseño, funcionalidad e innovación para espacios comerciales, hogares y oficinas. Desde su creación en 2010, la empresa ha sabido evolucionar, incorporando avances tecnológicos y adaptándose a las tendencias del mercado, ya que transforma cualquier espacio de oficina, hogares o locales comerciales en entornos inteligentes, implementando sistemas de iluminación, alarmas y otras soluciones tecnológicas de punta, creando ambientes modernos y funcionales.

### **Problemas por resolver en la empresa**

Actualmente se registran las horas extra de los colaboradores de manera manual en hojas de papel y en ocasiones desde ubicaciones remotas, lo que causa confusión y errores en los cálculos, todo esto se traduce en pagos incorrectos y en ocasiones en descontento por parte del equipo.

El método manual que se utiliza para calcular y asignar las vacaciones genera discrepancias en el saldo de días disponibles. Ocasiona malentendidos y problemas tanto en la planificación de vacaciones como en la administración interna.

El registro se realiza de forma manual, lo que provoca muchas veces olvidar marcar ya sea la entrada o salida, también dificulta el seguimiento de la puntualidad y ausencias de manera efectiva, sobre todo cuando se registra desde ubicaciones remotas.

Todo el proceso manual de generación de nómina ha provocado inconsistencias en el cálculo de salarios, deducciones y bonificaciones. Estas discrepancias se han reflejado en retrasos en los pagos y en situaciones de desconfianza por parte de los colaboradores.

Actualmente, las solicitudes de permiso o licencias se gestionan mediante formularios en papel o correos electrónicos, lo que retrasa la aprobación y dificulta el seguimiento de cada solicitud, generando confusión y pérdida de información.

La falta de un registro formal y estructurado del desempeño de los colaboradores dificulta evaluar su rendimiento de manera objetiva. Esto complica la toma de decisiones para ascensos, aumentos salariales o necesidades de capacitación.

El proceso para registrar incapacidades se realiza de forma manual, muchas veces también a través de papeleo o correos, lo que ocasiona demoras en la verificación y registro de días. Al final genera confusión sobre el saldo de días de incapacidad y afecta el cálculo correcto de la nómina.

El cálculo de las liquidaciones se realiza de manera manual, lo que puede generar errores en la determinación de montos por indemnización, vacaciones pendientes, bonificaciones y otros conceptos. Estos errores afectan la precisión de los pagos y pueden generar inconformidades entre los colaboradores, además de retrasos administrativos en Recursos Humanos.

El cálculo de aguinaldo se realiza de forma manual, lo que genera errores en los montos a pagar. Además, muchas veces las ausencias, incapacidades y días no laborados no siempre se registran correctamente, lo que ocasiona pagos incorrectos y reclamos por parte de los colaboradores. Esto retrasa el proceso, afectando la planificación financiera de la empresa.

## **Objetivos**

### **Objetivo General**

Desarrollar un sistema web de gestión del pago de planilla para la empresa Tendenzza & puerTalum, mediante el uso de Visual Studio Code con los lenguajes de programación PHP y JavaScript, el framework Laravel y Laravel Spatie, la biblioteca de JavaScript jQuery y la base de datos MySQL.

### **Objetivos Específicos**

Analizar los requerimientos funcionales y no funcionales del sistema web de gestión del pago de planilla, asegurando que cumpla con las necesidades de la empresa.

Diseñar la arquitectura general del sistema, lo que incluye la definición de la base de datos con sus tablas y relaciones, así como la planificación de la aplicación que se programará.

Programar los diferentes módulos del sistema con PHP y el framework Laravel para el backend, integrando jQuery, Bootstrap y Velzon.

Probar los módulos de manera independiente, realizando pruebas de integración entre el backend en Laravel y frontend basado en Blade, garantizando la correcta funcionabilidad del sistema y la comunicación entre los componentes del sistema.

### **Justificación**

Para que se lleve a cabo esta investigación, es fundamental considerar la necesidad de modernizar los procesos del área de Recursos Humanos, así como el potencial de mejora en la eficiencia, la productividad y la competitividad de Tendenzza & puertAlum mediante la implementación de soluciones tecnológicas innovadoras.

La automatización de tareas administrativas, el uso de plataformas digitales para la gestión del talento y la integración de herramientas de análisis de datos permitirán optimizar los flujos de trabajo, reducir errores humanos y facilitar la toma de decisiones estratégicas.

Este proyecto de investigación y desarrollo busca mejorar la eficiencia operativa de la empresa, garantizar el cumplimiento de normativas laborales vigentes y aumentar la satisfacción tanto de los colaboradores como de los líderes de cada departamento.

Además, se espera que la transformación digital del área de Recursos Humanos contribuya a fortalecer la cultura organizacional, mejorar la experiencia del empleado y posicionar a Tendenzza & puertAlum como una empresa moderna, ágil y orientada al crecimiento sostenible.

### **Viabilidad**

Para que este prototipo sea viable, debe presentar una mejora concreta en los procesos actuales de gestión de Recursos Humanos, especialmente en el manejo de planillas, asistencia y permisos. La empresa cuenta con los recursos técnicos necesarios, incluyendo cinco equipos portátiles disponibles, así como la disposición para invertir en infraestructura digital básica. Según Lifeder (2023), la viabilidad de una investigación implica que esta pueda

ejecutarse en la práctica con los recursos disponibles, lo que este caso se cumple satisfactoriamente.

### Viabilidad técnica

Para la elaboración del sistema se requiere una laptop HP RYZEN 5 con Windows 11 Home de sistema operativo, 16 GB de memoria RAM, un disco duro de 512 GB de almacenamiento. Este equipo, propiedad personal del estudiante universitario, está disponible para el desarrollo del proyecto y no requiere hardware adicional.

Adicionalmente, la empresa está dispuesta a realizar una inversión mínima en infraestructura digital para asegurar que el sistema esté disponible en línea. Esto abarca la compra de un dominio (t&pcr.com) y un servidor privado virtual (VPS) básico.

**Tabla 1**

#### *Infraestructura para la implementación del prototipo.*

Recurso	Especificaciones Técnicas
Dominio	Registro de dominio estándar para acceso público al sistema (t&pcr.com).
Servidor	RAM: 2 GB, Almacenamiento SSD: 30 GB, vCPU: 1 núcleo, Dirección IP: 1 IPv4 + IPv6 /80, Shared Uplink: 150 Mbps
Gestión de base de datos	MySQL 8.4, Workbench 8.0.43
Entorno de desarrollo	Visual Studio Code 1.104.2

Fuente: Elaboración Propia (2026)

Estas características son suficientes para alojar el prototipo funcional en un entorno seguro y accesible, garantizando un rendimiento adecuado para el número de usuarios previstos en la empresa.

Desde el punto de vista técnico, la creación del sistema es totalmente viable, dado que en Tendenzza & puertAlum todos los colaboradores cuentan con una computadora portátil laptops HP con procesador Intel Core i5 1335U, 16 GB de memoria RAM y disco duro sólido (SSD) de 256 GB, todas cuentan con Windows 11 Home como sistema operativo.

Esto posibilita la implementación sin tener que comprar nuevos dispositivos. Es importante señalar que no se necesita ninguna licencia adicional ni componentes físicos adicionales. En cuanto al Software se recomienda que la PC o laptop de los trabajadores tengan una versión 10 o superior de Windows.

**Tabla 2**

***Herramientas de software para la elaboración del prototipo.***

<b>Categoría</b>	<b>Herramienta</b>
Lenguajes y tecnologías	PHP 8.3, JavaScript, HTML 5, CSS 3
Frameworks y librerías	Laravel 12, jQuery, 3.7, Bootstrap 5.3, Velzon
Gestión de base de datos	MySQL 8.4, Workbench 8.0.43
Entorno de desarrollo	Visual Studio Code 1.104.2

Fuente: Elaboración Propia (2026)

Las herramientas mencionadas anteriormente son de licencia libre, excepto Velzon, que tiene un costo único de aproximadamente 25 dólares. Esta inversión se considera accesible y adecuada para los fines del proyecto.

**Viabilidad operativa**

La implementación del sistema no requiere conocimientos avanzados, sino únicamente nociones básicas sobre el uso de sistemas. Este sistema va a ser fácil de usar ya que incorpora interfaces de usuario intuitivas y accesibles, lo que facilita su manejo por parte de cualquier colaborador, independientemente de su nivel de experiencia tecnológica.

El personal de la empresa Tendenzza & puertAlum está calificado para el uso de este sistema, por lo tanto, no va a requerir de un entrenamiento extenso, sino de una inducción inicial que permita familiarizarse con sus funcionalidades. Es importante considerar que, la adopción de esta herramienta no implicará reducción del personal, ya que su propósito es optimizar procesos.

El prototipo representa una mejora significativa en la forma en que se gestionan las tareas relacionadas con Recursos Humanos. Permitirá automatizar solicitudes de permisos y otras funciones administrativas, contribuyendo a una gestión más eficiente, ordenada y estratégica dentro de la empresa.

### **Viabilidad económica**

Debido a que es un proyecto de graduación para poder optar el título de Bachillerato en Ingeniería en Sistemas de la Información, los resultados y el desarrollo de la propuesta son de manera totalmente gratuita para la empresa, por lo que el estudiante no recibirá ningún pago alguno.

Por otro lado, la seguridad física y lógica es esencial, se deben establecer acciones contra accesos no permitidos, ataques cibernéticos y pérdida de información. Esto abarca la gestión de accesos, cortafuegos, identificación de intrusiones y sistemas de recuperación y respaldo para desastres.

Finalmente, ante el incremento del uso de energía y la presión sobre el medio ambiente, las entidades optan por centros de datos ecológicos o sostenibles. Estos incluyen energías renovables, refrigeración eficaz y optimización de la utilización de recursos, disminuyendo el efecto en el medio ambiente y los gastos a largo plazo.

**Tabla 3**

#### ***Costo de Hardware.***

<b>Cantidad</b>	<b>Descripción</b>	<b>Valor Unitario</b>	<b>Valor total</b>
1	Laptop Ryzen 5 – 7535HS – 16,0 GB RAM Windows 11 Home Disco Duro 512 GB	€379.900	€379.900

Fuente: Costos extraídos del sitio web Compufax (2025)

El hardware mencionado anteriormente es de uso personal por parte del universitario para realizar el proyecto de graduación; por tal motivo, no hay que realizar un gasto adicional.

**Tabla 4*****Hardware empresarial.***

Cantidad	Descripción	Valor Unitario	Valor total
5	Laptop HP Intel Core i5, 16 GB RAM, Windows 11 Home, Disco Duro 256 GB	∅0	∅0

Fuente: Elaboración Propia (2026)

El hardware mencionado anteriormente es de uso empresarial por parte de la empresa Tendenzza & puertAlum; por tal motivo, no hay que realizar un gasto adicional.

**Tabla 5*****Costo Software.***

Cantidad	Descripción	Valor unitario	Valor total
1	PHP	∅0	∅0
1	Laravel JQuery Bootstrap	∅0	∅0
1	Velzon	∅12 600.00	∅12 600.00
1	MySQL WORKBENCH Visual Studio Code	∅0	∅0

Fuente: Elaboración Propia (2026)

Los lenguajes de programación, Front End y Back End, el Framework y la base de datos son herramientas de licencia libre, por lo que no se debe hacer gasto alguno con respecto al área del Software. Sin embargo, es importante señalar que Velzon, una plantilla administrativa avanzada que empleará el diseño de interfaces sí tiene un costo comercial de 25 dólares (12 600 colones). Este precio fue consultado directamente en el sitio oficial de distribución de la herramienta: ThemeForest – Velzon ASP.NET Core admin & dashboard template.

**Tabla 6****Costo de Infraestructura digital.**

Cantidad	Descripción	Valor unitario	Valor total
1	Dominio (t&pcr.com)	€7 163 (anual)	€7 163
1	Servidor (VPS)	€7 900 (mensual)	€7 900

Fuente: Elaboración Propia (2026)

Para la implementación en línea del sistema, la empresa tiene que hacer una inversión que incluye la adquisición de un dominio con un costo anual de 14,50 dólares (7 163 colones), lo que permitirá contar con una dirección única y profesional para el acceso al sistema. Este precio fue consultado directamente en el sitio oficial de distribución del dominio: Enom Web Site. <https://www.enom.com/reseller/domain-name-reseller-pricing-plans/>

Asimismo, se contratará un servidor privado virtual básico con un costo mensual de 15,99 dólares (7 900 colones), lo que permitirá tener memoria RAM, almacenamiento SSD, dirección IPv4 y un bloque IPv6, con un ancho de banda compartido de 150 Mbps. Este precio fue consultado directamente en el sitio oficial de distribución del VPS: Servidores Virtuales VPS en Costa Rica | CR Servers. (s. f.). Costa Rica Servers. <https://crservers.com/es/servidores-virtuales-privados>

Esta inversión puntual es considerada accesible y justificada por los beneficios que aporta en términos de diseño, funcionalidad y experiencia al usuario.

**Tabla 7****Costo de Personal**

Cantidad horas	Descripción	Costo por hora	Costo total
9	Planeación	€1,997.99	€17,981.91
450	Desarrollo	€1,997.99	€899,095.05
15	Pruebas	€1,997.99	€29,969.85
12	Documentación	€1,997.99	€23,975.88
486	<b>Total</b>		€931,498.56

Fuente: Costos obtenidos del sitio Web del Ministerio del Trabajo Costa Rica [MTSS] (2025)

El salario que se detalla para el desarrollador es un salario promedio, pero en este caso por ser un proyecto de graduación la empresa no presentará gasto real. El costo por hora es calculado según por la lista de salarios mínimos año 2025 por el Ministerio de Trabajo y Seguridad Social de Costa Rica, en su página web ([https://www.ymtss.go.cr/temas-laborales/salarios/Documentos-Salarios/lista\\_salarios\\_2025.pdf](https://www.ymtss.go.cr/temas-laborales/salarios/Documentos-Salarios/lista_salarios_2025.pdf)).

### **Viabilidad legal**

Para el desarrollo de este prototipo, se realizó una investigación sobre las normas legales vigentes en Costa Rica, asegurando que las herramientas a utilizar para el sistema son de uso Open Source, con excepción de Velzon, que tiene un costo comercial de 25 dólares. Además, se respetan las siguientes leyes:

Se respeta la Ley de Derechos de Autor y Derechos Conexos (Ley N° 6683) y la Ley de Derechos de Propiedad Intelectual (Ley N° 8039), al desarrollar el prototipo desde cero no se utilizan copias de código, respetando los principios de autoría y uso legítimo de referencias documentadas. Ambas leyes fueron consultadas a través del Sistema Costarricense de Información Jurídica (SCIJ, 2008).

Se respeta la Ley de Protección de la Persona frente al tratamiento de sus datos personales (Ley N° 8968), el desarrollador respetará todos los datos de los funcionarios de la empresa para el uso nada más del prototipo, Ley obtenida del Sistema Costarricense de Información Jurídica (SCIJ, 2023).

Ley N° 8148 que adiciona los artículos 196 BIS. 217 BIS Y 229 BIS al Código Penal (Ley N° 4573), para reprimir y sancionar los delitos informáticos: se respetan las disposiciones sobre el acceso, modificación y tratamiento de datos personales, así como los principios de seguridad informática establecidos por la Asamblea Legislativa en el año 2021.

Estas constituyen el marco legal que respalda el desarrollo ético y responsable del prototipo en Tendenzza & puerTalum, garantizando el cumplimiento normativo en materia de propiedad intelectual, protección de datos y seguridad informática.

## Proyecciones

Las proyecciones para el prototipo en Tendenzza & puerTalum son cruciales para comprender los resultados esperados y los beneficios que aporta a la organización. A continuación, se detallan las expectativas y alcances del prototipo:

- **Optimización de procesos críticos:** El prototipo está diseñado para automatizar tareas clave que actualmente presentan dificultades operativas, como la gestión de horas extra, control de vacaciones, asistencia, nómina, permisos, incapacidades, liquidaciones y aguinaldo. Esta automatización reducirá la carga administrativa, minimizará errores y agilizará la toma de decisiones.
- **Mayor eficiencia operativa:** Se proyecta una mejora significativa en la rapidez y precisión de los procesos de Recursos Humanos, permitiendo al personal enfocarse en funciones estratégicas en lugar de tareas repetitivas.
- **Mejora en la gestión de personal:** El sistema proporcionará una visión integral y actualizada del desempeño de los colaboradores, facilitando el seguimiento de métricas clave como uso de horas extra, solicitudes de permisos y cumplimiento de horarios. Esto permitirá tomar decisiones informadas y fortalecer la planificación del recurso humano.
- **Alcance del prototipo:** Aunque inicialmente se enfoca en resolver los procesos más críticos, se contempla su expansión para integrar otros módulos relacionados con la gestión personal, como capacitaciones, evaluaciones periódicas y desarrollo profesional.
- **Seguridad de datos:** Se implementan medidas de seguridad sólidas para garantizar la integridad y la confidencialidad de los datos de Recursos Humanos. Esto incluye la autenticación de usuarios y la protección de datos sensibles.

## Alcance funcional

Este proyecto de graduación se enfoca en la gestión de Recursos Humanos en la empresa Tendenzza & puerTalum., centralizando los siguientes módulos.

**Tabla 8**

***Módulos Gestión Recursos Humanos Tendenzza & puerTalum.***

Nombre del módulo	Descripción del módulo
GESTIONAR HORAS EXTRAS	Este módulo se encargará de realizar el cálculo de horas extra realizadas por los colaboradores. El colaborador deberá realizarla solicitud desde el sistema, esta será enviada a su jefatura inmediata, quien podrá aprobarla o rechazarla. En caso de aprobación, el sistema notificará automáticamente al departamento de Recursos Humanos, que a su vez notificará al colaborador sobre la confirmación, si la solicitud es rechazada, el sistema notificará al colaborador de manera inmediata.
GESTIONAR VACACIONES	Este módulo se encargará de gestionar el cálculo de vacaciones disponibles para cada colaborador. El sistema registra automáticamente un historial detallado de los días disfrutados y el saldo restante de días.  Los colaboradores pueden ver el saldo de días disponibles y realizar la solicitud desde el sistema.  Cada solicitud es enviada a la jefatura inmediata del colaborador para su aprobación. En caso de aprobación, el sistema notifica a Recursos Humanos, quien valida y notifica al colaborador vía sistema, en caso de ser rechazada, el colaborador también es notificado de forma inmediata.
GESTIONAR ASISTENCIA	Este módulo permitirá registrar y controlar los tiempos de entrada y salida de cada uno de los colaboradores a través de la aplicación web. Para garantizar la precisión del registro, el sistema emplea la geolocalización, validando que la marca se realice desde un área autorizada.

<p>GESTIONAR NÓMINAS</p>	<p>Este módulo se encarga de generar los cálculos del pago de planilla de la empresa, tomando en consideración, horas trabajadas, salario bruto, los cálculos de deducción de ley, el cálculo del rubro de renta, el salario neto, así como las incapacidades, y los permisos con goce y sin goce salarial</p>
<p>GESTIONAR PERMISOS</p>	<p>Este módulo se encargará de gestionar las solicitudes de permisos o licencias realizadas por los colaboradores.</p> <p>El colaborador deberá realizarla solicitud desde el sistema, esta será enviada a su jefatura inmediata, quien podrá aprobarla o rechazarla. En caso de aprobación, el sistema notificará automáticamente al departamento de Recursos Humanos, que a su vez notificará al colaborador sobre la confirmación, si la solicitud es rechazada, el sistema notificará al colaborador de manera inmediata</p>
<p>EVALUAR DESEMPEÑO</p>	<p>Este módulo se encargará de establecer indicadores de desempeño para cada colaborador, permitiendo aplicar evaluaciones periódicas en línea. A través de este instrumento evaluativo, los supervisores podrán registrar calificaciones y observaciones, generando un historial de rendimiento. Con base en estos resultados, se podrán asignar bonificaciones, identificar oportunidades de mejora y tomar decisiones relacionadas con el crecimiento profesional. Este módulo se encargará de establecer indicadores de desempeño para cada colaborador, permitiendo aplicar evaluaciones periódicas en línea. A través de este instrumento evaluativo, los supervisores podrán registrar calificaciones y observaciones, generando un historial de rendimiento. Con base en estos resultados, se podrán asignar bonificaciones, identificar oportunidades de mejora y tomar decisiones relacionadas con el crecimiento profesional.</p>
<p>GESTIONAR INCAPACIDADES</p>	<p>Este módulo se encargará de registrar y gestionar las incapacidades tramitadas por los colaboradores, ya sea por enfermedad o accidente, incorporando los datos correspondientes como tipo de incapacidad, diagnóstico, fechas de inicio y fin. El personal de recursos humanos será responsable de validar estas gestiones en el sistema.</p>

GESTIONAR LIQUIDACIÓN	Este módulo se encargará de generar y gestionar de forma automatizada el proceso completo de liquidación para cada colaborador, tomando en cuenta las condiciones salariales establecidas en su contrato laboral, así como las leyes nacionales vigentes. El sistema considera desde el primer día laborado todos los componentes aplicables, incluyendo días trabajados, vacaciones pendientes e indemnizaciones, permitiendo así un cálculo preciso desde el sistema web.
CALCULAR AGUINALDO	En este módulo, se encargará de realizar el cálculo automatizado del aguinaldo de cada colaborador, tomando en cuenta los meses efectivamente laborados durante el año, conforme a las normativas laborales vigentes. El proceso garantiza precisión en el monto a pagar y facilita su correcta incorporación en la nómina correspondiente.
MANTENIMIENTOS	Este módulo se encarga de realizar el borrado, inserción, modificación, actualización de datos.
CONSULTAS	Este módulo se encarga de generar información proporcionada de las diferentes tablas.
REPORTES	Este módulo se encarga de generar información proporcionada de las diferentes tablas y procesos, pero con un formato específico, según lo solicite el usuario. Podrá ser impreso o por pantalla.
SEGURIDAD	Este módulo se encarga de realizar la autenticación de contraseñas y definición de perfiles.

Fuente: Elaboración Propia (2026)

### Alcance metodológico

Para el desarrollo de este proyecto de graduación, se ha adoptado un enfoque metodológico basado en la comunicación continua y colaborativa con la empresa TendENZA & puerTalum. El propósito es garantizar que cada avance en el desarrollo del prototipo sea revisado y ajustado en conjunto, permitiendo que el cliente se sienta parte activa del proceso y que cada funcionalidad responda a sus necesidades reales.

El desarrollo de este aplicativo web se realiza bajo un modelo de ciclo de vida alternativo e incremental, lo cual permite construir el sistema por etapas, realizar pruebas constantes y aplicar mejoras continuas. Este enfoque facilita la adaptación del prototipo conforme se identifican nuevas necesidades o se recibe retroalimentación directa de los usuarios.

Cabe mencionar que se utiliza la arquitectura Modelo – Vista – Controlador (MVC), que ofrece una alta escalabilidad y modularidad. Esto permite agregar nuevas funciones sin necesidad de reescribir grandes bloques de código, optimizando el mantenimiento y evolución del sistema.

Además, se prioriza la seguridad de los datos personales de los funcionarios, incluyendo credenciales de acceso, mediante mecanismos de autenticación y protección de información sensible.

Finalmente, al construir el proceso de desarrollo, se realizará una sesión de capacitación única dirigida a todo el personal de la empresa, con el objetivo de explicar detalladamente cada funcionalidad del prototipo y asegurar su uso correcto y máximo aprovechamiento.

### **Alcance tecnológico**

Los fundamentos sobre los cuales se construye el prototipo, incluyendo los lenguajes de programación, frameworks, arquitecturas, y herramientas que aseguran su buen desempeño, son determinados por el alcance tecnológico del proyecto. Esta sección explica los elementos fundamentales que constituyen el desarrollo del Back End, la administración de la base de datos y los procedimientos de autenticación, asegurando que el sistema se sustente en tecnologías actuales, seguras y escalables, capaces de responder a las necesidades de gestión de Recursos Humanos en Tendenzza & puertAlum.

### **Back End**

El lenguaje principal utilizado en la lógica del software es PHP, implementado a través del framework Laravel. Esta elección permite un desarrollo ágil, seguro y escalable, con una estructura clara que facilita la integración de funcionalidades avanzadas.

## **Front End**

Este también se realizará con Laravel, complementando con tecnologías como jQuery que mejora la interacción dinámica en la interfaz, Bootstrap que proporciona una estructura visual responsiva y moderna y Velzon que es una plantilla administrativa premium permitiendo construir interfaces intuitivas y profesionales.

## **Arquitectura**

MVC (Modelo-Vista-Controlador): La arquitectura MVC se implementa en el sistema para separar claramente la lógica de negocio (Modelo), la interfaz de usuario (Vista) y el control de las interacciones (controlador). Esto garantiza una estructura organizada y facilita el mantenimiento futuro.

## **Framework**

El sistema se desarrolla bajo el framework Laravel, ampliamente conocido por su robustez, seguridad y comunidad activa. Laravel permite una gestión eficiente de rutas, controladores, vistas y modelos, optimizando el flujo de desarrollo.

## **Base de datos**

MySQL Workbench: La base de datos del sistema se gestiona mediante MySQL Workbench, un motor de base de datos ampliamente utilizado y confiable. Esta elección permite un almacenamiento seguro y eficiente de los datos de Recursos Humanos.

## Autenticación

Para la gestión de usuarios y permisos, implementará Laravel Spatie, una solución confiable y flexible que permite controlar el acceso a funcionalidades específicas del sistema, asegurando la protección de datos sensibles.

El alcance tecnológico establecido garantiza que el prototipo esté construido sobre una base tecnológica sólida y actualizada. La combinación de Laravel (PHP) para el desarrollo Back End, junto con herramientas Front End como jQuery, Bootstrap, Velzon y los lenguajes HTML, CSS y JavaScript, permite construir un sistema web eficiente, escalable y visualmente amigable. La gestión de datos se realiza mediante MySQL, utilizando Workbench como entorno de administración, lo que asegura un almacenamiento estructurado y seguro.

Esta arquitectura tecnológica responde a los requerimientos específicos de gestión de Recursos Humanos en Tendenzza & puerTalum, abordando procesos como control de asistencia, permisos, vacaciones, horas extra, incapacidades, liquidaciones, aguinaldo y evaluación de desempeño. Además, se garantiza la protección de datos personales mediante autenticación robusta con Laravel Spatie, cumpliendo con los estándares legales y técnicos vigentes.

**Tabla 9**

***Componentes de Desarrollo.***

Componentes	Especificaciones	Otros
Back End	Laravel (PHP)	Este será el lenguaje y frameworks para la lógica del sistema.
Front End	Laravel jQuery Bootstrap Velzon	Tecnologías para la interfaz visual y experiencia de usuario.
Arquitectura	MVC	Esta va a ser la arquitectura de software para escalabilidad y mantenimiento.

Framework	Laravel	Este es el principal del sistema
Base de datos	MySQL Workbench	Herramienta para gestión de base de datos
Autenticación	Laravel Spatie	Control de accesos y permisos de usuarios.

Fuente: Elaboración Propia (2026)

La fundamentación expuesta permite comprender la relevancia de la iniciativa y justifica su viabilidad técnica y organizacional. Con ello se establece un punto de partida sólido para avanzar hacia el análisis conceptual y técnico que dará forma al prototipo, asegurando que la solución planteada se eficiente, escalable y alineada con los objetivos institucionales.

## **CAPÍTULO II: MARCO REFERENCIAL**

### **Prototipado de Software en la Optimización Empresarial**

#### **Función en Procesos Internos.**

El prototipado del software permite simular funcionalidades clave antes de implementar un sistema definitivo. En el contexto de Recursos Humanos, esto incluye módulos como control de asistencia, permisos, incapacidades y planillas. La empresa puede validar ideas, detectar errores y ajustar procesos sin comprometer recursos. Este enfoque reduce riesgos y mejora la calidad del producto final. Además, facilita la retroalimentación directa de los usuarios internos. Esta interacción temprana con el sistema permite que los colaboradores se familiaricen con las funcionalidades y contribuyan activamente al diseño de una solución más efectiva.

#### **Prototipado rápido como estrategia de validación.**

El prototipado rápido se basa en la filosofía de “aprender haciendo”, donde se construyen versiones funcionales mínimas para validar la hipótesis. Según Miro (2025), esta técnica permite reducir las incertidumbres en las etapas iniciales del desarrollo. Se prioriza la funcionalidad sobre la estética, lo que agiliza la toma de decisiones. En entornos empresariales, esta estrategia es útil para probar flujos de trabajo y obtener retroalimentación inmediata.

#### **Etapas iterativas del desarrollo de prototipos.**

El desarrollo de prototipos se realiza en ciclos iterativos que incluyen, conceptualización, diseño, codificación, pruebas, y ajustes. Cada iteración permite mejorar el sistema según los resultados obtenidos. Esta dinámica favorece la evolución constante del producto. Además, permite incorporar cambios sin afectar la estructura general. Es una metodología flexible y centrada en el usuario.

### **Evaluación de usabilidad en prototipos.**

La usabilidad es un criterio esencial en sistemas administrativos. Un sistema usable permite que los colaboradores interactúen con facilidad, comprenden las funciones disponibles y ejecutan tareas sin errores. Según Nielsen (2023), “la usabilidad no es una cualidad única, sino un conjunto de atributos que determinan la facilidad de uso” (p. 25). En este proyecto, se aplicarán pruebas de tareas específicas como la solicitud de permisos y revisión de planillas. Estas pruebas permitirán identificar obstáculos en la experiencia del usuario y realizar ajustes que mejoren la eficiencia y satisfacción de los colaboradores al utilizar la plataforma.

### **Aplicaciones Web como Herramienta Corporativa**

#### **Aplicaciones web en sectores empresariales.**

Las aplicaciones web se han convertido en herramientas clave para la gestión empresarial. Su accesibilidad desde cualquier navegador permite que múltiples departamentos trabajen de forma integrada. En sectores como salud, educación, comercio y administración pública, estas plataformas agilizan procesos internos y mejoran la comunicación. Según Omatech (2023), las aplicaciones web permiten “aumentar tareas repetitivas y centralizar la información en tiempo real” (párr.4). Esta capacidad de centralización y automatización resulta especialmente útil en organizaciones que manejan grandes volúmenes de datos y requieren coordinación. En este sistema, la aplicación web permitirá que accedan a información actualizada de forma simultánea, mejorando la toma de decisiones y reduciendo los tiempos de respuesta.

#### **Beneficios operativos en áreas administrativas.**

En el área de Recursos Humanos, las aplicaciones web permiten gestionar permisos, planillas, horarios y expedientes de forma digital. Esto reduce el uso de papel, mejora la trazabilidad y facilita auditorías internas. Además, los jefes de departamento pueden acceder

a reportes actualizados sin depender de terceros. La automatización de procesos administrativos mejora la eficiencia y libera tiempo para tareas estratégicas.

### **Limitaciones técnicas y desafíos de implementación.**

A pesar de sus ventajas, las aplicaciones web presentan desafíos técnicos. Requieren conexión constante a internet, mantenimiento regular y actualizaciones de seguridad. También pueden enfrentar problemas de compatibilidad con navegadores antiguos. Según Hostinger (s.f.), “una aplicación web mal optimizada puede ralentizar el trabajo diario y generar frustración en los usuarios” (párr.6). Por ello, es fundamental realizar pruebas antes de su implementación definitiva. En este proyecto, se contempla una fase de pruebas funcionales y de rendimiento que permitirá identificar posibles fallos antes del despliegue oficial. Estas pruebas garantizarán que el sistema se adapte correctamente a cualquier dispositivo y navegador utilizado por la empresa, lo que minimiza riesgos operativos y asegurando una experiencia y confiable desde el primer día de uso.

### **Relación entre software y aplicaciones web.**

Toda aplicación web está respaldada por software que define su lógica, estructura y comportamiento. El software incluye lenguajes de programación, frameworks y bases de datos que permiten que la aplicación funcione correctamente. En este proyecto, Laravel y MySQL conforman la base tecnológica que da vida a la plataforma. Esta relación entre software y aplicación web es lo que permite automatizar procesos y garantizar la seguridad de los datos. La correcta integración entre estos componentes técnicos será determinante para lograr un sistema robusto, escalable y alineado con los objetivos operativos de Tendenzza & puertAlum.

## **Tecnologías Utilizadas**

### **Backend: Laravel como base Estructural.**

Laravel es el framework elegido para el desarrollo del backend. Ofrece una estructura clara y segura para gestionar rutas, modelos y controladores. Facilita la autenticación y validación de datos. Además, incluye herramientas para pruebas y gestión de información. Su uso mejora escalabilidad y mantenibilidad del sistema. (Laravel News, 2023). En este sistema, Laravel permite construir una base sólida que se adapta al crecimiento de la empresa. Su arquitectura modular facilita la incorporación de nuevos módulos como liquidaciones, reportes o evaluaciones sin comprometer la estabilidad del sistema existente.

### **Front End: Blade, jQuery, Bootstrap y Velzon.**

La interfaz se construye con Laravel Blade, jQuery, Bootstrap y Velzon. Blade permite integrar HTML y PHP de forma eficiente. jQuery facilita la manipulación del DOM y eventos. Bootstrap aporta diseño responsivo y moderno. Velzon mejora la estética y funcionalidad visual del sistema (Informatec Digital, 2025). Esta combinación tecnológica permite desarrollar una interfaz moderna, adaptable, y funcional, que responde a las exigencias visuales y operativas del entorno empresarial.

### **Interacción entre Front End y Back End.**

La comunicación entre frontend y backend es constante. El usuario interactúa con formularios y botones en la interfaz. El backend valida, procesa y responde a esas acciones. Por ejemplo, al solicitar vacaciones, se registra la solicitud y se muestra una confirmación. Esta integración garantiza fluidez y coherencia. Además, permite que los datos fluyan de forma segura entre capas del sistema. Dicha interacción asegura que cada acción realizada por el usuario tenga una respuesta inmediata y precisa, fortaleciendo la confianza en el sistema.

### **Arquitectura del Sistema y Control de Accesos**

#### **Patrón MVC como modelo de desarrollo.**

El sistema propuesto se estructura bajo el patrón Modelo-Vista-Controlador (MVC), ampliamente utilizado en el desarrollo web moderno. Este modelo permite separar la lógica de negocio, la interfaz de usuario y el control de flujo, lo que facilita el mantenimiento y la escalabilidad. Según Muñoz y Muñoz (2022), “el patrón MVC mejora la organización del código y permite que distintos desarrolladores trabajen en paralelo sin interferencias” (p.18). Esta arquitectura es ideal para proyectos que requieren modularidad y claridad estructural. En este sistema, la adopción del patrón MVC permitirá una mejor distribución de responsabilidades entre los componentes, facilitando tanto el trabajo colaborativo como la incorporación de nuevas funcionalidades sin afectar el núcleo del sistema.

### **Gestión de datos con MySQL y Workbench.**

La base de datos del sistema se gestiona mediante MySQL, un sistema relacional robusto y ampliamente adoptado en entornos empresariales. Para su diseño y administración se utiliza Workbench, herramienta que permite visualizar relaciones, aplicar restricciones y validar integridad referencial. Esta combinación facilita la estructuración jerárquica de los datos, especialmente en módulos como asistencia, permisos, y nómina. Además, permite realizar migraciones y respaldos de forma segura.

### **Control de roles y permisos con Laravel Spatie.**

La seguridad del sistema se refuerza mediante la implementación de roles y permisos utilizando la biblioteca Spatie, integrada en Laravel. Esta herramienta permite asignar privilegios específicos a cada tipo de usuario, como administración, jefe de departamento o colaborador. Según Spatie (s.f.), “la gestión granular de permisos permite proteger recursos sensibles y evitar accesos no autorizados” (párr.3). Esta funcionalidad es esencial en sistemas que manejan información confidencial como expedientes y planillas. En el sistema que se propone, esta gestión diferenciada de accesos permitirá establecer límites claros que información puede ser consultada o modificada por cada perfil. Esto no solo fortalece la seguridad de los datos, sino que también promueve la responsabilidad individual dentro del

entorno digital, al registrar y controlar cada acción realizada por los usuarios según su rol asignado.

## **Frameworks en el Desarrollo Web Empresarial**

### **Función estructural de los frameworks.**

Los frameworks son entornos de desarrollo que proporcionan una base estructurada para construir aplicaciones web. Permiten aplicar buenas prácticas, reutilizar componentes y mantener la coherencia del código. En el ámbito empresarial, su uso facilita la colaboración entre equipos técnicos y acelera los tiempos de entrega. Según González y Rivera (2021), “los frameworks permiten estandarizar procesos de desarrollo, reduciendo errores y mejorando la mentalidad del software” (p. 42). Esta estandarización será clave para mantener una lógica uniforme en el desarrollo de módulos como permisos, asistencia y nómina. Además, permitirá que distintos desarrolladores trabajen sobre una misma base sin generar conflictos, lo que agiliza el mantenimiento y facilita futuras actualizaciones.

### **Ventajas en seguridad, escalabilidad y mantenimiento.**

Los frameworks modernos ofrecen herramientas integradas para la gestión de seguridad, como protección contra ataques CSRF, validación de formularios y cifrado de contraseñas. Además, permiten escalar el sistema conforme crecen los requerimientos de la empresa. Gracias a su modularidad, facilita el mantenimiento, ya que cada componente puede actualizarse sin afectar el resto. Estas ventajas son especiales relevantes en sistemas que manejan datos sensibles como expedientes laborales y nómina.

### **Aplicación en sistemas de gestión de recursos humanos.**

En el contexto de Recursos Humanos, los frameworks permiten desarrollar módulos específicos como control de asistencias, gestión de permisos, evaluación de desempeño y generación de reportes. Laravel, por ejemplo, facilita la creación de rutas protegidas,

controladores personalizados y vistas adaptadas al perfil del usuario. Esta flexibilidad permite que el sistema se ajuste a las políticas internas de cada organización, manteniendo la seguridad y la eficiencia operativa.

## **Laravel como Framework Estratégico**

### **Estructura basada en Modelo-Vista-Controlador.**

Laravel se fundamenta en el patrón Modelo-Vista-Controlador (MVC), lo que permite una separación clara entre la lógica de negocio, la presentación y el control de flujo. Esta estructura mejora la organización del código, facilita el mantenimiento y permite que distintos desarrolladores trabajen en paralelo. En el contexto del sistema propuesto, esta arquitectura permite que los módulos de asistencia, permisos y nómina se desarrollen de forma independiente pero coherente.

### **Herramientas integradas para desarrollo ágil.**

Laravel incluye una serie de herramientas que aceleran el desarrollo de aplicaciones empresariales. Entre ellas destacan Artisan (para comandos automatizados), Eloquent (para gestión de bases de datos) y Blade (para plantillas dinámicas). Estas herramientas permiten construir funcionalidades complejas en menos tiempo, manteniendo la calidad de código. Según Laravel News (2023), “Laravel proporciona una experiencia de desarrollo elegante y expresiva, ideal para proyectos escalables” (párr. 2). En el desarrollo del sistema, estas herramientas permitirán implementar funcionalidades como la gestión de solicitudes, generación de reportes y validación de datos con mayor rapidez y precisión. Esto no solo optimiza el tiempo de entrega, sino que también garantiza una estructura de código limpia, reutilizable y fácil de mantener.

### **Seguridad y control de roles con librerías externas.**

La seguridad es un componente esencial en Laravel, reforzado mediante librerías como Spatie para la gestión de roles y permisos. Esta integración permite definir accesos específicos para cada tipo de usuario, protegiendo módulos sensibles como expedientes y reportes. Además, Laravel incluye protección contra ataques comunes como CSRF y XSS, lo que garantiza la integridad de los datos. Estas medidas son fundamentales en sistemas que manejan información laboral confidencial.

### **Ecosistema y comunidad de Laravel.**

Laravel cuenta con una comunidad activa y una amplia documentación que facilita la resolución de problemas y la incorporación de nuevas funcionalidades. El ecosistema incluye paquetes oficiales y externos que permiten extender el sistema sin necesidad de desarrollar desde cero. Esta comunidad garantiza soporte constante y actualizaciones periódicas, lo que convierte a Laravel en una opción confiable para proyectos institucionales y empresariales.

### **Diseño Responsivo e Interacción de Usuario**

#### **Adaptabilidad de interfaces en múltiples dispositivos.**

El diseño responsivo permite que una aplicación web se adapte automáticamente a distintos tamaños de pantalla, como computadoras, tabletas, y teléfonos móviles. Esta característica es fundamental en entornos laborales donde los usuarios acceden al sistema desde diversos dispositivos. Según Marcotte (2011), “el diseño responsivo no es una técnica, sino una filosofía que busca adaptarse al usuario en lugar de forzarlo a adaptarse al sistema” (p. 12). Esta adaptabilidad mejora la accesibilidad y la experiencia de uso. En la propuesta de este proyecto, esta filosofía se traduce en una interfaz que responde a las condiciones reales de uso, permitiendo que cada colaborador interactúe con los módulos de asistencia, permisos y planillas desde cualquier dispositivo sin pérdida de funcionalidad. Esto garantiza una experiencia inclusiva, eficiente y alineada con las dinámicas laborales actuales.

### **Bootstrap como herramienta de diseño responsivo.**

Bootstrap es un framework CSS que facilita la creación de interfaces responsivas mediante un sistema de rejillas, componentes reutilizables y clases predefinidas. En el sistema desarrollado, Bootstrap permite que los formularios, botones y tablas se ajusten automáticamente al ancho de la pantalla. Esto garantiza que los usuarios puedan interactuar con el sistema de forma cómoda, sin importar el dispositivo utilizado. Además, reduce el tiempo de desarrollo y mejora la coherencia visual.

### **Interactividad en tiempo real con Blade y jQuery.**

La combinación de Blade (motor de plantillas de Laravel) y jQuery permite crear interfaces dinámicas que responden a las acciones del usuario en tiempo real. Por ejemplo, al seleccionar un tipo de permiso, el sistema puede mostrar automáticamente los campos requeridos sin recargar la página. Esta interactividad mejora la fluidez del sistema y reduce la carga cognitiva del usuario. Además, permite validar formulario y mostrar mensajes de confirmación de manera inmediata.

### **Impacto en la experiencia del usuario.**

Un diseño responsivo y una interfaz interactiva incluye directamente en la percepción del sistema por parte del usuario. Una experiencia fluida, clara y adaptada a sus necesidades incrementa la satisfacción y la adopción del sistema. Según Rodríguez (2025)íiguez (2025), “la adaptabilidad será más crucial que nunca para los diseñadores y desarrolladores, ya que los usuarios esperan interfaces dinámicas, flexibles y eficientes” (párr. 3). En este proyecto, se prioriza la simplicidad, la legibilidad y la navegación intuitiva como pilares de la experiencia de usuario, especialmente en módulos como permisos, asistencia y planillas.

## **Bases de Datos y Gestión de Información**

### **Rol de la base de datos en sistemas administrativos.**

Las bases de datos son el núcleo estructural de los sistemas administrativos modernos. Permiten almacenar, organizar y recuperar información de forma eficiente, garantizando la integridad y disponibilidad de los datos. En el contexto de Recursos Humanos, una base de datos bien diseñada facilita la gestión de expedientes, de asistencia, permisos y nómina. Según Hernández (2022), “la base de datos es el componente que da sentido a la lógica del sistema, permitiendo que cada módulo funcione de manera integrada” (p. 33). En esta propuesta, la base de datos cumple un rol estratégico al permitir que los distintos módulos operen de forma sincronizada, evitando duplicidad de registros y facilitando la trazabilidad de cada acción.

### **Uso de MySQL y Workbench en el proyecto.**

Para este sistema se ha utilizado MySQL como motor de base de datos, por su estabilidad, escalabilidad y compatibilidad con Laravel. Workbench se emplea como herramienta de diseño y administración, permitiendo visualizar relaciones entre tables, aplicar restricciones y generar diagramas cantidad-relación. Esta combinación facilita la estructuración jerárquica de los datos, especialmente en módulos como asistencia, permisos y planillas.

### **Integridad de datos y validaciones estructurales.**

La integridad de los datos se garantiza mediante el uso de claves primarias, foráneas y restricciones de tipo. Estas validaciones evitan duplicados, errores de ingreso y relaciones inconsistentes entre módulos. Por ejemplo, un permiso no puede registrarse sino existe un colaborador asociado. Según Torres (2023), “la validación estructural es esencial para mantener la coherencia lógica del sistema y evitar fallos operativos” (p. 21). Para este proyecto, se han definido reglas estrictas de validación para asegurar que cada registro cumpla con los criterios establecidos y se relacione correctamente con los demás

componentes del sistema, previniendo errores y fortaleciendo la confiabilidad del sistema ante auditorías internas y procesos de control.

## **Normalización y Eficiencia de Datos**

### **Importancia de la normalización en bases de datos.**

La normalización de datos es un proceso técnico que permite organizar la información en estructuras lógicas, eliminando redundancias y asegurando la integridad referencial. En sistemas administrativos, este proceso garantiza que cada dato se almacene una sola vez y se relacione correctamente con otros módulos. Según Vargas (2021), “la normalización permite reducir inconsistencias y mejorar el rendimiento de las consultas en sistemas empresariales” (p. 29). En el sistema propuesto, se aplican formas normales hasta la tercera etapa para asegurar eficiencia operativa. Esta aplicación de principios de normalización permite que los módulos del sistema, como asistencia, permisos y nomina, funcionen de manera integrada, sin duplicidad de registros ni conflictos de relación.

### **Relación entre eficiencia y estructura de datos.**

Una base de datos bien normalizada permite que las consultas sean más rápidas, precisas y menos propensas a errores. Esto se traduce en eficiencia operativa, especialmente en módulos que requieren cálculos frecuentes como nómina, asistencia y reportes. Además, facilita la integración con otros sistemas y reduce el consumo de recursos del servidor. En este proyecto, se ha priorizado una estructura jerárquica clara, con claves primarias y foráneas bien definidas.

### **Validación de claves y compatibilidad referencial.**

La validación de claves es esencial para mantener la coherencia entre tablas relacionadas. Por ejemplo, un registro de asistencia debe estar vinculado a un colaborador existente, y un permiso debe estar asociado a un tipo definido. Estas relaciones se establecen

mediante claves foráneas que garantizan la compatibilidad referencial. Según Rojas (2023), “la validación cruzada entre tablas es un mecanismo preventivo que evita errores lógicos y fortalece la seguridad de los datos” (p. 17). En la propuesta, se han implementado reglas de validación que aseguran la consistencia de cada registro, evitando errores operativos y facilitando el control interno. Esta compatibilidad referencial no solo protege la integridad de la información, sino que también permite auditar el comportamiento del sistema con mayor precisión, fortaleciendo la confiabilidad de los procesos administrativos.

## **Automatización Mediante Procedimientos Almacenados**

### **Concepto y función de los procedimientos almacenados.**

Los procedimientos almacenados son bloques de código SQL que se ejecutan directamente en el servidor de base de datos. Permiten automatizar tareas repetitivas, como cálculos, validaciones o actualizaciones masivas. En sistemas administrativos, su uso mejora el rendimiento y reduce la carga sobre el backend. Según Méndez (2021), “los procedimientos almacenados permiten encapsular lógica de negocio dentro de la base de datos, reduciendo la dependencia del código externo” (p.14). En este sistema, esta lógica encapsulada permite que las operaciones críticas se ejecuten de forma más rápida y segura, sin depender de la capa de presentación. Esto no solo optimiza el uso de recursos, sino que también facilita el mantenimiento del sistema, ya que las reglas de negocio se concentran en un entorno controlado y fácilmente auditable.

### **Aplicación en módulos de Recursos Humanos.**

En el sistema propuesto, los procedimientos almacenados se utilizan para calcular montos de nómina, validar las fechas de permisos y generar reportes mensuales. Esta lógica se ejecuta directamente en MySQL, lo que reduce el tiempo de respuesta y mejora la eficiencia. Además, permite que los procesos críticos se mantengan estables, incluso si se actualiza el frontend o el backend. Esta estrategia garantiza consistencia operativa y facilita auditorías internas.

### **Ventajas frente a consultas tradicionales.**

A diferencia de las consultas SQL convencionales, los procedimientos almacenados pueden incluir estructuras condicionales, bucles y parámetros de entrada. Esto permite construir procesos más complejos y reutilizables. Según López (2023). “la automatización mediante procedimientos almacenados mejora la trazabilidad de los procesos y reduce el margen de error humano” (p. 22). Esta ventaja se traduce en una mayor confiabilidad de los resultados generados por el sistema, especialmente en operaciones sensibles como el cálculo de liquidaciones o la validación de ausencias. Al reducir la intervención manual y estandarizar los procesos, se fortalece la seguridad de los datos y se mejora la eficiencia general del sistema administrativo.

### **Seguridad de la Información y Cifrado de Contraseñas.**

#### **Importancia de la seguridad en sistemas administrativos.**

La seguridad de la información es un componente crítico en sistemas que gestionan datos sensibles, como expedientes laborales, nómina y permisos. Un sistema seguro protege la confidencialidad, integridad y disponibilidad de los datos. Según Ramírez (2023), “la seguridad no debe ser un módulo adicional, sino un principio transversal en todo el ciclo de desarrollo” (p. 19). En este proyecto, se han aplicado medidas preventivas desde la arquitectura hasta la interfaz. Esta visión integral de la seguridad permite que cada capa del sistema, desde la base de datos hasta la interacción del usuario, esté protegida contra amenazas internas y externas.

#### **Cifrado de contraseñas con algoritmo con algoritmos hash.**

El cifrado de contraseñas se realiza mediante algoritmos hash como bcrypt, que transforman la contraseña en una cadena irreconocible y no reversible. Esto impide que un atacante pueda recuperar la contraseña original, incluso si accede a la base de datos. Laravel incluye funciones nativas para aplicar este tipo de cifrado. Según Laravel Docs (2025),

“bcrypt es el algoritmo recomendado por defecto, ya que incluye sal aleatoria y es resistente a ataques por fuerza bruta” (párr. 5). Esta técnica se aplica de forma automática al momento de registrar o actualizar credenciales, lo que garantiza que ninguna contraseña se almacene en texto plano.

### **Gestión de roles y accesos diferenciados.**

La gestión de roles permite asignar permisos específicos a cada tipo de usuario, como administrador, jefe de departamento o colaborador. Esta diferenciación evita accesos no autorizados y protege módulos críticos. En este sistema, se ha implementado la biblioteca Spatie para controlar accesos de forma granular. Además, se han definido rutas protegidas y validaciones en el backend para reforzar la seguridad.

### **Prevención de vulnerabilidades comunes.**

El sistema incluye medidas contra vulnerabilidades frecuentes como inyecciones SQL, ataques XSS y falsificación de solicitudes (CSRF). Estas amenazas pueden comprometer la integridad del sistema si no se gestionan adecuadamente. Laravel proporciona protección automática contra CSRF y depuración de datos. Según Ali (2025), “aunque Laravel ofrece seguridad desde el inicio, los desarrolladores deben implementar buenas prácticas para proteger formularios, rutas y datos sensibles” (párr. 2). En este proyecto, se han adoptado dichas buenas prácticas como parte del protocolo de desarrollo, incluyendo la depuración de datos, la validación de formularios y el uso de tokens de seguridad. Estas acciones refuerzan la estabilidad del sistema y aseguran que los datos permanezcan íntegros ante intentos de explotación o manipulación externa.

### **Tendencias en la Digitalización de Recursos Humanos.**

#### **Transformación digital en procesos laborales.**

La digitalización ha redefinido la forma en que las organizaciones gestionan sus recursos humanos. Procesos como reclutamiento, evaluación, capacitación y control de asistencia se han automatizado mediante plataformas digitales. Esta transformación permite mayor trazabilidad, eficiencia y acceso remoto. Según Pérez (2024), “la digitalización no solo optimiza procesos, sino que transforma la cultura organizacional hacia modelos más ágiles y colaborativos” (p. 31). En el contexto del sistema propuesto, esta transformación digital se traduce en una herramienta que no solo automatiza tareas operativas, sino que también promueve una cultura organizacional basada en la transparencia, la autonomía y la mejora continua.

### **Plataformas integradas para gestión del talento.**

Las plataformas integradas permiten centralizar funciones como asistencia, permisos, expedientes, capacitaciones y reportes en un solo entorno digital. Esto mejora la experiencia del usuario y facilita la administración por parte del departamento de RRHH. Según Gómez (2023), “las plataformas integradas son clave para una gestión del talento eficiente, alineada con los objetivos estratégicos de la organización” (p. 22). En este proyecto, la integración de módulos en una sola plataforma permite una visión holística del desempeño y la gestión del personal. Esta centralización no solo optimiza el trabajo del área de Recursos Humanos, sino que también permite generar indicadores clave para la toma de decisiones estratégicas, alineando la gestión del talento con los objetivos institucionales de crecimiento y sostenibilidad.

### **Interoperabilidad y Escalabilidad en Sistemas Empresariales.**

#### **Concepto de interoperabilidad en entornos digitales.**

La interoperabilidad se refiere a la capacidad de distintos sistemas y aplicaciones para comunicarse, intercambiar datos y utilizar la información de manera efectiva. En el contexto empresarial, esta característica permite integrar plataformas como nómina, contabilidad y recursos humanos, evitando duplicidad de datos. Según Salazar (2023), “la interoperabilidad

es clave para lograr una visión unificada de los procesos organizacionales y mejorar la toma de decisiones” (p. 16). En este sistema, esta capacidad de integración permitirá que los módulos internos se conecten con otras plataformas institucionales, facilitando el flujo de información entre áreas y reduciendo los tiempos de procesamiento. Esta visión unificada de los datos fortalece la coordinación interdepartamental y mejora la eficiencia operativa en todos los niveles de la organización.

### **Escalabilidad como principio de sostenibilidad tecnológica.**

La escalabilidad permite que un sistema crezca en funcionalidad, usuarios o volumen de datos sin perder rendimiento. En proyectos institucionales, esta característica es esencial para adaptarse a nuevas necesidades sin rediseñar toda la arquitectura, Laravel y MySQL ofrecen herramientas para escalar horizontal o verticalmente, según el crecimiento de la organización. Esta capacidad garantiza que el sistema siga siendo útil y eficiente a largo plazo.

## **Impacto del Software en la Toma de Decisiones Administrativas**

### **Digitalización como soporte para decisiones estratégicas.**

El uso de software en entornos administrativos permite recopilar, procesar y visualizar datos relevantes para la toma de decisiones. Módulos como asistencia, permisos y nómina generan información que puede ser analizada para detectar patrones, evaluar desempeño y optimizar recursos. Según Cordero (2023), “el software no solo automatiza tareas, sino que transforman los datos en conocimiento útil para la gestión estratégica” (p.30). En este sistema esta capacidad de transformación se traduce en una herramienta que permite a la administración tomar decisiones informadas, basadas en evidencia concreta y actualizada. La digitalización no solo agiliza los procesos, sino que también fortalece la planificación institucional al ofrecer una visión clara del comportamiento organizacional.

### **Generación de reportes dinámicos y visuales.**

El sistema desarrollado permite generar reportes personalizados que incluyen gráficos, tabletas y filtros interactivos. Estos reportes facilitan la interpretación de datos por parte de jefaturas y directivos, permitiendo decisiones basadas en evidencia. Laravel y herramientas como Chart.js, permiten integrar visualizaciones en tiempo real. Esta funcionalidad mejora la transparencia y la capacidad de respuesta ante situaciones operativas.

### **Evaluación de indicadores clave de desempeño (KPI).**

Los indicadores clave de desempeño permiten medir la eficiencia de los procesos internos. En el sistema propuesto, se pueden evaluar métricas como puntualidad, frecuencia de permisos, carga laboral y cumplimiento de horarios. Estos datos se presentan en paneles visuales que permiten comparaciones históricas y segmentadas. Según Molina (2024), “los KPI bien definidos permiten alinear la operación con los objetivos estratégicos de la organización” (p. 18). En el proyecto, la implementación de KPI permite monitorear el comportamiento de los colaboradores y detectar áreas de mejora en tiempo real. Esta evaluación continua facilita la toma de decisiones correctivas y preventivas, alineando los esfuerzos operativos con los objetivos institucionales y promoviendo una gestión basada en resultados.

### **Cierre del Capítulo**

A lo largo de este capítulo se ha expuesto la fundamentación técnica del sistema propuesto, abordando desde el prototipo inicial hasta las tecnologías aplicadas en su desarrollo. Se detallaron los frameworks utilizados, la arquitectura basada en el patrón MVC, la gestión de base de datos con MySQL y Workbench, así como la implementación de seguridad mediante cifrado y control de accesos. Cada bloque temático ha contribuido a evidenciar la coherencia entre los componentes del sistema y su alineación con las necesidades del área de Recursos Humanos.

Asimismo, se analizaron aspectos como la normalización de datos, la automatización de procesos mediante procedimientos almacenados, y la generación de reportes para la toma de decisiones administrativas. Se destacó la importancia de la interoperabilidad y la escalabilidad como principios de sostenibilidad tecnológica, así como las tendencias actuales en digitalización del talento humano. Todo ello permite concluir que el sistema desarrollado no solo responde a requerimientos funcionales, sino que también incorpora buenas prácticas de ingeniería de software, garantizando eficiencia, seguridad y adaptabilidad.

## **CAPITULO III: MARCO METODOLÓGICO**

### **Enfoques de Investigación**

Los enfoques de investigación son estrategias que dan dirección al proceso de búsqueda y solución de problemáticas. Son herramientas esenciales para la sociedad, ya que permiten adquirir, interpretar y generar nuevos conocimientos o soluciones aplicables en distintos contextos, como la optimización de procesos empresariales. La elección del enfoque nunca es aleatoria, sino que se basa en las metas del estudio y naturaleza del problema (González Rivera, 2024). Esta elección metodológica permite al investigador seleccionar el camino más adecuado para alcanzar los objetivos del estudio.

#### **Enfoque Cuantitativo**

Este enfoque utiliza la recolección de datos para probar hipótesis con base en la medición numérica y el análisis estadístico. Su objetivo es establecer patrones de comportamiento, determinar la magnitud de un fenómeno o probar teorías. Este enfoque es típicamente estructurado y se aplica en investigaciones donde se busca la generalización de resultados a partir de una muestra poblacional (Hernández & Mendoza, 2018& Mendoza, 2018). Su aplicación es común en estudios donde se requiere precisión estadística y la posibilidad de repetir los resultados.

#### **Enfoque Cualitativo**

El enfoque cualitativo se centra en la recolección y el análisis de datos para afinar las preguntas de investigación, explotar fenómenos en profundidad y obtener perspectivas detalladas de los participantes. Se interesa por las experiencias, emociones y percepciones, buscando comprender el contexto y reconstruir una realidad tal como es percibida por los involucrados. Su flexibilidad permite adaptarse a los hallazgos emergentes del proceso.

#### **Enfoque Mixto**

Este enfoque combina elementos de los enfoques cuantitativo y cualitativo en un mismo estudio. Su propósito es lograr una comprensión más profunda y holística del problema al triangular los datos numéricos y las narrativas de los participantes. Es una estrategia potente que utiliza las fortalezas de ambos métodos para obtener una perspectiva completa de la investigación.

### **Enfoque de Investigación Seleccionado**

Para el desarrollo de este proyecto de graduación, se seleccionó el enfoque cualitativo. Esta decisión obedece a que el objetivo principal en la etapa de análisis no es medir la frecuencia de los problemas, sino comprender cómo y por qué ocurren los procesos de Recursos Humanos de manera actual en Tendenzza & puertAlum.

Se requiere la flexibilidad del enfoque cualitativo para estudiar a los colaboradores y jefaturas mediante técnicas abiertas, como entrevistas o grupos focales. Esto permitirá obtener perspectivas, experiencias y emociones detalladas sobre la gestión actual de módulos como permisos, vacaciones, horas extras e incapacidades. Se busca entender la experiencia de cada uno al solicitar un permiso o al gestionar una planilla, incluyendo cuánto tiempo y esfuerzo actualmente.

El uso de este enfoque ayuda a generar información abierta y necesaria para la optimización de los procesos de gestión de Recursos Humanos que serán integrados al nuevo sistema, tales como la gestión de planillas, cálculo de incapacidades y manejo de liquidaciones.

### **Tipos de Investigación**

Una vez definido el enfoque, es necesario determinar el alcance o tipo de investigación para precisar el alcance del estudio. El tipo de investigación establece la finalidad del estudio y el nivel de profundidad con el que se abordará el fenómeno. La elección del tipo depende directamente de las preguntas de investigación y las metas del proyecto.

## **Investigación Exploratoria**

Este tipo de investigación se utiliza para examinar un tema o problema que ha sido poco estudiado o del que se tiene escasa información. Su objetivo principal es familiarizarse con un fenómeno desconocido para obtener una visión general y desarrollar futuras líneas de investigación.

## **Investigación Descriptiva**

La investigación descriptiva busca especificar las propiedades, características y perfiles de personas, grupos, comunidades, procesos, objetos o cualquier otro fenómeno que se somete a análisis. En este tipo de estudio, el investigar se centra en describir las características del objeto de estudio sin establecer relaciones causales entre variables.

## **Investigación Explicativa**

La investigación explicativa va más allá de la descripción de conceptos o fenómenos y se enfoca en conocer la relación o el grado de asociación que existe entre dos o más variables. Su objetivo es responder a las causas de fenómeno, es decir, por qué ocurre un determinado evento y en qué condiciones se manifiesta.

## **Tipo de Investigación Seleccionado**

El tipo de investigación seleccionado para este proyecto es la investigación Explicativa. Se opta este enfoque para realizar una investigación profunda que permita entender la relación de causa y efecto en la gestión de Recursos Humanos. Donde se busca conocer:

- Causa (Proceso Actual) ¿Qué debe hacer exactamente un colaborador para solicitar un permiso o vacaciones?

- Efecto (Gestión) ¿Cuánto tiempo requiere la jefatura para procesar una solicitud y actualizar el registro de horas y saldo?

La información se obtendrá directamente de los colaboradores y jefaturas de Tendenzza & puertAlum, lo que permitirá conocer el proceso a profundidad y entender las causas de la ineficiencia o los cuellos de botella existentes en la gestión de aguinaldos o liquidaciones. Esta comprensión detallada es fundamental para que la solución del prototipo aborde y resuelva las causas raíz de los problemas operacionales.

### **Fuentes de Información**

Las fuentes de información son fundamentales en cualquier proceso investigativo, ya que permiten sustentar el análisis con datos relevantes, verificables y organizados. A través de ellas se accede al conocimiento necesario para desarrollar, justificar y validar los contenidos de un proyecto académico. Existen diversos tipos de fuentes, clasificadas según el origen y la forma en que se obtiene la información.

A continuación, se describen los tres tipos principales de fuentes de información utilizadas en investigaciones académicas:

#### **Fuente de Información Primaria**

Las fuentes primarias son aquellas que proporcionan información directa y original, sin haber sido interpretada o modificada por terceros. Este tipo de fuente incluye entrevistas, encuestas, observaciones documentos oficiales y registros institucionales. Se consideran fundamentales porque permiten obtener datos desde la fuente misma del fenómeno estudiado. Según LibreTexts español (2023), las fuentes primarias “proporcionan evidencia de primera mano sobre un tema o evento” (párr. 1). Este tipo de información es clave para comprender la realidad desde la perspectiva de los actores involucrados, fortaleciendo la validez interna del estudio.

## **Fuente de Información Secundaria**

Las fuentes secundarias se componen de materiales que analizan, interpretan o resumen información proveniente de fuente primaria. Entre ellas se encuentran libros, artículos científicos, informes técnicos, revistas especializadas y publicaciones académicas. Estas fuentes permiten contextualizar el objetivo de estudio y contrastar diferentes enfoques teóricos. Como explica Lacorzana (2021), las fuentes secundarias “recogen información ya procesada por otros autores, con el fin de analizarla o interpretarla” (párr. 3). Su uso en la investigación permite enriquecer el marco teórico y sustentar los argumentos con base de estudios previos.

## **Fuente de Información Terciaria**

Las fuentes terciarias agrupan y organizan información proveniente de fuentes primarias y secundarias. Su función principal es facilitar el acceso a contenidos ya existentes mediante herramientas de búsqueda y recopilación. Ejemplos comunes son bibliografías, catálogos, índices temáticos, bases de datos académicas y motores de búsqueda como Google Scholar. Según ClasificaciónDe.org (2022), estas fuentes “sirven como guías para localizar información en otras fuentes” (párr. 2). Aunque no aportan contenido original, son esenciales para localizar referencias, verificar citas y estructurar la documentación del proyecto de forma eficiente y ordenada.

## **Variables**

Las variables de investigación son elementos fundamentales que permiten medir, analizar y comprender los fenómenos estudiados. Se definen como características que pueden adoptar distintos valores o categorías dentro de un contexto determinado. Su correcta identificación y clasificación es esencial para el diseño metodológico del estudio. Según Dupuis (2024), “una variable es una propiedad que puede tomar diferentes valores o categorías en un determinado contexto” (párr. 1). Esta capacidad de cambio permite que las variables sean manipuladas, observadas o controladas según los objetivos del proyecto.

## **Tipos de Variables Consideradas**

En este proyecto se contemplan tres tipos de variables, cada una con una función específica en el proceso de recolección y análisis de datos:

### **Variables conceptuales.**

Son aquellas que se definen por su significado teórico, expresadas mediante conceptos abstractos relacionados con el objetivo de estudio. Estas variables permiten establecer el marco conceptual que orienta la investigación y delimitan el enfoque analítico del proyecto.

### **Variables operacionales.**

Las variables operacionales se vinculan con la aplicación práctica de los conceptos teóricos. En este proyecto, se utilizan para describir situaciones concretas de la empresa mediante instrumentos como entrevistas, facilitando la obtención de datos relevantes y medibles.

### **Variables instrumentales.**

Corresponden a los medios utilizados para recolectar información, como cuestionarios, formularios o guías de entrevista. Estas herramientas permiten transformar los conceptos en datos observables, facilitando el análisis y la interpretación de los resultados.

Próximamente se presenta la tabla de variables, donde se detallan sus definiciones y la relación directa que tienen con los objetivos específicos del proyecto de graduación.

Tabla 10

Cuadro de variables.

Objetivo específico	Variable	Variable conceptual	Variable operacional	Variable instrumental
Examinar los procedimientos vigentes en el área de Recursos Humanos, incluyendo la gestión de personal, el control de permisos y otros elementos clave vinculados al manejo interno del talento humano	<i>Requerimientos</i>	Según Lacruz (2022), “El departamento de recursos humanos se encarga de administrar todo lo relacionado con el capital humano de la organización, desde su ingreso hasta su desvinculación” (párr. 2).	Observación directa Entrevistas	Guía de observaciones Guía de entrevistas
Diseñar un sistema integral de automatización según las necesidades que se identifiquen durante el análisis.	<i>Diseño de la automatización</i>	IBM (2023) afirma que “la automatización es la aplicación de tecnología, programas o procesos para lograr resultados con una intervención humana mínima” (párr. 1).	Diagramas de flujo Diseño de base de datos	MySQL Workbench Diagrams.net Laravel (PHP) MVC
Programar la solución de automatizada conforme a los requerimientos técnicos definidos, integrando funcionalidades específicas para la gestión de Recursos Humanos.	<i>Programar Requisitos</i>	Tejera-Martínez et al. (2020) afirma que “la programación consiste en traducir ideas y soluciones en instrucciones comprensibles para un sistema informático, utilizando lenguajes estructurados”	Front End Back End Base de Datos	Visual Studio Code MySQL Workbench Laravel (PHP) Visual Studio Code jQuery Bootstrap Velzon MVC MySQL Workbench

		<p>(párr. 3).</p> <p>Por su parte, Sommerville (2020) define los requisitos como “una declaración de lo que el sistema debe hacer o de las características que debe tener” (p. 45).</p>		
<p>Realizar pruebas exhaustivas y evaluaciones de calidad en la solución de automatización de Recursos Humanos para el prototipo final.</p>	<p><i>Pruebas exhaustivas</i> <i>Calidad</i></p>	<p>Según Rodríguez y Méndez (2022), las pruebas exhaustivas buscan validar el comportamiento del sistema ante múltiples combinaciones de datos y escenarios posibles” (párr.2).</p> <p>Por su parte, López (2021) señala que “la calidad del software implica que el producto cumpla con los requisitos funcionales, sea eficiente y fácil de mantener” (párr. 3).</p>	<p>Pruebas funcionales Validación de Resultados</p>	<p>Guía de pruebas Registro de incidencias Laravel Spatie (autenticación y permisos)</p>

Fuente: Elaboración Propia (2026)

## **Instrumentos de Recolección de Datos**

Los instrumentos de recolección de datos permiten obtener información relevante y verificable para el desarrollo del proyecto. En esta investigación se aplicaron técnicas cualitativas que facilitaron el análisis de los procesos actuales en el área de Recursos Humanos, así como la identificación de oportunidades de automatización. Los instrumentos seleccionados fueron la entrevista, la observación directa y el análisis de documentos de archivo.

### **Entrevista**

Según Santos (2025) menciona que “este método consiste en recopilar información formulando preguntas. A través de la comunicación interpersonal, el emisor obtiene respuestas verbales del receptor sobre un tema o problema específico” (párr. 1).

En este proyecto, la entrevista se aplicó al personal de jefatura del área de Recursos Humanos, incluyendo al encargado de permisos, vacaciones y cálculo de aguinaldos. Las sesiones se realizaron de forma virtual y presencial, permitiendo conocer cómo se gestionan actualmente estos procesos. Gracias a esta técnica, se identificaron los procedimientos manuales que requieren automatización, como el registro de solicitudes, la validación por parte del jefe de área, y el cálculo de beneficios laborales.

### **Observación**

Según Santos (2025) la observación “es una forma discreta y sencilla de inspeccionar datos sin depender de un intermediario. Este método se caracteriza por no ser intrusivo y requiere evaluar el comportamiento del objeto de estudio por un tiempo continuo sin intervenir” (párr. 1). Durante el desarrollo del sistema, se aplicó observación directa en el entorno laboral. El investigador, en su rol de desarrollador, pudo evidenciar cómo los colaboradores realizan solicitudes de permisos directamente al jefe de área sin un sistema automatizado. También se observó el uso de hojas de cálculo para el cálculo de aguinaldos, lo cual representa una oportunidad clara para digitalizar y optimizar estos procesos.

## **Documentos de Archivo**

Se revisaron documentos internos como planillas de Excel, formularios físicos de solicitud de permisos, registros de asistencia y reportes de vacaciones. Estos archivos permitieron validar la información obtenida en entrevistas y observación, y sirvieron como base para el diseño de la base de datos y los flujos funcionales del sistema.

## **Procesos de Recolección y Análisis de Datos**

El proceso se desarrolló en tres etapas:

- **Planeación:** Se definieron los objetivos de la observación y entrevista, así como los instrumentos de registro (guía de entrevista y ficha de observación).
- **Aplicación:** Se realizaron entrevistas semiestructuradas y observaciones directas en el entorno laboral. Paralelamente, se recopilaron documentos de archivo relevantes.
- **Análisis:** Los datos fueron organizados, categorizados, analizados para identificar patrones, necesidades y oportunidades de mejora. Esta información fundamentó el diseño del sistema de automatización para el área de Recursos Humanos.

## CAPITULO IV: ANÁLISIS DE RESULTADOS

Para comprender a detalle la situación actual de Recursos Humanos de Tendenzza & puertAlum, fue necesario aplicar instrumentos de recolección de datos que permitieran obtener información real, verificable y directamente asociada con los procesos administrativos que se desea automatizar. El instrumento principal utilizado fue la entrevista estructurada, elaborada con una serie de preguntas específicas orientadas a conocer cómo se gestionan actualmente tareas como el registro de colaboradores, el manejo de permisos, las vacaciones, el cálculo de aguinaldo, la elaboración de la nómina, la administración de incapacidades y el control de asistencia.

La entrevista fue aplicada al personal encargado de estas funciones dentro del área de Recursos Humanos, lo que permitió obtener respuesta claras y detalladas acerca de los procedimientos que se llevan a cabo en el día a día. Las preguntas utilizadas entre las que se incluyen interrogantes de ¿Como se realiza actualmente el registro de nuevos colaboradores?, proporcionaron una visión precisa del funcionamiento actual y permitieron identificar las oportunidades de mejora.

Como complemento a la entrevista, también se utilizó observación directa en el entorno laboral, lo cual permitió corroborar aspectos mencionados durante la conversación, especialmente en lo relacionado con el uso de documentos físicos, el procedimiento manual, de datos y la dependencia de hojas de cálculo. Además, se revisaron documentos de archivos, tales como planillas, solicitudes de permisos, registros de asistencia y reportes de vacaciones, que sirvieron para validar la información obtenida.

La combinación de estos instrumentos permitió detectar que la mayoría de los procesos se realizan de forma manual, lo que genera inconsistencias, pérdida de información, duplicidad de datos y retrasos en tareas administrativas que afectan la eficiencia del departamento. Una vez recopilada la información, esta fue organizada y estudiada con el fin de identificar los requerimientos que deberá cumplir el sistema propuesto.

El análisis de los datos obtenidos mediante la entrevista y complemento con la observación y la revisión documental constituye la base para el diseño y desarrollo del prototipo funcional, permitiendo que la solución responda de manera adecuada a las necesidades reales de la empresa y contribuya el mejoramiento de sus procesos internos.

¿Cómo se realiza actualmente el registro de nuevos colaboradores?

El registro de nuevos colaboradores se lleva a cabo mediante procedimientos manuales. La información se anota en una hoja de formulario que creamos y se recopila a través de archivos individuales, donde se incluyen los datos personales, el puesto que se le asignó y algunos detalles básicos necesarios para su incorporación. Como mencione anteriormente esto se apoya principalmente en papel y en registros administrados por mí la encargada de Recursos Humanos.

¿Qué tipo de información se almacena sobre cada empleado?

Todo lo que es relacionado con la jornada laboral, como las horas trabajadas, horas extra, permisos solicitados, ausencias y registros de asistencia. También se incluyen datos sobre períodos de vacaciones, incapacidades, aspectos vinculados a la gestión de nómina. Esta información se organiza en archiveros de forma manual en carpetas que administra el departamento.

¿Existe algún sistema digital para gestionar expedientes laborales?

Todo se gestiona en expedientes laborales de forma manual. Los datos se conservan en carpetas y en documentos independientes almacenados por el área. Aunque pueda existir apoyo ocasional en herramientas como Excel, no se utiliza un sistema centralizado para integrar toda la información de los expedientes en una única plataforma digital.

¿Cuál es el procedimiento para solicitar permisos laborales?

Las solicitudes se realizan por medio de formularios en papel o través de correos o incluso por WhatsApp que van dirigidos hacia al encargado correspondiente. Cada colaborador envía una su solicitud indicando el motivo y las fechas requeridas, posteriormente se espera la revisión y la confirmación por parte del responsable.

¿Quién valida o aprueba estas solicitudes?

La aprobación de los permisos la realiza el encargado del departamento o la persona responsable de Recursos Humanos. Ellos revisan la solicitud, verifican la información proporcionada y confirman si procede o no, comunicando posteriormente la decisión al colaborador solicitante.

¿Se lleva un registro histórico de permisos otorgados?

Se puede decir que sí, pero este se conserva en formatos manuales, tales como notas internas, archivos físicos, comprobantes que traen e inclusive muchas se hacen de forma verbal o WhatsApp. En la medida de lo posible cada permiso que se guarda según lo que utilice el área y esto permite contar con un historial básico que refleja las solicitudes y aprobaciones realizadas.

¿Cómo se gestionan las solicitudes de vacaciones?

Las solicitudes también se gestionan a través de un proceso manual. El colaborador la realiza por medio de formularios en papel, correos, por WhatsApp que igualmente van dirigidas al encargado correspondiente, este a su vez revisa el tiempo disponible y los días acumulados. Una vez que se revisa la información, se confirma la disponibilidad de fechas y se procede a registrar la solicitud dependiendo del área que sea.

¿Existen algún calendario o sistema que controle los días acumulados por cada empleado?

El control se realiza de manera manual. La información se va registrando en documentos tipo lista que permite llevar un seguimiento de los días disponibles para cada colaborador.

¿Cómo se realiza el cálculo del aguinaldo actualmente?

El cálculo del aguinaldo se efectúa mediante un proceso manual que toma en cuenta la información recopilada sobre los días laborados, ausencias y otros datos relevantes del periodo. Este cálculo se realiza utilizando los registros disponibles y se apoya en métodos establecidos por el área de Recursos Humanos, basados en el código de trabajo.

¿Qué herramienta se utiliza para este proceso (Excel, software,)?

Para el cálculo de aguinaldo utilizamos herramientas básicas como hojas de cálculo creadas en Excel. Con esto organizamos la información necesaria y facilitamos los cálculos de forma práctica dentro del equipo de trabajo.

¿Qué aspectos considera que podrían mejorar si se automatizan estos procesos?

Nos podría aportar beneficios como rapidez en el procesamiento de información, una organización más estructurada de los registros y la posibilidad de acceder a datos actualizados de manera inmediata. También podría facilitar la gestión de solicitudes y mejorar la precisión en tareas administrativas relacionadas con horarios, permisos, vacaciones y cálculos periódicos.

¿Ha tenido experiencia previa con sistemas automatizados en otras áreas?

El área de Recursos Humanos realiza procesos principalmente de forma manual; sin embargo, la empresa sí ha tenido acercamientos a soluciones tecnológicas en otros departamentos, especialmente relacionados con la implementación de herramientas digitales para ambientes inteligentes y sistemas modernos dentro de sus operaciones generales.

¿Cómo se lleva el control de las incapacidades reportadas por los colaboradores?

Actualmente, el control de incapacidades se realiza de forma manual. Cuando un colaborador presenta una incapacidad, debe enviar el documento por correo, por WhatsApp o entregarlo de forma física, y a partir de ahí se registra la información en hojas o archivos independientes.

El seguimiento de los días utilizados se realiza revisando los documentos enviados y comparándolos con otros registros manuales del área. Esto requiere verificar cada caso individualmente para determinar cuántos días corresponden y cuántos aún tiene disponibles el colaborador.

Dado que no existe una plataforma centralizada, el control se apoya en papeleo, correos y anotaciones, lo que hace que el registro dependa de la revisión manual de la información enviada por cada empleado.

¿Cuál es el procedimiento actual para generar la nómina semanal o mensual?

El proceso de nómina se gestiona revisando varios documentos manuales, como hojas donde se registran horas extra, ausencias, deducciones aplicables y bonificaciones especiales.

Toda esta información se recopila desde diferentes fuentes, como marcas que se anotaron, correos de permisos, actualizaciones enviadas por supervisores y cálculos en hojas

individuales. Posteriormente, estos datos se procesan de forma manual para calcular salarios, montos por horas trabajadas, horas extraordinarias, deducciones y otros ajustes necesarios.

La elaboración final de la nómina requiere consolidar toda esta información y revisarla antes de generar los pagos correspondientes, lo que implica un proceso detallado y dependiente de múltiples pasos.

¿Se mantiene algún registro histórico de horas extra, ausencias o maracas realizadas por los colaboradores?

El registro histórico se maneja con anotaciones manuales que la empresa conserva en documentos físicos o archivos aislados.

En el caso de las horas extra, asistencia, permisos y ausencias, la información suele quedar distribuida en diferentes hojas o correos que se archivan para referencia en el futuro. Esto significa que, aunque sí existe un historial en lo que cabe y podemos, todo no está centralizado en una única plataforma, sino que depende de documentos separados que hay que revisar individualmente cuando se necesita consultar un dato específico.

Esto nos obliga a buscar la información manualmente entre los registros previos para verificar datos históricos o realizar comparaciones entre periodos.

¿Qué tipo de errores son más frecuentes durante el cálculo de la planilla en salarios, deducciones, bonificaciones entre otros?

Los más comunes son el cálculo de horas extra, deducciones por ausencias y ajustes de las bonificaciones ya que muchas veces los datos no coinciden porque se recopilan de diferentes hojas, correos, mensajes de WhatsApp y esto genera inconsistencias, mucho tiempo y pagos incorrectos.

¿Qué problemas suelen presentarse cuando un colaborador solicita vacaciones?

El principal problema es que los días acumulados no siempre están actualizados, lo que provoca que se aprueben solicitudes con saldos insuficientes. También se generan retrasos para aprobarlas ya que la información tiene que revisarse manualmente en listas, correos, mensajes de WhatsApp, y en ocasiones se aprueban solicitudes duplicadas.

¿Con que frecuencia se detectan pagos incorrectos en horas extra y cómo se corrigen actualmente?

Bueno esto se presenta con bastante frecuencia, especialmente cuando los registros se hacen desde ubicaciones remotas. La corrección se realiza revisando nuevamente las hojas y ajustando manualmente la planilla, lo que retrasa el pago y genera reclamos de los colaboradores.

¿Qué problemas se generan cuando se intenta consultar el historial de permisos, incapacidades o ausencias de un colaborador?

El historial está disperso en diferentes documentos físicos, correos, notas y WhatsApp. Esto hace que consultar la información sea lento y poco confiable, ya que no siempre se encuentra todo lo que se necesita y en ocasiones se pierde información.

### **Análisis Resultante del Proceso de Observación y Revisión Documental**

Como parte del proceso de recolección de datos, se realizó una observación directa en la empresa con el fin de verificar de primera mano cómo se gestionan los procesos administrativos del área de Recursos Humanos. Durante la visita, me fue posible analizar los procedimientos utilizados para el manejo de asistencia, permisos, vacaciones, incapacidades y elaboración de planillas. Asimismo, se revisaron los documentos y archivos que conforman la estructura operativa diaria del departamento.

Durante esta observación se constató que todos los cálculos relacionados con la planilla (incluyendo horas trabajadas, horas extra, deducciones, vacaciones acumuladas, incapacidades y aguinaldo) se realizan mediante hojas de cálculo elaboradas en Excel. Estos archivos son modificados y actualizados por varias personas del departamento, lo cual incrementa significativamente el riesgo de errores por manipulación simultánea, versiones distintas o cambios no registrados. Esta situación es especialmente crítica debido a que cualquier error en la planilla afecta directamente los salarios de los colaboradores, lo que genera consecuencias administrativas, reclamos internos, y fallos en la gestión financiera de la empresa. El manejo manual de estos documentos convierte en procesos vulnerables y potencialmente catastróficos en términos de exactitud y control de la información.

Adicionalmente se confirmó que los registros vinculados a asistencia, permisos e incapacidades se encuentran dispersos entre documentos físicos, notas informales, correos electrónicos y mensajes de WhatsApp. Esta variedad de fuentes obliga al personal del área a consultar diferentes medios para validar un solo dato, lo que ralentiza el trabajo y aumenta la probabilidad de omitir información importante.

Es importante señalar que, debido a políticas internas de confidencialidad, no fue permitido tomar fotografías de los documentos revisados. Sin embargo, se pudo observar de manera detallada cada formulario, planilla y archivo utilizado actualmente, lo cual permitió corroborar plenamente que los procedimientos carecen de una plataforma centralizada y dependen de métodos manuales para el control de datos laborales.

El análisis de esta información, complementado con la revisión documental y las entrevistas realizadas, permite concluir que los procesos actuales presentan una alta vulnerabilidad operativa y una dependencia excesiva del manejo manual, lo que justifica la necesidad urgente de implementar un sistema automatizado que mejore la trazabilidad, reduzca errores y centralice la gestión del recurso humano. Este análisis constituye la base fundamental para el diseño del prototipo funcional que busca optimizar los procesos internos del área de Recursos Humanos.

## **CAPITULO V: PROPUESTA**

### **Análisis**

En esta etapa se estudia y organiza toda la información recopilada durante la investigación, con el fin de tomar decisiones clave para la construcción del sistema. Este análisis constituye la base para el desarrollo del prototipo, ya que permite transformar los datos obtenidos en acciones concretas que guiarán las siguientes fases del proyecto.

El proceso analítico facilita la definición de una aplicación ajustada a las necesidades reales del cliente, en este caso, el área de Recursos Humanos de Tendenzza & puertAlum. Se busca que la solución propuesta se integre de forma natural a sus métodos de trabajo, sea intuitiva para los colaboradores y optimice los procedimientos que actualmente se realizan de manera manual. Por esta razón, cada decisión tomada en esta etapa se fundamenta en la información suministrada por el cliente y validada mediante entrevistas, observación y revisión documental.

### **Análisis Detallado del Software**

El prototipo planteado para este proyecto está conformado por varios módulos funcionales que fueron definidos según los requerimientos identificados durante la fase de recolección de datos. Cada módulo cumple una función específica dentro del sistema y está orientado a resolver problemas detectados en los procesos actuales del departamento de Recursos Humanos. A continuación, se detalla la función de cada uno:

#### **Módulo de Gestión de Horas Extra.**

Este módulo permite que los colaboradores registren sus horas extra directamente en el sistema. La solicitud se envía a jefatura para su aprobación o rechazo, y el sistema notifica automáticamente a Recursos Humanos y al colaborador según la resolución. Esto facilita el control y evita el uso de formularios físicos o mensajes informales.

### **Módulo de Gestión de Vacaciones.**

El sistema calcula automáticamente los días de vacaciones disponibles, registra los días disfrutados y muestra el saldo actualizado. Los pueden solicitar vacaciones desde la plataforma, y la jefatura valida la solicitud. El sistema mantiene un historial de todas las gestiones y elimina errores comunes en procesos manuales.

### **Módulo de Gestión de Asistencia.**

Este módulo registra las entradas y salidas mediante la aplicación web utilizando geolocalización para validar que las marcas se realicen en zonas autorizadas. Se genera un historial completo de asistencia, atrasos y ausencias, lo que genera un registro preciso y confiable sin necesidad de hojas manuales.

### **Módulo de Gestión de Nómina.**

El sistema automatiza el cálculo de la planilla considerando salario base, horas trabajadas, horas extra, deducciones legales, renta, incapacidades y permisos. Centraliza la información proveniente de otros módulos para generar cálculos exactos y reducir errores propios de la elaboración manual.

### **Módulo de Gestión de Permisos.**

Este módulo gestiona las solicitudes de permisos o licencias. El colaborador realiza la solicitud en el sistema, la jefatura la aprueba o rechaza, y el sistema notifica a Recursos Humanos y al colaborador. La trazabilidad evita pérdida de información y elimina la dependencia de métodos manuales.

### **Módulo de Evaluación de Desempeño.**

Permite aplicar evaluaciones periódicas registrando calificaciones y observaciones sobre el rendimiento de cada colaborador. El sistema genera un historial que facilita la toma de decisiones sobre bonificaciones, ascensos y necesidades de capacitación.

### **Módulo de Gestión de Incapacidades.**

Centraliza el registro de incapacidades, permitiendo almacenar información como diagnóstico y fechas. Recursos Humanos valida los datos y el sistema actualiza automáticamente los días aplicables y los ajustes en la nómina.

### **Módulo de Gestión de Liquidaciones.**

Automatiza el cálculo de liquidaciones considerando salario, antigüedad, vacaciones pendientes e indemnizaciones según la legislación vigente. Reduce errores y agiliza el proceso, ya que utiliza toda la información acumulada en el sistema.

### **Módulo de Cálculo de Aguinaldo.**

Calcula automáticamente el aguinaldo en función de los meses trabajados y los ingresos del año. El sistema asegura exactitud y permite integrar el resultado directamente a la nómina correspondiente.

### **Módulo de Mantenimientos.**

Permite administrar los registros de la base de datos MySQL mediante operaciones CRUD, para entidades como colaboradores, vacaciones, incapacidades, horas extra y permisos, su objetivo es mantener actualizada la información que utilizará el resto del sistema.

El diseño se basa en la arquitectura MVC (Modelo-Vista-Controlador), donde los modelos representan las tablas de la base de datos, controladores manejan la lógica del sistema, y las vistas muestran la información al usuario.

### **Módulo de Consultas.**

Este módulo facilita la visualización ordenada de la información almacenada de la base de datos. Los usuarios pueden consultar datos en tiempo real mediante pantallas diseñadas para una fácil comprensión y accesos rápido.

Se utiliza la arquitectura MVC para recuperar los datos desde la base de datos y mostrarlos en tablas interactivas con opciones de búsqueda, ordenamiento y paginación, usando los componentes visuales del tema Velzon.

### **Módulo de Reportes.**

Este módulo permite mostrar información resumida o filtrada según los criterios del usuario. Incluye reportes de permisos, vacaciones, incapacidades, horas extra, asistencias, nómina y evaluaciones de desempeño.

Además, ofrece la posibilidad de exportar los reportes en formatos de PDF y Excel para facilitar el respaldo y presentación de información. Se utilizan gráficas del tema Velzon para representar visualmente los indicadores más importantes.

### **Módulo de Seguridad.**

Este módulo se encarga del control de acceso al sistema. Valida que cada usuario tenga credenciales validas y los permisos necesarios para ingresar a las distintas secciones de la aplicación.

Los usuarios están asociados a roles (Administrador, jefe de RH, jefe de Operaciones, entre otros), los cuales determinan qué funciones pueden utilizar. La autenticación se maneja con las herramientas de seguridad de Laravel, incluyendo protección de contraseñas y control de sesiones.

## **Análisis Detallado del Software**

El prototipo funcional se desarrolla utilizando tecnologías web modernas que garantizan estabilidad, seguridad y facilidad de mantenimiento. Para la lógica del sistema se emplea PHP 8.3, mientras que la estructura y presentación de la interfaz se construyen con HTML 5, CSS 3 y JavaScript, permitiendo una interacción eficiente entre el usuario y aplicación.

El framework Laravel 12 se utiliza como base del desarrollo bajo la arquitectura MVC, complementado con jQuery 3.7, Bootstrap 5.3 y la plantilla administrativa Velzon para mejorar la experiencia de usuario. La gestión de información se realiza mediante MySQL 8.4, administrada con MySQL Workbench 8.0.43, y el desarrollo del sistema se lleva a cabo en el entorno Visual Studio Code 1.104.2, facilitando un proceso ordenado y controlado.

## **Análisis Detallado del Hardware**

Para el desarrollo del prototipo no se generaron costos asociados al hardware, ya que el desarrollador utiliza su propio equipo personal, el cual cuenta con las especificaciones necesarias para ejecutar todas las herramientas de software involucradas durante el proceso de construcción del sistema. Este equipo permitió realizar las actividades de programación, pruebas y documentación sin limitaciones técnicas. A continuación, se describe el equipo utilizado para el desarrollo:

- **Laptop:** HP Ryzen 5 7535HS
- **Memoria RAM:** 16 GB
- **Almacenamiento:** SSD de 512 GB
- **Sistema operativo:** Windows 11 Home

Estas características son suficientes para garantizar un entorno estable y eficiente durante el proceso de desarrollo del prototipo

En cuanto a la puesta en producción, la empresa Tendenzza & puertAlum cuenta con los recursos tecnológicos necesarios para implementar el sistema sin necesidad de realizar inversiones adicionales en hardware. Todos los colaboradores cuentan disponen de computadoras portátiles HP con procesador Intel Core I5 1335U, con 16 GB de memoria RAM y disco sólido (SSD) DE 256 GB, todas operando bajo Windows 11 Home, lo que asegura un rendimiento óptimo para el uso del sistema web propuesto.

Además, no se requiere adquirir licencias especiales ni componentes físicos adicionales, ya que las herramientas a utilizar en la etapa de operación son compatibles con la infraestructura existente. Únicamente se recomienda que los equipos de los colaboradores cuenten con Windows 10 o superior, lo cual garantiza compatibilidad total con la plataforma y una correcta experiencia de usuario.

Para la puesta en producción del sistema, se va a utilizar un servidor virtual privado (VPS) en la nube, el cual permitirá alojar la aplicación web y garantizar su disponibilidad para los usuarios de la empresa. El uso de infraestructura en la nube ofrece ventajas como acceso remoto, estabilidad del servicio, respaldo de la información y posibilidad de escalar los recursos si el sistema lo requiere en el futuro.

Las características mínimas recomendadas para el servidor son un procesador 2vCPU, memoria RAM de 4GB, almacenamiento de 45 GB SSD, una dirección IP pública IPv4, una velocidad de red de 150 Mbps compartidos, y un sistema operativo Linux (Ubuntu Server o similar), donde permitirá ejecutar en un mismo entorno el servidor web, el servidor de aplicación y el servidor de base de datos, recursos que son suficientes para soportar la carga del sistema en la empresa. Esta combinación de 2 núcleos de procesamiento y 4 GB de memoria RAM permite gestionar adecuadamente las solicitudes de los usuarios, mientras que el almacenamiento SSD garantiza un acceso rápido a la información almacenada en la base de datos.

### **Análisis Detallado de Telecomunicaciones**

Para la implementación en línea del sistema, la empresa realiza una inversión básica en infraestructura en la nube, mediante la adquisición de un dominio web y la contratación

de un servidor privado virtual (VPS). Esta infraestructura permitirá alojar el sistema y garantizar su acceso remoto de forma segura y continua para los colaboradores autorizados.

La empresa cuenta con un servicio de internet, al cual las computadoras se conectan mediante WIFI. Cuando los colaboradores realizan labores fuera de las instalaciones, utilizan dispositivos de conexión WIFI móvil proporcionados por la empresa, cuyo costo no se incluye en este proyecto. Tendenzza & puertAlum no dispone de un centro de TI propio, por lo que el uso de servicios en la nube resulta una solución adecuada al no requerir infraestructura física adicional ni personal técnico especializado.

### **Análisis Detallado de Herramientas y licencias con costos y capacidades.**

Para el desarrollo del sistema se emplearon los lenguajes PHP, JavaScript, HTML 5 y CSS 3, los cuales permiten la construcción de una aplicación web dinámica, accesible y compatible con distintos navegadores. La lógica del sistema se implementa del lado del servidor con PHP, mientras que las tecnologías del lado del cliente permiten una adecuada interacción con el usuario.

El framework utilizado es el Laravel 12, lo cual facilita la organización del código bajo el patrón Modelo-Vista-Controlador (MVC). Para la interfaz gráfica se utilizó Bootstrap 5.3 junto con la plantilla administrativa Velzon, la cual cuenta con costo único de aproximadamente 25 dólares. La librería jQuery 3.7 se empleó para mejorar la experiencia con el usuario. Para la gestión de datos se utiliza MySQL 8.4 Community Edition, junto con MySQL Workbench, ambos de uso gratuito. El entorno de desarrollo es Visual Studio Code, sin costos de licenciamiento.

### **Análisis Detallado del personal requerido para el uso del sistema.**

El sistema desarrollado cuenta con una interfaz gráfica amigable, intuitiva y orientada al usuario, en la cual los módulos, botones y formularios poseen nombres claros que facilitan su comprensión y uso. Para su correcta utilización, el personal requiere conocimientos básicos en el manejo de computadoras y aplicaciones web, tales como el uso de un navegador, ingreso de información en formularios y navegación entre distintas pantallas en el sistema.

En el caso de Tendenzza & puertAlum, estos requerimientos no representan una limitación, ya que la empresa se caracteriza por integrar soluciones tecnológicas avanzadas en sus procesos operativos. Sus colaboradores cuentan con experiencia en el uso de herramientas digitales y tecnologías modernas, debido a que la organización trabaja constantemente con sistemas inteligentes, automatización de espacios, entre otros en su operación diaria. Esto facilita la adopción del sistema sin generar una curva de aprendizaje compleja.

No obstante, se recomienda brindar una capacitación breve, ya sea presencial o virtual, enfocada en el uso de los módulos principales del sistema en los flujos de trabajo definidos, con el fin de asegurar que esta capacitación no forma parte del alcance del presente proyecto.

### **Análisis del Conocimiento del personal requerido para el mantenimiento del sistema.**

Para garantizar la correcta operación, continuidad y disponibilidad del sistema, se recomienda que la empresa cuente con un responsable del mantenimiento técnico. El perfil requerido puede ser el de un técnico o desarrollador junior en sistemas, ya que no se trata de un sistema de alta complejidad ni de infraestructura crítica.

Este perfil debe poseer conocimientos básicos en aplicaciones web, manejo de PHP, JavaScript, HTML, CSS, así como nociones del framework Laravel, uso de MySQL como motor de base de datos y administración básica del servidor donde se aloje el sistema. Además, se espera que tenga habilidades para realizar tareas de mantenimiento preventivo, respaldo de información, actualización de dependencias y atención de incidencias menores.

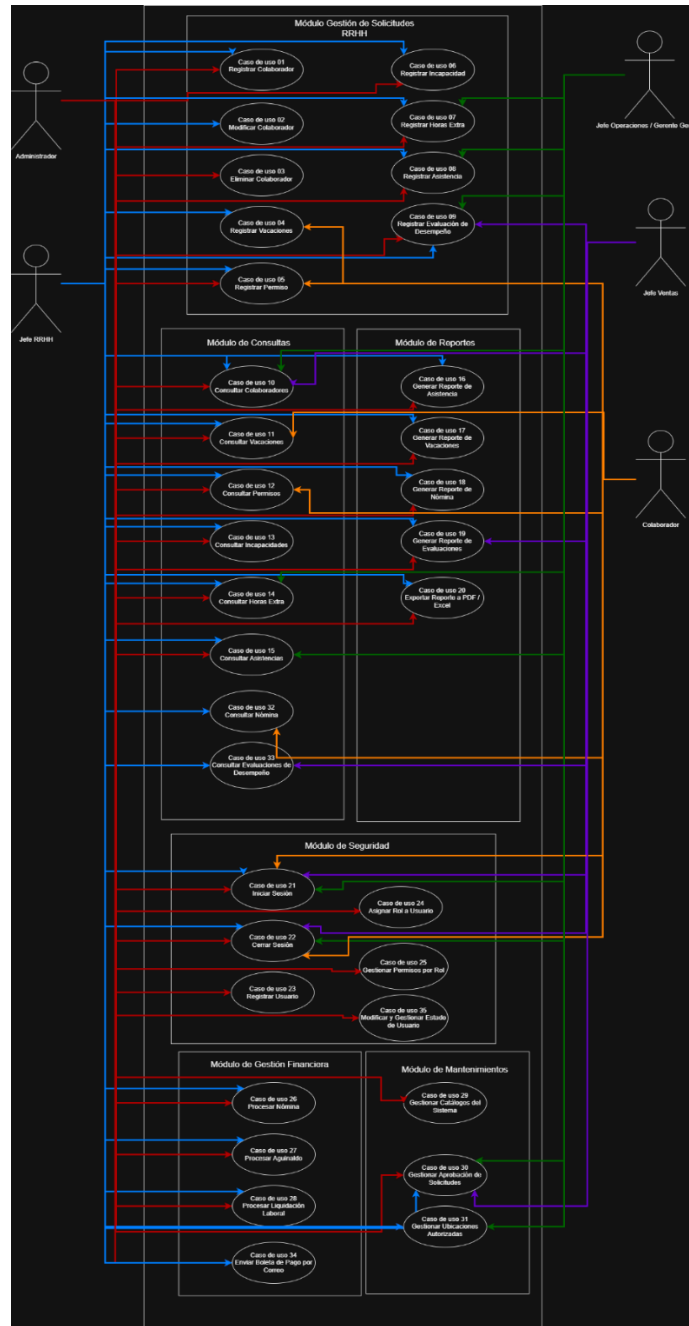
### **Diagrama de casos de uso.**

El siguiente diagrama de casos de uso representa las diferentes funcionalidades del sistema RRHH – Tendenzza y su interacción con los actores del sistema. Cada caso de uso describe una acción específica que los usuarios pueden realizar según su rol asignado. Los actores identificados son: Administrador, jefe de RRHH, jefe de Operaciones, jefe de Ventas,

y Colaborador, cada uno con permisos y accesos diferenciados según sus responsabilidades dentro de la organización.

**Figura 1**

*Diagrama Casos de Uso.*



Fuente: Elaboración Propia (2026)

A continuación, se representa la descripción detallada de cada caso de uso

## Casos de Uso

### Módulo Gestión de Solicitudes RRHH.

**Tabla 11**

*Caso de uso 1.*

<b>Prototipo: Sistema RRHH – Tendenza</b>	
<b>Número Caso de Uso: 01</b>	<b>Nombre del Caso de Uso:</b> Registrar Colaborador
<b>Fecha elaboración:</b>	13/12/2025
<b>Descripción Caso de Uso:</b>	Permite registrar un nuevo colaborador en el sistema con sus datos personales, de contacto y laborales.
<b>Autor caso de uso:</b>	Miguel
<b>Actores relacionados:</b>	Administrador, jefe RRHH
<b>Precondiciones:</b>	<ul style="list-style-type: none"> <li>- El usuario debe estar autenticado con el rol de Administrador o de jefe RRHH.</li> <li>- Deben existir los catálogos de género, tipo de identificación, ocupación y ubicación geográfica.</li> </ul>
<b>Flujo Básico del caso de uso</b>	
En este caso de uso comienza cuando el usuario desea registrar un nuevo colaborador en el sistema. El usuario accede al módulo de colaboradores, selecciona la opción para agregar un nuevo registro y completa el formulario con la información solicitada. El sistema valida los datos ingresados y, una vez confirmada la operación, guarda la información y muestra un mensaje de confirmación	
<b>Detallar el paso a paso del Flujo Básico</b>	
<ol style="list-style-type: none"> <li>1. El usuario accede al módulo de Colaboradores desde el menú principal.</li> <li>2. Selecciona la opción ‘Agregar colaborador’.</li> <li>3. El sistema muestra el formulario de registro.</li> <li>4. El usuario ingresa los datos personales (nombre, apellidos, identificación, género, fecha de nacimiento). <ul style="list-style-type: none"> <li>• Si la identificación ingresada ya existe en el sistema, se ejecuta el flujo alternativo 1.</li> </ul> </li> <li>5. El usuario ingresa los datos de contacto (teléfono, correo, dirección). <ul style="list-style-type: none"> <li>• Si existen campos obligatorios incompletos o con formato incorrecto, se ejecuta el flujo alternativo 2.</li> </ul> </li> <li>6. El usuario ingresa los datos laborales (fecha de ingreso, salario base, ocupación).</li> <li>7. El usuario selecciona ‘guardar’. <ul style="list-style-type: none"> <li>• Si el usuario selecciona la opción ‘Cancelar’, se ejecuta el sub-Flujo 1.</li> </ul> </li> <li>8. El sistema valida los datos ingresados.</li> <li>9. El sistema solicita confirmación para registrar el colaborador.</li> <li>10. El usuario confirma el registro.</li> <li>11. El sistema guarda el colaborador en la base de datos.</li> <li>12. El sistema muestra el mensaje ‘Colaborador registrado correctamente’.</li> </ol>	
<b>Sub-Flujos</b>	
<b>Sub-Flujo 1: Cancelar operación</b>	<ol style="list-style-type: none"> <li>1. El usuario selecciona ‘Cancelar’</li> <li>2. El sistema descarta la información ingresada.</li> <li>3. El sistema regresa a la pantalla de lista de colaboradores.</li> </ol>
<b>Flujos Alternos</b>	
Corresponde a lo que debe realizar el sistema ante posibles errores	

<b>Flujo Alternativo 1: Identificación duplicada</b>	<ol style="list-style-type: none"> <li>1. El sistema detecta que la identificación ingresada ya se encuentra registrada.</li> <li>2. El sistema muestra el mensaje 'La identificación ya está registrada'.</li> <li>3. El flujo retorna al paso 4 del flujo Básico.</li> </ol>
<b>Flujo Alternativo 2: Campos incompletos</b>	<ol style="list-style-type: none"> <li>1. El sistema detecta que existen campos obligatorios vacíos o con formato incorrecto.</li> <li>2. El sistema resalta los campos que requieren corrección.</li> <li>3. El flujo retorna al paso correspondiente del flujo Básico.</li> </ol>
<b>Requerimientos especiales</b>	
<ol style="list-style-type: none"> <li>1. La identificación debe ser única en la base de datos.</li> <li>2. El colaborador debe ser mayor de 18 años.</li> <li>3. El salario debe ser numérico y mayor a cero.</li> <li>4. El correo debe tener un formato válido.</li> <li>5. Solo roles autorizados pueden acceder a esta función.</li> </ol>	
<b>Postcondiciones</b>	
<ol style="list-style-type: none"> <li>1. El colaborador queda registrado en la base de datos.</li> <li>2. Se genera automáticamente su saldo de vacaciones.</li> <li>3. El colaborador está disponible para los módulos de asistencia, vacaciones, permisos y nómina.</li> </ol>	

**Tabla 12***Caso de uso 2.*

<b>Prototipo: Sistema RRHH – Tendenzza</b>	
<b>Número Caso de Uso: 02</b>	<b>Nombre del Caso de Uso:</b> Modificar Colaborador
<b>Fecha elaboración:</b>	13/12/2025
<b>Descripción Caso de Uso:</b>	Permite modificar los datos de un colaborador existente, asegurando que los cambios queden reflejados en todos los módulos relacionados.
<b>Autor caso de uso:</b>	Miguel
<b>Actores relacionados:</b>	Administrador, jefe RRHH
<b>Precondiciones:</b>	<ul style="list-style-type: none"> <li>- El usuario debe estar autenticado con el rol de Administrador o de jefe RRHH.</li> <li>- El colaborador debe existir en la base de datos.</li> </ul>
<b>Flujo Básico del caso de uso</b>	
Este caso de uso comienza cuando el usuario desea modificar la información de un colaborador existente. El usuario busca al colaborador, selecciona la opción editar, realiza los cambios necesarios y confirma la operación para que el sistema valide y actualice la información.	
<b>Detallar el paso a paso del Flujo Básico</b>	
<ol style="list-style-type: none"> <li>1- El usuario accede al módulo de colaboradores desde el menú principal.</li> <li>2- El sistema muestra la lista de colaboradores registrados.</li> <li>3- El usuario busca el colaborador que desea modificar. <ul style="list-style-type: none"> <li>• Si el colaborador no es encontrado, se ejecuta el Flujo Alternativo 1.</li> </ul> </li> <li>4- El sistema muestra el colaborador encontrado.</li> <li>5- El usuario selecciona la opción 'editar'.</li> <li>6- El sistema muestra el formulario con los datos actuales del colaborador.</li> <li>7- El usuario modifica los datos necesarios (personales, contacto o laborales).</li> <li>8- El usuario selecciona 'guardar cambios'. <ul style="list-style-type: none"> <li>• Si el usuario selecciona la opción 'Cancelar', se ejecuta el sub-Flujo 1.</li> </ul> </li> <li>9- El sistema valida los datos modificados.</li> <li>10- El sistema solicita confirmación para actualizar el colaborador.</li> <li>11- El usuario confirma la modificación.</li> </ol>	

12- El sistema actualiza los datos en la base de datos. 13- El sistema muestra el mensaje ‘Colaborador actualizado correctamente’.	
<b>Sub-Flujo 1: Cancelar operación</b>	
	1- El usuario selecciona ‘Cancelar’. 2- El sistema descarta los cambios. 3- El sistema regresa a la lista de colaboradores.
<b>Flujos Alternos</b>	
Corresponde a lo que debe realizar el sistema ante posibles errores	
<b>Flujo Alternativo 1: Colaborador no encontrado</b>	1- El sistema no encuentra resultados en la búsqueda realizada. 2- El sistema muestra el mensaje ‘No se encontraron colaboradores’. 3- El flujo regresa al paso 3 del Flujo Básico.
<b>Flujo Alternativo 2: Campos incompletos</b>	1- El sistema detecta campos obligatorios vacíos o con formato incorrecto. 2- El sistema resalta los campos que requieren corrección. 3- El flujo regresa al paso 7.
<b>Requerimientos especiales</b>	
1- Solo roles autorizados pueden modificar colaboradores. 2- El sistema debe mantener un registro de la fecha de última modificación. 3- Los cambios en el salario deben reflejarse en los cálculos de nómina.	
<b>Postcondiciones</b>	
1- Los datos del colaborador quedan actualizados en la base de datos. 2- Los cambios se reflejan en todos los módulos relacionados.	

**Tabla 13***Caso de uso 3.*

<b>Prototipo: Sistema RRHH - Tendenza</b>	
<b>Número Caso de Uso:03</b>	<b>Nombre del Caso de Uso:</b> Eliminar Colaborador
<b>Fecha elaboración:</b>	13/12/2025
<b>Descripción Caso de Uso:</b>	Permite eliminar un colaborador del sistema cuando ya no forma parte de la empresa, siempre que no existan registros pendientes asociados que impidan su eliminación.
<b>Autor caso de uso:</b>	Miguel
<b>Actores relacionados:</b>	Administrador
<b>Precondiciones:</b>	<ul style="list-style-type: none"> <li>- El usuario debe estar autenticado con rol administrador.</li> <li>- El colaborador debe existir en la base de datos.</li> <li>- El colaborador no debe tener registros pendientes (nóminas sin procesar, vacaciones activas).</li> </ul>
<b>Flujo Básico del caso de uso</b>	
Este caso de uso comienza cuando el usuario desea eliminar un colaborador del sistema. El usuario accede al módulo de colaboradores, busca al colaborador requerido y selecciona la opción de eliminación. El sistema solicita confirmación, valida que no existan registros pendientes y finalmente, elimina al colaborador y actualiza la lista correspondiente.	
<b>Detallar el paso a paso del Flujo Básico</b>	

<ol style="list-style-type: none"> <li>1- El usuario accede al módulo de colaboradores desde el menú principal.</li> <li>2- El sistema muestra la lista de colaboradores registrados.</li> <li>3- El usuario busca al colaborador que desea eliminar. <ul style="list-style-type: none"> <li>• Si el colaborador no es encontrado, se ejecuta el Flujo Alterno 1.</li> </ul> </li> <li>4- El sistema muestra el colaborador encontrado.</li> <li>5- El usuario selecciona la opción ‘eliminar’.</li> <li>6- El sistema muestra mensaje de advertencia ¿Está seguro de que desea eliminar este colaborador?</li> <li>7- El usuario confirma la eliminación. <ul style="list-style-type: none"> <li>• Si el usuario selecciona la opción ‘Cancelar’, se ejecuta el sub-Flujo 1.</li> </ul> </li> <li>8- El sistema verifica que el colaborador no tenga registros pendientes. <ul style="list-style-type: none"> <li>• Si el colaborador posee registros pendientes, se ejecuta el Flujo Alterno 2.</li> </ul> </li> <li>9- El sistema elimina al colaborador de la base de datos.</li> <li>10- El sistema muestra el mensaje ‘Colaborador eliminado correctamente’.</li> <li>11- El sistema actualiza la lista de colaboradores.</li> </ol>	
<b>Sub-Flujos</b>	
<b>Sub-Flujo 1: Cancelar eliminación.</b>	<ol style="list-style-type: none"> <li>1- El usuario selecciona ‘cancelar’ en el mensaje de advertencia.</li> <li>2- El sistema cierra el mensaje de confirmación.</li> <li>3- El sistema regresa a la lista de colaboradores sin eliminar al colaborador.</li> </ol>
<b>Flujos Alternos</b>	
Corresponde a lo que debe realizar el sistema ante posibles errores	
<b>Flujo Alterno 1: Colaborador no encontrado.</b>	<ol style="list-style-type: none"> <li>1- El sistema no encuentra resultados en la búsqueda.</li> <li>2- El sistema muestra el mensaje ‘No se encontraron colaboradores’.</li> <li>3- El flujo regresa al paso 3 del Flujo Básico.</li> </ol>
<b>Flujo Alterno 2: Colaboradores con registros pendientes.</b>	<ol style="list-style-type: none"> <li>1- El sistema detecta que el colaborador tiene nóminas o vacaciones pendientes.</li> <li>2- El sistema muestra el mensaje ‘No se puede eliminar el colaborador, tiene registros pendientes’.</li> <li>3- El flujo termina sin realizar la eliminación.</li> </ol>
<b>Requerimientos especiales</b>	
<ol style="list-style-type: none"> <li>1- Solo el rol Administrador puede eliminar colaboradores.</li> <li>2- El sistema debe verificar que no existan registros dependientes antes de permitir eliminar.</li> <li>3- Se recomienda realizar una liquidación antes de eliminar un colaborador.</li> </ol>	
<b>Postcondiciones</b>	
<p>El colaborador queda eliminado de la base de datos.  Los registros históricos asociados se mantienen para efectos de reportes.  El colaborador ya no aparece en las listas activas del sistema.</p>	

**Tabla 14***Caso de uso 4.*

<b>Prototipo: Sistema RRHH – Tendenzza</b>	
<b>Número Caso de Uso:04</b>	<b>Nombre del Caso de Uso:</b> Registrar vacaciones
<b>Fecha elaboración:</b>	13/12/2025
<b>Descripción Caso de Uso:</b>	Permite registrar una solicitud de vacaciones para un colaborador, validando el saldo de días disponibles y asegurando que las fechas ingresadas sean correctas antes de almacenar la información.
<b>Autor caso de uso:</b>	Miguel
<b>Actores relacionados:</b>	Administrador, jefe RRHH, colaborador

<b>Precondiciones:</b>	<ul style="list-style-type: none"> <li>- El usuario debe estar autenticado en el sistema.</li> <li>- El colaborador debe existir en la base de datos.</li> <li>- El colaborador debe tener saldo de vacaciones disponible.</li> </ul>
<b>Flujo Básico del caso de uso</b>	
Este caso de uso comienza cuando el usuario desea registrar la solicitud de vacaciones. El usuario accede al módulo de vacaciones, selecciona al colaborador e ingresa el período solicitado. El sistema calcula automáticamente los días, valida el saldo disponible y tras la confirmación del usuario, registra la solicitud.	
<b>Detallar el paso a paso del Flujo Básico</b>	
<ol style="list-style-type: none"> <li>1- El usuario accede al módulo de vacaciones desde el menú principal.</li> <li>2- Selecciona la opción 'Registrar vacaciones'.</li> <li>3- El sistema muestra el formulario de registro.</li> <li>4- El usuario busca y selecciona el colaborador. <ul style="list-style-type: none"> <li>• Si el colaborador no es encontrado, se ejecuta el Flujo Alterno 1.</li> </ul> </li> <li>5- El sistema muestra el saldo de vacaciones disponibles del colaborador.</li> <li>6- El usuario ingresa la fecha de inicio y fecha fin de las vacaciones.</li> <li>7- El sistema calcula automáticamente los días solicitados.</li> <li>8- El usuario selecciona 'Guardar'. <ul style="list-style-type: none"> <li>• Si el usuario selecciona la opción 'Cancelar', se ejecuta el sub-Flujo 1.</li> </ul> </li> <li>9- El sistema valida que los días solicitados no excedan el saldo disponible. <ul style="list-style-type: none"> <li>• Si el saldo es insuficiente, se ejecuta el Flujo Alterno 2.</li> </ul> </li> <li>10- El sistema solicita confirmación para registrar las vacaciones.</li> <li>11- El usuario confirma el registro.</li> <li>12- El sistema guarda la solicitud en la base de datos.</li> <li>13- El sistema muestra el mensaje 'vacaciones registradas correctamente'.</li> </ol>	
<b>Sub-Flujos</b>	
<b>Sub-Flujo 1: Cancelar operación</b>	<ol style="list-style-type: none"> <li>1- El usuario selecciona 'Cancelar'.</li> <li>2- El sistema descarta los datos.</li> <li>3- El sistema regresa a la lista de vacaciones.</li> </ol>
<b>Flujos Alternos</b>	
Corresponde a lo que debe realizar el sistema ante posibles errores	
<b>Flujo Alterno 1: Colaborador no encontrado</b>	<ol style="list-style-type: none"> <li>1- El sistema no encuentra resultados en la búsqueda.</li> <li>2- El sistema muestra el mensaje 'No se encontraron colaboradores'.</li> <li>3- El flujo regresa al paso 4 del flujo Básico.</li> </ol>
<b>Flujo Alterno 2: Saldo insuficiente</b>	<ol style="list-style-type: none"> <li>1- El sistema detecta que los días solicitados exceden el saldo disponible.</li> <li>2- El sistema muestra el mensaje 'Saldo de vacaciones insuficiente'.</li> <li>3- El flujo regresa al paso 6 del Flujo Básico.</li> </ol>
<b>Requerimientos especiales</b>	
<ol style="list-style-type: none"> <li>1- El saldo de vacaciones debe actualizarse automáticamente al aprobar la solicitud.</li> <li>2- Las fechas no deben solaparse con otras vacaciones ya registradas del mismo colaborador.</li> <li>3- El sistema debe calcular correctamente los días hábiles.</li> </ol>	
<b>Postcondiciones</b>	
<ol style="list-style-type: none"> <li>1- La solicitud de vacaciones queda registrada en la base de datos.</li> <li>2- El saldo de vacaciones del colaborador se descuenta según los días aprobados.</li> <li>3- Las vacaciones quedan disponibles para consulta y reportes.</li> </ol>	

**Tabla 15***Caso de uso 5.*

<b>Prototipo: Sistema RRHH – Tendenzza</b>	
<b>Número Caso de Uso:05</b>	<b>Nombre del Caso de Uso:</b> Registrar permiso
<b>Fecha elaboración:</b>	14/12/2025
<b>Descripción Caso de Uso:</b>	Permite registrar una solicitud de permisos para un colaborador, indicando el tipo de permiso, período solicitado y la justificación correspondiente, para su posterior aprobación y control.
<b>Autor caso de uso:</b>	Miguel
<b>Actores relacionados:</b>	Administrador, jefe RRHH, colaborador.
<b>Precondiciones:</b>	<ul style="list-style-type: none"> <li>- El usuario debe estar autenticado en el sistema.</li> <li>- El colaborador debe existir en la base de datos.</li> <li>- Deben existir los tipos de permiso en el catálogo.</li> </ul>
<b>Flujo Básico del caso de uso</b>	
Este caso de uso comienza cuando el usuario desea registrar una solicitud de permiso. El usuario accede al módulo de permisos, selecciona al colaborador, indica el tipo de permiso, las fechas y la justificación. El sistema valida la información ingresada y una vez confirmada la operación, registra la solicitud.	
<b>Detallar el paso a paso del Flujo Básico</b>	
<ol style="list-style-type: none"> <li>1- El usuario accede al módulo de permisos desde el menú principal.</li> <li>2- Selecciona la opción 'Registrar permiso'.</li> <li>3- El sistema muestra el formulario de registro.</li> <li>4- El usuario busca y selecciona el colaborador. <ul style="list-style-type: none"> <li>• Si el colaborador no es encontrado, se ejecuta el Flujo Alterno 1.</li> </ul> </li> <li>5- El sistema muestra los datos del colaborador.</li> <li>6- El usuario selecciona el tipo de permiso (médico, personal, académico, entre otros).</li> <li>7- El usuario ingresa la fecha de inicio y fecha fin del permiso. <ul style="list-style-type: none"> <li>• Si las fechas ingresadas no son válidas, se ejecuta el Flujo Alterno 3.</li> </ul> </li> <li>8- El usuario ingresa la justificación o motivo del permiso. <ul style="list-style-type: none"> <li>• Si el usuario selecciona la opción 'Cancelar', se ejecuta el Sub-Flujo 1.</li> </ul> </li> <li>9- El usuario selecciona 'guardar'.</li> <li>10- El sistema valida los datos ingresados.</li> <li>11- El sistema solicita confirmación para registrar el permiso.</li> <li>12- El usuario confirma el registro.</li> <li>13- El sistema guarda el permiso en la base de datos.</li> <li>14- El sistema muestra el mensaje 'Permiso registrado correctamente'.</li> </ol>	
<b>Sub-Flujos</b>	
<b>Sub-Flujo 1: Cancelar operación</b>	<ol style="list-style-type: none"> <li>1- El usuario selecciona 'Cancelar'.</li> <li>2- El sistema descarta los datos.</li> <li>3- El sistema regresa a la lista de permisos.</li> </ol>
<b>Flujos Alternos</b>	
Corresponde a lo que debe realizar el sistema ante posibles errores	
<b>Flujo Alterno 1: Colaborador no encontrado.</b>	<ol style="list-style-type: none"> <li>1- El sistema no encuentra resultados en la búsqueda.</li> <li>2- El sistema muestra el mensaje 'No se encontraron colaboradores'.</li> <li>3- El flujo regresa al paso 4 del flujo Básico.</li> </ol>
<b>Flujo Alterno 2: Campos incompletos.</b>	<ol style="list-style-type: none"> <li>1- El sistema detecta campos obligatorios vacíos.</li> <li>2- El sistema resalta los campos que requieren corrección.</li> <li>3- El flujo regresa al paso correspondiente del Flujo Básico.</li> </ol>
<b>Flujo Alterno 3: Fechas invalidas.</b>	<ol style="list-style-type: none"> <li>1- El sistema detecta que la fecha de fin es anterior a la fecha de inicio.</li> <li>2- El sistema muestra el mensaje 'Las fechas ingresadas no son válidas'.</li> <li>3- El flujo regresa al paso 7 del Flujo Básico.</li> </ol>

<b>Requerimientos especiales</b>
1- El tipo de permiso debe seleccionarse de una lista predefinida. 2- La justificación es obligatoria para todos los permisos. 3- El sistema debe registrar la fecha y la hora de creación de la solicitud.
<b>Postcondiciones</b>
1- El permiso queda registrado en la base de datos. 2- El permiso queda disponible para la aprobación por parte del jefe correspondiente. 3- El permiso aparece en el historial del colaborador.

**Tabla 16***Caso de uso 6.*

<b>Prototipo: Sistema RRHH - Tendenza</b>	
<b>Número Caso de Uso: 06</b>	<b>Nombre del Caso de Uso:</b> Registrar incapacidad.
<b>Fecha elaboración:</b>	13/12/2025
<b>Descripción Caso de Uso:</b>	Permite registrar una incapacidad médica para un colaborador, indicando el tipo de incapacidad, el periodo correspondiente y las observaciones necesarias, para su correcta gestión administrativa y de nómina.
<b>Autor caso de uso:</b>	Miguel
<b>Actores relacionados:</b>	Administrador, jefe RRHH
<b>Precondiciones:</b>	<ul style="list-style-type: none"> <li>- El usuario debe estar autenticado con el rol Administrador o jefe de RRHH.</li> <li>- El colaborador debe existir en la base de datos.</li> <li>- Deben existir los tipos de incapacidad en el catálogo.</li> </ul>
<b>Flujo Básico del caso de uso</b>	
Este caso de uso comienza cuando el usuario desea registrar una incapacidad médica de un colaborador. El usuario accede al módulo de incapacidades, selecciona al colaborador e indica el tipo de incapacidad, el período de ausencia y las observaciones correspondientes. El sistema valida la información y tras la confirmación del usuario, registra la incapacidad.	
<b>Detallar el paso a paso del Flujo Básico</b>	
1- El usuario accede al módulo de incapacidades desde el menú principal. 2- Selecciona la opción 'Registrar incapacidad'. 3- El sistema muestra el formulario de registro. 4- El usuario busca y selecciona al colaborador. <ul style="list-style-type: none"> <li>• Si el colaborador no es encontrado, se ejecuta el Flujo Alterno 1.</li> </ul> 5- El sistema muestra los datos del colaborador. 6- El usuario selecciona el tipo de incapacidad (enfermedad general, accidente laboral, maternidad, entre otros). 7- El usuario ingresa la fecha de inicio y fecha fin de la incapacidad. <ul style="list-style-type: none"> <li>• Si las fechas ingresadas no son válidas, se ejecuta el Flujo Alternativo 3.</li> </ul> 8- El usuario ingresa observaciones o diagnóstico. <ul style="list-style-type: none"> <li>• Si existen campos obligatorios incompletos, se ejecuta el Flujo Alterno 2.</li> </ul> 9- El usuario selecciona 'Guardar'. <ul style="list-style-type: none"> <li>• Si el usuario selecciona la opción 'Cancelar', se ejecuta el Sub-Flujo 1.</li> </ul> 10- El sistema valida los datos ingresados. 11- El sistema solicita confirmación para registrar la incapacidad. 12- El usuario confirma el registro. 13- El sistema guarda la incapacidad en la base de datos. 14- El sistema muestra el mensaje 'Incapacidad registrada correctamente'.	
<b>Sub-Flujos</b>	

<b>Sub-Flujo 1: Cancelar operación</b>	<ol style="list-style-type: none"> <li>1- El usuario selecciona 'Cancelar'.</li> <li>2- El sistema descarta los datos.</li> <li>3- El sistema regresa a la lista de incapacidades.</li> </ol>
<b>Flujos Alternos</b>	
Corresponde a lo que debe realizar el sistema ante posibles errores	
<b>Flujo Alternativo 1: Colaborador no encontrado.</b>	<ol style="list-style-type: none"> <li>1- El sistema no encuentra resultados de la búsqueda.</li> <li>2- El sistema muestra el mensaje 'No se encontraron colaboradores'.</li> <li>3- El flujo regresa al paso 4 del Flujo Básico.</li> </ol>
<b>Flujo Alternativo 2: Campos incompletos</b>	<ol style="list-style-type: none"> <li>1- El sistema detecta campos obligatorios vacíos.</li> <li>2- El sistema resalta los campos que requieren corrección.</li> <li>3- El flujo regresa al paso correspondiente del Flujo Básico.</li> </ol>
<b>Flujo Alternativo 3: Fechas inválidas.</b>	<ol style="list-style-type: none"> <li>1- El sistema detecta que la fecha de fin es inferior a la fecha de inicio.</li> <li>2- El sistema muestra el mensaje 'Las fechas ingresadas no son válidas'.</li> <li>3- El flujo regresa al paso 7 del Flujo Básico.</li> </ol>
<b>Requerimientos especiales</b>	
<ol style="list-style-type: none"> <li>1- El tipo de incapacidad debe seleccionarse de una lista predefinida.</li> <li>2- El sistema debe calcular automáticamente los días de incapacidad.</li> <li>3- Las incapacidades deben afectar el cálculo de nómina según el tipo.</li> </ol>	
<b>Postcondiciones</b>	
<ol style="list-style-type: none"> <li>1- La incapacidad queda registrada en la base de datos.</li> <li>2- El historial de incapacidades del colaborador se actualiza.</li> <li>3- La incapacidad se refleja en los reportes y cálculos de nómina.</li> </ol>	

**Tabla 17***Caso de uso 7.*

<b>Prototipo: Sistema RRHH – Tendenzza</b>	
<b>Número Caso de Uso: 07</b>	<b>Nombre del Caso de Uso:</b> Registrar horas extra
<b>Fecha elaboración:</b>	13/12/2025
<b>Descripción Caso de Uso:</b>	Permite registrar las horas extra trabajadas por un colaborador, con el fin de que dichas horas sean consideradas posteriormente en el cálculo y pago de la nómina correspondiente.
<b>Autor caso de uso:</b>	Miguel
<b>Actores relacionados:</b>	Administrador, jefe RRHH, jefe operaciones.
<b>Precondiciones:</b>	<ul style="list-style-type: none"> <li>- El usuario debe estar autenticado con el rol autorizado.</li> <li>- El colaborador debe existir en la base de datos.</li> </ul>
<b>Flujo Básico del caso de uso</b>	
Este caso de uso comienza cuando el usuario desea registrar horas extra trabajadas por un colaborador. El usuario accede al módulo correspondiente, selecciona al colaborador, ingresa la fecha, la cantidad de horas y el motivo. El sistema valida la información y tras la confirmación del usuario, registra las horas extra.	
<b>Detallar el paso a paso del Flujo Básico</b>	

<ol style="list-style-type: none"> <li>1- El usuario accede al módulo de Horas Extra desde el menú principal.</li> <li>2- Selecciona la opción 'Registrar horas extra'.</li> <li>3- El sistema muestra el formulario de registro.</li> <li>4- El usuario busca y selecciona al colaborador. <ul style="list-style-type: none"> <li>• Si el colaborador no es encontrado, se ejecuta el Flujo Alterno 1.</li> </ul> </li> <li>5- El sistema muestra los datos del colaborador y su salario base.</li> <li>6- El usuario ingresa la fecha en que se realizaron las horas extra.</li> <li>7- El usuario ingresa la cantidad de horas trabajadas. <ul style="list-style-type: none"> <li>• Si la cantidad de horas ingresada no es válida, se ejecuta el Flujo Alterno 3.</li> </ul> </li> <li>8- El usuario ingres el motivo de las horas extra. <ul style="list-style-type: none"> <li>• Si existen campos obligatorios incompletos, se ejecuta el Flujo Alterno 2.</li> </ul> </li> <li>9- El usuario selecciona 'guardar'. <ul style="list-style-type: none"> <li>• Si el usuario selecciona la opción 'Cancelar', se ejecuta el Sub-Flujo 1.</li> </ul> </li> <li>10- El sistema valida los datos ingresados.</li> <li>11- El sistema solicita confirmación para registrar las horas extra.</li> <li>12- El usuario confirma el registro.</li> <li>13- El sistema guarda las horas extra en la base de datos.</li> <li>14- El sistema muestra el mensaje 'Horas extra registradas correctamente'.</li> </ol>	
<b>Sub-Flujos</b>	
<b>Sub-Flujo 1: Cancelar operación</b>	<ol style="list-style-type: none"> <li>1- El usuario selecciona 'Cancelar'.</li> <li>2- El sistema descarta los datos.</li> <li>3- El sistema regresa a la lista de horas extra.</li> </ol>
<b>Flujos Alternos</b>	
Corresponde a lo que debe realizar el sistema ante posibles errores	
<b>Flujo Alterno 1: Colaborador no encontrado.</b>	<ol style="list-style-type: none"> <li>1- El sistema no encuentra resultados de la búsqueda.</li> <li>2- El sistema muestra el mensaje 'No se encontraron colaboradores'</li> <li>3- El flujo regresa al paso 4 del Flujo Básico.</li> </ol>
<b>Flujo Alterno 2: Campos incompletos</b>	<ol style="list-style-type: none"> <li>1- El sistema detecta campos obligatorios vacíos.</li> <li>2- El sistema resalta los campos que requieren corrección.</li> <li>3- El flujo regresa al paso correspondiente del Flujo Básico.</li> </ol>
<b>Flujo Alterno 3: Cantidad de horas invalidas</b>	<ol style="list-style-type: none"> <li>1- El sistema detecta que la cantidad de horas es menor o igual a cero.</li> <li>2- El sistema muestra el mensaje 'La cantidad de horas debe ser mayor a cero'.</li> <li>3- El flujo regresa al paso 7 del Flujo Básico.</li> </ol>
<b>Requerimientos especiales</b>	
<ol style="list-style-type: none"> <li>1- La cantidad de horas debe ser numérica y mayor a cero.</li> <li>2- El sistema debe calcular el monto a pagar según el salario del colaborador.</li> <li>3- Las horas extra deben reflejarse en el cálculo de nómina del período correspondiente.</li> </ol>	
<b>Postcondiciones</b>	
<ol style="list-style-type: none"> <li>1- Las horas extra quedan registradas en la base de datos.</li> <li>2- El registro queda disponible para el cálculo de nómina.</li> <li>3- Las horas extra aparecen en el historial del colaborador.</li> </ol>	

**Tabla 18***Caso de uso 8.*

<b>Prototipo: Sistema RRHH – Tendenzza</b>	
<b>Número Caso de Uso: 08</b>	<b>Nombre del Caso de Uso:</b> Registrar asistencia.
<b>Fecha elaboración:</b>	13/12/2025

<b>Descripción Caso de Uso:</b>	Permite que el colaborador registre su asistencia diaria mediante marcas de entrada y salida desde la aplicación web, utilizando validación de ubicación por GPS para verificar que se encuentre dentro de una zona Autorizada
<b>Autor caso de uso:</b>	Miguel
<b>Actores relacionados:</b>	Administrador, jefe RRHH, jefe Operaciones, colaborador.
<b>Precondiciones:</b>	<ul style="list-style-type: none"> <li>- El usuario debe estar autenticado con el rol autorizado.</li> <li>- El colaborador debe existir en la base de datos.</li> <li>- Deben existir ubicaciones autorizadas vigentes configuradas en el sistema.</li> <li>- El dispositivo del colaborador debe tener acceso a GPS o geolocalización del navegador.</li> <li>- Deben existir los estados de asistencia en el catálogo.</li> </ul>
<b>Flujo Básico del caso de uso</b>	
Este caso de uso comienza cuando el colaborador desea registrar su asistencia del día. El colaborador accede al módulo de asistencia, el sistema obtiene sus coordenadas GPS actuales y verifica que se encuentre dentro de una ubicación autorizada. Si la validación es exitosa, el sistema registra la marca de entrada o salida según corresponda.	
<b>Detallar el paso a paso del Flujo Básico</b>	
<ol style="list-style-type: none"> <li>1. El colaborador accede al módulo de asistencias desde el menú principal.</li> <li>2. El sistema muestra el historial de asistencias del mes actual.</li> <li>3. El sistema muestra los botones de acción según el estado del día: <ul style="list-style-type: none"> <li>• Si no hay entrada registrada hoy, se muestra: botón ‘Marcar Entrada’.</li> <li>• Si ya hay entrada, pero no salida se muestra: botón ‘Marcar Salida’.</li> </ul> </li> <li>4. El colaborador selecciona ‘Marcar Entrada’ <ul style="list-style-type: none"> <li>• Si el colaborador ya tiene una entrada registrada hoy, se ejecuta el Flujo Alterno 2.</li> </ul> </li> <li>5. El sistema solicita permiso de geolocalización al navegador. <ul style="list-style-type: none"> <li>• Si el navegador no otorga permiso o no hay GPS disponible, se ejecuta el Flujo Alterno 3.</li> </ul> </li> <li>6. El sistema obtiene las coordenadas GPS actuales (latitud y longitud).</li> <li>7. El sistema consulta las ubicaciones autorizadas vigentes para ese día. <ul style="list-style-type: none"> <li>• Si no existen ubicaciones autorizadas configuradas, se ejecuta el Flujo Alterno 4.</li> </ul> </li> <li>8. El sistema calcula la distancia entre las coordenadas actuales y cada ubicación autorizada <ul style="list-style-type: none"> <li>• Si el colaborador esta fuera del radio permitido de todas las ubicaciones, se ejecuta el Flujo Alterno 5.</li> </ul> </li> <li>9. El sistema registra la entrada con: hora actual, coordenadas GPS y estado ‘Presente’.</li> <li>10. El sistema muestra el mensaje ‘Entrada registrada correctamente’, con la hora registrada.</li> <li>11. El colaborador selecciona ‘Marcar Salida’. <ul style="list-style-type: none"> <li>• Si no existe una entrada registrada hoy, se ejecuta el Flujo Alterno 6.</li> </ul> </li> <li>12. El sistema obtiene las coordenadas GPS actuales (pasos 5-7 se repiten).</li> <li>13. El sistema registra la salida con: hora actual y coordenadas GPS.</li> <li>14. El sistema calcula automáticamente las horas trabajadas.</li> <li>15. El sistema muestra el mensaje ‘Salida registrada correctamente’ con las horas trabajadas.</li> </ol>	
<b>Sub-Flujos</b>	
<b>Sub-Flujo 1: Cancelar operación</b>	<ol style="list-style-type: none"> <li>1- El colaborador cierra el módulo o navega a otra sección.</li> <li>2- El sistema no registra ninguna marca.</li> <li>3- El sistema conserva el historial sin cambios.</li> </ol>
<b>Flujos Alternos</b>	
Corresponde a lo que debe realizar el sistema ante posibles errores	
<b>Flujo Alterno 1: Sin ubicaciones vigentes.</b>	<ol style="list-style-type: none"> <li>1- El sistema detecta que no existe ubicaciones autorizadas activas para la fecha actual.</li> <li>2- El sistema muestra el mensaje ‘No se hay ubicaciones autorizadas configuradas para hoy’.</li> <li>3- El flujo finaliza sin registrar la asistencia.</li> </ol>
<b>Flujo Alterno 2: Entrada ya registrada.</b>	<ol style="list-style-type: none"> <li>1- El sistema detecta que ya existe una marca de entrada para el colaborador el día actual.</li> <li>2- El sistema muestra el mensaje ‘Ya tiene una entrada registrada hoy’.</li> <li>3- El flujo regresa al paso 2 del Flujo Básico.</li> </ol>

<b>Flujo Alternativo 3: GPS no disponible</b>	<ol style="list-style-type: none"> <li>1- El navegador no otorga permiso de geolocalización o el dispositivo no tiene GPS.</li> <li>2- El sistema muestra el mensaje 'No se pudo obtener su ubicación. Por favor habilite el GPS e intente nuevamente.</li> <li>3- El flujo regresa al paso 4 del Flujo Básico.</li> </ol>
<b>Flujo Alternativo 4: Sin ubicaciones configuradas</b>	<ol style="list-style-type: none"> <li>1- El sistema detecta que no existen ubicaciones autorizadas vigentes.</li> <li>2- El sistema muestra el mensaje 'No hay ubicaciones autorizadas configuradas, contacte al administrador'.</li> <li>3- El flujo finaliza sin registrar la asistencia.</li> </ol>
<b>Flujo Alternativo 5: Fuera del radio autorizado</b>	<ol style="list-style-type: none"> <li>1- El sistema detecta que las coordenadas del colaborador están fuera del radio permitido de todas las ubicaciones autorizadas.</li> <li>2- El sistema muestra el mensaje 'Su ubicación actual no corresponde a una zona autorizada'.</li> <li>3- El flujo regresa al paso 4 del del Flujo Básico.</li> </ol>
<b>Flujo Alternativo 6: Sin entrada registrada</b>	<ol style="list-style-type: none"> <li>1- El sistema detecta que no existe una marca de entrada para el día actual.</li> <li>2- El sistema muestra el mensaje 'Debe registrar la entrada antes de marcar la salida'.</li> <li>3- El flujo regresa al paso 2 del Flujo Básico.</li> </ol>
<b>Requerimientos especiales</b>	
<ol style="list-style-type: none"> <li>1- Solo se puede registrar una entrada y una salida al colaborador por día.</li> <li>2- La validación de ubicación GPS es obligatoria para registrar la asistencia.</li> <li>3- El sistema debe calcular automáticamente las horas trabajadas al registrar la salida.</li> <li>4- El radio de validación GPS es configurable por ubicación autorizada.</li> <li>5- Las coordenadas registradas deben almacenar tanto para la entrada como para la salida.</li> <li>6- Las asistencias deben reflejarse en los reportes y control de puntualidad.</li> </ol>	
<b>Postcondiciones</b>	
<ol style="list-style-type: none"> <li>1- La asistencia queda registrada en la base de datos con coordenadas GPS.</li> <li>2- El registro queda disponible para reportes de asistencia.</li> <li>3- Las horas trabajadas se calculan automáticamente al registrar la salida.</li> </ol>	

**Tabla 19***Caso de uso 9.*

<b>Prototipo: Sistema RRHH – Tendenzza.</b>	
<b>Número Caso de Uso: 09</b>	<b>Nombre del Caso de Uso:</b> Registrar evaluación de desempeño
<b>Fecha elaboración:</b>	13/12/2025
<b>Descripción Caso de Uso:</b>	Permite registrar una evaluación de desempeño para un colaborador, calificando su rendimiento laboral durante un período determinado, con el fin de dar seguimiento a su desempeño y generar reportes.
<b>Autor caso de uso:</b>	Miguel
<b>Actores relacionados:</b>	Administrador, jefe RRHH, jefe de Operaciones, jefe de ventas.
<b>Precondiciones:</b>	<ul style="list-style-type: none"> <li>- El usuario debe estar autenticado con rol autorizado.</li> <li>- El colaborador debe existir en la base de datos.</li> <li>- Deben existir los estados de evaluación en el catálogo.</li> </ul>
<b>Flujo Básico del caso de uso</b>	
Este caso de uso comienza cuando el usuario desea registrar una evaluación de desempeño de un colaborador. El usuario accede al módulo de evaluaciones, selecciona al colaborador, ingresa el período, la calificación, las observaciones y el estado de la evaluación. El sistema valida la información y tras la confirmación del usuario, registra la evaluación.	
<b>Detallar el paso a paso del Flujo Básico</b>	

<ol style="list-style-type: none"> <li>1- El usuario accede al módulo de evaluaciones desde el menú principal.</li> <li>2- Selecciona la opción 'Registrar evaluación'.</li> <li>3- El sistema muestra el formulario de registro.</li> <li>4- El usuario busca y selecciona al colaborador. <ul style="list-style-type: none"> <li>• Si el colaborador no es encontrado, se ejecuta el Flujo Alterno 1.</li> </ul> </li> <li>5- El sistema muestra los datos del colaborador.</li> <li>6- El usuario ingresa el período de evaluación (fecha inicio y fecha fin).</li> <li>7- El usuario ingresa la calificación obtenida. <ul style="list-style-type: none"> <li>• Si la calificación ingresada no es válida, se ejecuta el Flujo Alterno 3.</li> </ul> </li> <li>8- El usuario ingresa las observaciones y comentarios sobre el desempeño. <ul style="list-style-type: none"> <li>• Si existen campos obligatorios incompletos, se ejecuta el Flujo Alterno 2.</li> </ul> </li> <li>9- El usuario selecciona el estado de evaluación (aprobado, reprobado, pendiente de mejora).</li> <li>10- El usuario selecciona 'Guardar'. <ul style="list-style-type: none"> <li>• Si el usuario selecciona la opción 'Cancelar', se ejecuta el Sub-Flujo 1.</li> </ul> </li> <li>11- El sistema valida los datos ingresados.</li> <li>12- El sistema solicita confirmación para registrar la evaluación.</li> <li>13- El usuario confirma el registro.</li> <li>14- El sistema guarda la evaluación en la base de datos.</li> <li>15- El sistema muestre el mensaje 'Evaluación registrada correctamente'.</li> </ol>	
<b>Sub-Flujos</b>	
<b>Sub-Flujo 1: Cancelar operación.</b>	<ol style="list-style-type: none"> <li>1- El usuario selecciona 'Cancelar'.</li> <li>2- El sistema descarta los datos.</li> <li>3- El sistema regresa a la lista de evaluaciones.</li> </ol>
<b>Flujos Alternos</b>	
Corresponde a lo que debe realizar el sistema ante posibles errores	
<b>Flujo Alterno 1: Colaborador no encontrado.</b>	<ol style="list-style-type: none"> <li>1- El sistema no encuentra resultados en la búsqueda.</li> <li>2- El sistema muestra el mensaje 'No se encontraron colaboradores'.</li> <li>3- El flujo regresa al paso 4 del Flujo Básico.</li> </ol>
<b>Flujo Alterno 2: Campos incompletos.</b>	<ol style="list-style-type: none"> <li>1- El sistema detecta campos obligatorios vacíos.</li> <li>2- El sistema muestra el mensaje 'No se encontraron colaboradores'.</li> <li>3- El flujo regresa al paso correspondiente del Flujo Básico.</li> </ol>
<b>Flujo Alterno 3: Calificación inválida.</b>	<ol style="list-style-type: none"> <li>1- El sistema detecta que la calificación está fuera del rango permitido.</li> <li>2- El sistema muestra el mensaje 'La calificación debe estar entre 0 y 100'.</li> <li>3- El flujo regresa al paso 7 del Flujo Básico.</li> </ol>
<b>Requerimientos especiales</b>	
<ol style="list-style-type: none"> <li>1- La calificación debe ser numérica entre 0 y 100.</li> <li>2- Las observaciones son obligatorias para documentar el desempeño.</li> <li>3- El sistema debe guardar quién realizó la evaluación.</li> </ol>	
<b>Postcondiciones</b>	
<ol style="list-style-type: none"> <li>1- La evaluación queda registrada en la base de datos.</li> <li>2- El historial de evaluaciones del colaborador se actualiza.</li> <li>3- La evaluación queda disponible para reportes de desempeño.</li> </ol>	

### Módulo de Consultas.

**Tabla 20***Caso de uso 10.*

<b>Prototipo: Sistema RRHH - Tendenzza</b>	
<b>Número Caso de Uso: 10</b>	<b>Nombre del Caso de Uso:</b> Consultar colaboradores
<b>Fecha elaboración:</b>	13/12/2025
<b>Descripción Caso de Uso:</b>	Permite consultar y visualizar la información de los colaboradores registrados en el sistema, utilizando opciones de búsqueda, filtrado, ordenamiento y paginación.
<b>Autor caso de uso:</b>	Miguel
<b>Actores relacionados:</b>	Administrador, jefe RRHH, jefe Operaciones, jefe Ventas
<b>Precondiciones:</b>	<ul style="list-style-type: none"> <li>- El usuario debe estar autenticado con el rol autorizado.</li> <li>- Deben existir colaboradores registrados en la base de datos.</li> </ul>
<b>Flujo Básico del caso de uso</b>	
Este caso de uso comienza cuando el usuario desea consultar información de los colaboradores. El usuario accede al módulo y el sistema muestra la lista de colaboradores y el sistema muestra la lista disponible, permitiendo realizar búsquedas, ordenar la información y visualizar el detalle de cada colaborador.	
<b>Detallar el paso a paso del Flujo Básico</b>	
<ol style="list-style-type: none"> <li>1- El usuario accede al módulo de colaboradores desde el menú principal.</li> <li>2- El sistema muestra la lista de colaboradores registrados en una tabla. <ul style="list-style-type: none"> <li>• Si no existen colaboradores registrados, se ejecuta el Flujo Alterno 2.</li> </ul> </li> <li>3- El usuario puede buscar un colaborador por nombre, identificación u ocupación. <ul style="list-style-type: none"> <li>• Si la búsqueda no devuelve resultados, se ejecuta el Flujo Alterno 2.</li> </ul> </li> <li>4- El sistema filtra y muestra los resultados de la búsqueda.</li> <li>5- El usuario selecciona un colaborador para ver más detalles. <ul style="list-style-type: none"> <li>• Si el usuario selecciona ordenar la lista, se ejecuta el Sub-Flujo 1.</li> </ul> </li> <li>6- El sistema muestra la información completa del colaborador (datos personales, contacto, laborales). <ul style="list-style-type: none"> <li>• Si el usuario selecciona cambiar de página o cantidad de registros, se ejecuta el Sub-Flujo 2.</li> </ul> </li> <li>7- El usuario revisa la información.</li> <li>8- El usuario cierra la vista de detalles.</li> <li>9- El sistema regresa a la lista de colaboradores.</li> </ol>	
<b>Sub-Flujos</b>	
<b>Sub-Flujo 1: Ordenar lista.</b>	<ol style="list-style-type: none"> <li>1- El usuario selecciona una columna para ordenar (nombre, fecha de ingreso, ocupación).</li> <li>2- El sistema ordena la lista según la columna seleccionada.</li> </ol>
<b>Sub-Flujo 2: Pagina resultados.</b>	<ol style="list-style-type: none"> <li>1- El usuario selecciona el número de página o cantidad de registros a mostrar.</li> <li>2- El sistema actualiza la vista con los registros correspondientes.</li> </ol>
<b>Flujos Alternos</b>	
Corresponde a lo que debe realizar el sistema ante posibles errores	
<b>Flujo Alterno 1: Sin resultados.</b>	<ol style="list-style-type: none"> <li>1- El sistema no encuentra colaboradores que coinciden con la búsqueda.</li> <li>2- El sistema muestra el mensaje 'No se encontraron resultados'.</li> <li>3- El flujo regresa al paso 3 del Flujo Básico.</li> </ol>
<b>Flujo Alterno 2: Lista vacía.</b>	<ol style="list-style-type: none"> <li>1- El sistema detecta que no existen colaboradores registrados.</li> <li>2- El sistema muestra el mensaje 'No se encontraron registrados'.</li> <li>3- El flujo termina.</li> </ol>
<b>Requerimientos especiales</b>	

<ol style="list-style-type: none"> <li>1- La tabla debe permitir ordenar por cualquier columna.</li> <li>2- La búsqueda debe ser en tiempo real mientras el usuario escribe.</li> <li>3- El sistema debe mostrar paginación cuando hay muchos registros.</li> </ol>
<b>Postcondiciones</b>
<ol style="list-style-type: none"> <li>1- El usuario visualiza la información solicitada.</li> <li>2- No se realizan cambios en la base de datos.</li> </ol>

**Tabla 21***Caso de uso 11.*

<b>Prototipo: Sistema RRHH - Tendenzza</b>	
<b>Número Caso de Uso: 11</b>	<b>Nombre del Caso de Uso:</b> Consultar vacaciones.
<b>Fecha elaboración:</b>	13/12/2025
<b>Descripción Caso de Uso:</b>	Permite consultar y visualizar las vacaciones registradas de los colaboradores en el sistema, utilizando opciones de búsqueda, filtrado y visualización de detalles.
<b>Autor caso de uso:</b>	Miguel
<b>Actores relacionados:</b>	Administrador, jefe RRHH, Colaborador.
<b>Precondiciones:</b>	<ul style="list-style-type: none"> <li>- El usuario debe estar autenticado con rol autorizado.</li> <li>- Deben existir registros de vacaciones en la base de datos.</li> </ul>
<b>Flujo Básico del caso de uso</b>	
Este caso de uso comienza cuando el usuario desea consultar las vacaciones registradas. El usuario accede al módulo y el sistema muestra la lista disponible, permitiendo realizar búsquedas, aplicar filtros y visualizar el detalle de cada registro.	
<b>Detallar el paso a paso del Flujo Básico</b>	
<ol style="list-style-type: none"> <li>1- El usuario accede al módulo de vacaciones desde el menú principal.</li> <li>2- El sistema muestra la lista de vacaciones registradas en una tabla. <ul style="list-style-type: none"> <li>• Si no existen registros de vacaciones, se ejecuta el Flujo Alterno 2.</li> </ul> </li> <li>3- El usuario puede buscar por nombre del colaborador o por fechas. <ul style="list-style-type: none"> <li>• Si la búsqueda no devuelve resultados, se ejecuta el Flujo Alterno 2.</li> </ul> </li> <li>4- El sistema filtra y muestra los resultados de la búsqueda.</li> <li>5- El usuario puede filtrar las vacaciones por estado. <ul style="list-style-type: none"> <li>• Si el usuario selecciona filtrar por estado, se ejecuta el Sub-Flujo 1.</li> </ul> </li> <li>6- El usuario puede filtrar las vacaciones por período. <ul style="list-style-type: none"> <li>• Si el usuario selecciona filtrar por período, se ejecuta el Sub-Flujo 2.</li> </ul> </li> <li>7- El usuario selecciona un registro para ver más detalles.</li> <li>8- El sistema muestra la información completa de las vacaciones (colaborador, fechas, días, estado, saldo).</li> <li>9- El usuario revisa la información.</li> <li>10- El usuario cierra la vista de detalles.</li> <li>11- El sistema regresa a la lista de vacaciones.</li> </ol>	
<b>Sub-Flujos</b>	
<b>Sub-Flujo 1: Filtrar por estado</b>	<ol style="list-style-type: none"> <li>1- El usuario selecciona un estado (aprobado, pendiente, rechazado).</li> <li>2- El sistema filtra y muestra solo las vacaciones con ese estado.</li> </ol>
<b>Sub-Flujo 2: Filtrar por periodo</b>	<ol style="list-style-type: none"> <li>1- El usuario selecciona el rango de fechas.</li> <li>2- El sistema muestra las vacaciones dentro del período seleccionado.</li> </ol>
<b>Flujos Alternos</b>	
Corresponde a lo que debe realizar el sistema ante posibles errores	
<b>Flujo Alterno 1: Sin resultados</b>	<ol style="list-style-type: none"> <li>1- El sistema no encuentra vacaciones que coincidan con los criterios de búsqueda.</li> <li>2- El sistema muestra el mensaje 'No se encontraron resultados'.</li> </ol>

	3- El flujo regresa al paso 3 del Flujo Básico.
<b>Flujo Alterno 2: Lista vacía.</b>	1- El sistema detecta que no existen vacaciones registradas en el sistema. 2- El sistema muestra el mensaje 'No hay vacaciones registradas'. 3- El flujo termina
<b>Requerimientos especiales</b>	
1- La tabla debe permitir ordenar por cualquier columna. 2- El sistema debe mostrar el saldo de vacaciones disponible por colaborador. 3- La búsqueda debe ser en tiempo real mientras el usuario escribe.	
<b>Postcondiciones</b>	
1- El usuario visualiza la información solicitada. 2- No se realizan cambios en la base de datos.	

**Tabla 22***Caso de uso 12.*

<b>Prototipo: Sistema RRHH - TendENZA</b>	
<b>Número Caso de Uso: 12</b>	<b>Nombre del Caso de Uso:</b> Consultar permisos
<b>Fecha elaboración:</b>	13/12/2025
<b>Descripción Caso de Uso:</b>	Permite consultar y visualizar los permisos registrados de los colaboradores en el sistema, utilizando opciones de búsqueda, filtrado y visualización de detalles.
<b>Autor caso de uso:</b>	Miguel
<b>Actores relacionados:</b>	Administrador, jefe RRHH, Colaborador
<b>Precondiciones:</b>	<ul style="list-style-type: none"> <li>- El usuario debe estar autenticado con el rol autorizado.</li> <li>- Deben existir registros de permisos en la base de datos.</li> </ul>
<b>Flujo Básico del caso de uso</b>	
Este caso de uso comienza cuando el usuario desea consultar los permisos registrados. El usuario accede al módulo de permisos y el sistema muestra la lista disponible, permitiendo realizar búsquedas, aplicar filtros y visualizar el detalle de cada permiso.	
<b>Detallar el paso a paso del Flujo Básico</b>	
1- El usuario accede al módulo de permisos desde el menú principal. 2- El sistema muestra la lista de permisos registrados en una tabla. <ul style="list-style-type: none"> <li>• Si no existen permisos registrados, se ejecuta el Flujo Alterno 2.</li> </ul> 3- El usuario ingresa criterios de búsqueda por nombre del colaborador o por tipo de permiso. <ul style="list-style-type: none"> <li>• Si la búsqueda no devuelve resultados, se ejecuta el Flujo Alterno 1.</li> </ul> 4- El sistema filtra y muestra los resultados de la búsqueda. 5- El usuario puede filtrar los permisos por tipo. <ul style="list-style-type: none"> <li>• Si el usuario selecciona filtrar por tipo de permiso, se ejecuta el Sub-Flujo 1.</li> </ul> 6- El usuario puede filtrar los permisos por estado. <ul style="list-style-type: none"> <li>• Si el usuario selecciona filtrar por estado, se ejecuta el Sub-Flujo 2.</li> </ul> 7- El usuario selecciona un registro para ver más detalles. 8- El sistema muestra la información completa del permiso (colaborador, tipo, fechas, justificación, estado). 9- El usuario revisa la información. 10- El usuario cierra la vista de detalles. 11- El sistema regresa a la lista de permisos.	
<b>Sub-Flujos</b>	

<b>Sub-Flujo 1: Filtrar por tipo</b>	<ol style="list-style-type: none"> <li>1- El usuario selecciona un tipo de permiso (médico, personal, académico, entre otros).</li> <li>2- El sistema filtra y muestra solo los permisos de ese tipo.</li> </ol>
<b>Sub-Flujo 2: Filtrar por estado</b>	<ol style="list-style-type: none"> <li>1- El usuario selecciona un estado (aprobado, pendiente, rechazado).</li> <li>2- El sistema muestra solo los permisos con ese estado.</li> </ol>
<b>Flujos Alternos</b>	
Corresponde a lo que debe realizar el sistema ante posibles errores	
<b>Flujo Alternativo 1: Sin resultados</b>	<ol style="list-style-type: none"> <li>1- El sistema no encuentra permisos que coincidan con la búsqueda.</li> <li>2- El sistema muestra el mensaje 'No se encontraron resultados'.</li> <li>3- El flujo regresa al paso 3 del Flujo Básico.</li> </ol>
<b>Flujo Alternativo 2: Lista vacía</b>	<ol style="list-style-type: none"> <li>1- El sistema detecta que no existen permisos registrados en el sistema.</li> <li>2- El sistema muestra el mensaje 'No hay permisos registrados'.</li> <li>3- El flujo termina.</li> </ol>
<b>Requerimientos especiales</b>	
<ol style="list-style-type: none"> <li>1- La tabla debe permitir ordenar por cualquier columna.</li> <li>2- El sistema debe mostrar el tipo de permiso con un distintivo visual.</li> <li>3- La búsqueda debe ser en tiempo real mientras el usuario escribe.</li> </ol>	
<b>Postcondiciones</b>	
<ol style="list-style-type: none"> <li>1- El usuario visualiza la información solicitada.</li> <li>2- No se realizan cambios en la base de datos.</li> </ol>	

**Tabla 23***Caso de uso 13.*

<b>Prototipo: Sistema RRHH - Tendenza</b>	
<b>Número Caso de Uso: 13</b>	<b>Nombre del Caso de Uso:</b> Consultar incapacidades
<b>Fecha elaboración:</b>	13/12/2025
<b>Descripción Caso de Uso:</b>	Permite consultar y visualizar las incapacidades registradas de los colaboradores en el sistema, utilizando opciones de búsqueda, filtrado y visualización de detalles.
<b>Autor caso de uso:</b>	Miguel
<b>Actores relacionados:</b>	Administrador, jefe RRHH
<b>Precondiciones:</b>	<ul style="list-style-type: none"> <li>- El usuario debe estar autenticado con el rol de Administrador o jefe RRHH.</li> <li>- Deben existir registros de incapacidades en la base de datos.</li> </ul>
<b>Flujo Básico del caso de uso</b>	
Este caso de uso comienza cuando el usuario desea consultar las incapacidades registradas. El usuario accede al módulo de incapacidades y el sistema muestra la lista disponible, permitiendo realizar búsquedas, aplicar filtros y visualizar el detalle de cada incapacidad.	
<b>Detallar el paso a paso del Flujo Básico</b>	

<ol style="list-style-type: none"> <li>1. El usuario accede al módulo de incapacidades desde el menú principal.</li> <li>2. El sistema muestra la lista de incapacidades registradas en una tabla. <ul style="list-style-type: none"> <li>• Si no existen incapacidades registradas, se ejecuta el Flujo Alterno 2.</li> </ul> </li> <li>3. El usuario ingresa criterios de búsqueda por nombre del colaborador o por tipo de incapacidad. <ul style="list-style-type: none"> <li>• Si la búsqueda no devuelve resultados, se ejecuta el Flujo Alterno 1.</li> </ul> </li> <li>4. El sistema filtra y muestra los resultados de la búsqueda.</li> <li>5. El usuario puede filtrar las incapacidades por tipo. <ul style="list-style-type: none"> <li>• Si el usuario selecciona filtrar por tipo de incapacidad, se ejecuta el Sub-Flujo 1.</li> </ul> </li> <li>6. El usuario puede filtrar las incapacidades por período. <ul style="list-style-type: none"> <li>• Si el usuario selecciona filtrar por rango de fechas, se ejecuta el Sub-Flujo 2.</li> </ul> </li> <li>7. El usuario selecciona un registro para ver más detalles.</li> <li>8. El sistema muestra la información completa de la incapacidad (colaborador, tipo de incapacidad, fechas, total de días, observaciones).</li> <li>9. El usuario revisa la información.</li> <li>10. El usuario cierra la vista de detalles.</li> <li>11. El sistema regresa a lista de incapacidades.</li> </ol>	
<b>Sub-Flujos</b>	
<b>Sub-Flujo 1: Filtrar por tipo</b>	<ol style="list-style-type: none"> <li>1. El usuario selecciona un tipo de incapacidad (enfermedad general, accidente laboral, maternidad, otro).</li> <li>2. El sistema filtra y muestra solo las incapacidades de ese tipo.</li> </ol>
<b>Sub-Flujo 2: Filtrar por período</b>	<ol style="list-style-type: none"> <li>1. El usuario selecciona un rango de fechas.</li> <li>2. El sistema filtra y muestra solo las incapacidades de ese tipo.</li> </ol>
<b>Flujos Alternos</b>	
Corresponde a lo que debe realizar el sistema ante posibles errores	
<b>Flujo Alterno 1: Sin resultados</b>	<ol style="list-style-type: none"> <li>1. El sistema no encuentra incapacidades que coincidan con los criterios de búsqueda.</li> <li>2. El sistema muestra el mensaje 'No se encontraron resultados'.</li> <li>3. El flujo regresa al paso 3 del Flujo Básico.</li> </ol>
<b>Flujo Alterno 2: Lista vacía</b>	<ol style="list-style-type: none"> <li>1- El sistema detecta que no existen incapacidades registradas en el sistema.</li> <li>2- El sistema muestra el mensaje 'No hay incapacidades registradas'.</li> <li>3- El flujo termina.</li> </ol>
<b>Requerimientos especiales</b>	
<ol style="list-style-type: none"> <li>1- La tabla debe permitir ordenar por cualquier columna.</li> <li>2- El sistema debe mostrar el total de días de incapacidad por registro.</li> <li>3- La búsqueda debe ser en tiempo real mientras el usuario escribe.</li> </ol>	
<b>Postcondiciones</b>	
<ol style="list-style-type: none"> <li>1- El usuario visualiza la información solicitada.</li> <li>2- No se realizan cambios en la base de datos.</li> </ol>	

## Tabla 24

### Caso de uso 14.

<b>Prototipo: Sistema RRHH - Tendenzza</b>	
<b>Número Caso de Uso: 14</b>	<b>Nombre del Caso de Uso:</b> Consultar horas extra
<b>Fecha elaboración:</b>	13/12/2025
<b>Descripción Caso de Uso:</b>	Permite consultar y visualizar las horas extra registradas de los colaboradores en el sistema, utilizando opciones de búsqueda, filtrado y visualización del detalle de cada registro.
<b>Autor caso de uso:</b>	Miguel
<b>Actores relacionados:</b>	Administrador, jefe RRHH, jefe de Operaciones.

<b>Precondiciones:</b>	<ul style="list-style-type: none"> <li>- El usuario debe estar autenticado con el rol autorizado.</li> <li>- Deben existir registros de horas extra en la base de datos.</li> </ul>
<b>Flujo Básico del caso de uso</b>	
Este caso de uso comienza cuando el usuario desea consultar las horas extra registradas. El usuario accede al módulo de horas extra y el sistema muestra la lista disponible, permitiendo realizar búsquedas, aplicar filtros y visualizar el detalle de los registros.	
<b>Detallar el paso a paso del Flujo Básico</b>	
<ol style="list-style-type: none"> <li>1- El usuario accede al módulo de Horas Extra desde el menú principal.</li> <li>2- El sistema muestra la lista de horas extra registradas en una tabla. <ul style="list-style-type: none"> <li>• Si no existen registros de horas extra, se ejecuta el Flujo Alterno 2.</li> </ul> </li> <li>3- El usuario ingresa criterios de búsqueda por nombre del colaborador o por fecha. <ul style="list-style-type: none"> <li>• Si la búsqueda no devuelve resultados, se ejecuta el Flujo Alterno 1.</li> </ul> </li> <li>4- El sistema filtra y muestra los resultados de la búsqueda.</li> <li>5- El usuario puede filtrar las horas extra por estado. <ul style="list-style-type: none"> <li>• Si el usuario selecciona filtrar por estado, se ejecuta el Sub-Flujo 1.</li> </ul> </li> <li>6- El usuario puede filtrar las horas extra por período. <ul style="list-style-type: none"> <li>• Si el usuario selecciona filtrar por rango de fechas, se ejecuta el Sub-Flujo 2.</li> </ul> </li> <li>7- El usuario selecciona un registro para ver más detalles.</li> <li>8- El sistema muestra la información completa de las horas extra (colaborador, fecha, cantidad de horas, motivo, estado).</li> <li>9- El usuario revisa la información.</li> <li>10- El usuario cierra la vista de detalles.</li> <li>11- El sistema regresa a la lista de horas extra.</li> </ol>	
<b>Sub-Flujos</b>	
<b>Sub-Flujo 1: Filtrar por estado</b>	<ol style="list-style-type: none"> <li>1- El usuario selecciona un estado (aprobado, pendiente, rechazado).</li> <li>2- El sistema filtra y muestra solo las horas extra con ese estado.</li> </ol>
<b>Sub-Flujo 2: Filtrar por período</b>	<ol style="list-style-type: none"> <li>1- El usuario selecciona un rango de fechas.</li> <li>2- El sistema muestra las horas extra dentro de ese período.</li> </ol>
<b>Flujos Alternos</b>	
Corresponde a lo que debe realizar el sistema ante posibles errores	
<b>Flujo Alterno 1: Sin resultados</b>	<ol style="list-style-type: none"> <li>1- El sistema no encuentra horas extra que coincidan con la búsqueda y muestra el mensaje 'No se encontraron resultados'.</li> <li>2- El flujo regresa al paso 3.</li> </ol>
<b>Flujo Alterno 2: Lista vacía</b>	<ol style="list-style-type: none"> <li>1- No existen horas extra registradas en el sistema y muestra el mensaje 'No hay horas extra registradas'</li> <li>2- El flujo termina.</li> </ol>
<b>Requerimientos especiales</b>	
<ol style="list-style-type: none"> <li>1- La tabla debe permitir ordenar por cualquier columna.</li> <li>2- El sistema debe mostrar el total de horas por colaborador.</li> <li>3- La búsqueda debe ser en tiempo real mientras el usuario escribe.</li> </ol>	
<b>Postcondiciones</b>	
<ol style="list-style-type: none"> <li>1- El usuario visualiza la información solicitada.</li> <li>2- No se realizan cambios en la base de datos.</li> </ol>	

**Tabla 25***Caso de uso 15.*

<b>Número Caso de Uso: 15</b>	<b>Nombre del Caso de Uso:</b> Consultar asistencias
<b>Fecha elaboración:</b>	13/12/2025
<b>Descripción Caso de Uso:</b>	Permite consultar y visualizar las asistencias registradas de los colaboradores en el sistema, utilizando opciones de búsqueda, filtrado y visualización del detalle de cada registro.
<b>Autor caso de uso:</b>	Miguel
<b>Actores relacionados:</b>	Administrador, jefe RRHH, jefe Operaciones
<b>Precondiciones:</b>	<ul style="list-style-type: none"> <li>- El usuario debe estar autenticado con rol autorizado.</li> <li>- Deben existir registros de asistencias en la base de datos.</li> </ul>
<b>Flujo Básico del caso de uso</b>	
Este caso de uso comienza cuando el usuario desea consultar las asistencias registradas. El usuario accede al módulo y el sistema muestra la lista disponible, permitiendo realizar búsquedas, aplicar filtros y visualizar el detalle de los registros.	
<b>Detallar el paso a paso del Flujo Básico</b>	
<ol style="list-style-type: none"> <li>1. El usuario accede al módulo de Asistencias desde el menú principal.</li> <li>2. El sistema muestra la lista de asistencias registradas en una tabla. <ul style="list-style-type: none"> <li>• Si no existen registros de asistencias, se ejecuta el Flujo Alterno 2.</li> </ul> </li> <li>3. El usuario ingresa criterios de búsqueda por nombre del colaborador o por fecha. <ul style="list-style-type: none"> <li>• Si la búsqueda no devuelve resultados, se ejecuta el Flujo Alterno 1.</li> </ul> </li> <li>4. El sistema filtra y muestra los resultados de la búsqueda.</li> <li>5. El usuario puede filtrar las asistencias por estado. <ul style="list-style-type: none"> <li>• Si el usuario selecciona filtrar por estado, se ejecuta el Sub-Flujo 1</li> </ul> </li> <li>6. El usuario puede filtrar las asistencias por período. <ul style="list-style-type: none"> <li>• Si el usuario selecciona filtrar por rango de fechas, se ejecuta el Sub-Flujo 2.</li> </ul> </li> <li>7. El usuario selecciona un registro para ver más detalles.</li> <li>8. El sistema muestra la información completa de la asistencia (colaborador, fecha, hora de entrada, hora de salida, horas trabajadas, estado).</li> <li>9. El usuario revisa la información.</li> <li>10. El usuario cierra la vista de detalles.</li> <li>11. El sistema regresa a la lista de asistencias.</li> </ol>	
<b>Sub-Flujos</b>	
<b>Sub-Flujo 1: Filtrar por estado</b>	<ol style="list-style-type: none"> <li>1- El usuario selecciona un estado (presente, tardanza, ausencia justificada, ausencia injustificada).</li> <li>2- El sistema filtra y muestra solo las asistencias con ese estado.</li> </ol>
<b>Sub-Flujo 2: Filtrar por período</b>	<ol style="list-style-type: none"> <li>1- El usuario selecciona un rango de fechas.</li> <li>2- El sistema muestra las asistencias dentro de ese período.</li> </ol>
<b>Flujos Alternos</b>	
Corresponde a lo que debe realizar el sistema ante posibles errores	
<b>Flujo Alterno 1: Sin resultados</b>	<ol style="list-style-type: none"> <li>1- El sistema no encuentra asistencias que coincidan con la búsqueda.</li> <li>2- El sistema muestra el mensaje 'No se encontraron resultados'.</li> <li>3- El flujo regresa al paso 3 del Flujo Básico.</li> </ol>
<b>Flujo Alterno 2: Lista vacía</b>	<ol style="list-style-type: none"> <li>1- El sistema detecta que no existen asistencias registradas en el sistema.</li> <li>2- El sistema muestra el mensaje 'No hay asistencias registradas'.</li> <li>3- El flujo termina.</li> </ol>
<b>Requerimientos especiales</b>	
<ol style="list-style-type: none"> <li>1- La tabla debe permitir ordenar por cualquier columna.</li> <li>2- El sistema debe calcular automáticamente las horas trabajadas.</li> <li>3- La búsqueda debe realizarse en tiempo real mientras el usuario escribe.</li> </ol>	
<b>Postcondiciones</b>	
<ol style="list-style-type: none"> <li>1- El usuario visualiza la información solicitada.</li> <li>2- No se realizan cambios en la base de datos.</li> </ol>	

## Módulo de Reportes

**Tabla 26**

*Caso de uso 16.*

<b>Prototipo: Sistema RRHH - Tendenzza</b>	
<b>Número Caso de Uso: 16</b>	<b>Nombre del Caso de Uso:</b> Generar reporte de asistencias
<b>Fecha elaboración:</b>	13/12/2025
<b>Descripción Caso de Uso:</b>	Permite generar un reporte de asistencias de los colaboradores según los filtros seleccionados, así como exportar dicho reporte en diferentes formatos.
<b>Autor caso de uso:</b>	Miguel
<b>Actores relacionados:</b>	Administrador, jefe RRHH, jefe Operaciones
<b>Precondiciones:</b>	<ul style="list-style-type: none"> <li>- El usuario debe estar autenticado con el rol autorizado.</li> <li>- Deben existir registros de asistencias en la base de datos.</li> </ul>
<b>Flujo Básico del caso de uso</b>	
Este caso de uso comienza cuando el usuario desea generar un reporte de asistencias. El usuario accede al módulo de reportes, selecciona los filtros deseados y el sistema genera el reporte con la información correspondiente.	
<b>Detallar el paso a paso del Flujo Básico</b>	
<ol style="list-style-type: none"> <li>1- El usuario accede al módulo de Reportes desde el menú principal.</li> <li>2- El usuario selecciona la opción 'Reporte de asistencias'.</li> <li>3- El sistema muestra el formulario de filtros disponibles.</li> <li>4- El usuario selecciona el rango de fechas (fecha inicio y fecha fin). <ul style="list-style-type: none"> <li>• Si el rango de fechas no es válido, se ejecuta el Flujo Alterno 2.</li> </ul> </li> <li>5- El usuario selecciona un colaborador específico o la opción 'Todos los colaboradores'.</li> <li>6- El usuario selecciona el estado de asistencia o la opción 'Todos los estados'.</li> <li>7- El usuario selecciona 'Generar reporte'.</li> <li>8- El sistema valida los filtros ingresados. <ul style="list-style-type: none"> <li>• Si no existen resultados con los filtros seleccionados, se ejecuta el Flujo Alterno 1.</li> </ul> </li> <li>9- El sistema consulta la información en la base de datos.</li> <li>10- El sistema genera y muestra el reporte de asistencias en pantalla.</li> <li>11- El usuario visualiza el reporte generado.</li> <li>12- El usuario puede seleccionar exportar el reporte. <ul style="list-style-type: none"> <li>• Si el usuario selecciona 'Exportar PDF', se ejecuta el Sub-Flujo 1.</li> <li>• Si el usuario selecciona 'Exportar Excel', se ejecuta el Sub-Flujo 2.</li> </ul> </li> </ol>	
<b>Sub-Flujos</b>	
<b>Sub-Flujo 1: Exportar PDF</b>	<ol style="list-style-type: none"> <li>1- El usuario selecciona la opción 'Exportar PDF'.</li> <li>2- El sistema genera el archivo en formato PDF.</li> <li>3- El sistema descarga el archivo en el dispositivo del usuario.</li> </ol>
<b>Sub-Flujo 1: Exportar a Excel</b>	<ol style="list-style-type: none"> <li>1- El usuario selecciona 'Exportar Excel'.</li> <li>2- El sistema genera el archivo en formato Excel.</li> <li>3- El sistema descarga el archivo en el dispositivo del usuario.</li> </ol>
<b>Flujos Alternos</b>	
Corresponde a lo que debe realizar el sistema ante posibles errores	
<b>Flujo Alterno 1: Sin resultados</b>	<ol style="list-style-type: none"> <li>1- El sistema no encuentra asistencias con los filtros seleccionados.</li> <li>2- El sistema muestra el mensaje 'No se encontraron datos para los filtros seleccionados'.</li> <li>3- El flujo regresa al paso 4 del Flujo Básico.</li> </ol>

<b>Flujo Alternativo 2: Fechas inválidas</b>	<ol style="list-style-type: none"> <li>1- El sistema detecta que la fecha fin es anterior a la fecha de inicio.</li> <li>2- El sistema muestra el mensaje 'El rango de fechas no es válido'.</li> <li>3- El flujo regresa al paso 4 del Flujo Básico.</li> </ol>
<b>Requerimientos especiales</b>	
<ol style="list-style-type: none"> <li>1- El reporte debe mostrar totales de horas trabajadas por el colaborador.</li> <li>2- El reporte debe incluir gráficas de asistencia.</li> <li>3- Los archivos exportados deben incluir el logo de la empresa.</li> </ol>	
<b>Postcondiciones</b>	
<ol style="list-style-type: none"> <li>1- El usuario visualiza el reporte generado.</li> <li>2- El usuario puede descargar el reporte en formato PDF o Excel.</li> <li>3- No se realizan cambios en la base de datos.</li> </ol>	

**Tabla 27***Caso de uso 17.*

<b>Prototipo: Sistema RRHH – Tendenzza</b>	
<b>Número Caso de Uso: 17</b>	<b>Nombre del Caso de Uso:</b> Generar reporte de vacaciones
<b>Fecha elaboración:</b>	13/12/2025
<b>Descripción Caso de Uso:</b>	Permite generar un reporte de vacaciones de los colaboradores según los filtros seleccionados, así como exportar dicho reporte en diferentes formatos.
<b>Autor caso de uso:</b>	Miguel
<b>Actores relacionados:</b>	Administrador, jefe RRHH
<b>Precondiciones:</b>	<ul style="list-style-type: none"> <li>- El usuario debe estar autenticado con rol Administrador o jefe RRHH.</li> <li>- Deben existir registros de vacaciones en la base de datos.</li> </ul>
<b>Flujo Básico del caso de uso</b>	
Este caso de uso comienza cuando el usuario desea generar un reporte de vacaciones. El usuario selecciona los filtros deseados y el sistema genera el reporte con la información correspondiente.	
<b>Detallar el paso a paso del Flujo Básico</b>	
<ol style="list-style-type: none"> <li>1- El usuario accede al módulo de Reportes desde el menú principal.</li> <li>2- El usuario selecciona la opción 'Reporte de vacaciones'.</li> <li>3- El sistema muestra el formulario de filtros disponibles.</li> <li>4- El usuario selecciona el rango de fechas (fecha inicio y fecha fin). <ul style="list-style-type: none"> <li>• Si el rango de fechas no es válido, se ejecuta el Flujo Alternativo 2.</li> </ul> </li> <li>5- El usuario selecciona un colaborador específico o todos los colaboradores.</li> <li>6- El usuario selecciona el estado de vacaciones o la opción todos los estados.</li> <li>7- El usuario selecciona la opción 'Generar Reporte'.</li> <li>8- El sistema valida los filtros ingresados. <ul style="list-style-type: none"> <li>• Si no existen registros con los filtros seleccionados, se ejecuta el Flujo Alternativo 1.</li> </ul> </li> <li>9- El sistema consulta la información en la base de datos.</li> <li>10- El sistema genera y muestra el reporte de vacaciones en pantalla.</li> <li>11- El usuario visualiza el reporte generado. <ul style="list-style-type: none"> <li>• Si el usuario selecciona 'Exportar PDF', se ejecuta el Sub-Flujo 1.</li> <li>• Si el usuario selecciona 'Exportar Excel', se ejecuta el Sub-Flujo 2.</li> </ul> </li> </ol>	
<b>Sub-Flujos</b>	
<b>Sub-Flujo 1: Exportar PDF</b>	<ol style="list-style-type: none"> <li>1- El usuario selecciona 'Exportar PDF'.</li> <li>2- El sistema genera el archivo en formato PDF.</li> <li>3- El sistema descarga el archivo en el dispositivo del usuario.</li> </ol>

<b>Sub-Flujo 2: Exportar a Excel</b>	<ol style="list-style-type: none"> <li>1- El usuario selecciona 'Exportar Excel.</li> <li>2- El sistema genera el archivo en formato Excel.</li> <li>3- El sistema descarga el archivo en el dispositivo el usuario.</li> </ol>
<b>Flujos Alternos</b>	
Corresponde a lo que debe realizar el sistema ante posibles errores	
<b>Flujo Alternativo 1: Sin resultados</b>	<ol style="list-style-type: none"> <li>1- El sistema no encuentra vacaciones con los filtros seleccionados.</li> <li>2- El sistema muestra el mensaje 'No se encontraron datos para los filtros seleccionados'.</li> <li>3- El flujo regresa al paso 4 del Flujo Básico.</li> </ol>
<b>Flujo Alternativo 2: Fechas invalidas</b>	<ol style="list-style-type: none"> <li>1- El sistema detecta que la fecha fin es anterior a la fecha de inicio.</li> <li>2- El sistema muestra el mensaje 'El rango de fechas no es válido'.</li> <li>3- El flujo regresa al paso 4 del Flujo Básico.</li> </ol>
<b>Requerimientos especiales</b>	
<ol style="list-style-type: none"> <li>1- El reporte debe mostrar el saldo de vacaciones disponible por colaborador.</li> <li>2- El reporte debe incluir el total de días tomados y días pendientes.</li> <li>3- Los archivos exportados deben incluir el logo de la empresa.</li> </ol>	
<b>Postcondiciones</b>	
<ol style="list-style-type: none"> <li>1- El usuario visualiza el reporte generado.</li> <li>2- El usuario puede descargar el reporte en formato PDF o Excel.</li> <li>3- No se realizan cambios en la base de datos.</li> </ol>	

**Tabla 28***Caso de uso 18.*

<b>Prototipo: Sistema RRHH – Tendenzza</b>	
<b>Número Caso de Uso: 18</b>	<b>Nombre del Caso de Uso:</b> Generar reporte de nómina.
<b>Fecha elaboración:</b>	13/12/2025
<b>Descripción Caso de Uso:</b>	Permite generar un reporte de nómina de los colaboradores según el período seleccionado, mostrando el detalle de los ingresos, deducciones y totales, así como exportar dicho reporte en diferentes formatos.
<b>Autor caso de uso:</b>	Miguel
<b>Actores relacionados:</b>	Administrador, jefe RRHH
<b>Precondiciones:</b>	<ul style="list-style-type: none"> <li>- El usuario debe estar autenticado con el rol de Administrador o jefe RRHH.</li> <li>- Deben existir registros de nómina en la base de datos.</li> </ul>
<b>Flujo Básico del caso de uso</b>	
Este caso de uso comienza cuando el usuario desea generar un reporte nómina. El usuario accede al módulo de reportes, selecciona el período y los filtros correspondientes, y el sistema genera el reporte con la información de nómina.	
<b>Detallar el paso a paso del Flujo Básico</b>	
<ol style="list-style-type: none"> <li>1- El usuario accede al módulo de Reportes desde el menú principal.</li> <li>2- El usuario selecciona la opción 'Reporte de nómina'.</li> <li>3- El sistema muestra el formulario de filtros disponibles.</li> <li>4- El usuario selecciona el período de nómina (mes y año). <ul style="list-style-type: none"> <li>• Si el periodo seleccionado no tiene nómina procesada, se ejecuta el Flujo Alternativo 2.</li> </ul> </li> <li>5- El usuario selecciona el colaborador específico o la opción Todos los colaboradores.</li> <li>6- El usuario selecciona 'Generar reporte'.</li> <li>7- El sistema valida los filtros ingresados. <ul style="list-style-type: none"> <li>• Si no existen registros de nómina</li> </ul> </li> <li>8- El sistema consulta la información en la base de datos.</li> <li>9- El sistema genera y muestra el reporte en la pantalla.</li> </ol>	

10- El usuario visualiza el reporte con el detalle de nómina (salario base, deducciones, horas extra, total a pagar).	
11- El usuario puede seleccionar exportar el reporte. <ul style="list-style-type: none"> <li>• Si el usuario selecciona 'Exportar PDF', se ejecuta el Sub-Flujo 1.</li> <li>• Si el usuario selecciona 'Exportar Excel', se ejecuta el Sub-Flujo 2.</li> </ul>	
<b>Sub-Flujos</b>	
<b>Sub-Flujo 1: Exportar a PDF</b>	<ol style="list-style-type: none"> <li>1- El usuario selecciona 'Exportar PDF'.</li> <li>2- El sistema genera el archivo en formato PDF.</li> <li>3- El sistema descarga el archivo en el dispositivo del usuario.</li> </ol>
<b>Sub-Flujo 2: Exportar a Excel</b>	<ol style="list-style-type: none"> <li>1- El usuario selecciona 'Exportar a Excel'.</li> <li>2- El sistema genera el archivo en formato Excel.</li> <li>3- El sistema descarga el archivo en el dispositivo del usuario.</li> </ol>
<b>Flujos Alternos</b>	
Corresponde a lo que debe realizar el sistema ante posibles errores	
<b>Flujo Alternativo 1: Sin resultados</b>	<ol style="list-style-type: none"> <li>1- El sistema no encuentra nóminas con los filtros seleccionados.</li> <li>2- El sistema muestra el mensaje 'No se encontraron datos para el período seleccionado'.</li> <li>3- El flujo regresa al paso 4 del Flujo Básico.</li> </ol>
<b>Flujo Alternativo 2: Período no procesado</b>	<ol style="list-style-type: none"> <li>1- El sistema detecta que el período seleccionado no tiene nómina procesada.</li> <li>2- El sistema muestra el mensaje 'No existe nómina procesada para este período'.</li> <li>3- El flujo regresa al paso 4 del Flujo Básico.</li> </ol>
<b>Requerimientos especiales</b>	
<ol style="list-style-type: none"> <li>1- El reporte debe mostrar el desglose completo (salario base, horas extra, deducciones, total).</li> <li>2- El reporte debe incluir totales generales de la nómina.</li> <li>3- Los archivos exportados deben incluir el logo de la empresa.</li> </ol>	
<b>Postcondiciones</b>	
<ol style="list-style-type: none"> <li>1- El usuario visualiza el reporte generado.</li> <li>2- El usuario puede descargar el reporte en formato PDF o Excel.</li> <li>3- No se realizan cambios en la base de datos.</li> </ol>	

**Tabla 29***Caso de uso 19.*

<b>Prototipo: Sistema RRHH – Tendenzza</b>	
<b>Número Caso de Uso: 19</b>	<b>Nombre del Caso de Uso:</b> Generar reporte de evaluaciones
<b>Fecha elaboración:</b>	13/12/2025
<b>Descripción Caso de Uso:</b>	Permite generar un reporte de evaluaciones de desempeño de los colaboradores según los filtros seleccionados, mostrando información relevante como período evaluado, calificación, estado y observaciones, así como exportar el reporte en distintos formatos.
<b>Autor caso de uso:</b>	Miguel

<b>Actores relacionados:</b>	Administrador, jefe RRHH, jefe de Operaciones, jefe de Ventas
<b>Precondiciones:</b>	<ul style="list-style-type: none"> <li>- El usuario debe estar autenticado con el rol autorizado.</li> <li>- Deben existir registros de evaluaciones en la base de datos.</li> </ul>
<b>Flujo Básico del caso de uso</b>	
Este caso de uso comienza cuando el usuario desea generar un reporte de evaluaciones de desempeño. El usuario accede al módulo de reportes, selecciona los filtros correspondientes y el sistema genera el reporte con la información solicitada.	
<b>Detallar el paso a paso del Flujo Básico</b>	
<ol style="list-style-type: none"> <li>1- El usuario accede al módulo de Reportes desde el menú principal.</li> <li>2- El usuario selecciona la opción 'Reporte de evaluaciones'.</li> <li>3- El sistema muestra el formulario de filtros disponibles.</li> <li>4- El usuario selecciona el rango de fechas (fecha inicio y fecha fin). <ul style="list-style-type: none"> <li>• Si la fecha fin es anterior a la fecha inicio, se ejecuta el Flujo Alterno 2.</li> </ul> </li> <li>5- El usuario selecciona un colaborador específico o la opción Todos los colaboradores.</li> <li>6- El usuario selecciona el estado de evaluación o la opción Todos los estados.</li> <li>7- El usuario selecciona la opción 'Generar reporte'.</li> <li>8- El sistema valida los filtros ingresados. <ul style="list-style-type: none"> <li>• Si no existen evaluaciones que coincidan con los filtros seleccionados, se ejecuta el Flujo Alterno 1.</li> </ul> </li> <li>9- El sistema consulta la información en la base de datos.</li> <li>10- El sistema genera y muestra el reporte en pantalla.</li> <li>11- El usuario visualiza el reporte con los datos de evaluaciones (colaboradores, período, calificación, estado y observaciones).</li> <li>12- El usuario puede seleccionar exportar el reporte. <ul style="list-style-type: none"> <li>• Si el usuario selecciona 'Exportar PDF', se ejecuta el Sub-Flujo 1.</li> <li>• Si el usuario selecciona 'Exportar Excel', se ejecuta el Sub-Flujo 2.</li> </ul> </li> </ol>	
<b>Sub-Flujos</b>	
<b>Sub-Flujo 1: Exportar a PDF</b>	<ol style="list-style-type: none"> <li>1- El usuario selecciona la opción 'Exportar PDF'.</li> <li>2- El sistema genera el archivo en formato PDF.</li> <li>3- El sistema descarga el archivo en el dispositivo del usuario.</li> </ol>
<b>Sub-Flujo 2: Exportar a Excel</b>	<ol style="list-style-type: none"> <li>1- El usuario selecciona la opción 'Exportar Excel'.</li> <li>2- El sistema genera el archivo en formato Excel.</li> <li>3- El sistema descarga el archivo en el dispositivo del usuario.</li> </ol>
<b>Flujos Alternos</b>	
Corresponde a lo que debe realizar el sistema ante posibles errores	
<b>Flujo Alterno 1: Sin resultados</b>	<ol style="list-style-type: none"> <li>1- El sistema no encuentra evaluaciones con los filtros seleccionados.</li> <li>2- El sistema muestra el mensaje 'No se encontraron datos para los filtros seleccionados'.</li> <li>3- El flujo regresa al paso 4 del Flujo Básico.</li> </ol>
<b>Flujo Alterno 2: Fechas inválidas</b>	<ol style="list-style-type: none"> <li>1- El sistema detecta que la fecha fin es anterior a la fecha inicio.</li> <li>2- El sistema muestra el mensaje 'El rango de fechas no es válido'.</li> <li>3- El flujo regresa al paso 4 del Flujo Básico.</li> </ol>
<b>Requerimientos especiales</b>	
<ol style="list-style-type: none"> <li>1- El reporte debe mostrar el promedio de calificaciones por colaborador.</li> <li>2- El reporte debe incluir gráficas de desempeño.</li> <li>3- Los archivos exportados deben incluir el logo de la empresa.</li> </ol>	
<b>Postcondiciones</b>	
<ol style="list-style-type: none"> <li>1- El usuario visualiza el reporte generado.</li> <li>2- El usuario puede descargar el reporte en formato PDF O Excel.</li> <li>3- No se realizan cambios en la base de datos.</li> </ol>	

**Tabla 30***Caso de uso 20.*

<b>Prototipo: Sistema RRHH – Tendenzza</b>	
<b>Número Caso de Uso: 20</b>	<b>Nombre del Caso de Uso:</b> Exportar reporte a PDF / Excel
<b>Fecha elaboración:</b>	13/12/2025
<b>Descripción Caso de Uso:</b>	Permite exportar cualquier reporte generado en el sistema a formato PDF o Excel para su descarga y almacenamiento en el dispositivo del usuario.
<b>Autor caso de uso:</b>	Miguel
<b>Actores relacionados:</b>	Administrador, jefe RRHH, jefe Operaciones, jefe de Ventas
<b>Precondiciones:</b>	<ul style="list-style-type: none"> <li>- El usuario debe estar autenticado con rol autorizado.</li> <li>- Debe existir un reporte previamente generado y visualizado en pantalla.</li> </ul>
<b>Flujo Básico del caso de uso</b>	
Este caso de uso comienza cuando el usuario desea exportar un reporte que ya fue generado en pantalla. El usuario selecciona el formato de exportación y el sistema genera el archivo correspondiente.	
<b>Detallar el paso a paso del Flujo Básico</b>	
<ol style="list-style-type: none"> <li>1- El usuario genera un reporte en el sistema (asistencias, vacaciones, nómina o evaluaciones).</li> <li>2- El sistema muestra el reporte en generado en pantalla.</li> <li>3- El usuario selecciona la opción 'Exportar'.</li> <li>4- El sistema muestra las opciones de formato disponibles (PDF, Excel).</li> <li>5- El usuario selecciona el formato deseado.</li> <li>6- El sistema valida que el reporte contenga información. <ul style="list-style-type: none"> <li>• Si el reporte no contiene datos, se ejecuta el Flujo Alterno 2.</li> </ul> </li> <li>7- El sistema genera el archivo en el formato seleccionado. <ul style="list-style-type: none"> <li>• Si ocurre un error durante la generación del archivo, se ejecuta el Flujo Alterno 1.</li> </ul> </li> <li>8- El sistema muestra el mensaje 'Archivo exportado correctamente'.</li> <li>9- El sistema descarga el archivo en el dispositivo del usuario.</li> <li>10- El flujo finaliza.</li> </ol>	
<b>Sub-Flujos</b>	
<b>Sub-Flujo 1: Exportar a PDF</b>	<ol style="list-style-type: none"> <li>1- El usuario selecciona 'PDF'.</li> <li>2- El sistema genera el archivo en formato PDF, incluyendo encabezado, logo de la empresa y fecha de generación.</li> <li>3- El sistema descarga el archivo PDF en el dispositivo del usuario.</li> </ol>
<b>Sub-Flujo 2: Exportar a Excel</b>	<ol style="list-style-type: none"> <li>1- El usuario selecciona la opción 'Excel'.</li> <li>2- El sistema genera el archivo con formato de hoja de cálculo con estructura de tabla y filtros habilitados.</li> <li>3- El sistema descarga el archivo en formato .xlsx en el dispositivo del usuario.</li> </ol>
<b>Flujos Alternos</b>	
Corresponde a lo que debe realizar el sistema ante posibles errores	
<b>Flujo Alterno 1: Error al generar archivo</b>	<ol style="list-style-type: none"> <li>1- El sistema no puede generar el archivo en el formato seleccionado.</li> <li>2- El sistema muestra el mensaje 'Error al exportar el reporte, intente nuevamente'.</li> <li>3- El flujo regresa al paso 3 del Flujo Básico.</li> </ol>
<b>Flujo Alterno 2: Reporte vacío</b>	<ol style="list-style-type: none"> <li>1- El sistema detecta que el reporte no contiene datos.</li> <li>2- El sistema muestra del mensaje 'No hay datos para exportar'.</li> <li>3- El flujo finaliza.</li> </ol>
<b>Requerimientos especiales</b>	

<ol style="list-style-type: none"> <li>1- Los archivos PDF deben incluir encabezado con logo de la empresa y fecha de generación.</li> <li>2- Los archivos Excel deben presentarse en formato de tabla con filtros habilitados.</li> <li>3- El nombre del archivo debe incluir el tipo de reporte y la fecha de exportación.</li> </ol>
<b>Postcondiciones</b>
<ol style="list-style-type: none"> <li>1- El usuario obtiene el archivo en el formato seleccionado.</li> <li>2- El archivo queda guardado en el dispositivo del usuario.</li> <li>3- No se realizan cambios en la base de datos.</li> </ol>

## Módulo de Seguridad

**Tabla 31**

*Caso de uso 21.*

<b>Prototipo: Sistema RRHH – Tendenzza</b>	
<b>Número Caso de Uso: 21</b>	<b>Nombre del Caso de Uso:</b> Iniciar sesión
<b>Fecha elaboración:</b>	13/12/2025
<b>Descripción Caso de Uso:</b>	Permite a un usuario autenticarse en el sistema mediante sus credenciales para acceder a las funcionalidades disponibles según su rol asignado.
<b>Autor caso de uso:</b>	Miguel
<b>Actores relacionados:</b>	Administrador, jefe RRHH, jefe Operaciones, jefe Ventas, Colaborador
<b>Precondiciones:</b>	<ul style="list-style-type: none"> <li>- El usuario debe estar registrado en el sistema.</li> <li>- El usuario debe tener un rol asignado.</li> <li>- El usuario debe estar activo en el sistema.</li> </ul>
<b>Flujo Básico del caso de uso</b>	
Este caso de uso comienza cuando el usuario desea ingresar al sistema. El usuario ingresa sus credenciales y el sistema valida la información para permitir el acceso.	
<b>Detallar el paso a paso del Flujo Básico</b>	
<ol style="list-style-type: none"> <li>1- El usuario accede a la página de inicio de sesión del sistema.</li> <li>2- El sistema muestra el formulario de inicio de sesión.</li> <li>3- El usuario ingres su nombre de usuario.</li> <li>4- El usuario ingresa su contraseña.</li> <li>5- El usuario selecciona ‘Iniciar Sesión’.</li> <li>6- El sistema valida que los campos obligatorios estén completados. <ul style="list-style-type: none"> <li>• Si existen campos vacíos, se ejecuta el Flujo Alterno 3.</li> </ul> </li> <li>7- El sistema valida las credenciales ingresadas. <ul style="list-style-type: none"> <li>• Si las credenciales son incorrectas, se ejecuta el Flujo Alterno 1.</li> </ul> </li> <li>8- El sistema verifica que el usuario se encuentre activo. <ul style="list-style-type: none"> <li>• Si el usuario está inactivo, se ejecuta el Flujo Alterno 2.</li> </ul> </li> <li>9- El sistema verifica el rol asignado al usuario.</li> <li>10- El sistema crea la sesión del usuario.</li> <li>11- El sistema redirige al usuario al dashboard principal según su rol.</li> <li>12- El sistema muestra el mensaje ‘Bienvenido al sistema’.</li> <li>13- El flujo finaliza.</li> </ol>	
<b>Sub-Flujos</b>	
<b>Sub-Flujo 1: Recordar credenciales</b>	<ol style="list-style-type: none"> <li>1- El usuario marca la opción ‘Recordarme’.</li> <li>2- El sistema guarda de forma segura las credenciales en el navegador del usuario.</li> <li>3- En el próximo ingreso, el sistema autocompleta los campos de inicio de sesión.</li> </ol>

<b>Flujos Alternos</b>	
Corresponde a lo que debe realizar el sistema ante posibles errores	
<b>Flujo Alternativo 1: Credenciales Incorrectos</b>	1- El sistema detecta que el usuario o contraseña no coinciden. 2- El sistema muestra el mensaje 'Usuario o contraseña incorrectos'. 3- El flujo regresa al paso 3 del Flujo Básico.
<b>Flujo Alternativo 2: Usuario inactivo</b>	1- El sistema detecta que el usuario está inactivo. 2- El sistema muestra el mensaje 'Su cuenta está deshabilitada, contacte al administrador'. 3- El flujo finaliza.
<b>Flujo Alternativo 3: Campos vacíos</b>	1- El sistema detecta campos obligatorios sin completar. 2- El sistema muestra el mensaje 'Debe completar todos los campos'. 3- El flujo regresa al paso 3 del Flujo Básico.
<b>Requerimientos especiales</b>	
1- La contraseña debe mostrarse oculta, con opción de visualizarla. 2- El sistema debe bloquear la cuenta después de 3 intentos fallidos. 3- La sesión debe expirar después de 30 minutos de inactividad.	
<b>Postcondiciones</b>	
1- El usuario queda autenticado en el sistema. 2- Se crea una sesión activa para el usuario. 3- El usuario accede a las funcionalidades según su rol asignado.	

**Tabla 32***Caso de uso 22.*

<b>Prototipo: Sistema RRHH – Tendenzza</b>	
<b>Número Caso de Uso: 22</b>	<b>Nombre del Caso de Uso:</b> Cerrar sesión
<b>Fecha elaboración:</b>	13/12/2025
<b>Descripción Caso de Uso:</b>	Permite a un usuario finalizar su sesión activa en el sistema.
<b>Autor caso de uso:</b>	Miguel
<b>Actores relacionados:</b>	Administrador, jefe RRHH, jefe de Operaciones, jefe de Ventas, Colaborador
<b>Precondiciones:</b>	- El usuario debe tener una sesión activa en el sistema.
<b>Flujo Básico del caso de uso</b>	
Este caso de uso comienza cuando el usuario desea salir del sistema. El usuario selecciona la opción de cerrar sesión y el sistema finaliza la sesión activa de forma segura.	
<b>Detallar el paso a paso del Flujo Básico</b>	
1- El usuario selecciona el nombre de usuario en la barra superior. 2- El sistema muestra el menú de opciones de usuario. 3- El usuario selecciona la opción 'Cerrar sesión'. 4- El sistema solicita confirmación para cerrar sesión. 5- El usuario confirma la acción. <ul style="list-style-type: none"> <li>• Si el usuario cancela la acción, se ejecuta la Su-Flujo 1.</li> </ul> 6- El sistema elimina y destruye la sesión activa del usuario. 7- El sistema elimina los datos de sesión almacenados en el navegador. 8- El sistema redirige al usuario a la página de inicio de sesión. 9- El sistema muestra el mensaje 'Sesión cerrada correctamente'. 10- El flujo finaliza.	
<b>Sub-Flujos</b>	

<b>Sub-Flujo 1: Cancelar cierre</b>	<ol style="list-style-type: none"> <li>1- El usuario selecciona 'Cancelar' en la confirmación.</li> <li>2- El sistema cierra el mensaje de confirmación.</li> <li>3- El usuario permanece en la pantalla actual con su sesión activa.</li> </ol>
<b>Flujos Alternos</b>	
Corresponde a lo que debe realizar el sistema ante posibles errores	
<b>Flujo Alternativo 1: Sesión expirada</b>	<ol style="list-style-type: none"> <li>1- El sistema detecta que la sesión ha expirado por inactividad.</li> <li>2- El sistema redirige automáticamente al usuario a la página de inicio de sesión.</li> <li>3- El sistema muestra el mensaje 'Su sesión ha expirado'.</li> <li>4- El flujo finaliza.</li> </ol>
<b>Requerimientos especiales</b>	
<ol style="list-style-type: none"> <li>1- El sistema debe eliminar completamente todos los datos de sesión del navegador.</li> <li>2- El usuario no debe poder acceder a páginas protegidas después de cerrar sesión.</li> <li>3- El cierre debe ejecutarse de forma inmediata.</li> </ol>	
<b>Postcondiciones</b>	
<ol style="list-style-type: none"> <li>1- La sesión del usuario queda completamente destruida.</li> <li>2- El usuario es redirigido a la página de inicio de sesión.</li> <li>3- El usuario debe autenticarse nuevamente para acceder al sistema.</li> </ol>	

**Tabla 33***Caso de uso 23.*

<b>Prototipo: Sistema RRHH – Tendenzza</b>	
<b>Número Caso de Uso: 23</b>	<b>Nombre del Caso de Uso:</b> Registrar un usuario
<b>Fecha elaboración:</b>	13/12/2025
<b>Descripción Caso de Uso:</b>	Permite al administrador crear un nuevo usuario en el sistema, asociándolo a un colaborador existente y asignándole credenciales de acceso.
<b>Autor caso de uso:</b>	Miguel
<b>Actores relacionados:</b>	Administrador
<b>Precondiciones:</b>	<ul style="list-style-type: none"> <li>- El usuario debe estar autenticado con el rol de administrador.</li> <li>- El colaborador debe existir en la base de datos.</li> <li>- El colaborador no debe tener un usuario asignado previamente.</li> </ul>
<b>Flujo Básico del caso de uso</b>	
Este caso de uso comienza cuando el administrador desea crear un nuevo usuario en el sistema. El administrador selecciona un colaborador, asigna las credenciales y el sistema registra el usuario.	
<b>Detallar el paso a paso del Flujo Básico</b>	
<ol style="list-style-type: none"> <li>1- El administrador accede al módulo de Usuarios desde el menú principal.</li> <li>2- Selecciona la opción 'Registrar usuario'.</li> <li>3- El sistema muestra el formulario de registro.</li> <li>4- El administrador busca y selecciona un colaborador. <ul style="list-style-type: none"> <li>• Si el colaborador no existe, se ejecuta el Flujo Alternativo 1.</li> </ul> </li> <li>5- El sistema muestra los datos del colaborador seleccionado.</li> <li>6- El administrador ingresa el nombre de usuario. <ul style="list-style-type: none"> <li>• Si el nombre de usuario ya existe, se ejecuta el Flujo Alternativo 2.</li> </ul> </li> <li>7- El administrador ingresa la contraseña.</li> <li>8- El administrador confirma la contraseña. <ul style="list-style-type: none"> <li>• Si las contraseñas no coinciden, se ejecuta el Flujo Alternativo 3.</li> </ul> </li> <li>9- El administrador selecciona el estado del usuario (activo, inactivo).</li> </ol>	

10- El administrador selecciona 'Guardar'. 11- El sistema valida los datos ingresados. 12- El sistema solicita confirmación para registrar el usuario. 13- El administrador confirma la operación. 14- El sistema guarda el usuario en la base de datos y lo asocia al colaborador. 15- El sistema muestra el mensaje 'Usuario registrado correctamente'. 16- El flujo finaliza.	
<b>Sub-Flujos</b>	
<b>Sub-Flujo 1: Cancelar operación</b>	1- El administrador selecciona 'Cancelar'. 2- El sistema descarta la información ingresada. 3- El sistema regresa a la lista de usuarios.
<b>Flujos Alternos</b>	
Corresponde a lo que debe realizar el sistema ante posibles errores	
<b>Flujo Alterno 1: Colaborador no encontrado</b>	1- El sistema no encuentra resultados en la búsqueda. 2- El sistema muestra el mensaje 'No se encontraron colaboradores'. 3- El flujo regresa al paso 4.
<b>Flujo Alterno 2: El usuario ya existe</b>	1- El sistema detecta que el nombre de usuario ya está registrado. 2- El sistema muestra el mensaje 'El nombre de usuario ya existe'. 3- El flujo regresa al paso 6.
<b>Flujo Alterno 3: Contraseñas no coinciden</b>	1- El sistema detecta que las contraseñas no coinciden. 2- El sistema muestra el mensaje 'Las contraseñas no coinciden'. 3- El flujo regresa al paso 7.
<b>Flujo Alterno 4: Colaborador ya tiene usuario</b>	1- El sistema detecta que el colaborador ya tiene un usuario asignado. 2- El sistema muestra el mensaje 'Este colaborador ya tiene un usuario asignado'. 3- El flujo regresa al paso 4.
<b>Requerimientos especiales</b>	
1- El nombre de usuario debe ser único en el sistema. 2- La contraseña debe tener al menos 8 caracteres. 3- La contraseña se almacena encriptada en la base de datos. 4- Solo el rol Administrador puede registrar usuarios.	
<b>Postcondiciones</b>	
1- El usuario queda registrado en la base de datos. 2- El usuario queda asociado al colaborador seleccionado. 3- El usuario puede iniciar sesión con las credenciales asignadas.	

**Tabla 34***Caso de uso 24.*

<b>Prototipo: Sistema RRHH – Tendenzza</b>	
<b>Número Caso de Uso: 24</b>	<b>Nombre del Caso de Uso: Asignar rol a usuario</b>
<b>Fecha elaboración:</b>	13/12/2025

<b>Descripción Caso de Uso:</b>	Permite asignar o modificar el rol de un usuario existente en el sistema para definir sus privilegios de acceso.
<b>Autor caso de uso:</b>	Miguel
<b>Actores relacionados:</b>	Administrador
<b>Precondiciones:</b>	<ul style="list-style-type: none"> <li>- El usuario debe estar autenticado con el rol Administrador.</li> <li>- El usuario al que se asignará el rol debe existir en la base de datos.</li> <li>- Deben existir roles definidos en el sistema.</li> </ul>
<b>Flujo Básico del caso de uso</b>	
Este caso de uso comienza cuando el administrador desea asignar o modificar el rol de un usuario existente. El administrador busca el usuario, selecciona el rol y el sistema actualiza la información.	
<b>Detallar el paso a paso del Flujo Básico</b>	
<ol style="list-style-type: none"> <li>1- El administrador accede al módulo de Usuarios desde el menú principal.</li> <li>2- El sistema muestra la lista de usuarios registrados.</li> <li>3- El administrador busca el usuario al que desea asignar o modificar el rol. <ul style="list-style-type: none"> <li>• Si el usuario no existe, se ejecuta el Flujo Alterno 1.</li> </ul> </li> <li>4- El sistema muestra el usuario encontrado.</li> <li>5- El administrador selecciona la opción 'Asignar rol'.</li> <li>6- El sistema muestra el formulario con el rol actual del usuario.</li> <li>7- El sistema muestra la lista de roles disponibles (Administrador, jefe RRHH, jefe Operaciones, jefe Ventas, Empleado).</li> <li>8- El administrador selecciona el nuevo rol. <ul style="list-style-type: none"> <li>• Si no se selecciona ningún rol, se ejecuta el Flujo Alterno 2.</li> </ul> </li> <li>9- El administrador selecciona 'Guardar'.</li> <li>10- El sistema solicita confirmación para asignar el rol.</li> <li>11- El administrador confirma la operación.</li> <li>12- El sistema actualiza el rol del usuario en la base de datos.</li> <li>13- El sistema muestra mensaje 'Rol asignado correctamente'.</li> <li>14- El flujo finaliza.</li> </ol>	
<b>Sub-Flujos</b>	
<b>Sub-Flujo 1: Cancelar operación</b>	<ol style="list-style-type: none"> <li>1- El administrador selecciona la opción 'Cancelar'.</li> <li>2- El sistema descarta los cambios realizados.</li> <li>3- El sistema regresa a la lista de usuarios.</li> </ol>
<b>Flujos Alternos</b>	
Corresponde a lo que debe realizar el sistema ante posibles errores	
<b>Flujo Alterno 1: Usuario no encontrado</b>	<ol style="list-style-type: none"> <li>1- El sistema no encuentra usuarios que coincidan con la búsqueda.</li> <li>2- El sistema muestra el mensaje 'No se encontraron usuarios'.</li> <li>3- El flujo regresa al paso 3.</li> </ol>
<b>Flujo Alterno 2: Rol no seleccionado</b>	<ol style="list-style-type: none"> <li>1- El sistema detecta que no se ha seleccionado un rol.</li> <li>2- El sistema muestra el mensaje 'Debe seleccionar un rol'.</li> <li>3- El flujo regresa al paso 8.</li> </ol>
<b>Requerimientos especiales</b>	
<ol style="list-style-type: none"> <li>1- Solo el rol administrador puede asignar o modificar roles de usuario.</li> <li>2- Un usuario solo puede tener un rol asignado a la vez.</li> <li>3- Los cambios de rol se aplican a partir de la próxima sesión del usuario.</li> </ol>	
<b>Postcondiciones</b>	
<ol style="list-style-type: none"> <li>1- El rol del usuario queda actualizado en la base de datos.</li> <li>2- Los permisos del usuario cambian según el nuevo rol asignado.</li> <li>3- El usuario debe cerrar e iniciar sesión para que los nuevos permisos se apliquen.</li> </ol>	

**Tabla 35***Caso de uso 25.*

<b>Prototipo: Sistema RRHH – Tendenzza</b>	
<b>Número Caso de Uso: 25</b>	<b>Nombre del Caso de Uso:</b> Gestionar permisos por rol
<b>Fecha elaboración:</b>	13/12/2025
<b>Descripción Caso de Uso:</b>	Permite al administrador configurar los permisos de acceso a las funcionalidades del sistema para cada rol definido.
<b>Autor caso de uso:</b>	Miguel
<b>Actores relacionados:</b>	Administrador
<b>Precondiciones:</b>	<ul style="list-style-type: none"> <li>- El usuario debe estar autenticado con el rol Administrador.</li> <li>- Deben existir roles definidos en el sistema.</li> <li>- Deben existir permisos disponibles en el catálogo del sistema.</li> </ul>
<b>Flujo Básico del caso de uso</b>	
Este caso de uso comienza cuando el administrador desea configurar los permisos asociados a un rol. El administrador selecciona el rol, define los permisos correspondientes y el sistema guarda la configuración.	
<b>Detallar el paso a paso del Flujo Básico</b>	
<ol style="list-style-type: none"> <li>1- El administrador accede al módulo de Roles y Permisos desde el menú principal.</li> <li>2- El sistema muestra la lista de roles disponibles.</li> <li>3- El administrador selecciona el rol que desea configurar.</li> <li>4- El sistema muestra la lista de permisos disponibles, organizados por módulo.</li> <li>5- El sistema marca automáticamente los permisos que el rol tiene actualmente asignados.</li> <li>6- El administrador marca o desmarca los permisos según corresponda. <ul style="list-style-type: none"> <li>• Puede ejecutar el Sub-Flujo 2: Seleccionar todos.</li> <li>• Puede ejecutar el Sub-Flujo 3: Desmarcar todos.</li> </ul> </li> <li>7- El administrador selecciona la opción ‘Guardar cambios’. <ul style="list-style-type: none"> <li>• Si no hay permisos seleccionados, se ejecuta el Flujo Alterno 1.</li> <li>• Si el rol es Administrador y se intenta quitar permisos críticos, se ejecuta el Flujo Alterno 2.</li> </ul> </li> <li>8- El sistema valida la configuración de permisos ingresada.</li> <li>9- El sistema solicita confirmación para guardar los cambios.</li> <li>10- El administrador confirma la operación.</li> <li>11- El sistema actualiza los permisos del rol en la base de datos.</li> <li>12- El sistema muestra el mensaje ‘Permisos actualizados correctamente’.</li> <li>13- El flujo finaliza.</li> </ol>	
<b>Sub-Flujos</b>	
<b>Sub-Flujo 1: Cancelar operación</b>	<ol style="list-style-type: none"> <li>1- El administrador selecciona ‘Cancelar’.</li> <li>2- El sistema descarta los cambios realizados.</li> <li>3- El sistema regresa a la lista de roles.</li> </ol>
<b>Sub-Flujo 2: Seleccionar todos</b>	<ol style="list-style-type: none"> <li>1- El administrador selecciona ‘Marcar todos’ dentro de un módulo.</li> <li>2- El sistema marca automáticamente todos los permisos de ese módulo.</li> </ol>
<b>Sub-Flujo 3: Desmarcar todos</b>	<ol style="list-style-type: none"> <li>1- El administrador selecciona la opción ‘Desmarcar todos’ en un módulo.</li> <li>2- El sistema desmarca todos los permisos de ese módulo.</li> </ol>
<b>Flujos Alternos</b>	
Corresponde a lo que debe realizar el sistema ante posibles errores	
<b>Flujo Alterno 1: Sin permisos seleccionados</b>	<ol style="list-style-type: none"> <li>1- El sistema detecta que no se ha seleccionado ningún permiso.</li> <li>2- El sistema muestra el mensaje ‘Debe seleccionar al menos un permiso’.</li> <li>3- El flujo regresa al paso 6.</li> </ol>
<b>Flujo Alterno 2: Rol protegido</b>	<ol style="list-style-type: none"> <li>1- El sistema detecta que el rol Administrador no puede quedar sin permisos obligatorios.</li> <li>2- El sistema muestra el mensaje ‘El rol Administrador debe mantener</li> </ol>

	<p>todos los permisos’.</p> <p>3- El flujo regresa al paso 6.</p>
<b>Requerimientos especiales</b>	
<p>1- Solo el rol Administrador puede gestionar permisos del sistema.</p> <p>2- Los permisos deben estar organizados por módulo (Mantenimientos, Consultas, Reportes, Seguridad).</p> <p>3- El rol Administrador siempre debe tener acceso completo al sistema.</p> <p>4- Los cambios se aplican de inmediato para todos los usuarios con ese rol.</p>	
<b>Postcondiciones</b>	
<p>1- Los permisos del rol quedan actualizados en la base de datos.</p> <p>2- Los usuarios con ese rol acceden únicamente a las funcionalidades permitidas.</p> <p>3- Los cambios se reflejan en la próxima sesión de los usuarios afectados.</p>	

## Módulo Gestión Financiera

**Tabla 36**

*Caso de uso 26.*

<b>Prototipo: Sistema RRHH – Tendenzza</b>	
<b>Número Caso de Uso: 26</b>	<b>Nombre del Caso de Uso:</b> Procesar Nómina
<b>Fecha elaboración:</b>	13/12/2025
<b>Descripción Caso de Uso:</b>	Permite generar el cálculo del pago de planilla de los colaboradores, considerando horas trabajadas, salario bruto, deducciones de ley, cálculo de renta, horas extra, incapacidades y permisos con o sin goce salarial.
<b>Autor caso de uso:</b>	Miguel
<b>Actores relacionados:</b>	Administrador, jefe RRHH
<b>Precondiciones:</b>	<ul style="list-style-type: none"> <li>- El usuario debe estar autenticado con el rol Administrador o jefe RRHH.</li> <li>- Deben existir colaboradores activos en la base de datos.</li> <li>- Deben existir registros de asistencia del período a procesar.</li> <li>- Los parámetros de deducciones de ley deben estar configurados en el sistema.</li> </ul>
<b>Flujo Básico del caso de uso</b>	
Este caso de uso comienza cuando el usuario desea generar el procesamiento de nómina de un período determinado. El usuario accede al módulo de nóminas, selecciona el período, el sistema calcula automáticamente todos los rubros y genera la planilla para su revisión y aprobación.	
<b>Detallar el paso a paso del Flujo Básico</b>	
<p>1- El usuario accede al módulo de Nóminas desde el menú principal.</p> <p>2- Selecciona la opción ‘Procesar Nómina’.</p> <p>3- El sistema muestra el formulario del Paso 1 (Periodo), con dos opciones de modo de generación.</p> <p>4- El usuario selecciona el modo de generación:</p> <ul style="list-style-type: none"> <li>• Todos los Colaboradores: genera nómina del mes completo para todos los colaboradores activos que aún no tienen nómina en ese período.</li> <li>• Colaborador Específico: permite seleccionar un colaborador y definir un rango de fechas personalizado para reparto proporcional del salario a la fecha.</li> </ul> <p>5- El usuario selecciona el mes y el año del período.</p> <ul style="list-style-type: none"> <li>• Si el modo es individual, el usuario también selecciona el colaborador, la fecha de inicio y la fecha fin del periodo.</li> </ul>	

- 6- El usuario selecciona 'Calcular Vista Previa'.
- 7- El sistema valida los datos ingresados.
  - Si ya existe una nómina procesada para todos los colaboradores del período, se ejecuta el Flujo Alterno 1.
- 8- El sistema obtiene los colaboradores a procesar, excluyendo automáticamente los que ya tienen nómina en ese período.
- 9- Para cada colaborador, el sistema calcula, días de permiso sin goce aprobados, días de incapacidad aprobados en el período, días efectivos de trabajo, salario calculado según días efectivos de trabajados, horas extra, bonificaciones, entre otros.
- 10- El sistema muestra el Paso 2, la vista previa con una tabla detallada por colaborador.
- 11- El sistema muestra los totales consolidados del período.
- 12- El usuario revisa los cálculos.
  - Si el usuario selecciona 'Cambiar período', se ejecuta el Sub-Flujo 1.
- 13- El usuario selecciona 'Confirmar y Generar Nóminas'.
- 14- El sistema muestra una confirmación mediante SweetAlert con el detalle de la operación.
- 15- El usuario confirma la operación.
- 16- El sistema guarda las nóminas y sus líneas de detalle en la base de datos.
- 17- El sistema vincula automáticamente las horas extra, vacaciones, incapacidades y evaluaciones del período a las nóminas generadas.
- 18- El sistema redirige al listado de nóminas del período con el mensaje 'Nóminas generadas correctamente'

#### Sub-Flujos

##### **Sub-Flujo 1: Cancelar operación**

- 1- El usuario selecciona 'Cambiar período'.
- 2- El sistema descarta los cálculos realizados.
- 3- El sistema regresa al paso 1.

#### Flujos Alternos

Corresponde a lo que debe realizar el sistema ante posibles errores

##### **Flujo Alterno 1: Nómina ya procesada.**

- 1- El sistema detecta que todos los colaboradores ya cuentan con una nómina para el período seleccionado.
- 2- El sistema muestra el mensaje 'Ya existe una nómina procesada para este período'.
- 3- El sistema oculta el botón de confirmación y muestra el botón 'Ver nóminas del período'.
- 4- El usuario puede consultar las nóminas existentes o regresar al paso 1 para seleccionar otro período.

##### **Flujo Alterno 2: Error en el cálculo**

- 1- El sistema detecta un error durante el procesamiento.
- 2- El sistema muestra el mensaje 'Error al procesar la nómina' con el detalle del error.
- 3- El flujo regresa al paso 6 del Flujo Básico.

#### Requerimientos especiales

- 1- El sistema debe calcular automáticamente las deducciones según la legislación vigente.
- 2- Los cálculos deben considerar el salario base, horas extra, incapacidades y permisos.
- 3- El sistema debe generar un desglose detallado por cada colaborador.
- 4- Solo roles autorizados pueden procesar nóminas.

#### Postcondiciones

- 1- La nómina queda registrada en la base de datos.
- 2- Las horas extra, vacaciones, incapacidades y evaluaciones del período quedan vinculadas a las nóminas correspondientes.
- 3- Los montos calculados quedan disponibles para reportes.
- 4- Se genera el historial de nómina del período.

**Tabla 37***Caso de uso 27.*

<b>Prototipo: Sistema RRHH – Tendenzza</b>	
<b>Número Caso de Uso: 27</b>	<b>Nombre del Caso de Uso:</b> Procesar Aguinaldo
<b>Fecha elaboración:</b>	13/12/2025
<b>Descripción Caso de Uso:</b>	Permite calcular y procesar el aguinaldo de los colaboradores activos de forma general o individual. El sistema calcula automáticamente el monto correspondiente basándose en los salarios devengados del 1 de diciembre al 30 de noviembre del año en curso. El administrador o jefe RRHH decide cuándo realizar el pago dentro del mes de diciembre. Se genera una boleta de aguinaldo que puede enviarse por correo electrónico al colaborador.
<b>Autor caso de uso:</b>	Miguel
<b>Actores relacionados:</b>	Administrador, jefe RRHH.
<b>Precondiciones:</b>	<ul style="list-style-type: none"> <li>- El usuario debe estar autenticado con el rol de Administrador o jefe RRHH.</li> <li>- Deben existir nóminas procesadas del período diciembre a noviembre para los colaboradores activos.</li> <li>- El aguinaldo del período en curso no debe haber sido procesado previamente.</li> </ul>
<b>Flujo Básico del caso de uso</b>	
Este caso de uso comienza cuando el usuario desea procesar el aguinaldo del período. El usuario accede al módulo de aguinaldo, selecciona el modo de generación (todos los colaboradores o uno específico) y el año a procesar. El sistema calcula automáticamente el monto correspondiente, muestra una vista previa, y el usuario confirma el procesamiento. Posteriormente decide cuándo marcarlo como pagado.	
<b>Detallar el paso a paso del Flujo Básico</b>	
<ol style="list-style-type: none"> <li>1- El usuario accede al módulo de Aguinaldo desde el menú principal.</li> <li>2- El sistema muestra la lista de aguinaldos procesados por año y el botón 'Calcular Aguinaldo'.</li> <li>3- El usuario selecciona 'Procesar Aguinaldo' e indica el año a procesar.</li> <li>4- El sistema muestra el formulario del paso 1 del Wizard con dos modos de generación disponibles: todos los colaboradores o un colaborador específico.</li> <li>5- El usuario selecciona el modo generación y el año a procesar.</li> <li>6- En modo general: el sistema verifica si todos los colaboradores activos ya tienen aguinaldo del período. Si es así, se ejecuta el Flujo Alterno 2. Si solo algunos tienen, el sistema continúa procesando únicamente los que faltan. En modo individual: el sistema verifica que ese colaborador específico no tenga aguinaldo del período. Si ya tiene, se ejecuta el Flujo Alterno 2.</li> <li>7- El selecciona 'Calcular Vista Previa'.</li> <li>8- El sistema calcula para cada colaborador pendiente: <ul style="list-style-type: none"> <li>• Suma de todos los salarios devengados.</li> <li>• Monto de aguinaldo = suma de salarios ÷ 12 (siempre dividido entre 12, independientemente de los meses laborados).</li> <li>• Si no existen nóminas del período, el sistema estima el monto usando el salario base actual multiplicado por los meses estimados desde la fecha de ingreso, dividido entre 12.</li> <li>• No se aplican deducciones de ningún tipo (ni CCSS, ni renta) conforme a la Ley 2412.</li> </ul> </li> <li>9- El sistema muestra la tabla de vista previa con: colaborador, meses laborados, salario promedio, monto de aguinaldo y neto a pagar, junto con los totales generales.</li> <li>10- El usuario revisa los montos y selecciona "Confirmar y Generar Aguinaldo(s)". Si los datos no son válidos, se ejecuta el Flujo Alterno 1.</li> <li>11- El sistema guarda los aguinaldos con estado "Pendiente" y muestra el mensaje de éxito indicando cuántos aguinaldos fueron generados.</li> <li>12- Cuando el usuario decide realizar el pago, selecciona "Marcar como Pagado" individual o "Pagar Todos". Se ejecuta el Sub-Flujo 1.</li> <li>13- El flujo finaliza.</li> </ol>	
<b>Sub-Flujos</b>	

<b>Sub-Flujo 1: Marcar como pagado y notificar</b>	<ol style="list-style-type: none"> <li>1- El usuario selecciona 'Marcar como Pagado' desde la lista de aguinaldos.</li> <li>2- El sistema solicita confirmación de la acción.</li> <li>3- El usuario confirma.</li> <li>4- El sistema actualiza el estado a 'Pagado', registra la fecha de pago y el usuario que autorizó.</li> <li>5- El sistema genera la boleta de aguinaldo individual para cada colaborador.</li> <li>6- El usuario puede seleccionar 'Enviar correos' para notificar a todos los colaboradores con su boleta.</li> <li>7- El sistema envía el correo con la boleta adjunta y registra el estado de envío.</li> <li>8- El sub-flujo finaliza.</li> </ol>
<b>Sub-Flujo 2: Exportar a Excel</b>	<ol style="list-style-type: none"> <li>1- El usuario selecciona 'Exportar a Excel' desde la vista del aguinaldo procesado.</li> <li>2- El sistema genera el archivo .xlsx con el detalle de cada colaborador: nombre, salarios del período y monto de aguinaldo.</li> <li>3- El sistema descarga el archivo automáticamente.</li> <li>4- El sub-flujo finaliza.</li> </ol>
<b>Flujos Alternos</b>	
Corresponde a lo que debe realizar el sistema ante posibles errores	
<b>Flujo Alternativo 1: Sin nóminas del período</b>	<ol style="list-style-type: none"> <li>1- El sistema detecta que no existen nóminas procesadas para el período seleccionado.</li> <li>2- El sistema muestra el mensaje 'No existen nóminas procesadas para el período seleccionado'.</li> <li>3- El flujo finaliza.</li> </ol>
<b>Flujo Alternativo 2: Aguinaldo ya procesado</b>	<ol style="list-style-type: none"> <li>1- El sistema detecta que ya existe un aguinaldo procesado para el período seleccionado.</li> <li>2- El sistema muestra el mensaje 'El aguinaldo de este período ya fue procesado'.</li> <li>3- El sistema redirige al usuario a la vista de detalle del aguinaldo existente.</li> <li>4- El flujo finaliza.</li> </ol>
<b>Flujo Alternativo 3: Error al enviar correo</b>	<ol style="list-style-type: none"> <li>1- El sistema detecta que uno o más colaboradores no tienen correo registrado o el envío falla.</li> <li>2- El sistema muestra un resumen indicando cuántos correos se enviaron exitosamente y cuántos fallaron.</li> <li>3- El aguinaldo permanece en estado 'Pagado'. El usuario puede reintentar los envíos fallidos individualmente.</li> </ol>
<b>Requerimientos especiales</b>	
<ol style="list-style-type: none"> <li>1- Solo los roles Administrador y jefe RRHH pueden procesar y marcar como pagado el aguinaldo.</li> <li>2- El período de cálculo es del 1 de diciembre al 30 de noviembre del año en curso.</li> <li>3- Colaboradores con menos de un año se calculan proporcional a los meses trabajados en el período.</li> <li>4- No se puede procesar el aguinaldo dos veces para el mismo período.</li> <li>5- El pago puede realizarse en cualquier momento de diciembre según decisión del jefe o administrador de RRHH.</li> <li>6- Una vez marcado como pagado, no puede revertirse el estado.</li> </ol>	
<b>Postcondiciones</b>	
<ol style="list-style-type: none"> <li>1- El aguinaldo queda registrado en la base de datos con el monto calculado por colaborador.</li> <li>2- Cuando se marca como pagado, queda registrada la fecha y el usuario que autorizó el pago.</li> <li>3- Se genera la boleta de aguinaldo disponible para descarga e impresión.</li> <li>4- Si se envió por correo, queda registrado el estado y la fecha de envío por colaborador.</li> </ol>	

**Tabla 38***Caso de uso 28.*

<b>Prototipo: Sistema RRHH – Tendenzza</b>	
<b>Número Caso de Uso: 28</b>	<b>Nombre del Caso de Uso:</b> Procesar Liquidación Laboral
<b>Fecha elaboración:</b>	13/12/2025
<b>Descripción Caso de Uso:</b>	Permite calcular y registrar la liquidación laboral de un colaborador cuando termina su relación laboral con la empresa. El sistema calcula automáticamente todos los rubros correspondientes según la legislación costarricense: preaviso, cesantía, vacaciones no disfrutadas y aguinaldo proporcional, tomando en cuenta los años, meses y días trabajados, así como horas extra pendientes. Se genera una boleta de liquidación que puede exportarse a Excel y enviarse por correo.
<b>Autor caso de uso:</b>	Miguel
<b>Actores relacionados:</b>	Administrador, jefe RRHH
<b>Precondiciones:</b>	<ul style="list-style-type: none"> <li>- El usuario debe estar autenticado con rol Administrador o jefe RRHH.</li> <li>- El colaborador por liquidar debe estar registrado en el sistema y activo.</li> <li>- Deben existir registros de asistencia, horas extra, vacaciones y nóminas previas del colaborador.</li> </ul>
<b>Flujo Básico del caso de uso</b>	
Este caso de uso comienza cuando el usuario desea registrar la liquidación de un colaborador. El usuario accede al módulo de liquidaciones, selecciona el colaborador, ingresa la fecha y motivo de salida, y el sistema calcula automáticamente todos los rubros correspondientes según la ley costarricense. El usuario revisa los montos y confirma el procesamiento.	
<b>Detallar el paso a paso del Flujo Básico</b>	
<ol style="list-style-type: none"> <li>1- El usuario accede al módulo de Liquidaciones desde el menú principal.</li> <li>2- El sistema muestra la lista de liquidaciones procesadas y el botón 'Nueva Liquidación'.</li> <li>3- El usuario selecciona 'Nueva Liquidación'.</li> <li>4- El sistema muestra el formulario con un selector de colaborador, fecha de salida y tipo de terminación.</li> <li>5- El usuario selecciona el colaborador a liquidar.</li> <li>6- El sistema carga automáticamente: fecha de ingreso, salario base, días de vacaciones no disfrutadas y horas extra pendientes.</li> <li>7- El usuario ingresa la fecha de salida y selecciona el tipo de terminación (renuncia o despido sin responsabilidad patronal).</li> <li>8- El usuario selecciona 'Calcular'. Si los datos no son válidos, se ejecuta el Flujo Alterno 1.</li> <li>9- El sistema calcula automáticamente: <ul style="list-style-type: none"> <li>• Preaviso: según tiempo trabajado (Art. 28 Código de Trabajo de Costa Rica).</li> <li>• Cesantía: según tabla legal, máximo 8 años (solo aplica en despido sin responsabilidad patronal).</li> <li>• Vacaciones no disfrutadas: proporcional a los días acumulados no tomados.</li> <li>• Aguinaldo proporcional: del 1 de diciembre hasta la fecha de salida.</li> <li>• Horas extra pendientes: suma de horas extra aprobadas y no incluidas en nómina previa.</li> </ul> </li> <li>10- El sistema muestra el desglose detallado de cada rubro calculado.</li> <li>11- El usuario revisa los montos. Si desea modificar algún dato, regresa al paso 7.</li> <li>12- El usuario selecciona 'Confirmar y Guardar'. Se ejecuta el Sub-Flujo 1.</li> <li>13- El sistema guarda la liquidación con estado 'Procesada' y muestra el mensaje 'Liquidación registrada correctamente'.</li> <li>14- El flujo finaliza.</li> </ol>	
<b>Sub-Flujos</b>	
<b>Sub-Flujo 1: Generar boleta y exportar</b>	<ol style="list-style-type: none"> <li>1- El sistema genera la boleta de liquidación en formato PDF con todos los rubros desglosados.</li> <li>2- El usuario puede seleccionar 'Exportar a Excel' para descargar el desglose en formato .xlsx.</li> <li>3- El usuario puede seleccionar 'Enviar por correo' para enviar la</li> </ol>

	<p>boleta al correo registrado del colaborador.</p> <p>4- El sistema actualiza el estado del colaborador a inactivo (borrado lógico).</p> <p>5- El sub-flujo finaliza.</p>
<b>Sub-Flujo 2: Marcar como pagada</b>	<p>1- El usuario localiza la liquidación procesada en la lista.</p> <p>2- El usuario selecciona 'Marcar como pagada'.</p> <p>3- El sistema solicita confirmación.</p> <p>4- El usuario confirma.</p> <p>5- El sistema actualiza el estado a 'Pagada' y registra la fecha y el usuario que autorizó el pago.</p> <p>6- El sub-flujo finaliza.</p>
<b>Flujos Alternos</b>	
Corresponde a lo que debe realizar el sistema ante posibles errores	
<b>Flujo Alternativo 1: Datos inválidos</b>	<p>1- El sistema detecta campos obligatorios vacíos o con formato incorrecto.</p> <p>2- El sistema resalta los campos que requieren corrección.</p> <p>3- El flujo regresa al paso 7 del Flujo Básico.</p>
<b>Flujo Alternativo 2: Colaborador ya liquidado</b>	<p>1- El sistema detecta que el colaborador seleccionado ya tiene una liquidación registrada.</p> <p>2- El sistema muestra el mensaje 'Este colaborador ya fue liquidado anteriormente'.</p> <p>3- El sistema ofrece la opción de ver la liquidación existente.</p> <p>4- El flujo finaliza.</p>
<b>Flujo Alternativo 3: Error al enviar correo</b>	<p>1- El sistema detecta que el colaborador no tiene correo registrado o el envío falla.</p> <p>2- El sistema muestra el mensaje 'No se pudo enviar el correo. Verifique el correo registrado del colaborador'.</p> <p>3- La liquidación permanece guardada. El usuario puede reintentar el envío desde la lista.</p>
<b>Requerimientos especiales</b>	
<p>1- Solo los roles Administrador y jefe RRHH pueden procesar liquidaciones.</p> <p>2- El cálculo del preaviso sigue el Artículo 28 del Código de Trabajo de Costa Rica.</p> <p>3- La cesantía aplica únicamente en caso de despido sin responsabilidad patronal, con tope máximo de 8 años.</p> <p>4- El aguinaldo proporcional se calcula desde el 1 de diciembre hasta la fecha de salida.</p> <p>5- No se puede modificar ni eliminar una liquidación que ya fue marcada como pagada.</p> <p>6- La exportación a Excel debe incluir el desglose completo de cada rubro con sus fórmulas.</p>	
<b>Postcondiciones</b>	
<p>1- La liquidación queda registrada en la base de datos con todos los rubros calculados.</p> <p>2- El colaborador queda inactivo en el sistema (borrado lógico).</p> <p>3- Se genera la boleta de liquidación disponible para descarga e impresión.</p> <p>4- Si se envió por correo, queda registrado el estado y la fecha de envío.</p>	

## Módulo de Mantenimientos

**Tabla 39**

*Caso de uso 29.*

<b>Prototipo: Sistema RRHH – Tendenzza</b>	
<b>Número Caso de Uso: 29</b>	<b>Nombre del Caso de Uso:</b> Gestionar Catálogos del Sistema
<b>Fecha elaboración:</b>	13/12/2025
<b>Descripción Caso de Uso:</b>	Permite realizar la gestión completa de los catálogos del sistema, incluyendo la creación, modificación, consulta y eliminación de registros en los diferentes

	catálogos como países, géneros, tipos de permiso, tipos de incapacidad, estados y demás datos de referencia.
<b>Autor caso de uso:</b>	Miguel
<b>Actores relacionados:</b>	Administrador
<b>Precondiciones:</b>	- El usuario debe estar autenticado con el rol Administrador.
<b>Flujo Básico del caso de uso</b>	
Este caso de uso comienza cuando el usuario desea gestionar los catálogos del sistema. El usuario accede al módulo de mantenimientos, selecciona el catálogo a gestionar y puede crear, modificar o eliminar registros según sea necesario.	
<b>Detallar el paso a paso del Flujo Básico</b>	
<ol style="list-style-type: none"> <li>1- El usuario accede al módulo de Mantenimientos desde el menú principal.</li> <li>2- El sistema muestra la lista de catálogos disponibles (países, provincias, cantones, distritos géneros, tipos de identificación, ocupaciones, tipos de permiso, tipos de incapacidad, estados de solicitud, estados de asistencia, estados de evaluación, motivos de salida, entre otros).</li> <li>3- El usuario selecciona el catálogo que desea gestionar.</li> <li>4- El sistema muestra la lista de registros del catálogo seleccionado.</li> <li>5- El usuario selecciona la acción a realizar. <ul style="list-style-type: none"> <li>• Si selecciona 'Agregar', se ejecuta el Sub-Flujo 1.</li> <li>• Si selecciona 'Editar', se ejecuta el Sub-Flujo 2.</li> <li>• Si selecciona 'Eliminar', se ejecuta el Sub-Flujo 3.</li> </ul> </li> <li>6- El sistema actualiza la lista del catálogo.</li> <li>7- El flujo finaliza.</li> </ol>	
<b>Sub-Flujos</b>	
<b>Sub-Flujo 1: Agregar registro</b>	<ol style="list-style-type: none"> <li>1- El usuario selecciona 'Agregar'.</li> <li>2- El sistema muestra el formulario del registro.</li> <li>3- El usuario ingresa el nombre o descripción del registro.</li> <li>4- El usuario ingresa los datos adicionales según el catálogo (código, abreviatura, etc.)</li> <li>5- El usuario selecciona el estado (activo/inactivo).</li> <li>6- El usuario selecciona 'Guardar'.</li> <li>7- El sistema valida que el nombre no esté duplicado.</li> <li>8- El sistema guarda el registro.</li> <li>9- El sistema muestra 'Registro agregado correctamente'.</li> </ol>
<b>Sub-Flujo 2: Editar registro</b>	<ol style="list-style-type: none"> <li>1- El usuario selecciona un registro de la lista.</li> <li>2- El usuario selecciona 'Editar'.</li> <li>3- El sistema muestra el formulario con los datos actuales.</li> <li>4- El usuario modifica los datos necesarios.</li> <li>5- El usuario selecciona 'Guardar cambios'.</li> <li>6- El sistema actualiza el registro.</li> <li>7- El sistema muestra 'Registro actualizado correctamente'.</li> </ol>
<b>Sub-Flujo 3: Eliminar registro</b>	<ol style="list-style-type: none"> <li>1- El usuario selecciona un registro de la lista.</li> <li>2- El usuario selecciona 'Eliminar'.</li> <li>3- El sistema verifica que el registro no esté siendo utilizado.</li> <li>4- El sistema solicita confirmación.</li> <li>5- El usuario confirma la eliminación.</li> <li>6- El sistema elimina el registro.</li> <li>7- El sistema muestra 'Registro eliminado correctamente'.</li> </ol>
<b>Flujos Alternos</b>	
Corresponde a lo que debe realizar el sistema ante posibles errores	
<b>Flujo Alternativo 1: Nombre duplicado</b>	<ol style="list-style-type: none"> <li>1- El sistema detecta que el nombre ya existe en el catálogo.</li> <li>2- El sistema muestra el mensaje 'Ya existe un registro con ese nombre'.</li> <li>3- El flujo regresa al paso correspondiente del Sub-Flujo.</li> </ol>

<b>Flujo Alterno 2: Registro en uso</b>	<ol style="list-style-type: none"> <li>1- El sistema detecta que el registro está siendo utilizado por otros módulos.</li> <li>2- El sistema muestra el mensaje 'No se puede eliminar, el registro está en uso'.</li> <li>3- El flujo finaliza sin eliminar.</li> </ol>
<b>Requerimientos especiales</b>	
<ol style="list-style-type: none"> <li>1- Solo el rol Administrador puede gestionar catálogos.</li> <li>2- El nombre de cada registro debe ser único dentro de su catálogo.</li> <li>3- No se puede eliminar un registro que esté siendo utilizado por otros módulos.</li> <li>4- Los catálogos geográficos (provincia, cantón, distrito) tienen dependencia jerárquica.</li> </ol>	
<b>Postcondiciones</b>	
<ol style="list-style-type: none"> <li>1- El catálogo queda actualizado en la base de datos.</li> <li>2- Los cambios se reflejan en los formularios que utilizan el catálogo.</li> <li>3- Los registros eliminados ya no aparecen en las listas de selección.</li> </ol>	

**Tabla 40***Caso de uso 30.*

<b>Prototipo: Sistema RRHH – Tendenzza</b>	
<b>Número Caso de Uso: 30</b>	<b>Nombre del Caso de Uso:</b> Gestionar aprobación de solicitudes.
<b>Fecha elaboración:</b>	13/12/2025
<b>Descripción Caso de Uso:</b>	Permite a la jefatura inmediata revisar, aprobar o rechazar las solicitudes realizadas por los colaboradores a su cargo, incluyendo solicitudes de vacaciones, permisos y horas extra.
<b>Autor caso de uso:</b>	Miguel
<b>Actores relacionados:</b>	Administrador, jefe RRHH, jefe Operaciones, jefe Ventas
<b>Precondiciones:</b>	<ul style="list-style-type: none"> <li>- El usuario debe estar autenticado con el rol de jefatura o superior.</li> <li>- Debe existir al menos una solicitud en estado pendiente.</li> <li>- El colaborador solicitante debe estar bajo la supervisión del usuario.</li> </ul>
<b>Flujo Básico del caso de uso</b>	
Este caso de uso comienza cuando el usuario desea gestionar las solicitudes pendientes de los colaboradores a su cargo. El usuario accede al módulo correspondiente, selecciona el tipo de solicitud, revisa el detalle y decide aprobarla o rechazarla.	
<b>Detallar el paso a paso del Flujo Básico</b>	
<ol style="list-style-type: none"> <li>1- El usuario accede al módulo de solicitudes pendientes desde el menú principal.</li> <li>2- El sistema muestra los tipos de solicitudes disponibles (vacaciones, permisos, horas extra).</li> <li>3- El usuario selecciona el tipo de solicitud que desea gestionar. <ul style="list-style-type: none"> <li>• Si selecciona 'Vacaciones', se ejecuta el Sub-Flujo 1.</li> <li>• Si selecciona 'Permisos', se ejecuta el Sub-Flujo 2.</li> <li>• Si selecciona 'Horas extra', se ejecuta el Sub-Flujo 3.</li> </ul> </li> <li>4- El sistema muestra la lista de solicitudes pendientes del tipo seleccionado. <ul style="list-style-type: none"> <li>• Si no existen solicitudes pendientes, se ejecuta el Flujo Alterno 1.</li> </ul> </li> <li>5- El usuario selecciona una solicitud para revisar.</li> <li>6- El sistema muestra el detalle el detalle completo de solicitud.</li> <li>7- El usuario revisa la información de la solicitud.</li> <li>8- El usuario selecciona 'Aprobar o Rechazar'. <ul style="list-style-type: none"> <li>• Si selecciona 'Aprobar', se ejecuta el Sub-Flujo 4.</li> <li>• Si selecciona 'Rechazar', se ejecuta el Sub-Flujo 5.</li> </ul> </li> <li>9- El sistema actualiza el estado de la solicitud en la base de datos.</li> </ol>	

<p>10- El sistema muestra el mensaje de confirmación. 11- El sistema actualiza la lista de solicitudes pendientes.</p>	
<b>Sub-Flujos</b>	
<b>Sub-Flujo 1: Solicitudes de vacaciones</b>	<ol style="list-style-type: none"> <li>1- El sistema muestra la lista de solicitudes de vacaciones pendientes.</li> <li>2- Al seleccionar una solicitud, el sistema muestra: colaborador, tipo de permisos, fechas solicitadas, justificación y documentos adjuntos.</li> <li>3- Al aprobar, el sistema registra el tipo de goce salarial.</li> </ol>
<b>Sub-Flujo 2: Solicitudes de permisos</b>	<ol style="list-style-type: none"> <li>1- El sistema muestra la lista de solicitudes de permisos pendientes.</li> <li>2- Al seleccionar una solicitud, el sistema muestra: colaborador, tipo de permiso, fechas solicitadas, justificación, documentos adjuntos.</li> <li>3- Al aprobar, el usuario debe indicar si el permiso es con goce o sin goce salarial.</li> </ol>
<b>Sub-Flujo 3: Solicitud de horas extra</b>	<ol style="list-style-type: none"> <li>1- El sistema muestra la lista de solicitudes de horas extra pendientes.</li> <li>2- Al seleccionar una solicitud, el sistema muestra: colaborador, fecha, cantidad de horas, motivo, salario base de colaborador.</li> <li>3- Al aprobar, el sistema calcula automáticamente el monto a pagar.</li> </ol>
<b>Sub-Flujo 4: Aprobar solicitud</b>	<ol style="list-style-type: none"> <li>1- El usuario selecciona 'Aprobar'.</li> <li>2- El sistema solicita confirmación.</li> <li>3- El usuario confirma la aprobación.</li> <li>4- El sistema cambia el estado de solicitud a 'Aprobado'.</li> <li>5- El sistema ejecuta las acciones correspondientes según el tipo: <ul style="list-style-type: none"> <li>• Vacaciones: descuenta los días del saldo del colaborador.</li> <li>• Permisos: registra el tipo de goce salarial.</li> <li>• Horas extra: calcula el monto a pagar según el salario.</li> </ul> </li> <li>6- El sistema muestra el mensaje 'Solicitud aprobada correctamente'.</li> </ol>
<b>Sub-Flujo 5: Rechazar solicitud</b>	<ol style="list-style-type: none"> <li>1- El usuario selecciona 'Rechazar'.</li> <li>2- El sistema solicita el motivo del rechazo. Si el motivo está vacío, se ejecuta el Flujo Alterno 2.</li> <li>3- El usuario ingresa el motivo y confirma el rechazo.</li> <li>4- El sistema cambia el estado de la solicitud a 'Rechazado'.</li> <li>5- El sistema muestra el mensaje 'Solicitud rechazada correctamente'.</li> </ol>
<b>Flujos Alternos</b>	
Corresponde a lo que debe realizar el sistema ante posibles errores	
<b>Flujo Alterno 1: Sin solicitudes pendientes</b>	<ol style="list-style-type: none"> <li>1- El sistema detecta que no existen solicitudes pendientes del tipo seleccionado.</li> <li>2- El sistema muestra el mensaje 'No hay solicitudes pendientes de aprobación'.</li> <li>3- El flujo regresa al paso 2 del Flujo Básico.</li> </ol>
<b>Flujo Alterno 2: Motivo de rechazo vacío</b>	<ol style="list-style-type: none"> <li>1- El sistema detecta que el motivo de rechazo está vacío.</li> <li>2- El sistema muestra el mensaje 'Debe ingresar el motivo del rechazo'.</li> <li>3- El flujo regresa al paso correspondiente del Sub-Flujo 5.</li> </ol>
<b>Flujo Alterno 3: Saldo insuficiente (vacaciones)</b>	<ol style="list-style-type: none"> <li>1- El sistema detecta que el colaborador no tiene saldo suficiente de vacaciones.</li> <li>2- El sistema muestra el mensaje 'El colaborador no tiene saldo suficiente para aprobar esta solicitud'.</li> </ol>

	3- El flujo regresa al paso 7 del Flujo Básico.
<b>Requerimientos especiales</b>	
1- Solo las jefaturas pueden aprobar o rechazar solicitudes de sus colaboradores. 2- El motivo de rechazo es obligatorio al rechaza una solicitud. 3- Para vacaciones, el sistema debe validar el saldo disponible antes de aprobar. 4- Para horas extra, el sistema debe calcular automáticamente el monto a pagar. 5- Las solicitudes aprobadas deben reflejarse en el cálculo de nómina. 6- El sistema debe mantener un historial de todas las aprobaciones y rechazos.	
<b>Postcondiciones</b>	
1- El estado de la solicitud queda actualizado en la base de datos. 2- El colaborador es notificado del resultado. 3- Los registros eliminados ya no aparecen en las listas de selección. <ul style="list-style-type: none"> <li>• Vacaciones: el saldo se actualiza automáticamente.</li> <li>• Permisos: queda registrado para consideración en nómina.</li> <li>• Horas extra: queda disponible para el cálculo de nómina.</li> </ul> 4- El historial de solicitudes del colaborador se actualiza.	

**Tabla 41***Caso de uso 31.*

<b>Prototipo: Sistema RRHH – Tendenzza</b>	
<b>Número Caso de Uso: 31</b>	<b>Nombre del Caso de Uso:</b> Gestionar Ubicaciones Autorizadas
<b>Fecha elaboración:</b>	13/12/2025
<b>Descripción Caso de Uso:</b>	Permite gestionar las ubicaciones geográficas autorizadas desde las cuales los colaboradores pueden registrar su asistencia mediante GPS. Incluye la creación, edición, visualización y eliminación de ubicaciones con su radio de validación y período de vigencia.
<b>Autor caso de uso:</b>	Miguel
<b>Actores relacionados:</b>	Administrador, jefe RRHH, jefe Operaciones
<b>Precondiciones:</b>	<ul style="list-style-type: none"> <li>- El usuario debe estar autenticado con el rol de Administrador o jefe RRHH.</li> <li>- El usuario debe tener el permiso 'Gestionar Ubicaciones' asignado.</li> </ul>
<b>Flujo Básico del caso de uso</b>	
Este caso de uso comienza cuando el usuario desea administrar las ubicaciones autorizadas del sistema. El usuario accede al módulo, selecciona la acción a realizar (crear, editar o eliminar) y el sistema ejecuta la operación correspondiente.	
<b>Detallar el paso a paso del Flujo Básico</b>	

<ol style="list-style-type: none"> <li>1- El usuario accede al módulo de Ubicaciones Autorizadas desde el menú principal.</li> <li>2- El sistema muestra la lista de ubicaciones registradas con su estado y vigencia.</li> <li>3- El usuario selecciona la acción a realizar. <ul style="list-style-type: none"> <li>• Si selecciona 'Agregar', se ejecuta el Sub-Flujo 1.</li> <li>• Si selecciona 'Editar', se ejecuta el Sub-Flujo 2.</li> <li>• Si selecciona 'Eliminar', se ejecuta el Sub-Flujo 3.</li> <li>• Si selecciona 'Ver detalle', se ejecuta el Sub-Flujo 4.</li> </ul> </li> <li>4- El sistema ejecuta la acción seleccionada.</li> <li>5- El sistema actualiza la lista de ubicaciones.</li> <li>6- El flujo finaliza.</li> </ol>	
<b>Sub-Flujos</b>	
<b>Sub-Flujo 1: Crear ubicación</b>	<ol style="list-style-type: none"> <li>1- El usuario selecciona 'Agregar ubicación'.</li> <li>2- El sistema muestra el formulario de registro con mapa interactivo.</li> <li>3- El usuario ingresa el nombre descriptivo de la ubicación.</li> <li>4- El usuario selecciona las coordenadas en el mapa o las ingresa manualmente (latitud, longitud).</li> <li>5- El usuario define el radio en metros permitido para la validación GPS.</li> <li>6- El usuario define las fechas de vigencia (inicio y fin).</li> <li>7- El usuario indica si la ubicación requiere validación GPS obligatoria.</li> <li>8- El usuario selecciona 'Guardar'. Si los datos no son válidos, se ejecuta el Flujo Alternativo 1.</li> <li>9- El sistema guarda la ubicación con estado activo y muestra 'Ubicación registrada correctamente'.</li> </ol>
<b>Sub-Flujo 2: Editar ubicación</b>	<ol style="list-style-type: none"> <li>1- El usuario selecciona una ubicación de la lista y elige 'Editar'.</li> <li>2- El sistema muestra el formulario con los datos actuales prellenados.</li> <li>3- El usuario modifica los campos necesarios.</li> <li>4- El usuario selecciona 'Guardar cambios'. Si los datos no son válidos, se ejecuta el Flujo Alternativo 1.</li> <li>5- El sistema actualiza la ubicación y muestra 'Ubicación actualizada correctamente'.</li> </ol>
<b>Sub-Flujo 3: Eliminar ubicación</b>	<ol style="list-style-type: none"> <li>1- El usuario selecciona una ubicación y elige 'Eliminar'.</li> <li>2- El sistema muestra mensaje de confirmación.</li> <li>3- El usuario confirma la eliminación.</li> <li>4- El sistema verifica si la ubicación tiene asistencias asociadas.</li> <li>5- Si tiene asistencias, el sistema la inactiva en lugar de eliminarla físicamente.</li> <li>6- El sistema muestra 'Ubicación eliminada correctamente'.</li> </ol>
<b>Sub-Flujo 4: Ver detalle</b>	<ol style="list-style-type: none"> <li>1- El usuario selecciona una ubicación y elige 'Ver detalle'.</li> <li>2- El sistema muestra los datos completos: nombre, coordenadas, radio, vigencia, cantidad de asistencias registradas.</li> <li>3- El usuario revisa la información y cierra la vista.</li> </ol>
<b>Flujos Alternos</b>	
Corresponde a lo que debe realizar el sistema ante posibles errores	
<b>Flujo Alternativo 1: Datos inválidos</b>	<ol style="list-style-type: none"> <li>1- El sistema detecta campos obligatorios vacíos o con formato incorrecto.</li> <li>2- El sistema resalta los campos que requieren corrección.</li> <li>3- El flujo regresa al paso correspondiente del Sub-Flujo activo.</li> </ol>
<b>Flujo Alternativo 2: Lista vacía</b>	<ol style="list-style-type: none"> <li>1- El sistema detecta que no existen ubicaciones registradas.</li> <li>2- El sistema muestra el mensaje 'No hay ubicaciones registradas'.</li> <li>3- El sistema ofrece la opción de crear la primera ubicación.</li> </ol>
<b>Requerimientos especiales</b>	

<ol style="list-style-type: none"> <li>1- Solo los roles Administrador y jefe RRHH pueden gestionar ubicaciones.</li> <li>2- Las coordenadas deben ser valores numéricos válidos (latitud entre -90 y 90, longitud entre -180 y 180).</li> <li>3- El radio mínimo permitido es de 50 metros.</li> <li>4- No se puede eliminar físicamente una ubicación con asistencias asociadas; solo se puede inactivar.</li> <li>5- La fecha de fin de vigencia debe ser posterior a la fecha de inicio.</li> </ol>
<b>Postcondiciones</b>
<ol style="list-style-type: none"> <li>1- La ubicación queda registrada o actualizada en la base de datos.</li> <li>2- Los cambios se reflejan inmediatamente en el módulo de asistencia.</li> <li>3- Las ubicaciones inactivas no aparecen disponibles para el registro de asistencia.</li> </ol>

**Tabla 42***Caso de uso 32.*

<b>Prototipo: Sistema RRHH – Tendenza</b>	
<b>Número Caso de Uso: 32</b>	<b>Nombre del Caso de Uso:</b> Consultar Nómina
<b>Fecha elaboración:</b>	13/12/2025
<b>Descripción Caso de Uso:</b>	Permite consultar y visualizar las nóminas procesadas en el sistema. El administrador y jefe RRHH pueden ver todas las nóminas, mientras que el colaborador solo puede ver sus propias boletas de pago.
<b>Autor caso de uso:</b>	Miguel
<b>Actores relacionados:</b>	Administrador, jefe RRHH, Colaborador
<b>Precondiciones:</b>	<ul style="list-style-type: none"> <li>- El usuario debe estar autenticado en el sistema.</li> <li>- Deben existir nóminas procesadas en la base de datos.</li> </ul>
<b>Flujo Básico del caso de uso</b>	
Este caso de uso comienza cuando el usuario desea consultar las nóminas registradas. El usuario accede al módulo de nóminas y el sistema muestra la lista disponible según su rol, permitiendo filtrar por período y visualizar el detalle de cada boleta.	
<b>Detallar el paso a paso del Flujo Básico</b>	
<ol style="list-style-type: none"> <li>1- El usuario accede al módulo de Nóminas desde el menú principal.</li> <li>2- El sistema muestra la lista de nóminas según el rol del usuario: <ul style="list-style-type: none"> <li>• Administrador / jefe RRHH: ve todas las nóminas de todos los colaboradores.</li> <li>• Colaborador: ve únicamente sus propias boletas de pago.</li> <li>• Si no existen nóminas registradas, se ejecuta el Flujo Alterno 2.</li> </ul> </li> <li>3- El usuario puede filtrar las nóminas por mes y año.</li> <li>4- El sistema filtra y actualiza la lista con los resultados correspondientes. <ul style="list-style-type: none"> <li>• Si la búsqueda no devuelve resultados, se ejecuta el Flujo Alterno 1.</li> </ul> </li> <li>5- El usuario selecciona una nómina para ver el detalle.</li> <li>6- El sistema verifica el permiso de acceso: <ul style="list-style-type: none"> <li>• Si el colaborador intenta ver una nómina que no le pertenece, se ejecuta el Flujo Alterno 3.</li> </ul> </li> <li>7- El sistema muestra el detalle completo de la boleta: colaborador, período, salario base, horas extra, deducciones CCSS, impuesto de renta y salario neto.</li> <li>8- El usuario revisa la información.</li> <li>9- El usuario cierra el detalle.</li> <li>10- El sistema regresa a la lista de nóminas.</li> </ol>	
<b>Sub-Flujos</b>	
<b>Sub-Flujo 1: Filtrar por período</b>	<ol style="list-style-type: none"> <li>1- El usuario selecciona el mes y año deseados.</li> <li>2- El sistema filtra y muestra únicamente las nóminas del período seleccionado.</li> <li>3- El sistema muestra el total de colaboradores y el monto total de la planilla del período.</li> </ol>
<b>Flujos Alternos</b>	
Corresponde a lo que debe realizar el sistema ante posibles errores	

<b>Flujo Alternativo 1: Sin resultados</b>	<ol style="list-style-type: none"> <li>1- El sistema no encuentra nóminas que coincidan con los filtros seleccionados.</li> <li>2- El sistema muestra el mensaje 'No se encontraron nóminas para el período seleccionado'.</li> <li>3- El flujo regresa al paso 3 del Flujo Básico.</li> </ol>
<b>Flujo Alternativo 2: Lista vacía</b>	<ol style="list-style-type: none"> <li>1- El sistema detecta que no existen nóminas procesadas.</li> <li>2- El sistema muestra el mensaje 'No hay nóminas procesadas'.</li> <li>3- El flujo termina.</li> </ol>
<b>Flujo Alternativo 3: Acceso no autorizado</b>	<ol style="list-style-type: none"> <li>1- El sistema detecta que el colaborador intenta acceder a una nómina que no le pertenece.</li> <li>2- El sistema muestra el mensaje 'No tiene permiso para ver esta boleta'.</li> <li>3- El flujo regresa al paso 2 del Flujo Básico.</li> </ol>
<b>Requerimientos especiales</b>	
<ol style="list-style-type: none"> <li>1- El sistema debe filtrar automáticamente las nóminas según el rol del usuario.</li> <li>2- Los colaboradores solo pueden visualizar sus propias boletas de pago.</li> <li>3- La tabla debe permitir ordenar por cualquier columna.</li> <li>4- El sistema debe mostrar el estado de la nómina (procesada, pagada).</li> <li>5- La búsqueda y filtros deben actualizarse en tiempo real.</li> </ol>	
<b>Postcondiciones</b>	
<ol style="list-style-type: none"> <li>1- El usuario visualiza la información solicitada según su rol.</li> <li>2- No se realizan cambios en la base de datos.</li> </ol>	

**Tabla 43***Caso de uso 33.*

<b>Prototipo: Sistema RRHH – Tendenzza</b>	
<b>Número Caso de Uso: 33</b>	<b>Nombre del Caso de Uso:</b> Consultar Evaluaciones de Desempeño
<b>Fecha elaboración:</b>	13/12/2025
<b>Descripción Caso de Uso:</b>	Permite consultar y visualizar las evaluaciones de desempeño registradas en el sistema, utilizando opciones de búsqueda, filtrado por estado o período, y visualización del detalle de cada evaluación.
<b>Autor caso de uso:</b>	Miguel
<b>Actores relacionados:</b>	Administrador, jefe RRHH, jefe Operaciones, jefe Ventas
<b>Precondiciones:</b>	<ul style="list-style-type: none"> <li>- El usuario debe estar autenticado con el rol autorizado.</li> <li>- Deben existir registros de evaluaciones de desempeño en la base de datos.</li> </ul>
<b>Flujo Básico del caso de uso</b>	
Este caso de uso comienza cuando el usuario desea consultar las evaluaciones de desempeño registradas. El usuario accede al módulo y el sistema muestra la lista disponible, permitiendo búsquedas, filtros y la visualización del detalle de cada evaluación.	
<b>Detallar el paso a paso del Flujo Básico</b>	

<ol style="list-style-type: none"> <li>1- El usuario accede al módulo de Evaluación de Desempeño desde el menú principal.</li> <li>2- El sistema muestra la lista de evaluaciones registradas en una tabla. <ul style="list-style-type: none"> <li>• Si no existen evaluaciones, se ejecuta el Flujo Alterno 2.</li> </ul> </li> <li>3- El usuario puede buscar una evaluación por nombre del colaborador.</li> <li>4- El sistema filtra y muestra los resultados de la búsqueda. <ul style="list-style-type: none"> <li>• Si la búsqueda no devuelve resultados, se ejecuta el Flujo Alterno 1.</li> </ul> </li> <li>5- El usuario puede filtrar las evaluaciones por estado. <ul style="list-style-type: none"> <li>• Si el usuario selecciona filtrar por estado, se ejecuta el Sub-Flujo 1.</li> </ul> </li> <li>6- El usuario puede filtrar las evaluaciones por período. <ul style="list-style-type: none"> <li>• Si el usuario selecciona filtrar por período, se ejecuta el Sub-Flujo 2.</li> </ul> </li> <li>7- El usuario selecciona una evaluación para ver más detalles.</li> <li>8- El sistema muestra la información completa: colaborador, período, calificación general, criterios evaluados, observaciones, estado y bonificación asignada si corresponde.</li> <li>9- El usuario revisa la información.</li> <li>10- El usuario cierra la vista de detalles.</li> <li>11- El sistema regresa a la lista de evaluaciones.</li> </ol>	
<b>Sub-Flujos</b>	
<b>Sub-Flujo 1: Filtrar por estado</b>	<ol style="list-style-type: none"> <li>1- El usuario selecciona un estado (Pendiente, Completada).</li> <li>2- El sistema filtra y muestra solo las evaluaciones con ese estado.</li> </ol>
<b>Sub-Flujo 1: Filtrar por periodo</b>	<ol style="list-style-type: none"> <li>1- El usuario selecciona un rango de fechas.</li> <li>2- El sistema muestra las evaluaciones dentro del período seleccionado.</li> </ol>
<b>Flujos Alternos</b>	
Corresponde a lo que debe realizar el sistema ante posibles errores	
<b>Flujo Alterno 1: Sin resultados</b>	<ol style="list-style-type: none"> <li>1- El sistema no encuentra evaluaciones que coincidan con los criterios de búsqueda.</li> <li>2- El sistema muestra el mensaje 'No se encontraron resultados'.</li> <li>3- El flujo regresa al paso 3 del Flujo Básico.</li> </ol>
<b>Flujo Alterno 2: Lista vacía</b>	<ol style="list-style-type: none"> <li>1- El sistema detecta que no existen evaluaciones registradas.</li> <li>2- El sistema muestra el mensaje 'No hay evaluaciones registradas'.</li> <li>3- El flujo termina.</li> </ol>
<b>Requerimientos especiales</b>	
<ol style="list-style-type: none"> <li>1- La tabla debe permitir ordenar por cualquier columna.</li> <li>2- El sistema debe mostrar la calificación general con un indicador visual (color según rango).</li> <li>3- La búsqueda debe realizarse en tiempo real mientras el usuario escribe.</li> <li>4- Solo los roles autorizados pueden visualizar evaluaciones de desempeño.</li> <li>5- El sistema debe mostrar si la evaluación tiene bonificación asignada.</li> </ol>	
<b>Postcondiciones</b>	
<ol style="list-style-type: none"> <li>1- El usuario visualiza la información solicitada.</li> <li>2- No se realizan cambios en la base de datos.</li> </ol>	

**Tabla 44***Caso de uso 34.*

<b>Prototipo: Sistema RRHH – Tendenzza</b>	
<b>Número Caso de Uso: 34</b>	<b>Nombre del Caso de Uso:</b> Enviar boleta de pago por correo electrónico
<b>Fecha elaboración:</b>	13/12/2025
<b>Descripción Caso de Uso:</b>	Permite enviar la boleta de pago de una o todas las nóminas de un período determinado al correo electrónico registrado de cada colaborador, adjuntando el documento en formato PDF.
<b>Autor caso de uso:</b>	Miguel

<b>Actores relacionados:</b>	Administrador, jefe RRHH
<b>Precondiciones:</b>	<ul style="list-style-type: none"> <li>- El usuario debe estar autenticado con el rol de Administrador o jefe RRHH.</li> <li>- Deben existir nóminas procesadas en la base de datos.</li> <li>- Los colaboradores deben tener un correo electrónico registrado en el sistema.</li> <li>- El servidor de correo debe estar configurado correctamente.</li> </ul>
<b>Flujo Básico del caso de uso</b>	
Este caso de uso comienza cuando el usuario desea enviar las boletas de pago por correo. El usuario accede al módulo de nóminas, selecciona la nómina o el período correspondiente y el sistema genera el PDF y lo envía al correo del colaborador.	
<b>Detallar el paso a paso del Flujo Básico</b>	
<ol style="list-style-type: none"> <li>1- El usuario accede al módulo de Nóminas desde el menú principal.</li> <li>2- El usuario filtra las nóminas por el período deseado.</li> <li>3- El usuario selecciona la acción de envío: <ul style="list-style-type: none"> <li>• Si desea enviar una boleta individual, se ejecuta el Sub-Flujo 1.</li> <li>• Si desea enviar todas las boletas del período, se ejecuta el Sub-Flujo 2.</li> </ul> </li> <li>4- El sistema procesa el envío según la opción seleccionada.</li> <li>5- El sistema registra la fecha de envío y marca la nómina como 'Correo enviado'.</li> <li>6- El sistema muestra la confirmación del envío.</li> <li>7- El flujo finaliza.</li> </ol>	
<b>Sub-Flujos</b>	
<b>Sub-Flujo 1: Enviar boleta individual</b>	<ol style="list-style-type: none"> <li>1- El usuario selecciona una nómina específica.</li> <li>2- El usuario selecciona la opción 'Enviar correo'.</li> <li>3- El sistema genera el PDF de la boleta con el desglose completo.</li> <li>4- El sistema envía el correo al colaborador con el PDF adjunto.</li> <li>5- El sistema actualiza el campo correo_enviado = true y registra la fecha de envío.</li> <li>6- El sistema muestra el mensaje 'Correo enviado correctamente'.</li> </ol>
<b>Sub-Flujo 2: Enviar todos los correos del periodo</b>	<ol style="list-style-type: none"> <li>1- El usuario selecciona la opción 'Enviar correos a todos' del período activo.</li> <li>2- El sistema consulta todas las nóminas del período que aún no han recibido correo.</li> <li>3- Por cada nómina pendiente, el sistema genera el PDF y envía el correo.</li> <li>4- El sistema registra el envío de cada boleta.</li> <li>5- El sistema muestra un resumen: 'Se enviaron X correos correctamente'.</li> </ol>
<b>Flujos Alternos</b>	
Corresponde a lo que debe realizar el sistema ante posibles errores	
<b>Flujo Alterno 1: Colaborador sin correo registrado</b>	<ol style="list-style-type: none"> <li>1- El sistema detecta que el colaborador no tiene correo electrónico registrado.</li> <li>2- El sistema omite ese colaborador y lo registra en un listado de errores.</li> <li>3- El flujo continúa con los demás colaboradores.</li> </ol>
<b>Flujo Alterno 2: Error en el envío del correo</b>	<ol style="list-style-type: none"> <li>1- El sistema detecta un error al enviar el correo (servidor no disponible, correo inválido).</li> <li>2- El sistema registra el error y no marca la nómina como enviada.</li> <li>3- El sistema muestra el mensaje 'Error al enviar correo al colaborador [nombre]. Intente nuevamente'.</li> <li>4- El flujo continúa con las demás boletas pendientes.</li> </ol>
<b>Flujo Alterno 3: Nómina ya enviada</b>	<ol style="list-style-type: none"> <li>1- El sistema detecta que la boleta ya fue enviada anteriormente.</li> <li>2- El sistema solicita confirmación para reenviar.</li> <li>3- Si el usuario confirma, el flujo continúa normalmente desde el Sub-Flujo correspondiente.</li> </ol>
<b>Requerimientos especiales</b>	

<ol style="list-style-type: none"> <li>1- Solo los roles Administrador y jefe RRHH pueden enviar boletas por correo.</li> <li>2- El PDF generado debe incluir el logo de la empresa y el desglose completo de la nómina.</li> <li>3- El sistema debe registrar la fecha y hora de cada envío.</li> <li>4- El sistema debe manejar errores de envío sin interrumpir el proceso de los demás colaboradores.</li> <li>5- El sistema debe mostrar un indicador visual en la lista de nóminas para las boletas ya enviadas.</li> </ol>
<b>Postcondiciones</b>
<ol style="list-style-type: none"> <li>1- Las boletas enviadas quedan marcadas con correo_enviado = true y la fecha de envío registrada.</li> <li>2- El colaborador recibe su boleta de pago en formato PDF en su correo electrónico.</li> <li>3- No se realizan cambios en los montos de la nómina.</li> </ol>

**Tabla 45***Caso de uso 35.*

<b>Prototipo: Sistema RRHH – Tendenza</b>	
<b>Número Caso de Uso: 35</b>	<b>Nombre del Caso de Uso:</b> Modificar y gestionar estado de usuario
<b>Fecha elaboración:</b>	13/12/2025
<b>Descripción Caso de Uso:</b>	Permite al administrador modificar los datos de un usuario existente, cambiar su estado (activo/bloqueado) y restablecer su contraseña en caso de que el usuario no pueda acceder al sistema.
<b>Autor caso de uso:</b>	Miguel
<b>Actores relacionados:</b>	Administrador
<b>Precondiciones:</b>	<ul style="list-style-type: none"> <li>- El usuario debe estar autenticado con el rol de Administrador.</li> <li>- El usuario por gestionar debe existir en la base de datos.</li> </ul>
<b>Flujo Básico del caso de uso</b>	
Este caso de uso comienza cuando el administrador desea gestionar un usuario existente. El administrador busca el usuario, selecciona la acción a realizar (editar, bloquear, desbloquear o restablecer contraseña) y el sistema ejecuta la operación.	
<ol style="list-style-type: none"> <li>1- Detallar el paso a paso del Flujo Básico</li> <li>2- El administrador accede al módulo de Usuarios desde el menú principal.</li> <li>3- El sistema muestra la lista de usuarios registrados.</li> <li>4- El administrador busca el usuario que desea gestionar. <ul style="list-style-type: none"> <li>• Si el usuario no es encontrado, se ejecuta el Flujo Alterno 1.</li> </ul> </li> <li>5- El administrador selecciona la acción a realizar: <ul style="list-style-type: none"> <li>• Si selecciona 'Editar', se ejecuta el Sub-Flujo 1.</li> <li>• Si selecciona 'Bloquear' o 'Desbloquear', se ejecuta el Sub-Flujo 2.</li> <li>• Si selecciona 'Restablecer contraseña', se ejecuta el Sub-Flujo 3.</li> </ul> </li> <li>6- El sistema ejecuta la acción seleccionada.</li> <li>7- El sistema muestra el mensaje de confirmación correspondiente.</li> <li>8- El sistema actualiza la lista de usuarios.</li> <li>9- El flujo finaliza.</li> </ol>	
<b>Sub-Flujos</b>	
<b>Sub-Flujo 1: Editar usuario</b>	<ol style="list-style-type: none"> <li>1- El administrador selecciona 'Editar'.</li> <li>2- El sistema muestra el formulario con los datos actuales del usuario.</li> <li>3- El administrador modifica los campos necesarios (nombre de usuario, nivel de acceso, permisos).</li> <li>4- El administrador selecciona 'Guardar cambios'. Si los datos no son válidos, se ejecuta el Flujo Alterno 2.</li> <li>5- El sistema actualiza el usuario y muestra 'Usuario actualizado correctamente'.</li> </ol>
<b>Sub-Flujo 2: Bloquear / Desbloquear usuario</b>	<ol style="list-style-type: none"> <li>1- El administrador selecciona 'Bloquear' si el usuario está activo, o 'Desbloquear' si está bloqueado.</li> <li>2- El sistema muestra mensaje de confirmación.</li> <li>3- El administrador confirma la acción.</li> </ol>

	<ul style="list-style-type: none"> <li>4- Si se bloquea: el sistema cambia Bloqueado = true. Si se desbloquea: el sistema cambia Bloqueado = false e Intentos_Fallidos = 0.</li> <li>5- El sistema muestra 'Estado del usuario actualizado correctamente'.</li> </ul>
<b>Sub-Flujo 3: Restablecer contraseña</b>	<ul style="list-style-type: none"> <li>1- El administrador selecciona 'Restablecer contraseña'.</li> <li>2- El sistema muestra un mensaje de confirmación indicando que se generará una nueva contraseña.</li> <li>3- El administrador confirma la operación.</li> <li>4- El sistema genera una nueva contraseña temporal y la almacena encriptada.</li> <li>5- El sistema muestra la nueva contraseña temporal al administrador para que la comunique al usuario.</li> <li>6- El sistema muestra 'Contraseña restablecida correctamente'.</li> </ul>
<b>Flujos Alternos</b>	
Corresponde a lo que debe realizar el sistema ante posibles errores	
<b>Flujo Alternativo 1: Usuario no encontrado</b>	<ul style="list-style-type: none"> <li>1- El sistema no encuentra usuarios que coincidan con la búsqueda.</li> <li>2- El sistema muestra el mensaje 'No se encontraron usuarios'.</li> <li>3- El flujo regresa al paso 3 del Flujo Básico.</li> </ul>
<b>Flujo Alternativo 2: Datos inválidos en edición</b>	<ul style="list-style-type: none"> <li>1- El sistema detecta campos obligatorios vacíos o con formato incorrecto.</li> <li>2- El sistema resalta los campos que requieren corrección.</li> <li>3- El flujo regresa al paso correspondiente del Sub-Flujo 1.</li> </ul>
<b>Flujo Alternativo 3: Intento de bloquear al propio usuario</b>	<ul style="list-style-type: none"> <li>1- El sistema detecta que el administrador intenta bloquear su propia cuenta.</li> <li>2- El sistema muestra el mensaje 'No puede bloquear su propia cuenta de usuario'.</li> <li>3- El flujo regresa al paso 3 del Flujo Básico.</li> </ul>
<b>Requerimientos especiales</b>	
<ul style="list-style-type: none"> <li>1- Solo el rol Administrador puede modificar y gestionar el estado de los usuarios.</li> <li>2- El administrador no puede bloquear su propia cuenta.</li> <li>3- Al desbloquear un usuario, el contador de intentos fallidos debe reiniciarse a cero.</li> <li>4- Las contraseñas restablecidas deben almacenarse encriptadas.</li> <li>5- Los cambios en permisos se aplican a partir de la próxima sesión del usuario.</li> </ul>	
<b>Postcondiciones</b>	
<ul style="list-style-type: none"> <li>1- Los datos del usuario quedan actualizados en la base de datos.</li> <li>2- Si se bloqueó, el usuario no puede iniciar sesión hasta que sea desbloqueado.</li> <li>3- Si se restableció la contraseña, el usuario puede iniciar sesión con la nueva contraseña temporal.</li> <li>4- Los cambios en permisos se reflejan en la próxima sesión del usuario.</li> </ul>	

## Arquitectura del Sistema.

El objetivo de la arquitectura del sistema es poder visualizar cómo se implementará el prototipo del Sistema RRHH – Tendenzza dentro de la infraestructura de recursos tecnológicos de la empresa Tendenzza & puertAlum. Esto abarca dónde estará localizado el sistema, cómo será accedido por los usuarios, cuáles recursos utiliza para su ejecución y qué servicios en la nube se requieren para su funcionamiento.

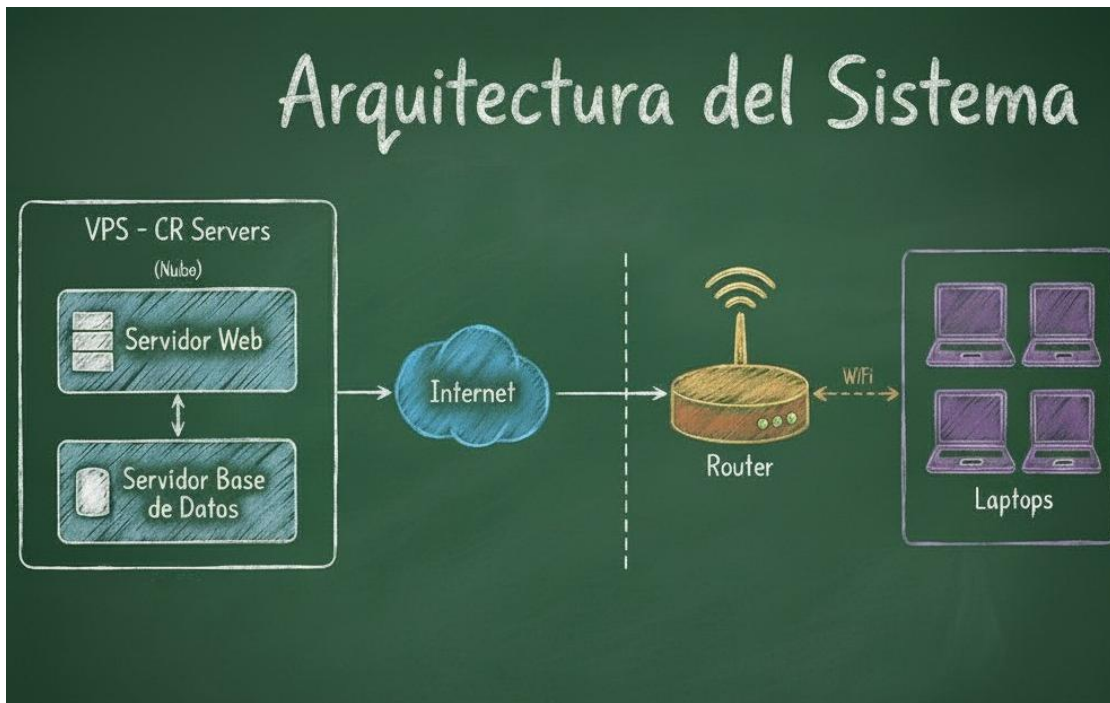
En la figura 2 se representa la arquitectura del sistema propuesta, donde la aplicación es alojada en un servidor privado virtual (VPS) contratado con el proveedor CR Servers,

ubicado en Costa Rica. Esta decisión responde a que la empresa no cuenta con un centro de tecnologías de información propio ni con infraestructura física para alojar servidores locales, por lo que el uso de servicios en la nube resulta la solución más adecuada al no requerir equipos adicionales ni personal técnico especializado para su mantenimiento.

El servidor VPS cuenta con las especificaciones básicas necesarias para ejecutar el sistema: memoria RAM suficiente, almacenamiento SSD, dirección IPv4 estática y un ancho de banda compartido de 150 Mbps. Dentro de este servidor se encuentran dos componentes principales: el servidor web, que ejecuta la aplicación desarrollada en Laravel y PHP, y el servidor de base de datos MySQL, que almacena toda la información del sistema de recursos humanos.

Los usuarios acceden al sistema a través de Internet mediante el dominio t&pcr.com, registrado con el proveedor Enom. La conexión desde las instalaciones de la empresa se realiza mediante un router WiFi al cual se conectan las computadoras portátiles de los colaboradores. Cuando los empleados realizan labores fuera de las instalaciones, utilizan dispositivos de conexión WiFi móvil proporcionados por la empresa, lo que garantiza el acceso remoto al sistema desde cualquier ubicación con cobertura de internet.

Esta arquitectura basada en la nube ofrece ventajas significativas como la disponibilidad del sistema las 24 horas de día, los 7 días de la semana, sin depender de la infraestructura física de la empresa. Además, permite escalabilidad futura en caso de que la organización requiera mayores recursos de procesamiento o almacenamiento.

**Figura 2***Arquitectura del Sistema*

Fuente: Elaboración Propia (2026)

**Arquitectura del Software**

La arquitectura del software del Sistema RRHH – Tendenzza & puertAlum se fundamenta en el patrón de diseño MVC (Modelo-Vista-Controlador), el cual permite una separación clara de responsabilidades entre las diferentes capas de la aplicación. Esta estructura facilita el mantenimiento, la escalabilidad y la correcta organización del código fuente, asegurando un desarrollo ordenado y alineado con las buenas prácticas de aplicaciones web modernas.

En la figura 3 se presenta la arquitectura del software del sistema donde se observa la división en capas correspondiente al patrón MVC y la interacción entre los componentes de modelo, vista y controlador que conforman la aplicación.

**Capa de Modelo.**

Es la responsable de la gestión de la información y de la interacción directa con la base de datos MySQL 8.4. En esta capa se definen las entidades principales del sistema, tales como usuarios, colaboradores, permisos, vacaciones, horas extra, nómina, incapacidades y aguinaldo, así como sus relaciones y restricciones.

El sistema de control de acceso y permisos se implementan de forma nativa, utilizando PHP 8.3 y el ORM Laravel Eloquent, sin depender de paquetes externos. Para ello se utilizan tablas diseñadas específicamente para el proyecto, permitiendo un control total del modelo relacional, y el cumplimiento de los requerimientos académicos. Los modelos incluyen métodos que permiten validar permisos, determinar niveles de acceso y controlar las operaciones permitidas para cada usuario.

### **Capa de Controlador.**

La capa del controlador gestiona la lógica de negocio del sistema y coordina la comunicación entre las vistas y los modelos. Los controladores reciben las solicitudes realizadas por los usuarios, validan los datos ingresados y verifican los permisos antes de ejecutar cualquier acción.

En esta capa se implementan funcionalidades como la gestión de colaboradores, aprobación o rechazo de permisos y vacaciones, cálculo de horas extra, generación de nómina y administración de usuarios. Los controladores utilizan métodos definidos en los modelos para validar el acceso a cada funcionalidad, garantizando que solo los usuarios autorizados puedan ejecutar determinadas acciones dentro del sistema.

### **Capa de Vista.**

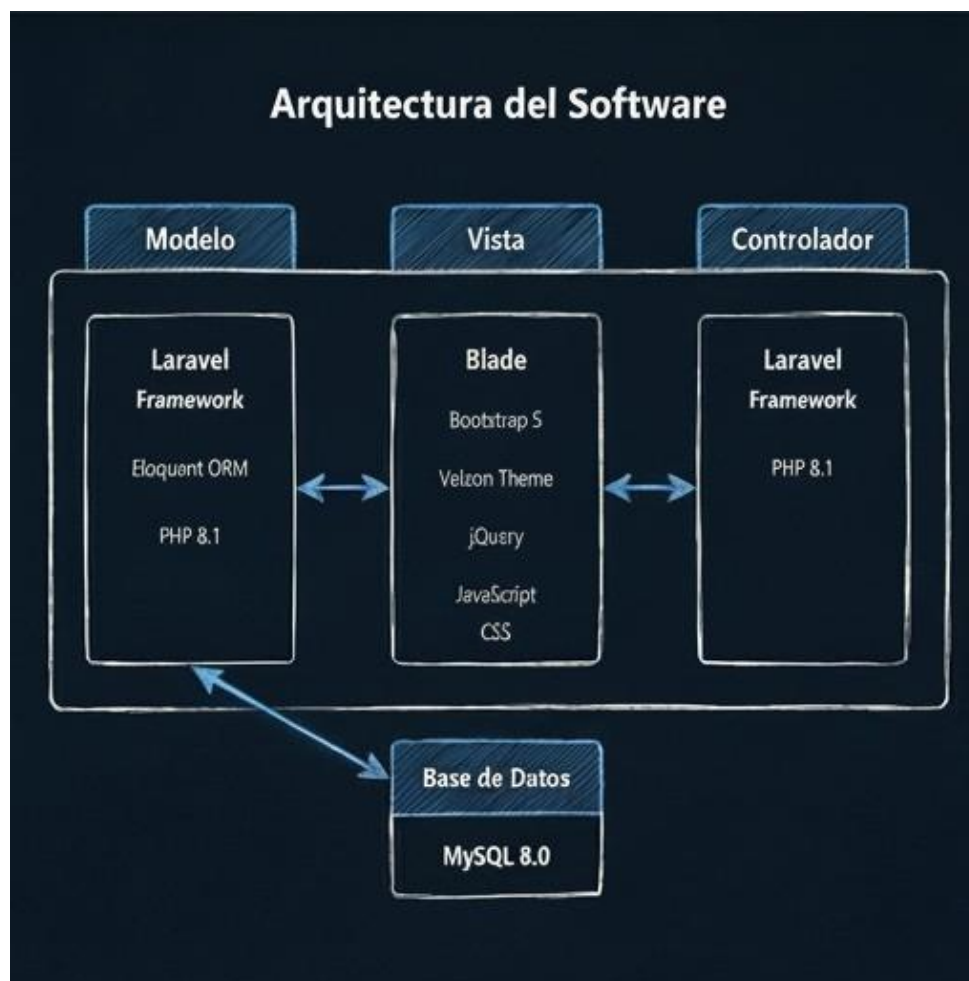
La capa de vista se encarga de la presentación de la información y de la interacción directa con el usuario. A través de esta capa se muestran formularios, listados, reportes y mensajes del sistema, proporcionando una interfaz clara, intuitiva y responsiva.

Esta capa se desarrolla utilizando el motor de plantillas Blade de Laravel adaptadas a lo que necesitamos, junto con HTML 5, CSS 3 y JavaScript. Para el diseño visual se emplea Bootstrap 5.3 y el tema administrativo Velzon, mientras que jQuery 3.7 permite mejorar la

experiencia del usuario mediante validaciones y comportamientos dinámicos. Además, la vista adapta los elementos visibles según los permisos del usuario, ocultando o mostrando opciones de acuerdo con su nivel de acceso.

**Figura 3**

*Arquitectura del Software.*



Fuente: Elaboración Propia (2026)

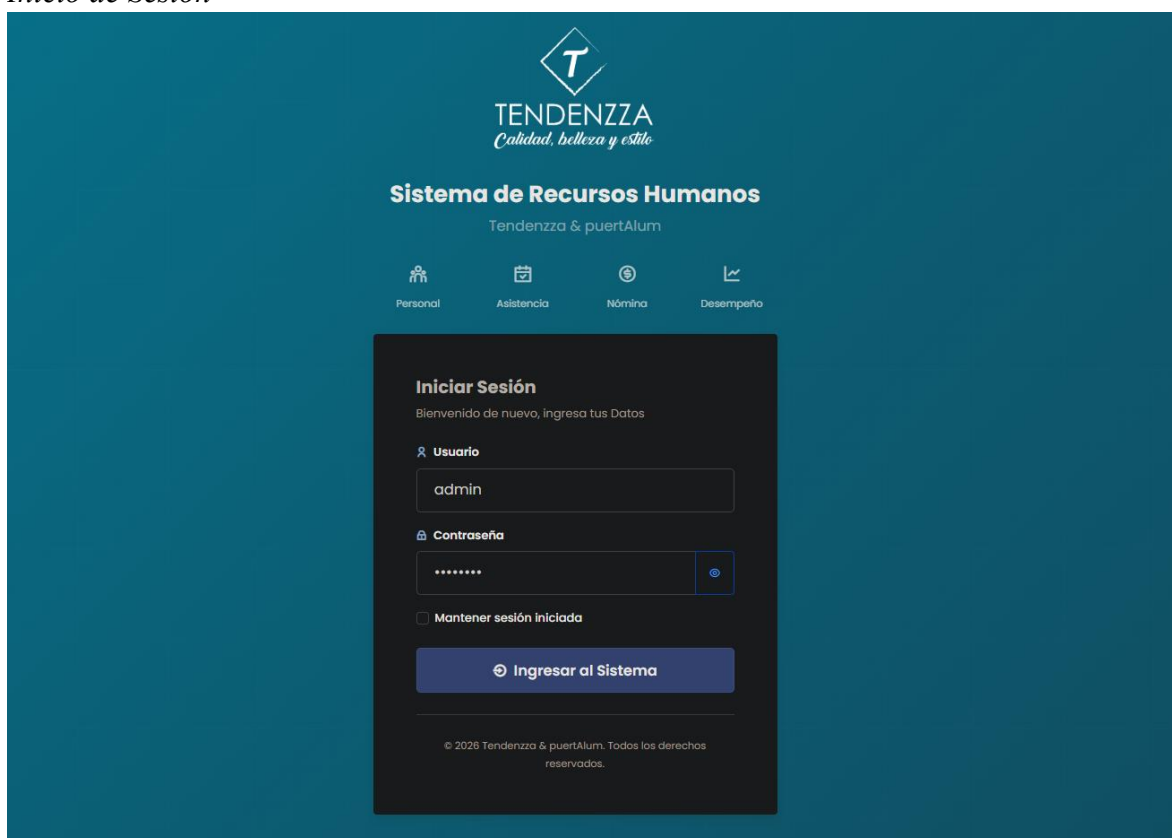
### Diseño de Interfaz

A continuación, se presentan los diseños de las interfaces de entrada del prototipo funcional del Sistema de Recursos Humanos para la empresa Tendenzza & puertAlum.

Estas interfaces permiten la interacción directa de los usuarios con el sistema para el registro, consulta y gestión de la información relacionada con los procesos de Recursos Humanos.

## Figura 4

### *Inicio de Sesión*



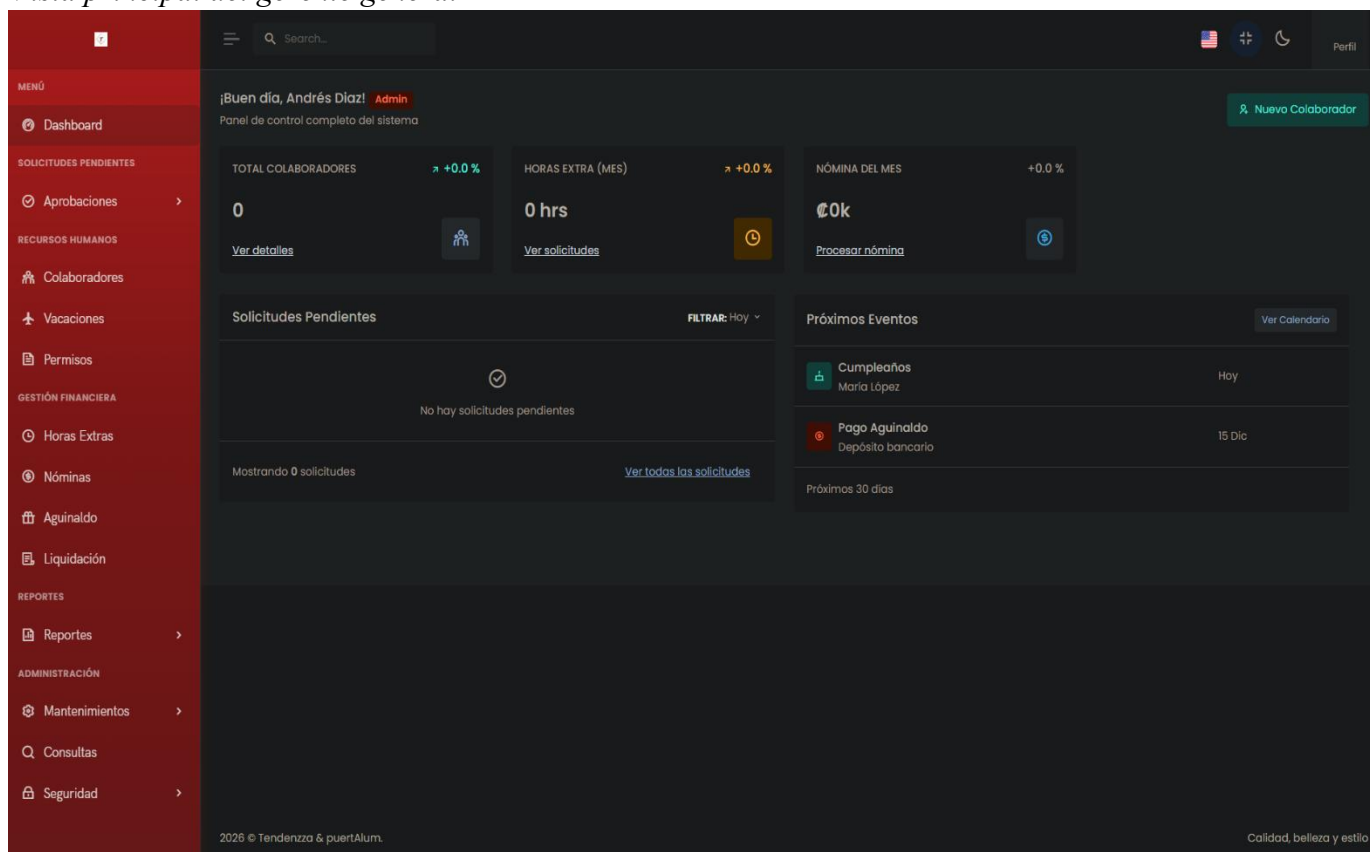
Fuente: Elaboración Propia (2026)

En la figura anterior se muestra la vista de inicio de sesión del Sistema RRHH – Tendenzza & puertAlum, la cual es presentada a todo el personal de la empresa al ingresar a la aplicación. En esta interfaz se visualizan los campos correspondientes al ID del colaborador y la contraseña, datos que deben ser ingresados por el usuario para acceder al sistema.

El usuario y la contraseña son asignados por el área de Recursos Humanos o por el administrador del sistema al momento de crear la cuenta del colaborador, garantizando así un acceso controlado y seguro a la plataforma.

## Figura 5

### Vista principal del gerente general

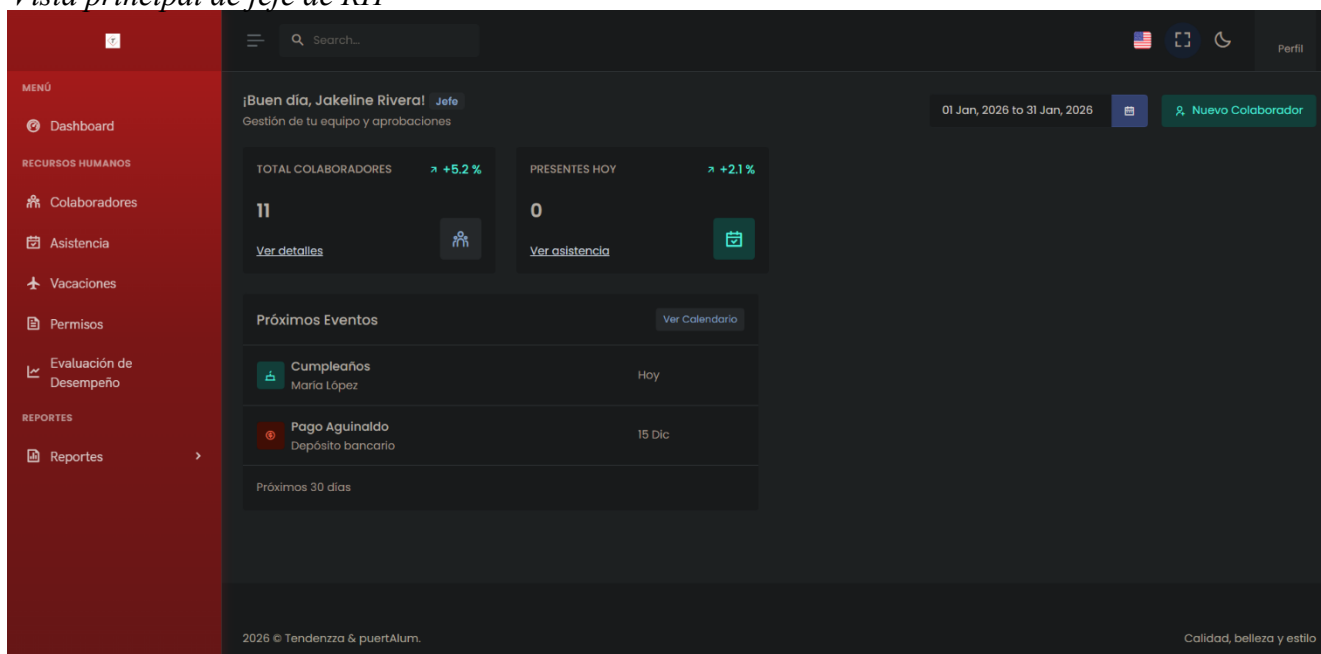


Fuente: Elaboración Propia (2026)

En la figura anterior se muestra la vista principal que visualiza el gerente general al iniciar sesión en el sistema. En esta pantalla se presenta el panel principal junto en el menú lateral (sidebar), desde el cual el usuario puede acceder a las distintas funcionalidades y acciones disponibles según su acceso.

## Figura 6

### Vista principal de jefe de RH

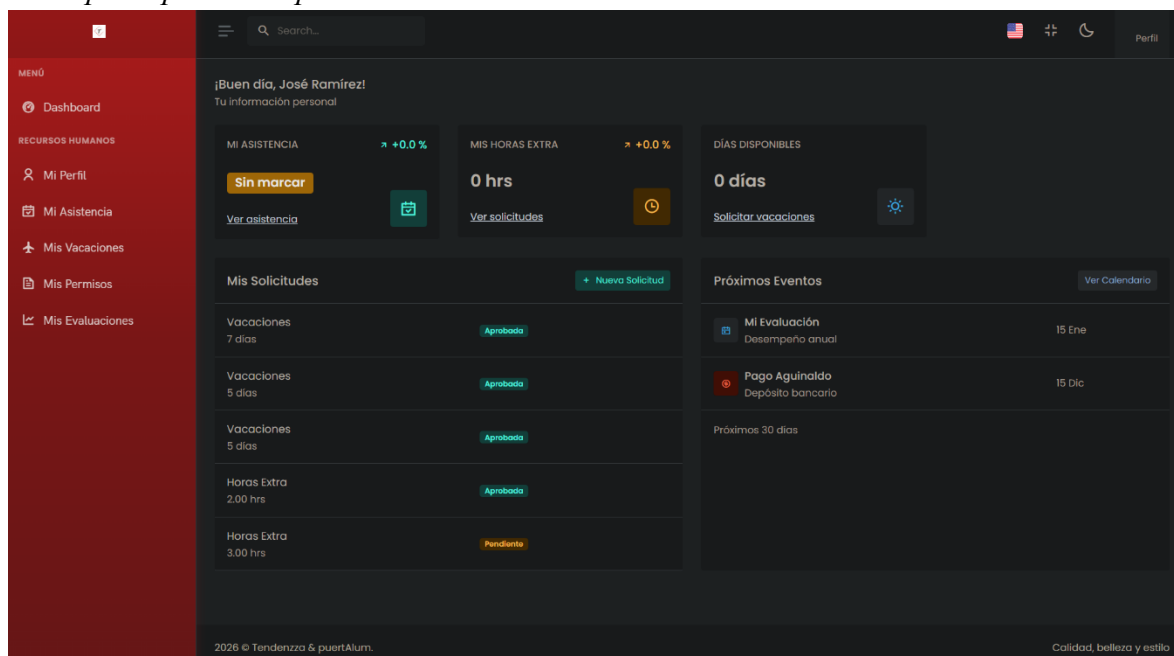


Fuente: Elaboración Propia (2026)

En la figura anterior se muestra la vista principal correspondiente al jefe de Recursos Humanos al iniciar sesión en el sistema. Esta interfaz presenta el panel de control y el menú lateral, desde donde el usuario puede gestionar los procesos propios del área, tales como colaboradores, permisos, vacaciones, incapacidades, horas extra, nómina y demás funciones asignadas según su rol.

## Figura 7

### Vista principal del empleado



Fuente: Elaboración Propia (2026)

En la figura 7, se muestra la vista principal para el empleado después de haber iniciado sesión, mostrándole sus funciones en el ‘sidebar’, información relevante y accesos rápidos para gestiones.

## Figura 8

### Agregar Colaborador

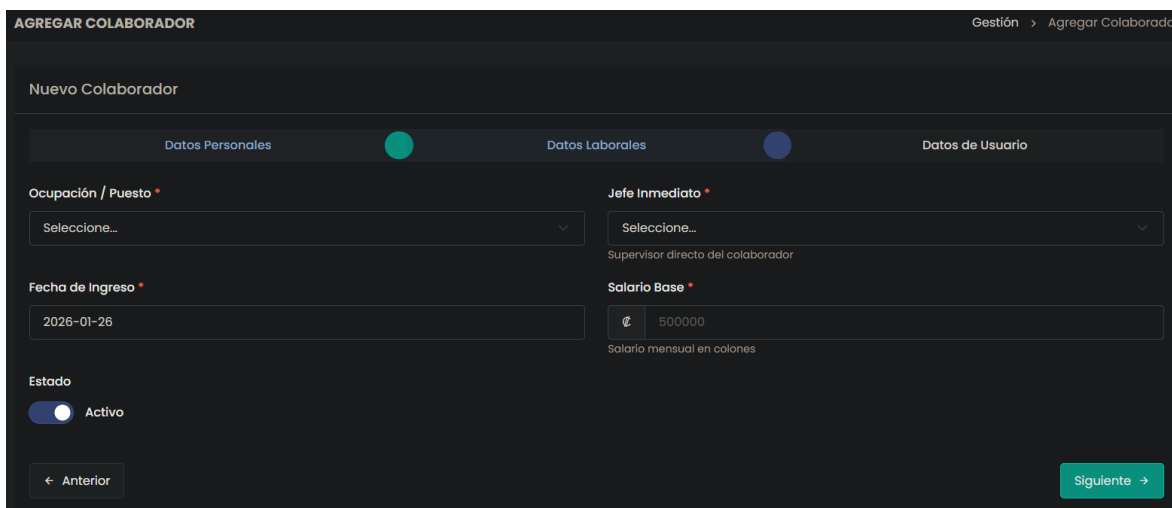
The screenshot shows a form titled 'AGREGAR COLABORADOR' with three tabs: 'Datos Personales', 'Datos Laborales', and 'Datos de Usuario'. The 'Datos Personales' tab is active, showing fields for 'Tipo de Identificación', 'Número de Identificación', 'Nombre', 'Primer Apellido', 'Segundo Apellido', 'Fecha de Nacimiento', 'Género', and 'País'. A 'Cancelar' button is on the left and a 'Siguiete' button is on the right.

Fuente: Elaboración Propia (2026)

En esta vista el jefe de RH o el administrador pueden ingresar a un colaborador, ingresando los datos correspondientes, es se divide en opciones, datos personales, datos laborales, y datos de usuario para que pueda ingresar al sistema.

## Figura 9

### *Agregar datos laborales*



The screenshot shows a web interface for adding a new employee. The title is 'AGREGAR COLABORADOR' and the breadcrumb is 'Gestión > Agregar Colaborador'. The main heading is 'Nuevo Colaborador'. There are three tabs: 'Datos Personales' (selected), 'Datos Laborales', and 'Datos de Usuario'. The 'Datos Laborales' tab is active, showing the following fields:

- Ocupación / Puesto \***: A dropdown menu with 'Seleccione...' as the current selection.
- Fecha de Ingreso \***: A date input field containing '2026-01-26'.
- Jefe Inmediato \***: A dropdown menu with 'Seleccione...' as the current selection. Below it, the text 'Supervisor directo del colaborador' is visible.
- Salario Base \***: A text input field containing '500000' with a currency symbol (₡) on the left. Below it, the text 'Salario mensual en colones' is visible.

At the bottom left, there is a toggle switch for 'Estado' set to 'Activo'. At the bottom, there are two buttons: '← Anterior' and 'Siguiete →'.

Fuente: Elaboración Propia (2026)

En esta opción se agregará los datos laborales del colaborador que previamente fue creado, aquí es donde se agrega el jefe inmediato, el puesto, salario y demás.

## Figura 10

### Agregar usuario

AGREGAR COLABORADOR

Gestión > Agregar Colaborador

Nuevo Colaborador

Datos Personales Datos Laborales Datos de Usuario

Opcional: Si no desea crear usuario ahora, puede hacerlo después desde el módulo de Usuarios.

Crear usuario de acceso al sistema

Nombre de Usuario \*

Ej: jperez, emp001

Máximo 20 caracteres, sin espacios

Contraseña \*

Confirmar Contraseña \*

Mínimo 8 caracteres

Los permisos se asignan desde el módulo Configuración → Permisos. Por defecto el usuario se crea con nivel de Empleado.

2026 © Tendenzza & puertAlum. Calidad, belleza y estilo

Fuente: Elaboración Propia (2026)

En esta última opción se puede asignar el usuario, pero solo para nivel empleado si el jefe de RH lo desea, de lo contrario tendría que avisar al administrador que cree el usuario como tal con niveles superiores.

## Figura 11




























### Lista de colaboradores

COLABORADORES Gestión > Colaboradores

Lista de Colaboradores  
Total: 14 | Activos: 11 | Inactivos: 3

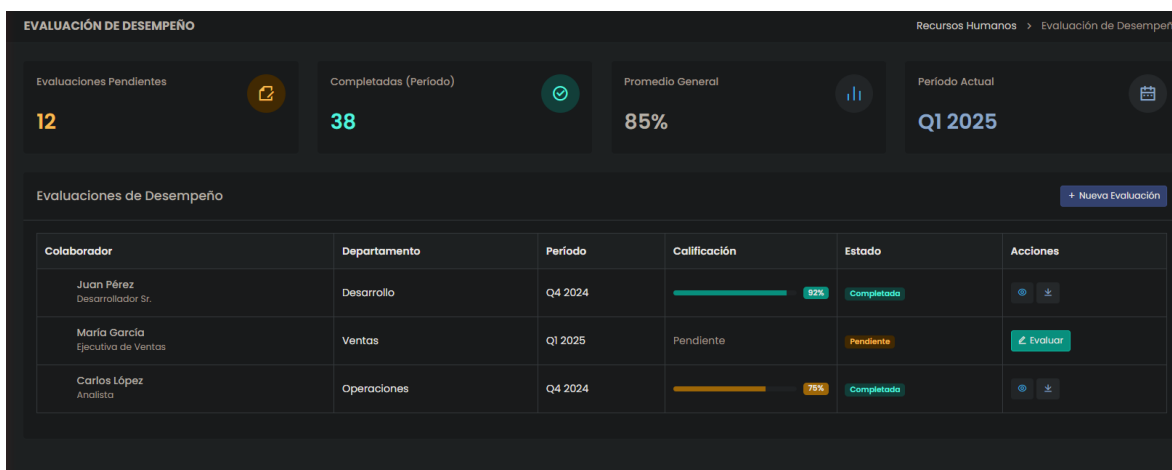
+ Agregar Colaborador Exportar

Q Buscar por nombre o apellidos... Todos Filtrar Limpiar

Colaborador	Identificación	Ocupación	Fecha de Ingreso	Salario Base	Estado	Acciones
AD Andrés Díaz Mora ID: 1	118520647	Gerente General	15/01/2020	€1,500,000.00	Activo	  
JR Jakeline Rivera Zúñiga ID: 2	205840321	Jefe RRHH	01/03/2020	€900,000.00	Activo	  
LH Luis Hernández Soto ID: 3	304560789	Jefe Operaciones	01/03/2020	€950,000.00	Activo	  
PV Patricia Vargas Jiménez ID: 4	402350198	Jefe Ventas	01/03/2020	€920,000.00	Activo	  
JR José Ramírez Pérez ID: 5	501230456	Asistente RRHH	10/01/2021	€550,000.00	Activo	  
MG María González Solís ID: 6	603450987	Analista	15/02/2021	€530,000.00	Activo	  
LM Laura Morales Vega ID: 7	702340123	Operario	10/03/2021	€520,000.00	Activo	  
DC Diana Campos Navarro ID: 8	801450678	Operario	12/04/2021	€490,000.00	Activo	  
RM Roberto Martínez Cruz ID: 9	155000123456	Vendedor	10/05/2021	€800,000.00	Activo	  

Fuente: Elaboración Propia (2026)

En la figura anterior se muestra el listado de los colaboradores activos, en el cual el gerente general y el jefe de recursos humanos puede ver la información completa de los colaboradores, editar los datos e inactivar al colaborador.

**Figura 12***Evaluación de desempeño*

Fuente: Elaboración Propia (2026)

**Figura 13***Realizar evaluación*

☆ **Evaluar Colaborador** ✕

Colaborador: María García      Período: Q1 2025  
 Puesto: Ejecutiva de Ventas      Evaluador: jrhh

Criterios de Evaluación

- Cumplimiento de Objetivos**  
 Deficiente 85% Excelente
- Trabajo en Equipo**  
 Deficiente 90% Excelente
- Puntualidad y Asistencia**  
 Deficiente 95% Excelente
- Iniciativa y Proactividad**  
 Deficiente 80% Excelente

**Observaciones y Retroalimentación**

Ingrese observaciones sobre el desempeño del colaborador...

**Áreas de Mejora Identificadas**

Identifique áreas donde el colaborador puede mejorar...

Cancelar Guardar Evaluación

Fuente: Elaboración Propia (2026)

En las figuras 12 y 13 se muestra el procedimiento de realizar una evaluación donde el jefe de RH o jefes directos pueden realizarla, utilizando barras con porcentajes para dar una calificación.

## Figura 14

### *Solicitar vacaciones*

The screenshot displays a user interface for vacation management. At the top, it says "GESTIÓN DE VACACIONES". Below that, there's a section titled "Mi Saldo de Vacaciones" with a greeting "Hola, María López". A large green circle shows the number "7" under the heading "Días Disponibles" for the "Periodo 2026". A summary table shows: "Días otorgados: 14", "Días disfrutados: 5", and "Días pendientes: 2". The "Saldo actual:" is "7 días", with a progress bar indicating "36% utilizado". Below this is a "Nueva Solicitud" form with fields for "Fecha Inicio" and "Fecha Fin" (both with "Seleccione" dropdowns), "Días Solicitados" (set to "0" with a "días hábiles" dropdown), and an "Observaciones" text area with the placeholder "Motivo (opcional)". There are buttons for "Enviar Solicitud" and "Limpiar". At the bottom, a "Política de Vacaciones" section lists rules: "Mínimo 3 días de anticipación", "Máximo 10 días consecutivos", "Requiere aprobación del jefe", and "No acumulables al siguiente año".

Categoría	Cantidad
Días otorgados	14
Días disfrutados	5
Días pendientes	2

**Saldo actual:** 7 días

36% utilizado

**Nueva Solicitud**

Fecha Inicio \*      Fecha Fin \*

Seleccione      Seleccione

Días Solicitados: 0 días hábiles

Observaciones: Motivo (opcional)

Enviar Solicitud      Limpiar

**Política de Vacaciones**

- Mínimo 3 días de anticipación
- Máximo 10 días consecutivos
- Requiere aprobación del jefe
- No acumulables al siguiente año

Fuente: Elaboración Propia (2026)



## Diccionario de la base de datos

A continuación, se desarrolla el diccionario de base de datos, mostrando todas las características de las tablas que se maneja en la base de datos.

**Tabla 46**

### Diccionario de datos - Adelanto Salario

Column name	Data Type	PK	NN	UQ	BIN	UN	ZE	AI	Default	Comment
idadelanto_salario	INT	✓	✓							
idColaborador	INT		✓							
monto	DECIMAL(10,2)		✓						'90000.00'	
mes	INT		✓							
anio	INT		✓							
idcatalogo_estado_solicitud	INT		✓						'1'	
aprobado_por	VARCHAR(20)		✓						'N/A'	
fecha_aprobacion	DATETIME		✓						'1900-01-01 00:00:00'	
fecha_creacion	TIMESTAMP		✓						CURRENT_TIMESTAMP	
fecha_actualizacion	TIMESTAMP		✓						CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP	

Fuente: Elaboración Propia (2026)

**Tabla 47**

### Diccionario de datos - Aguinaldo

Column name	Data Type	PK	NN	UQ	BIN	UN	ZE	AI	Default	Comment
idaguinaldo	INT	✓	✓							
idColaborador	INT		✓							
anio	INT		✓							
meses_laborados	DECIMAL(10,2)		✓							
salario_promedio	DECIMAL(10,2)		✓							
monto_aguinaldo	DECIMAL(10,2)		✓							
deducciones	DECIMAL(10,2)		✓							
monto neto	DECIMAL(10,2)		✓							
idcatalogo_estado_solicitud	INT		✓							
fecha_calculo	DATE		✓							
fecha_pago	DATE		✓							
observaciones	VARCHAR(500)		✓							
fecha_creacion	TIMESTAMP		✓						CURRENT_TIMESTAMP	
fecha_actualizacion	TIMESTAMP		✓						CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP	

Fuente: Elaboración Propia (2026)

**Tabla 48**

### Diccionario de datos - Asistencia

Column name	Data Type	PK	NN	UQ	BIN	UN	ZE	AI	Default	Comment
idasistencia	INT	✓	✓							
colaborador_idColaborador	INT		✓							
fecha	DATE		✓							
hora_entrada	TIME		✓							
hora_salida	TIME		✓							
estado_asistencia	VARCHAR(45)		✓							
latitud_entrada	DECIMAL(10,8)		✓							
longitud_entrada	DECIMAL(11,8)		✓							
ubicacion_entrada_id	INT		✓					'999'		FK: desde id de marc entrada (oficina/proyecto)
latitud_salida	DECIMAL(10,8)		✓							
longitud_salida	DECIMAL(11,8)		✓							
ubicacion_salida_id	INT		✓					'999'		FK: desde id de marc salida (oficina/proyecto)
idcatalogo_estado_asistencia	INT		✓							
observaciones	VARCHAR(500)		✓							
horas_trabajadas	DECIMAL(5,2)		✓							
fecha_creacion	TIMESTAMP		✓						CURRENT_TIMESTAMP	
fecha_actualizacion	TIMESTAMP		✓						CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP	
idnomina	INT		✓					'0'		

Fuente: Elaboración Propia (2026)

### Tabla 49

#### Diccionario de datos - Catalogo\_canton

catalogo_canton										
Column name	Data Type	PK	NN	UQ	BIN	UN	ZF	AI	Default	Comment
idCatalogo_Canton	INT	✓	✓							
Nombre_Canton	VARCHAR(45)		✓							
idCatalogo_Provincia	INT	✓	✓							
Activo	BIT(1)		✓							

Fuente: Elaboración Propia (2026)

### Tabla 50

#### Diccionario de datos - Catalogo\_categoria

catalogo_categoria										
Column name	Data Type	PK	NN	UQ	BIN	UN	ZF	AI	Default	Comment
idCatalogo_Categoria	INT	✓	✓							
Descripcion_Categoria	VARCHAR(45)		✓							
Activo	BIT(1)		✓							

Fuente: Elaboración Propia (2026)

Tabla

51

#### Diccionario de datos - Catalogo\_Credito\_Fiscal\_Hijo\_Conyugue

catalogo_credito_fiscal_hijo_conyugue										
Column name	Data Type	EK	NN	UQ	BIN	UN	ZF	AI	Default	Comment
idcatalogo_dedaccion_personal_renta	INT	✓	✓							
concepto	VARCHAR(50)		✓							
monto_mensual	DECIMAL(10,2)		✓							
activo	BIT(1)		✓							0'
fecha_creacion	TIMESTAMP		✓							CURRENT_TIMESTAMP
fecha_actualizacion	TIMESTAMP		✓							CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP

Fuente: Elaboración Propia (2026)

### Tabla 52

#### Diccionario de datos - Catalogo\_distrito

catalogo_distrito										
Column name	Data Type	PK	NN	UQ	BIN	UN	ZF	AI	Default	Comment
idCatalogo_Distrito	INT	✓	✓							
Nombre_Distrito	VARCHAR(45)		✓							
idCatalogo_Canton	INT	✓	✓							
idCatalogo_Provincia	INT	✓	✓							
Activo	BIT(1)		✓							

Fuente: Elaboración Propia (2026)

**Tabla 53***Diccionario de datos - Catalogo\_estado\_asistencia*

catalogo_estado asistencia											
Column name	Data Type	PK	NN	UQ	BIN	UN	ZF	AI	Default	Comment	
idcatalogo_estado_asistencia	INT	✓	✓								
descripcion	VARCHAR(50)		✓								
activo	BIT		✓								

Fuente: Elaboración Propia (2026)

**Tabla 54***Diccionario de datos - Catalogo\_estado\_evaluacion*

catalogo_estado evaluacion											
Column name	Data Type	PK	NN	UQ	BIN	UN	ZF	AI	Default	Comment	
idcatalogo_estado_evaluacion	INT	✓	✓								
descripcion	VARCHAR(50)		✓								
activo	BIT		✓								

Fuente: Elaboración Propia (2026)

**Tabla 55***Diccionario de datos - Catalogo\_estado\_liquidacion*

catalogo_estado liquidacion											
Column name	Data Type	PK	NN	UQ	BIN	UN	ZF	AI	Default	Comment	
idcatalogo_estado_liquidacion	INT	✓	✓								
descripcion	VARCHAR(100)		✓								
activo	BIT		✓								

Fuente: Elaboración Propia (2026)

**Tabla 56***Diccionario de datos - Catalogo\_estado\_solicitud*

catalogo_estado solicitud											
Column name	Data Type	PK	NN	UQ	BIN	UN	ZF	AI	Default	Comment	
idcatalogo_estado_solicitud	INT	✓	✓								
descripcion	VARCHAR(50)		✓								
activo	BIT		✓								

Fuente: Elaboración Propia (2026)

**Tabla 57***Diccionario de datos – Catalogo\_genero*

catalogo_genero											
Column name	Data Type	PK	NN	UQ	BIN	UN	ZF	AI	Default	Comment	
idCatalogo_Genero	INT	✓	✓								
Descripcion_Genero	VARCHAR(45)		✓								
Activo	BIT(1)		✓								

Fuente: Elaboración Propia (2026)

**Tabla 58***Diccionario de datos - Catalogo\_motivo\_salida*

catalogo_motivo_salida											
Column name	Data Type	PK	NN	UQ	BIN	UN	ZF	AI	Default	Comment	
idcatalogo_motivo_salida	INT	✓	✓								
descripcion	VARCHAR(100)		✓								
activo	BIT		✓								

Fuente: Elaboración Propia (2026)

**Tabla 59***Diccionario de datos - Catalogo\_nivel*

catalogo_nivel											
Column name	Data Type	PK	NN	UQ	BIN	UN	ZF	AI	Default	Comment	
idCatalogo_Nivel	INT	✓	✓								
Descripcion_Nivel	VARCHAR(45)		✓								
Activo	BIT(1)		✓								

Fuente: Elaboración Propia (2026)

**Tabla 60***Diccionario de datos - Catalogo\_ocupacion*

catalogo_ocupacion											
Column name	Data Type	PK	NN	UQ	BIN	UN	ZF	AI	Default	Comment	
idCatalogo_Ocupacion	INT	✓	✓								
Descripcion	VARCHAR(45)		✓								
Activo	BIT(1)		✓								

Fuente: Elaboración Propia (2026)

**Tabla 61***Diccionario de datos - Catalogo\_pais*

catalogo_pais											
Column name	Data Type	PK	NN	UQ	BIN	UN	ZF	AI	Default	Comment	
idcatalogo_pais	INT	✓	✓								
descripcion	VARCHAR(50)		✓								
activo	BIT		✓								

Fuente: Elaboración Propia (2026)

**Tabla 62***Diccionario de datos - Catalogo\_provincia*

catalogo_provincia											
Column name	Data Type	PK	NN	UQ	BIN	UN	ZF	AI	Default	Comment	
idCatalogo_Provincia	INT	✓	✓								
Nombre_Provincia	VARCHAR(45)		✓								
Activo	BIT(1)		✓								

Fuente: Elaboración Propia (2026)

**Tabla 63***Diccionario de datos - Catalogo\_tasa\_ccss*

catalogo_tasa_ccss												
Column name	DataType	PK	NN	UQ	BIN	UN	ZF	AI	Default	Comment		
idcatalogo_tasa_ccss	INT	✓	✓									
concepto	VARCHAR(50)		✓									
porcentaje_patrono	DECIMAL(5,2)		✓						0.00'			
porcentaje_trabajador	DECIMAL(5,2)		✓						0.00'			
activo	BIT(1)		✓						b'1			
fecha_creacion	TIMESTAMP		✓						CURRENT_TIMESTAMP			
fecha_actualizacion	TIMESTAMP		✓						CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP			

Fuente: Elaboración Propia (2026)

**Tabla 64***Diccionario de datos - Catalogo\_tipo\_concepto*

catalogo_tipo_concepto												
Column name	DataType	PK	NN	UQ	BIN	UN	ZF	AI	Default	Comment		
idcatalogo_tipo_concepto	INT	✓	✓									
descripcion	VARCHAR(50)		✓									
activo	BIT		✓									

Fuente: Elaboración Propia (2026)

**Tabla 65***Diccionario de datos - Catalogo\_tipo\_correo*

catalogo_tipo_correo												
Column name	DataType	PK	NN	UQ	BIN	UN	ZF	AI	Default	Comment		
idCatalogo_Tipo_Correo	INT	✓	✓									
Descripcion	VARCHAR(45)		✓									
ACTIVO	BIT(1)		✓									

Fuente: Elaboración Propia (2026)

**Tabla 66***Diccionario de datos - Catalogo\_tipo\_direccion*

catalogo_tipo_direccion												
Column name	DataType	PK	NN	UQ	BIN	UN	ZF	AI	Default	Comment		
idCatalogo_Tipo_Direccion	INT	✓	✓									
Tipo_Direccion	VARCHAR(45)		✓									
Activo	BIT		✓									

Fuente: Elaboración Propia (2026)

**Tabla 67***Diccionario de datos - Catalogo\_tipo\_identificacion*

catalogo_tipo_identificacion												
Column name	DataType	PK	NN	UQ	BIN	UN	ZF	AI	Default	Comment		
idcatalogo_tipo_identificacion	INT	✓	✓									
descripcion	VARCHAR(4550)		✓									
activo	BIT		✓									

Fuente: Elaboración Propia (2026)

**Tabla 68***Diccionario de datos - Catalogo\_tipo\_incapacidad*

catalogo_tipo_incapacidad												
Column name	Data Type	PK	NN	UQ	BIN	UN	ZF	AI	Default	Comment		
idcatalogo_tipo_incapacidad	INT	✓	✓									
descripcion	VARCHAR(50)		✓									
activo	BIT		✓									

Fuente: Elaboración Propia (2026)

**Tabla 69***Diccionario de datos - Catalogo\_tipo\_permiso*

catalogo_tipo_permiso												
Column name	Data Type	PK	NN	UQ	BIN	UN	ZF	AI	Default	Comment		
idcatalogo_tipo_permiso	INT	✓	✓									
nombre_permiso	VARCHAR(100)		✓									
con_goce_salarial	BIT		✓									
descripcion	VARCHAR(255)		✓									
activo	BIT		✓									

Fuente: Elaboración Propia (2026)

**Tabla 70***Diccionario de datos - Catalogo\_tipo\_telefono*

catalogo_tipo_telefono												
Column name	Data Type	PK	NN	UQ	BIN	UN	ZF	AI	Default	Comment		
idCatalogo_Tipo_Telefono	INT	✓	✓									
Description	VARCHAR(45)		✓									
Activo	BIT(1)		✓									

Fuente: Elaboración Propia (2026)

**Tabla 71***Diccionario de datos - Catalogo\_tramo\_renta*

catalogo_tramo_renta												
Column name	Data Type	PK	NN	UQ	BIN	UN	ZF	AI	Default	Comment		
idcatalogo_tramo_renta	INT	✓	✓									
tramo_desde	DECIMAL(12,2)		✓									
tramo_hasta	DECIMAL(12,2)		✓									
porcentaje	DECIMAL(5,2)		✓									
activo	BIT(1)		✓							B'I		
fecha_creacion	TIMESTAMP		✓							CURRENT_TIMESTAMP		
fecha_actualizacion	TIMESTAMP		✓							CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP		

Fuente: Elaboración Propia (2026)

**Tabla 72***Diccionario de datos - Cliente*

cliente											
Column name	DataType	PK	NN	LQ	BIN	UN	ZF	AI	Default	Comment	
idCliente	INT	✓	✓								
Fecha_Apertura	DATETIME		✓								
Activo	BIT(1)		✓								
idCatalogo_Categoria	INT		✓								
idPersona	INT		✓								

**Tabla 73***Diccionario de datos - Colaborador*

colaborador											
Column name	DataType	PK	NN	LQ	BIN	UN	ZF	AI	Default	Comment	
idColaborador	INT	✓	✓								
Fecha_Ingreso	DATETIME		✓								
activo	BIT		✓						1		
salario_base	DECIMAL(10,2)		✓						0		
fecha_creacion	TIMESTAMP		✓						CURRENT_TIMESTAMP		
fecha_actualizacion	TIMESTAMP		✓						CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP		
idCatalogo_Ocupacion	INT		✓								
Persona_idPersona	INT		✓								
Jefatura_idColaborador	INT		✓								
idUsuario	VARCHAR(20)		✓								

Fuente: Elaboración Propia (2026)

**Tabla 74***Diccionario de datos – Colaborador\_Credito\_Fiscal\_Hijo\_Conyuge*

colaborador_credito_fiscal_hijo_conyuge											
Column name	DataType	PK	NN	LQ	BIN	UN	ZF	AI	Default	Comment	
idcolaborador_deducccion_personal	INT	✓	✓								
idColaborador	INT		✓								
idcatalogo_deducccion_personal_renta	INT		✓								
cantidad	INT		✓						1		
activo	BIT(1)		✓						b'1'		
fecha_creacion	TIMESTAMP		✓						CURRENT_TIMESTAMP		
fecha_actualizacion	TIMESTAMP		✓						CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP		

Fuente: Elaboración Propia (2026)

**Tabla 75***Diccionario de datos – Colaborador\_Información\_Familiar*

colaborador_informacion_familiar											
Column name	DataType	PK	NN	LQ	BIN	UN	ZF	AI	Default	Comment	
idcolaborador_cargas_familiares	INT	✓	✓								
idColaborador	INT		✓								
num_hijos	INT		✓						0		
tiene_conyuge	BIT(1)		✓						b'0'		
fecha_creacion	TIMESTAMP		✓						CURRENT_TIMESTAMP		
fecha_actualizacion	TIMESTAMP		✓						CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP		

Fuente: Elaboración Propia (2026)

**Tabla 76***Diccionario de datos - Correo*

correo											
Column name	Data Type	PK	NN	UQ	BIN	UN	ZF	AI	Default	Comment	
Direccion_Correo	VARCHAR(100)	✓	✓								
Fecha_Ingreso	DATETIME		✓								
ACTIVO	BIT(1)		✓								
idPersona	INT	✓	✓								
idCatalogo_Tipo_Correo	INT		✓								

Fuente: Elaboración Propia (2026)

**Tabla 77***Diccionario de datos - Direccion*

direccion											
Column name	Data Type	PK	NN	UQ	BIN	UN	ZF	AI	Default	Comment	
idDireccion	INT	✓	✓								
Detalle	VARCHAR(200)		✓								
Persona_idPersona	INT	✓	✓								
idCatalogo_Distrito	INT	✓	✓								
idCatalogo_Canton	INT	✓	✓								
idCatalogo_Provincia	INT	✓	✓								
idCatalogo_Tipo_Direccion	INT		✓								

Fuente: Elaboración Propia (2026)

**Tabla 78***Diccionario de datos - Evaluacion\_desempeño*

evaluacion_desempeno											
Column name	Data Type	PK	NN	UQ	BIN	UN	ZF	AI	Default	Comment	
idEvaluacion_desempeno	INT	✓	✓								
idColaborador	INT	✓	✓								
periodo	VARCHAR(45)		✓								
fecha_evaluacion	DATE		✓								
evaluador	VARCHAR(20)		✓								
calificacion_general	DECIMAL(5,2)		✓								
cumplimiento_objetivos	INT		✓								
calidad_trabajo	INT		✓								
trabajo_equipo	INT		✓								
puntualidad	INT		✓								
iniciativa	INT		✓								
resolucion_problemas	INT		✓								
observaciones_supervisor	VARCHAR(2000)		✓							N/A	
fortalezas	VARCHAR(2000)		✓							N/A	
areas_mejora	VARCHAR(2000)		✓							N/A	
plan_desarrollo	VARCHAR(2000)		✓							N/A	
bonificacion_asignada	DECIMAL(10,2)		✓								
idcatalogo_estado_evaluacion	INT		✓								
fecha_creacion	TIMESTAMP		✓							CURRENT_TIMESTAMP	
fecha_actualizacion	TIMESTAMP		✓							CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP	
idnomina	INT		✓							∅	

Fuente: Elaboración Propia (2026)

**Tabla 79***Diccionario de datos - Horas\_extra*

horas_extra										
Column name	Data Type	PK	NN	UQ	BIN	UN	ZF	AI	Default	Comment
idHoras_extra	INT	✓	✓							
idColaborador	INT	✓	✓							
fecha_solicitud	DATE		✓							
fecha_trabajo	DATE		✓							
horas_solicitadas	DECIMAL(5,2)		✓							
descripcion_trabajo	VARCHAR(500)		✓							
idcatalogo_estado_solicitud	INT		✓							
aprobado_por	VARCHAR(20)		✓							
fecha_aprobacion	DATETIME		✓						'1900-01-01 00:00:00'	
motivo_rechazo	VARCHAR(100)		✓						'N/A'	
monto_pagar	DECIMAL(10,2)		✓							
fecha_creacion	TIMESTAMP		✓						CURRENT_TIMESTAMP	
fecha_actualizacion	TIMESTAMP		✓						CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP	
idnomina	INT		✓						'0'	

Fuente: Elaboración Propia (2026)

**Tabla 80***Diccionario de datos - Incapacidades*

incapacidades										
Column name	Data Type	PK	NN	UQ	BIN	UN	ZF	AI	Default	Comment
idincapacidades	INT	✓	✓							
colaborador_idColaborador	INT	✓	✓							
idcatalogo_tipo_incapacidad	INT		✓							
fecha_inicio	DATE		✓							
fecha_fin	DATE		✓							
dias_incapacidad	INT		✓							
diagnostico	VARCHAR(500)		✓							
numero_documento	VARCHAR(100)		✓							
documento_respaldo	VARCHAR(255)		✓							
porcentaje_pago	DECIMAL(5,2)		✓						'100.00'	
validado_por	VARCHAR(20)		✓							
idcatalogo_estado_solicitud	INT		✓							
observaciones	VARCHAR(500)		✓						'N/A'	
fecha_creacion	TIMESTAMP		✓						CURRENT_TIMESTAMP	
fecha_actualizacion	TIMESTAMP		✓						CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP	
idnomina	INT		✓						'0'	

Fuente: Elaboración Propia (2026)

**Tabla 81***Diccionario de datos - Liquidación*

Liquidacion										
Column name	Data Type	PK	NN	UQ	BIN	UN	ZF	AI	Default	Comment
idliquidacion	INT	✓	✓							
idColaborador	INT	✓	✓							
fecha_ingreso	DATE		✓							
fecha_salida	DATE		✓							
idmotivo_salida	INT		✓							
salario_mensual	DECIMAL(10,2)		✓							
dias_trabajados	INT		✓							
preaviso	DECIMAL(10,2)		✓						0	
cesantia	DECIMAL(10,2)		✓						0	
vacaciones_proporcionales	DECIMAL(10,2)		✓						0	
aguinaldo_proporcional	DECIMAL(10,2)		✓						0	
salario_pendiente	DECIMAL(10,2)		✓						0	
total_indemnizacion	DECIMAL(10,2)		✓						0	
deducciones	DECIMAL(10,2)		✓						0	
total_netto	VARCHAR(45)		✓							
idestado_liquidacion	INT		✓							
fecha_calculo	DATE		✓							
fecha_pago	DATE		✓						'1900-01-01'	
calculado_por	VARCHAR(20)		✓							
observaciones	VARCHAR(500)		✓						'N/A'	
fecha_creacion	TIMESTAMP		✓						CURRENT_TIMESTAMP	
fecha_actualizacion	TIMESTAMP		✓						CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP	

Fuente: Elaboración Propia (2026)

**Tabla 82***Diccionario de datos - Nomina*

nomina										
Column name	DataType	PK	NN	UQ	BIN	UN	ZF	AI	Default	Comment
idnomina	INT	✓	✓							
idColaborador	INT		✓							
periodo	VARCHAR(20)		✓							
anio	INT		✓							
mes	INT		✓							
salario_base	DECIMAL(10,2)		✓						'0.00'	
total_horas_extra	DECIMAL(10,2)		✓						'0.00'	
total_bonificaciones	DECIMAL(10,2)		✓						'0.00'	
total_ingresos	DECIMAL(10,2)		✓						'0.00'	
deduccion_ccsa_sem	DECIMAL(10,2)		✓						'0.00'	
deduccion_ccsa_ivm	DECIMAL(10,2)		✓						'0.00'	
deduccion_banco_popular	DECIMAL(10,2)		✓						'0.00'	
deduccion_renta	DECIMAL(10,2)		✓						'0.00'	
credito_fiscal_hijos	DECIMAL(10,2)		✓						'0.00'	
credito_fiscal_conyuge	DECIMAL(10,2)		✓						'0.00'	
total_deducciones	DECIMAL(10,2)		✓						'0.00'	
salario_netto	DECIMAL(10,2)		✓						'0.00'	
dias_laborados	INT		✓						'0'	
dias_vacaciones	INT		✓						'0'	
dias_permisos	INT		✓						'0'	
dias_incapacidad	INT		✓						'0'	
monto_incapacidad	DECIMAL(10,2)		✓						'0.00'	
idcatalogo_tasa_ccsa_sem	INT		✓						'1'	
idcatalogo_tasa_ccsa_ivm	INT		✓						'2'	
idcatalogo_tasa_ccsa_bp	INT		✓						'3'	
idcatalogo_tramo_renta	INT		✓						'1'	
idcatalogo_estado_solicitud	INT		✓						'1'	
pagado_por	VARCHAR(20)		✓						'N/A'	
fecha_pago	DATETIME		✓						'1900-01-01 00:00:00'	
fecha_creacion	TIMESTAMP		✓						CURRENT_TIMESTAMP	
fecha_actualizacion	TIMESTAMP		✓						CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP	

Fuente: Elaboración Propia (2026)

**Tabla 83***Diccionario de datos - Nomina\_detalle*

nomina_detalle										
Column name	DataType	PK	NN	UQ	BIN	UN	ZF	AI	Default	Comment
idnomina_detalle	INT	✓	✓							
idnomina	INT		✓							
idColaborador	INT		✓							
concepto	VARCHAR(50)		✓							
monto	DECIMAL(10,2)		✓						'0.00'	
idcatalogo_tipo_concepto	INT		✓							
fecha_creacion	TIMESTAMP		✓						CURRENT_TIMESTAMP	
fecha_actualizacion	TIMESTAMP		✓						CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP	

Fuente: Elaboración Propia (2026)

**Tabla 84***Diccionario de datos - Permisos\_por\_usuario*

permiso_por_usuario											
Column name	DataType	PK	NN	UQ	BIN	UN	ZF	AI	Default	Comment	
idUsuario	VARCHAR(20)		✓	✓							
idPermisos_Disponibles	INT		✓	✓							
Activo	BIT(1)		✓	✓							
Fecha_Creacion	DATETIME		✓	✓							
Fecha_Actualizacion	DATETIME		✓	✓							

Fuente: Elaboración Propia (2026)

**Tabla 85***Diccionario de datos - Permisos*

permisos											
Column name	Data Type	PK	NN	UQ	BIN	UN	ZE	AI	Default	Comment	
idpermisos	INT	✓	✓								
idColaborador	INT	✓	✓								
idcatalogo_tipo_permiso	INT		✓								
fecha_solicitud	DATE		✓								
fecha_inicio	DATE		✓								
fecha_fin	DATE		✓								
horas	DECIMAL(5,2)		✓								
dias	INT		✓								
motivo	VARCHAR(500)		✓								
idcatalogo_estado_solicitud	INT		✓					1			
aprobado_por	VARCHAR(20)		✓								
fecha_aprobacion	DATETIME		✓						'1900-01-01 00:00:00'		
motivo_rechazo	VARCHAR(500)		✓						'N/A'		
documento_respaldo	VARCHAR(255)		✓						'N/A'		
fecha_ereccion	TIMESTAMP		✓						CURRENT_TIMESTAMP		
fecha_actualizacion	TIMESTAMP		✓						CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP		

Fuente: Elaboración Propia (2026)

**Tabla 86***Diccionario de datos - Permisos\_disponibles*

permisos disponibles											
Column name	Data Type	PK	NN	UQ	BIN	UN	ZE	AI	Default	Comment	
idPermisos_Disponibles	INT	✓	✓								
Nombre_Permiso	VARCHAR(45)		✓								
Activo	BIT(1)		✓								
idCatalogo_Nivel	INT		✓								
ID_Padre	INT		✓								

Fuente: Elaboración Propia (2026)

**Tabla 87***Diccionario de datos - Persona*

persona											
Column name	Data Type	PK	NN	UQ	BIN	UN	ZE	AI	Default	Comment	
idPersona	INT	✓	✓								
numero_identificacion	VARCHAR(50)		✓								
Nombre	VARCHAR(45)		✓								
Apellido1	VARCHAR(45)		✓								
Apellido2	VARCHAR(45)		✓								
Fecha_Nacimiento	DATE		✓								
idCatalogo_Genero	INT		✓								
idcatalogo_tipo_identificacion	INT		✓								
idcatalogo_pais	INT		✓								

Fuente: Elaboración Propia (2026)

**Tabla 88***Diccionario de datos - Saldo\_vacaciones*

saldo_vacaciones										
Column name	Data Type	PK	NN	UQ	BIN	UN	ZE	AI	Default	Comment
idsaldo_vacaciones	INT	✓	✓							
idColaborador	INT	✓	✓							
anio	INT		✓							
dias_totales	INT		✓							
dias_disfrutados	INT		✓						0	
dias_disponibles	INT		✓							
dias_pendientes_anio_anterior	INT		✓							
fecha_calculo	DATE		✓							
created_at	TIMESTAMP		✓						CURRENT_TIMESTAMP	
updated_at	TIMESTAMP		✓						CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP	

Fuente: Elaboración Propia (2026)

**Tabla 89***Diccionario de datos - Telefono*

telefono										
Column name	Data Type	PK	NN	UQ	BIN	UN	ZE	AI	Default	Comment
Numero_Telefono	INT	✓	✓							
Fecha_Ingreso	DATETIME		✓							
Activo	BIT(1)		✓							
idPersona	INT	✓	✓							
idCatalogo_Tipo_Telefono	INT		✓							

Fuente: Elaboración Propia (2026)

**Tabla 90***Diccionario de datos - Ubicación\_Autorizada*

ubicacion_autorizada										
Column name	Data Type	PK	NN	UQ	BIN	UN	ZE	AI	Default	Comment
idubicacion_autorizada	INT	✓	✓							
nombre	VARCHAR(100)		✓							
latitud	DECIMAL(10,8)		✓							
longitud	DECIMAL(11,8)		✓							
radio_metros	INT		✓							
requiere_validacion_gps	BIT(1)		✓						b'0'	
fecha_inicio	DATE		✓							
fecha_fin	DATE		✓							
activo	BIT(1)		✓						b'1'	
registrado_por	INT		✓							
fecha_creacion	TIMESTAMP		✓						CURRENT_TIMESTAMP	
fecha_actualizacion	TIMESTAMP		✓						CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP	

**Tabla 91***Diccionario de datos - Usuario*

usuario										
Column name	Data Type	PK	NN	UQ	BIN	UN	ZE	AI	Default	Comment
idUsuario	VARCHAR(20)	✓	✓							
Clave	VARCHAR(191)		✓							
Intentos_Fallidos	TINYINT		✓							
Bloqueo	BIT(1)		✓							
fecha_creacion	TIMESTAMP		✓						CURRENT_TIMESTAMP	
fecha_actualizacion	TIMESTAMP		✓						CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP	

Fuente: Elaboración Propia (2026)

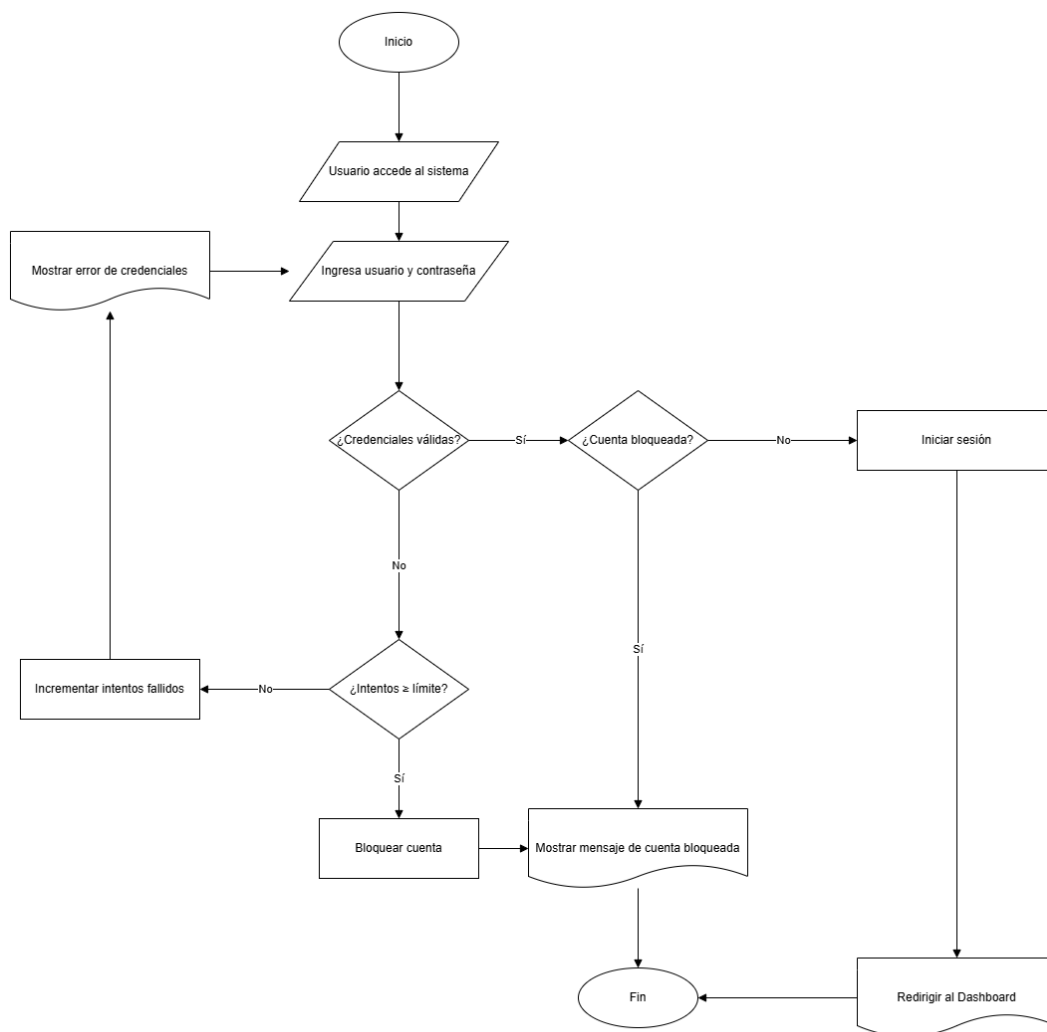
**Tabla 92***Diccionario de datos - Vacaciones*

vacaciones										
Columna name	Data Type	PK	NN	UQ	BIN	UN	ZF	AI	Default	Comment
idvacaciones	INT	✓	✓							
idColaborador	INT	✓	✓							
fecha_solicitud	DATE		✓							
fecha_inicio	DATE		✓							
fecha_fin	DATE		✓							
dias_solicitados	INT		✓							
dias_disponibles	INT		✓							
anio	INT		✓							
idcatalogo_estado_solicitud	INT		✓						'1'	
aprobado_por	VARCHAR(20)		✓							
fecha_aprobacion	DATETIME		✓						'1900-01-01 00:00:00'	
motivo_rechazo	VARCHAR(500)		✓						'N/A'	
observaciones	VARCHAR(500)		✓							
fecha_creacion	TIMESTAMP		✓						CURRENT_TIMESTAMP	
fecha_actualizacion	TIMESTAMP		✓						CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP	
idnomina	INT		✓						'0'	

Fuente: Elaboración Propia (2026)

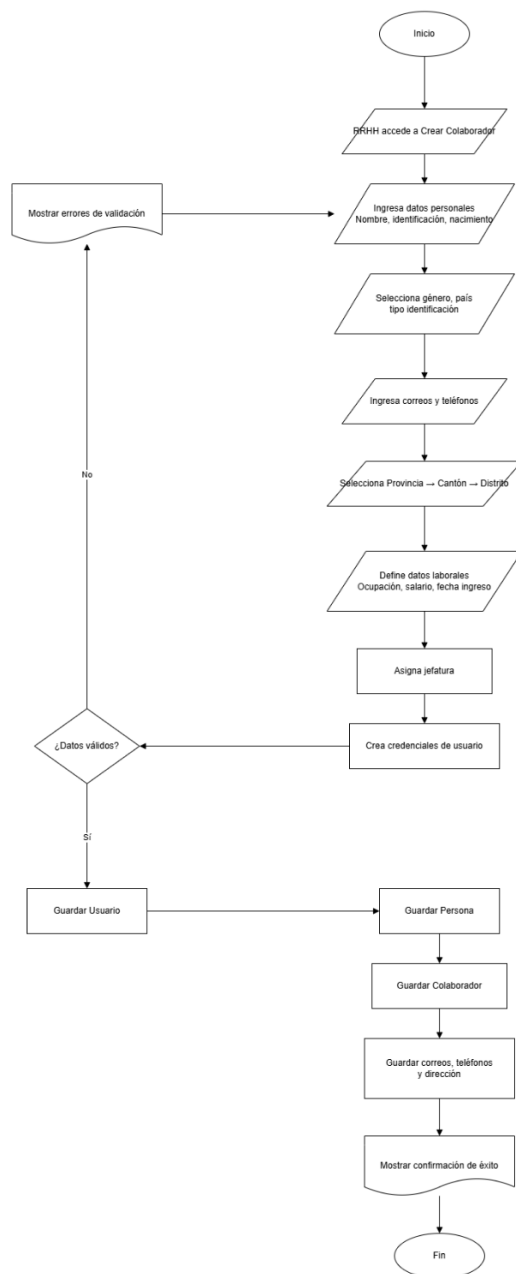
**Diseño de procesos**

En el siguiente apartado, se detallan los diseños de procesos implementados para este proyecto de graduación, con el fin de evidenciar la estructura y el enfoque metodológico adoptado.

**Figura 16***Diseño de procesos - Autenticación*

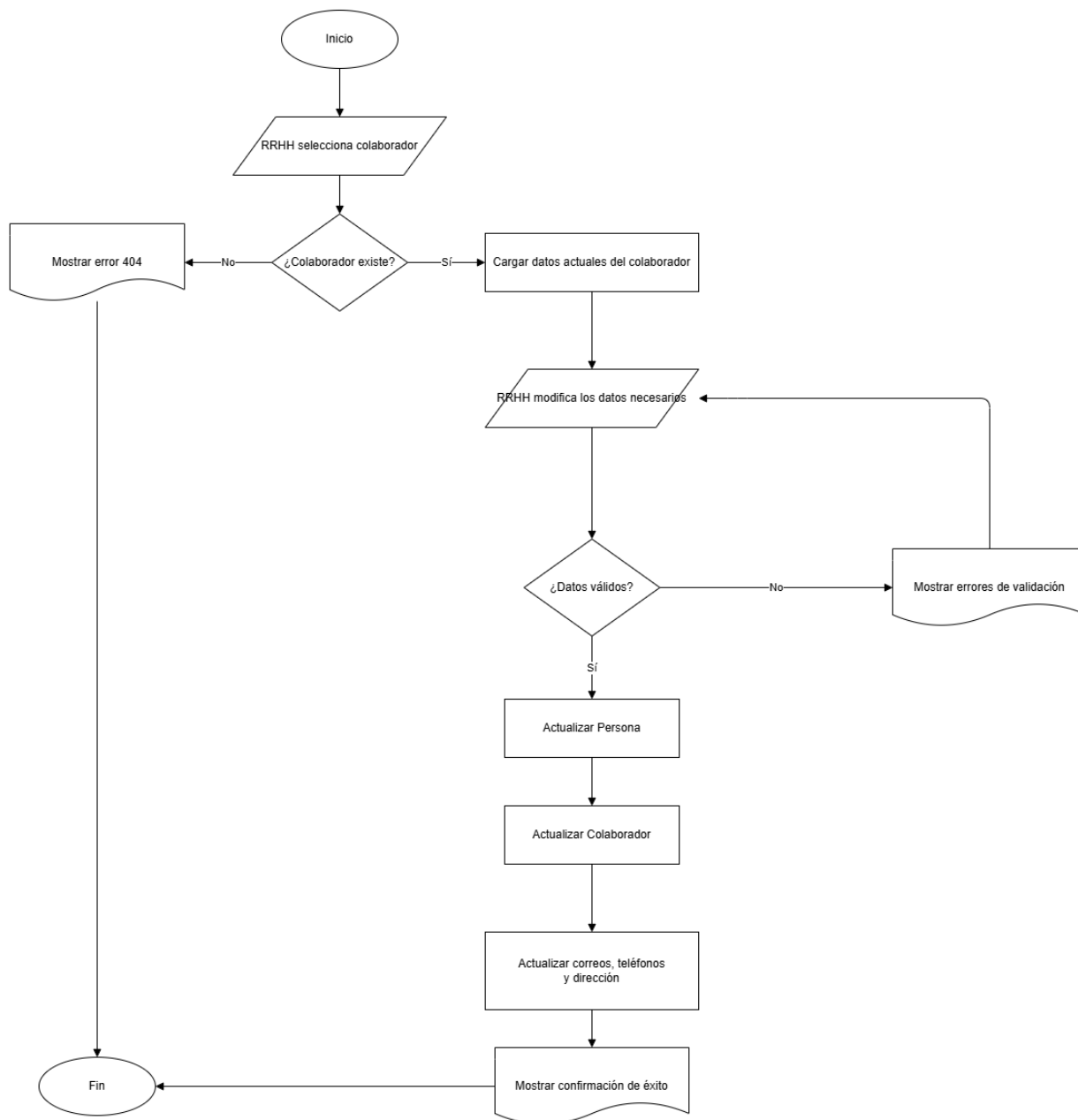
Fuente: Elaboración Propia (2026)

El diseño anterior corresponde al proceso de autenticación de un usuario dentro del sistema. En él se muestra el flujo desde que el usuario ingresa sus credenciales hasta su validación por parte del sistema. El proceso contempla la verificación de intentos fallidos y el posible bloqueo de la cuenta si se supera el límite permitido. Si las credenciales son correctas y la cuenta no está bloqueada, el usuario puede iniciar sesión y acceder al panel principal del sistema.

**Figura 17***Diseño de procesos - Agregar Colaborador*

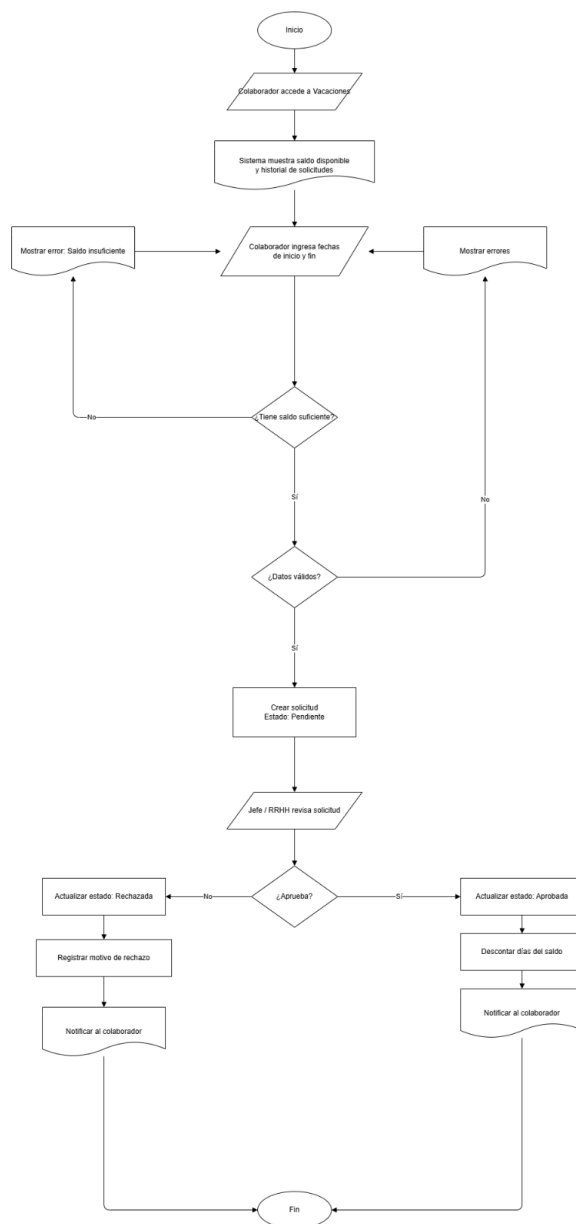
Fuente: Elaboración Propia (2026)

La imagen anterior corresponde al proceso de registro de un colaborador en el sistema. En él se muestra cómo el área de Recursos Humanos ingresa los datos personales, de contacto y laborales del colaborador. El proceso incluye una validación de la información antes de guardar los datos en el sistema y finalizar con una confirmación de registro exitoso.

**Figura 18***Diseño de procesos – Editar Colaborador*

Fuente: Elaboración Propia (2026)

El diseño anterior corresponde al proceso de actualización de información de un colaborador en el sistema. En él se muestra cómo el área de Recursos Humanos selecciona al colaborador, verifica su existencia y procede a modificar sus datos.

**Figura 19***Diseño de procesos – Vacaciones*

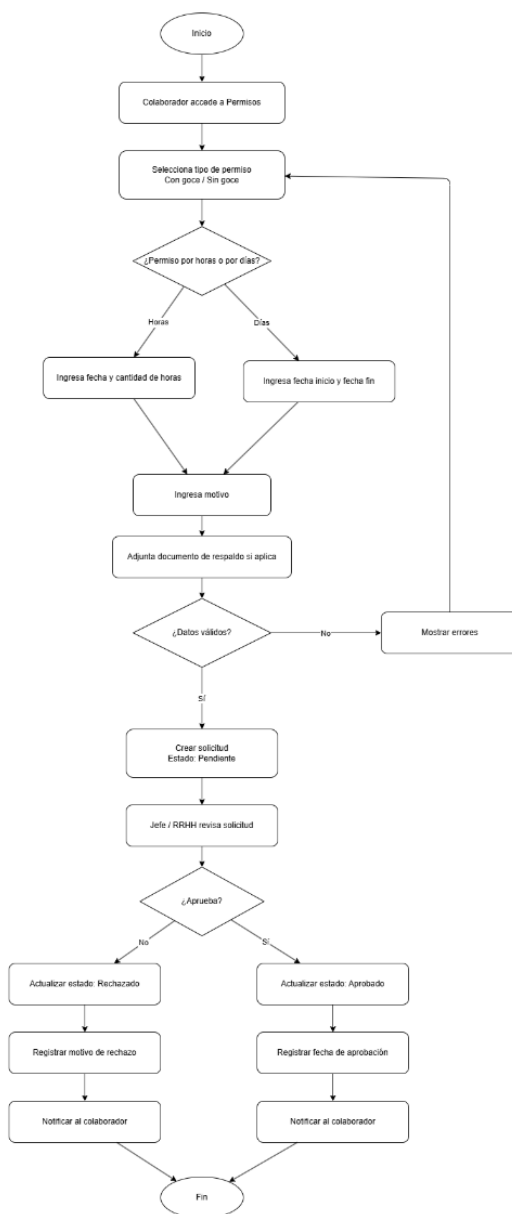
Fuente: Elaboración Propia (2026)

La figura anterior corresponde al proceso de solicitud de vacaciones de un colaborador dentro del sistema. En él se muestra cómo el colaborador ingresa las fechas solicitadas. El sistema verifica el saldo disponible y valida la información. Posteriormente, la solicitud es revisada por el jefe o RRHH, quienes pueden aprobarla o rechazarla, notificando finalmente al colaborador el resultado.

**Figura 20***Diseño de procesos – Horas Extra*

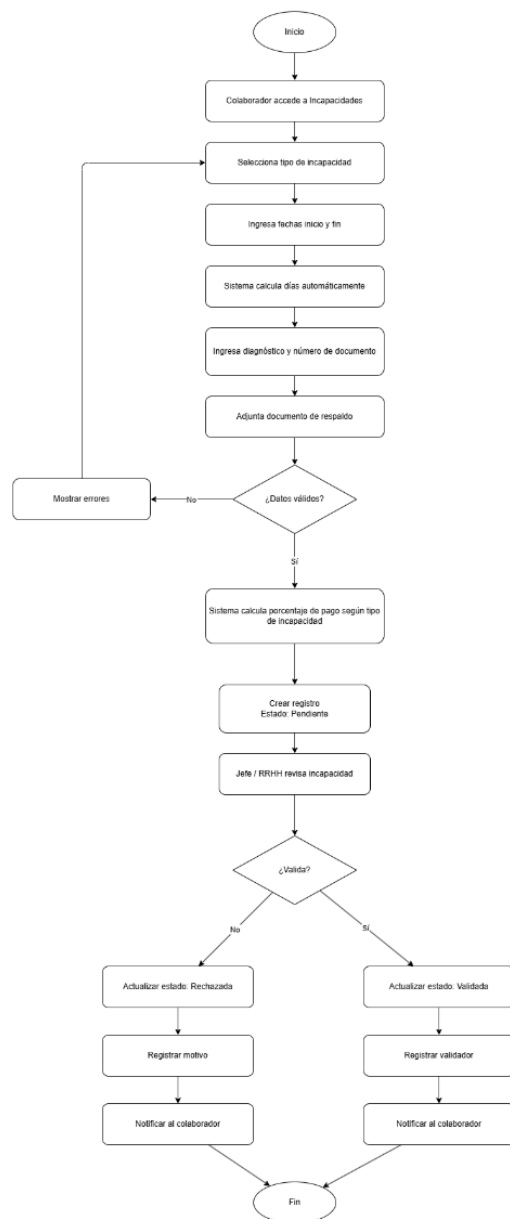
Fuente: Elaboración Propia (2026)

El diagrama mostrado anteriormente corresponde al proceso de solicitud de horas extra de un colaborador en el sistema. En él se muestra cómo el colaborador ingresa la fecha y la cantidad de horas solicitadas, el sistema valida la información y calcula el monto correspondiente. Posteriormente, la solicitud es revisada por el jefe, quien puede aprobarla o rechazarla y finalmente se notifica al colaborador del resultado.

**Figura 21***Diseño de procesos - Permisos*

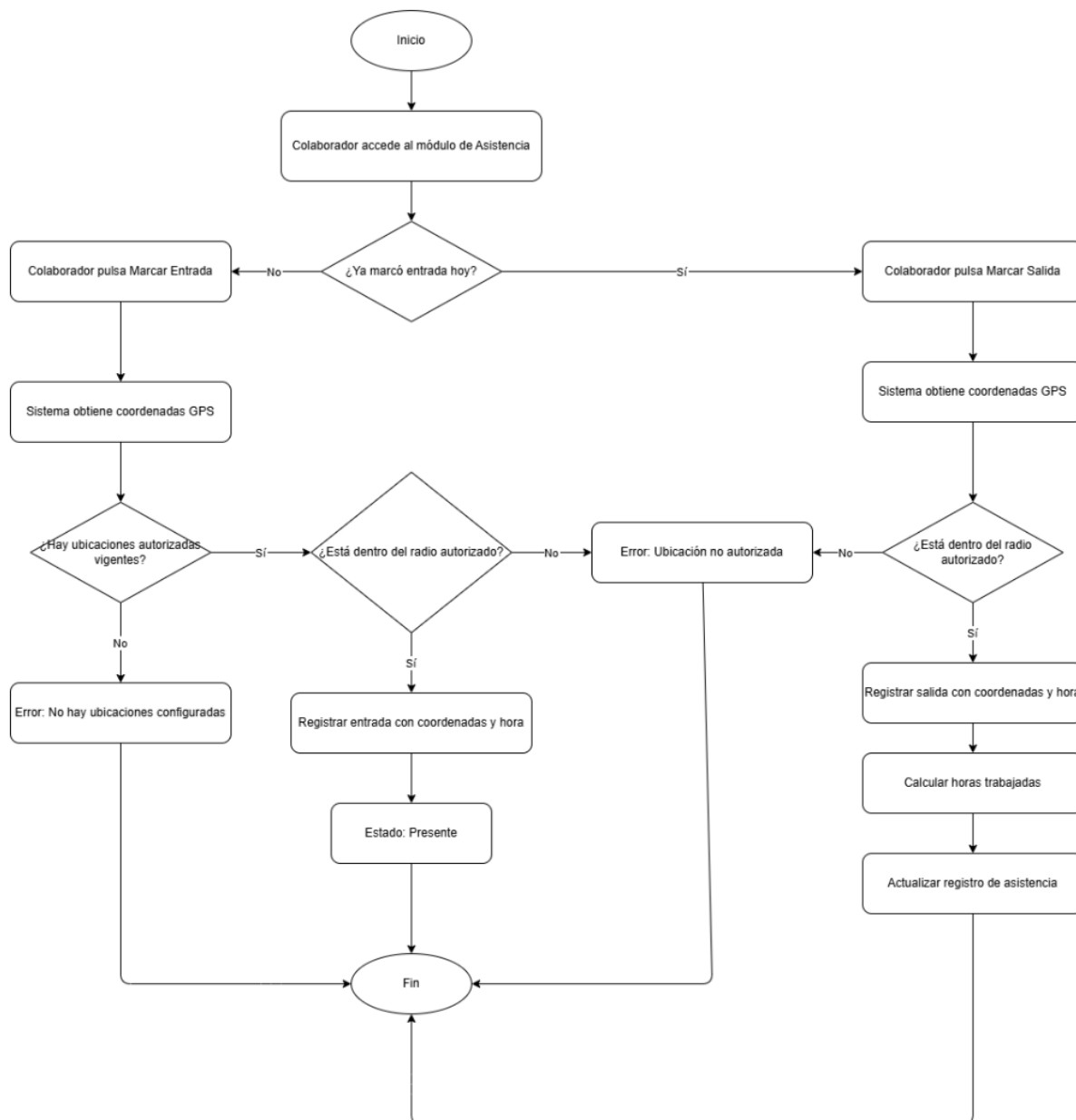
Fuente: Elaboración Propia (2026)

El esquema mostrado corresponde al proceso de solicitud de permisos de un colaborador en el sistema. En él se muestra cómo el colaborador selecciona el tipo de permiso, ingresa las fechas u horas y el motivo de la solicitud. Posteriormente, el sistema valida la información y la solicitud es revisada por el jefe o recursos humanos, quienes pueden aprobarla o rechazarla y finalmente notificar al colaborador el resultado.

**Figura 22***Diseño de procesos - Incapacidades*

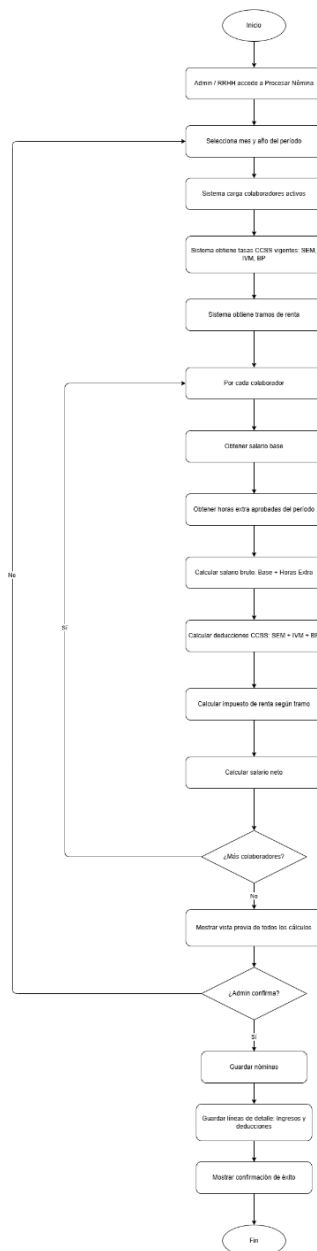
Fuente: Elaboración Propia (2026)

La figura anterior corresponde al proceso de registro y validación de incapacidades dentro del sistema. En él se muestra cómo el colaborador ingresa la información de la incapacidad y adjunta el documento de respaldo. Posteriormente, el sistema valida los datos y el jefe o recursos humanos revisa la solicitud para aprobarla o rechazarla, notificando finalmente al colaborador el resultado.

**Figura 23***Diseño de procesos - Asistencia*

Fuente: Elaboración Propia (2026)

El diseño anterior corresponde al proceso de registro de asistencia de un colaborador en el sistema. En él se muestra cómo el colaborador marca su entrada o salida, mientras el sistema obtiene las coordenadas GPS y verifica si se encuentra dentro del área autorizada.

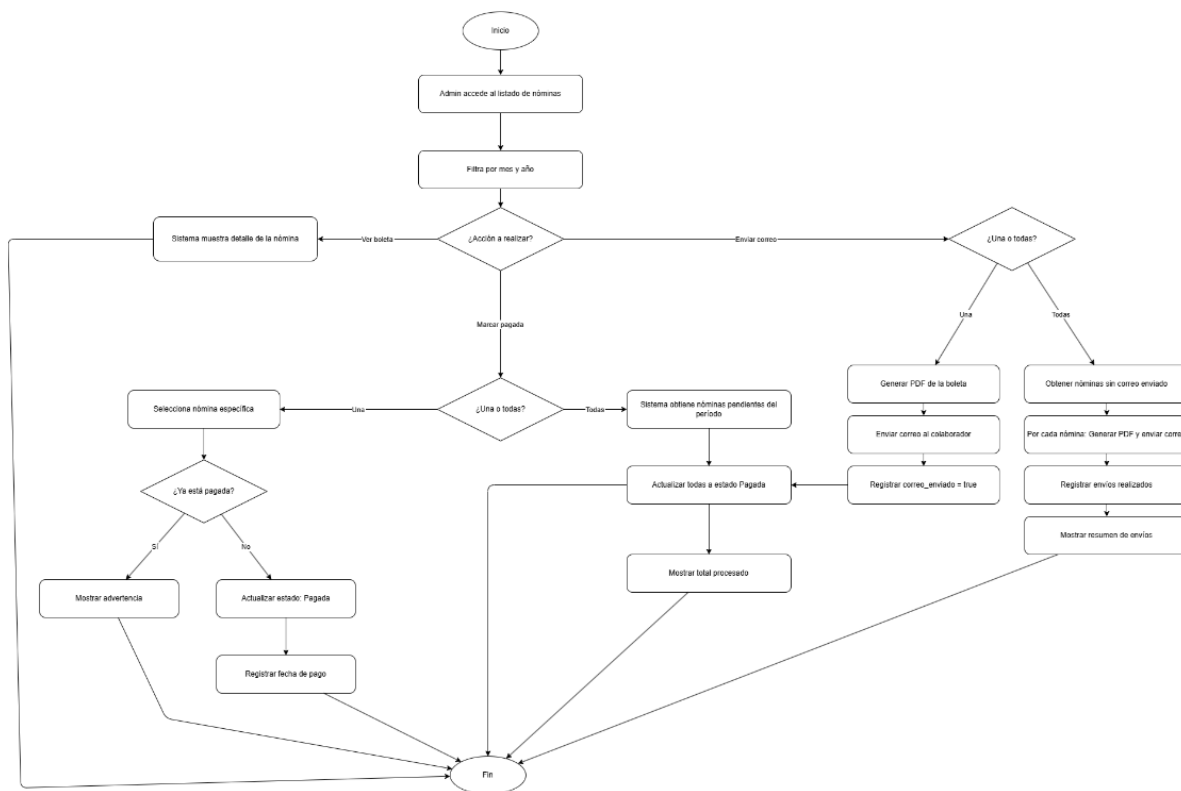
**Figura 24***Diseño de procesos – Procesar Nómina*

Fuente: Elaboración Propia (2026)

La figura ilustrada corresponde al proceso de procesamiento de nómina dentro del sistema. En él se muestra cómo el administrador o el área de recursos humanos selecciona el período a procesar y el sistema calcula el salario bruto, deducciones de CCSS, impuesto de renta y salario neto para cada colaborador.

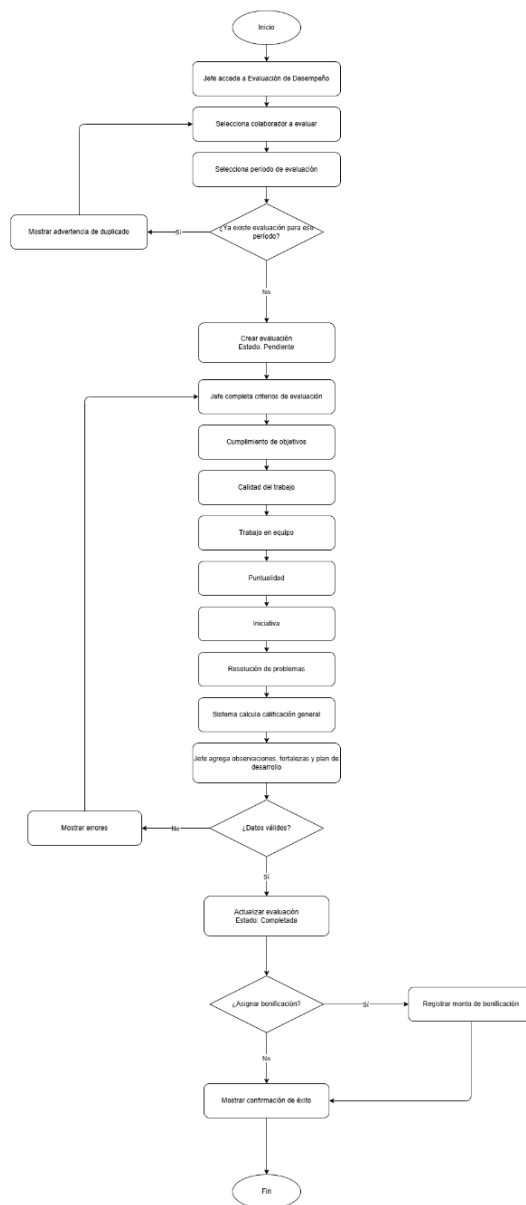
**Figura 25**

*Diseño de procesos – Gestión de Nómina*



Fuente: Elaboración Propia (2026)

El diagrama de gestión de nóminas muestra cómo el administrador accede al listado de nóminas, filtra por período y puede realizar distintas acciones como ver el detalle de las boletas, marcar una o varias nóminas como pagadas o enviar las boletas por correo a los colaboradores. El sistema valida el estado de las nóminas, actualiza la información correspondiente y finalmente muestra un resumen de las acciones realizadas.

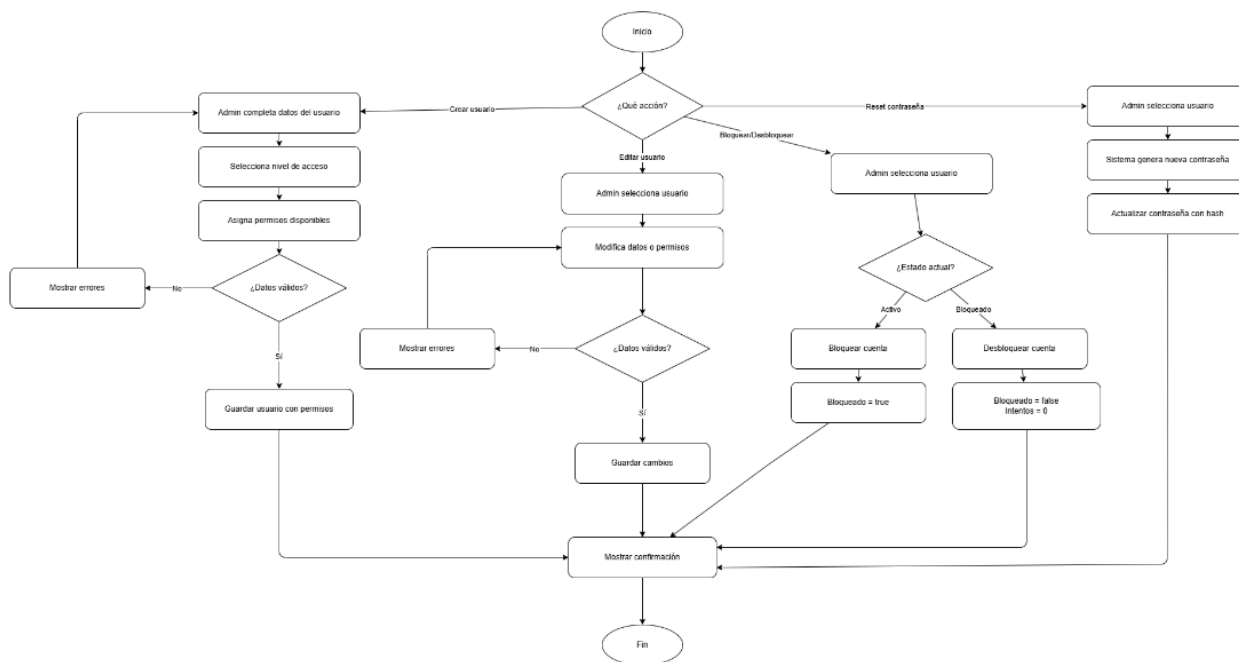
**Figura 26***Diseño de procesos – Evaluación Desempeño*

Fuente: Elaboración Propia (2026)

La figura anterior corresponde al proceso de evaluación de desempeño de un colaborador en el sistema. En él se muestra cómo el jefe selecciona al colaborador y el período de evaluación, completa los criterios establecidos y el sistema calcula la calificación general. Posteriormente, se registran observaciones y, si los datos son válidos, la evaluación se guarda como completada y opcionalmente se puede asignar una bonificación.

Figura 27

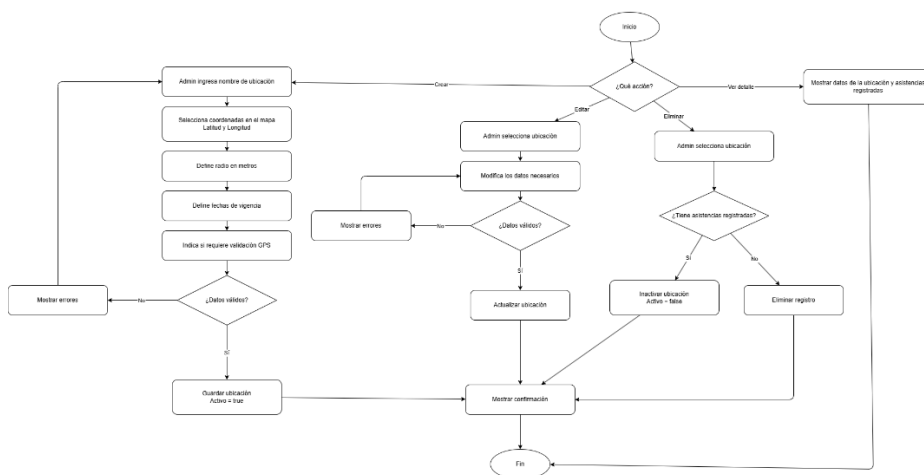
## Diseño de procesos – Ubicaciones Autorizadas



Fuente: Elaboración Propia (2026)

El diseño anterior corresponde al proceso de gestión de usuarios del sistema. En él se muestra cómo el administrador puede crear, editar o administrar cuentas de usuario, asignando niveles de acceso y permisos. Además, permite bloquear cuentas y restablecer contraseñas.

Figura 28

***Diseño de procesos – Ubicaciones Autorizadas***

Fuente: Elaboración Propia (2026)

El diagrama presentado corresponde al proceso de gestión de ubicaciones de trabajo en el sistema. En él se muestra cómo el administrador puede crear, editar, eliminar o consultar ubicaciones utilizadas para el registro de asistencia, definiendo coordenadas GPS, radio permitido y fechas de vigencia.

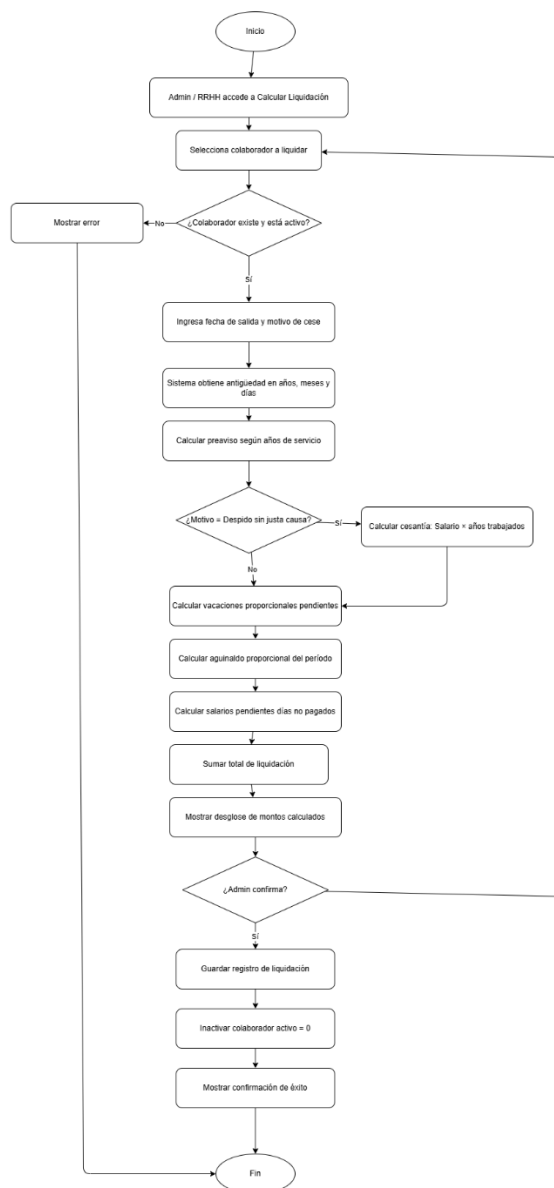
Figura 29

## Diseño de procesos – Aguinaldos



Fuente: Elaboración Propia (2026)

El siguiente diagrama representa el proceso de cálculo de aguinaldo en el sistema. En él se muestra cómo el administrador o el área de recursos humanos selecciona el año correspondiente y el sistema verifica si ya existe un registro previo. Posteriormente, se analizan las nóminas de cada colaborador durante el período establecido para calcular el monto proporcional del aguinaldo según los salarios y meses trabajados.

**Figura 30***Diseño de procesos – Liquidación*

Fuente: Elaboración Propia (2026)

Este diagrama representa el proceso de cálculo de liquidación de un colaborador dentro del sistema. En él se muestra cómo el administrador o el área de recursos humanos selecciona al colaborador e ingresa la fecha y el motivo de salida. El sistema calcula automáticamente los montos correspondientes como preaviso, cesantía si aplicara, vacaciones y aguinaldo proporcionales, así como salarios pendientes.

## Diseño de salidas

En el siguiente apartado, se muestran las pantallas del sistema web usados para la salida de información de cada módulo.

### Figura 31

#### *Diseño de Salidas - Colaboradores*

The screenshot displays a dashboard for 'COLABORADORES'. At the top, it shows 'TOTAL COLABORADORES 20', 'ACTIVOS 17', and 'INACTIVOS 3'. A search bar and a filter button are present. The main table lists the following data:

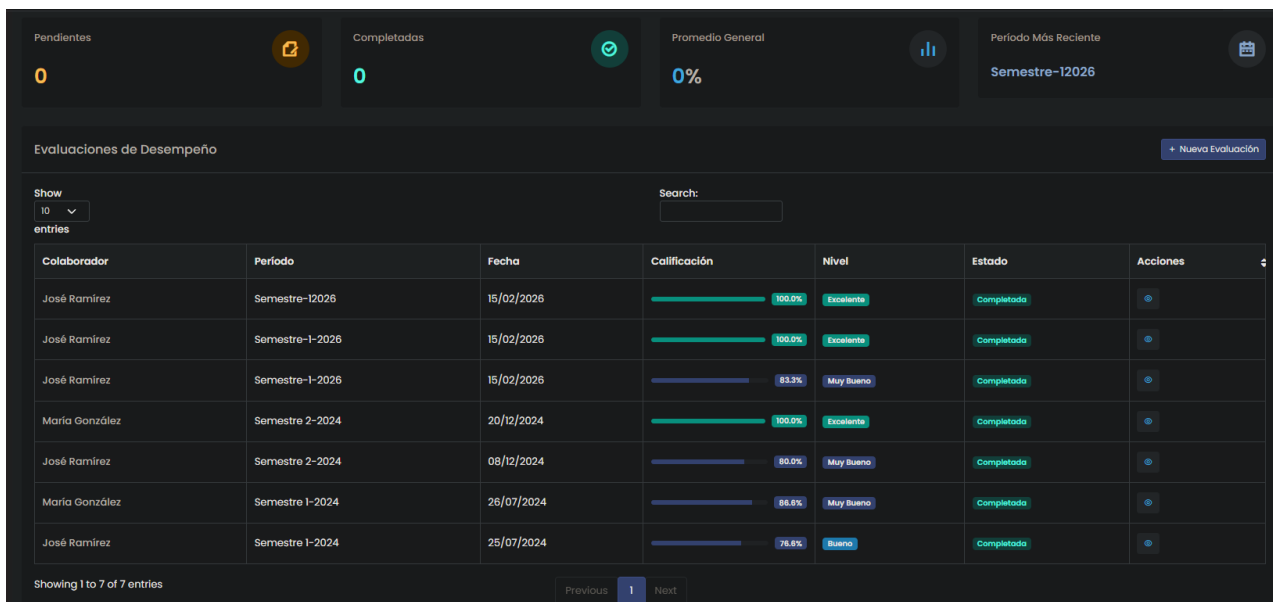
ID	Colaborador	Cedula	Ocupación	Fecha Ingreso	Salario	Estado	Acciones
#1	AD Andrés Diaz	118520647	Gerente General	15/01/2020	€1,500,000.00	ACTIVO	[Edit] [Delete]
#2	JR Jakeline Rivera	205840321	Jefa RRHH	01/03/2020	€900,000.00	ACTIVO	[Edit] [Delete]
#3	LH Luis Hernández	304560789	Jefe Operaciones	01/03/2020	€950,000.00	ACTIVO	[Edit] [Delete]
#4	PV Patricia Vargas	402350198	Jefa Ventas	01/03/2020	€920,000.00	ACTIVO	[Edit] [Delete]
#5	JR José Ramírez	501230456	Asistente RRHH	10/01/2021	€550,000.00	ACTIVO	[Edit] [Delete]
#6	MG María González	603450987	Analista	15/02/2021	€530,000.00	ACTIVO	[Edit] [Delete]
#7	LM Laura Morales	702340123	Opcario	10/03/2021	€520,000.00	ACTIVO	[Edit] [Delete]

Fuente: Elaboración Propia (2026)

En la figura anterior, se muestra la salida de los datos de los colaboradores registrados y activos de la empresa. Se muestra así mismo el nombre del colaborador, su ID, nombre, cedula, ocupación, fecha de ingreso, salario, estado, y las acciones a realizar.

## Figura 32

### Diseño de Salidas - Evaluaciones



Fuente: Elaboración Propia (2026)

En la figura, se muestra el listado de evaluaciones realizadas de cada colaborador, el nombre del colaborador, el departamento, periodo, la calificación, estado y las acciones a seguir.

Figura 33

## Diseño de salidas - Vacaciones

Colaborador	Solicitud	Periodo	Días	Estado	Acciones
DC Diana Campos Navarro Sin puesto	02/01/2026 1 month ago	23/12/2024 29/12/2024	7	Aprobada	
FS Fernando Silva Medina Sin puesto	02/01/2026 1 month ago	23/12/2024 31/12/2024	9	Aprobada	
AD Andrés Díaz Mora Sin puesto	02/01/2026 1 month ago	24/12/2024 31/12/2024	7	Aprobada	
JR José Ramírez Pérez Sin puesto	02/01/2026 1 month ago	26/12/2024 31/12/2024	5	Aprobada	
MG María González Solís Sin puesto	02/01/2026 1 month ago	23/12/2024 27/12/2024	5	Aprobada	
DC Diana Campos Navarro Sin puesto	02/01/2026 1 month ago	23/12/2024 29/12/2024	7	Aprobada	
FS Fernando Silva Medina Sin puesto	02/01/2026 1 month ago	23/12/2024 31/12/2024	9	Aprobada	
AD Andrés Díaz Mora Sin puesto	02/01/2026 1 month ago	24/12/2024 31/12/2024	7	Aprobada	
JR José Ramírez Pérez Sin puesto	02/01/2026 1 month ago	26/12/2024 31/12/2024	5	Aprobada	
MG María González Solís Sin puesto	02/01/2026 1 month ago	23/12/2024 27/12/2024	5	Aprobada	
AD Andrés Díaz Mora Sin puesto	02/01/2026 1 month ago	25/03/2024 31/03/2024	7	Aprobada	
JR Jakeline Rivera Zúñiga Sin puesto	02/01/2026 1 month ago	25/03/2024 31/03/2024	7	Aprobada	
LH Luis Hernández Soto Sin puesto	02/01/2026 1 month ago	25/03/2024 31/03/2024	7	Aprobada	
PV Patricia Vargas Jiménez Sin puesto	02/01/2026 1 month ago	25/03/2024 31/03/2024	7	Aprobada	

Fuente: Elaboración Propia (2026)

En la figura anterior, se muestra un listado de las vacaciones solicitadas por los colaboradores, donde se muestran los datos importantes como el periodo solicitado, los días solicitados, el estado y por últimos las acciones.

### Figura 34

#### Diseño de salidas - Horas Extra

Colaborador	Solicitud	Fecha Trabajo	Horas	Monto	Estado	Acciones
JR José Ramírez Pérez Sin puesto	22/02/2026 2 weeks ago	18/02/2026	3.5 hrs	₡17,500.00	Pendiente	👁️ ✓ ✕
MG María González Solís Sin puesto	22/02/2026 2 weeks ago	20/02/2026	2.0 hrs	₡10,000.00	Pendiente	👁️ ✓ ✕
JR José Ramírez Pérez Sin puesto	02/01/2026 2 months ago	08/12/2024	3.0 hrs	₡0.00	Pendiente	👁️ ✓ ✕
JR José Ramírez Pérez Sin puesto	02/01/2026 2 months ago	22/11/2024	2.0 hrs	₡6,875.00	Aprobada	👁️
MG María González Solís Sin puesto	02/01/2026 2 months ago	24/11/2024	3.5 hrs	₡11,593.75	Aprobada	👁️
MG María González Solís Sin puesto	02/01/2026 2 months ago	02/12/2024	2.0 hrs	₡0.00	Rechazada	👁️

Fuente: Elaboración Propia (2026)

En la figura anterior, se observa el listado de las solicitudes de hora extra de los colaboradores.

## Figura 35

### Diseño de salidas - Solicitud de Permisos

Colaborador	Período	Tipo	Duración	Motivo	Estado	Acciones
JR José Ramírez	23/02/2026	Permiso Médico	1 día	asasasasasqas	Aprobado	🔍
JR José Ramírez	16/02/2026	Permiso Personal	1 día	A	Rechazado	🔍
JR José Ramírez	12/11/2024	Permiso Personal	2.00 horas	Exámenes laboratorio	Aprobado	🔍
JR José Ramírez	26/11/2024	Permiso Personal	3.00 horas	Renovación licencia c...	Aprobado	🔍
MG María González	15/12/2024	Permiso Personal	4.00 horas	Cita ginecológica	Pendiente	🔍 ✓ ✕
MG María González	28/11/2024	Permiso Personal	8.00 horas	Viaje personal	Rechazado	🔍

Fuente: Elaboración Propia (2026)

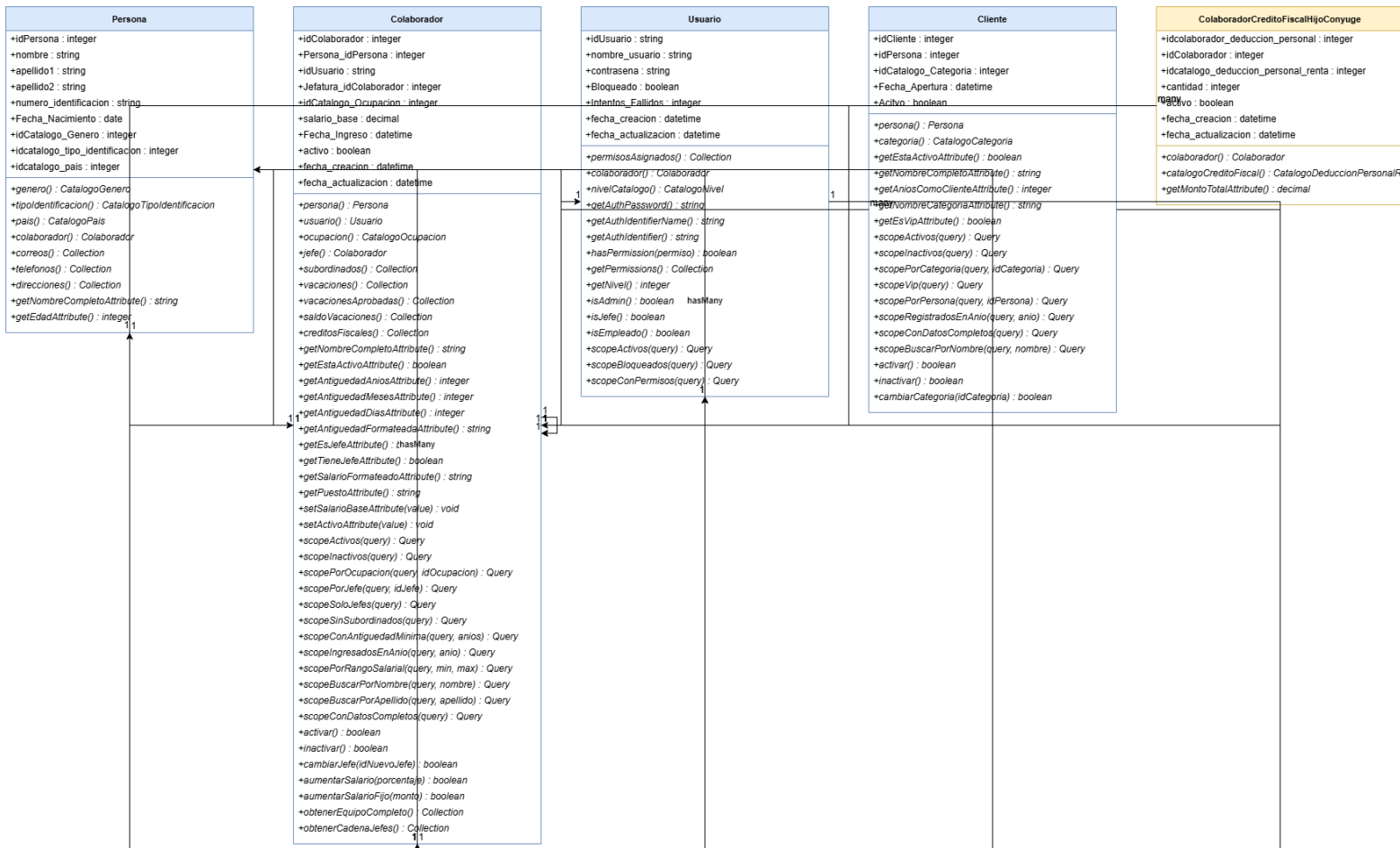
En la figura 31, se detalla la lista de la solicitud de permisos realizadas por los colaboradores, información como el colaborador, fecha del permiso, tipo de permiso, el horario, el motivo, el estado, y las acciones.

## Diagrama UML

En la siguiente figura, se muestra el diagrama de clases en al que se trabaja para esta aplicación.

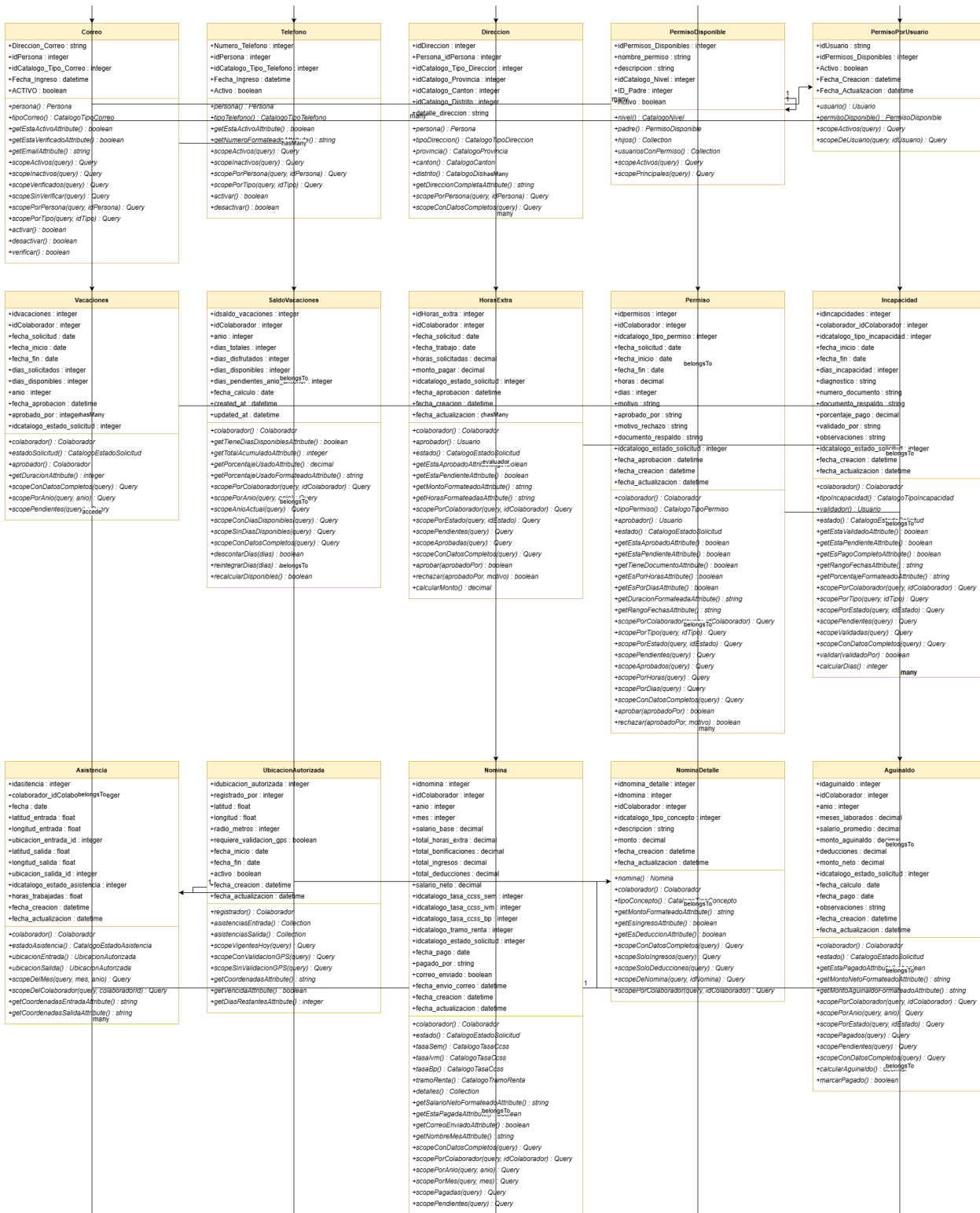
Figura 36

### Diagrama de Clases



Fuente: Elaboración Propia (2026)

Figura  
Diagrama de Clases



Fuente: Elaboración Propia (2026)

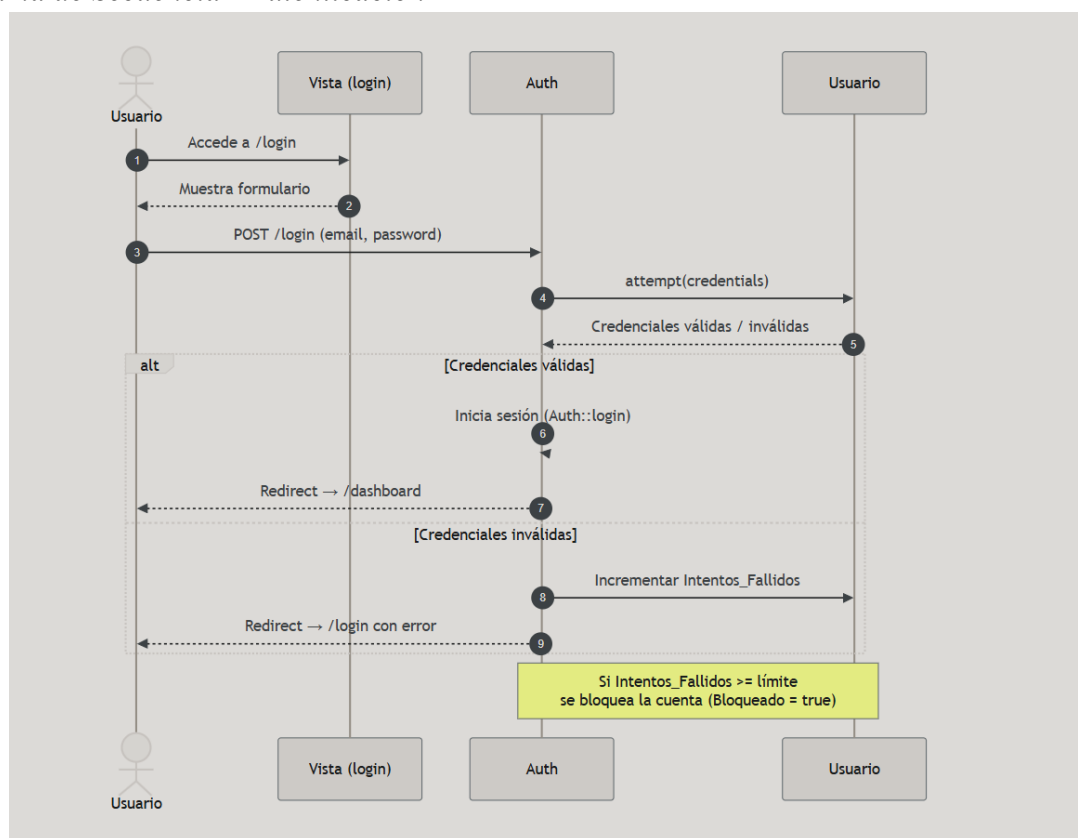


## Diagramas de secuencia

En el siguiente apartado, se muestran los diagramas de secuencia, los cuales permiten entender de una forma clara y fácil el cómo interactúan las diferentes partes del sistema para lograr un objetivo en común.

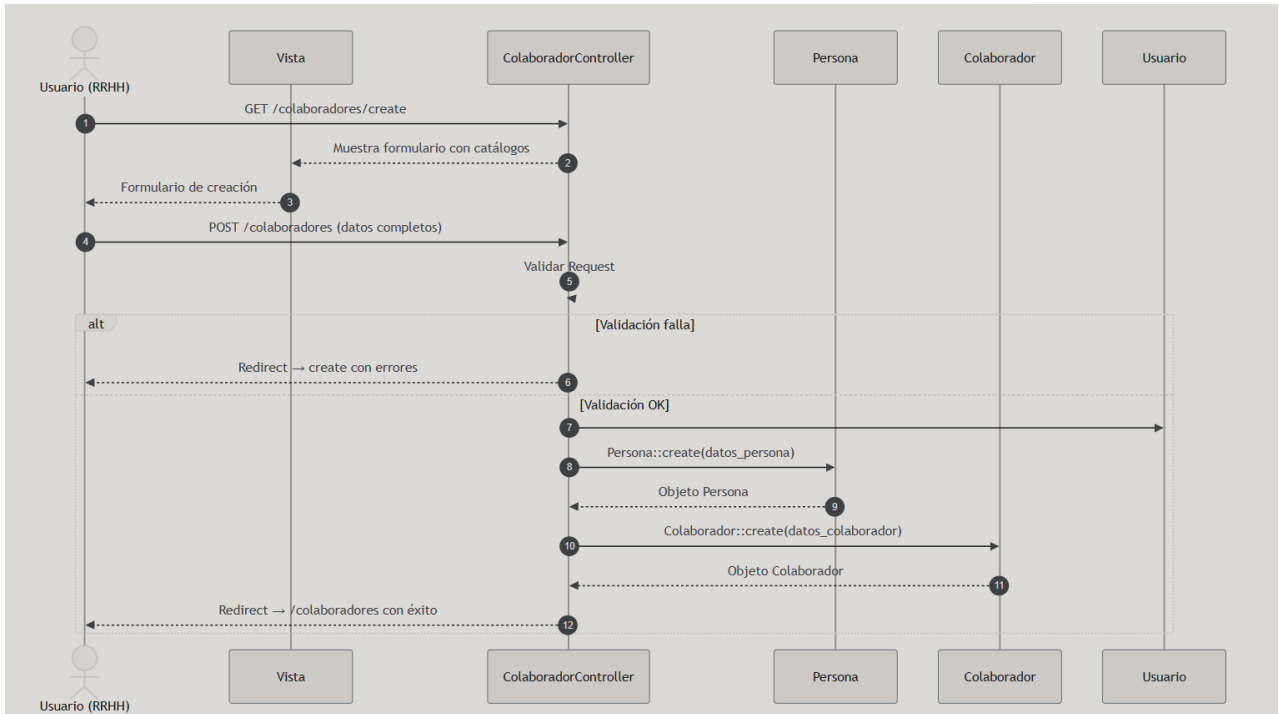
**Figura 39**

*Diagrama de Secuencia - Autenticación*



Fuente: Elaboración Propia (2026)

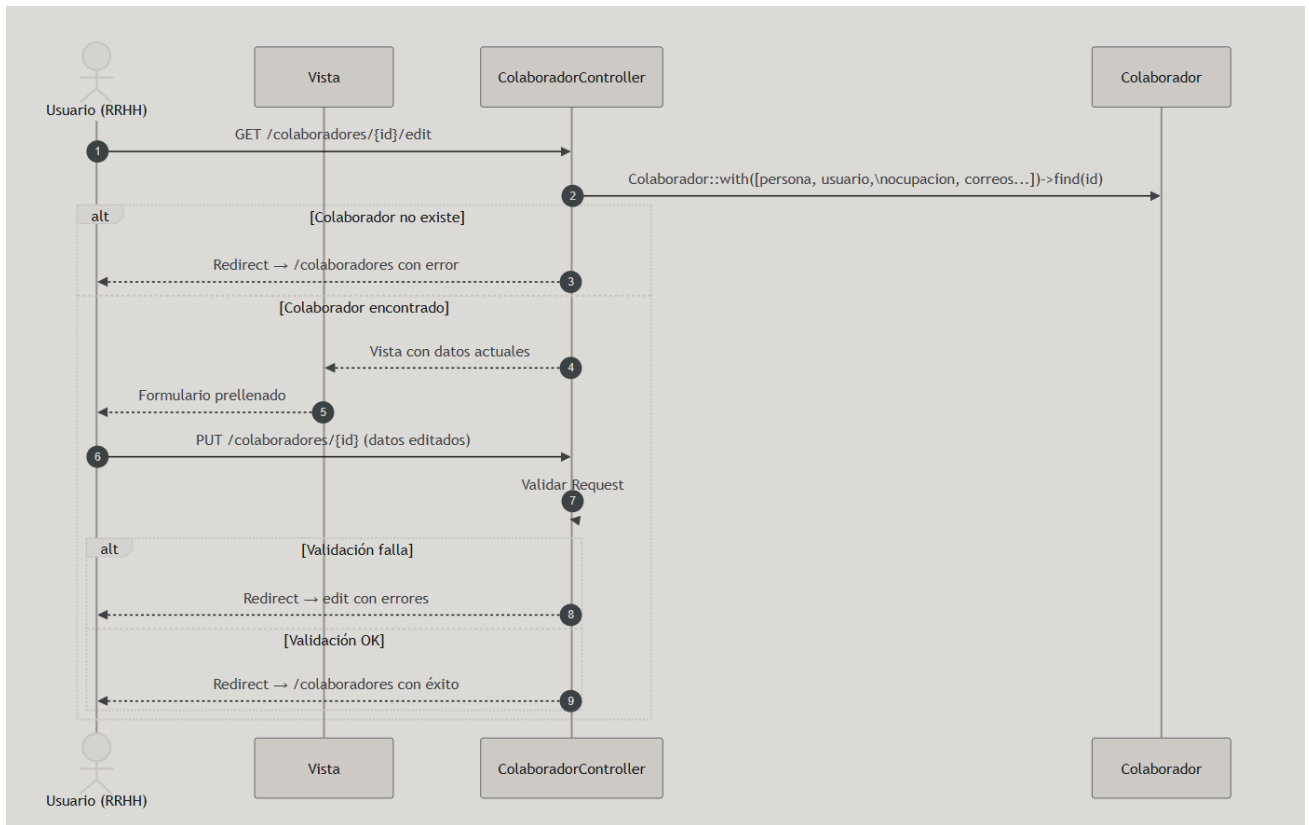
Autenticación muestra como el sistema verifica las credenciales de un usuario al iniciar sesión. Si los datos son correctos accede al sistema y si falla varias veces la cuenta se bloquea automáticamente como medida de seguridad.

**Figura 40***Diagrama de Secuencia - Crear Colaborador*

Fuente: Elaboración Propia (2026)

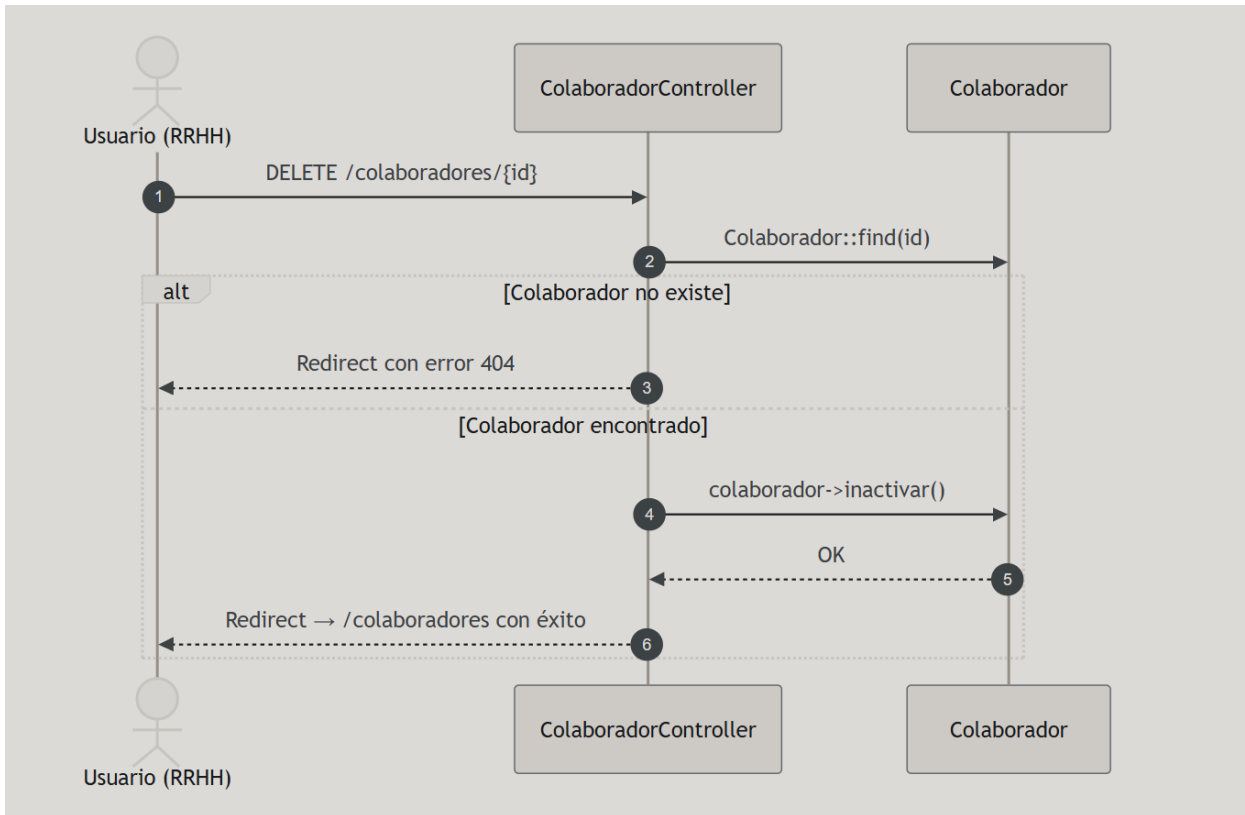
Registrar Colaborador describe los pasos que sigue el sistema para guardar un nuevo colaborador, creando primero sus datos personales, y finalmente su información laboral. Si algún dato está mal, el sistema lo indica y puede corrección antes de continuar.

Figura 41  
**Diagrama de Secuencia - Editar Colaborador**



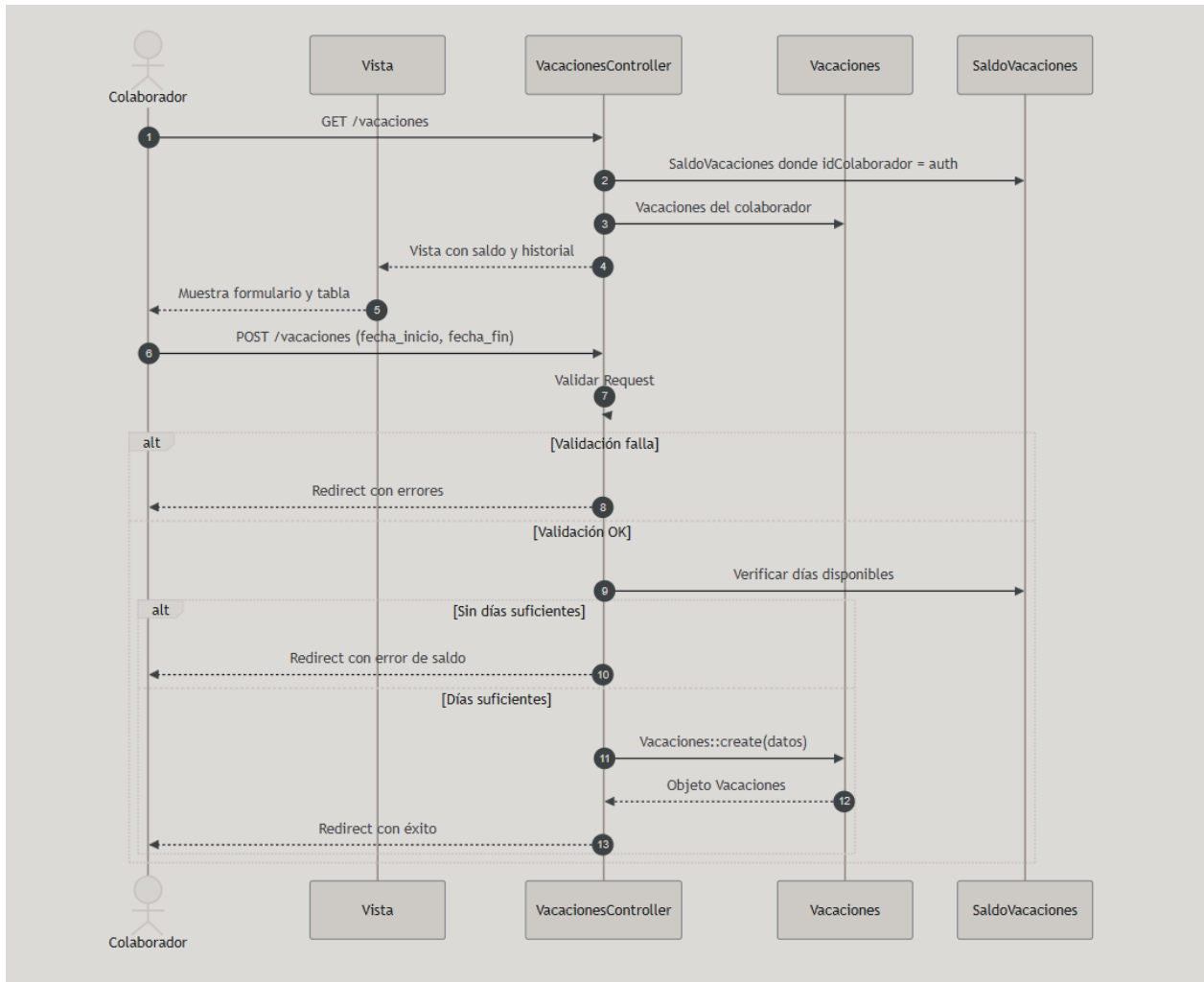
Fuente: Elaboración Propia (2026)

Editar colaborador explica cómo el sistema carga los datos actuales de un colaborador para puedan ser editados. Una vez realizados los cambios y confirmados, el sistema actualiza toda la información relacionada al mismo tiempo.

**Figura 42***Diagrama de Secuencia - Eliminar Colaborador*

Fuente: Elaboración Propia (2026)

Eliminar colaborador representa la desactivación de un colaborador en el sistema, sin borrar su historial. Si tiene registros pendientes como nóminas o vacaciones activas, el sistema impide la eliminación hasta que estén resueltos.

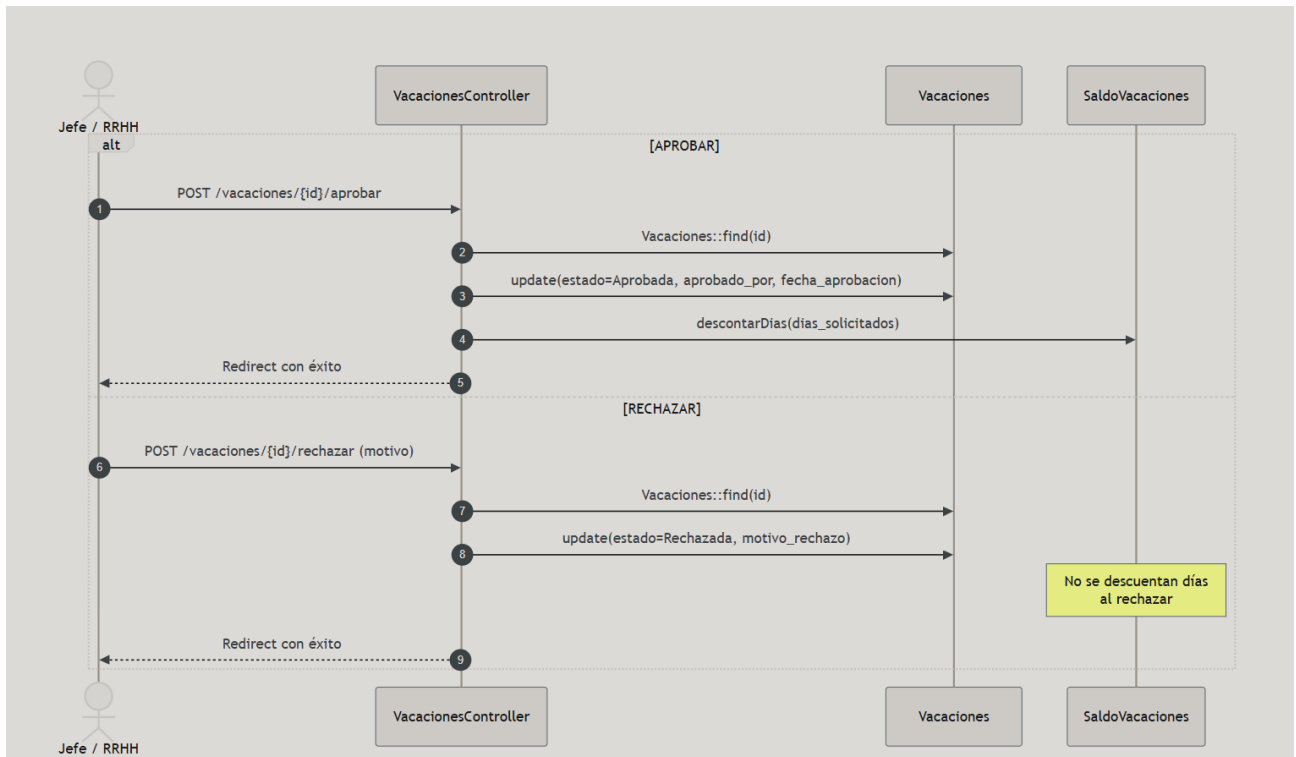
**Figura 43***Diagrama de Secuencia - Solicitar Vacaciones*

Fuente: Elaboración Propia (2026)

Solicitar Vacaciones muestra cómo un colaborador solicita días de vacaciones, verificando primero que tenga saldo disponible suficiente. Si todo está correcto, la solicitud se guarda como pendiente a la espera de aprobación por parte de su jefatura.

Figura 44

## Diagrama de Secuencia - Aprobar / Rechazar Vacaciones

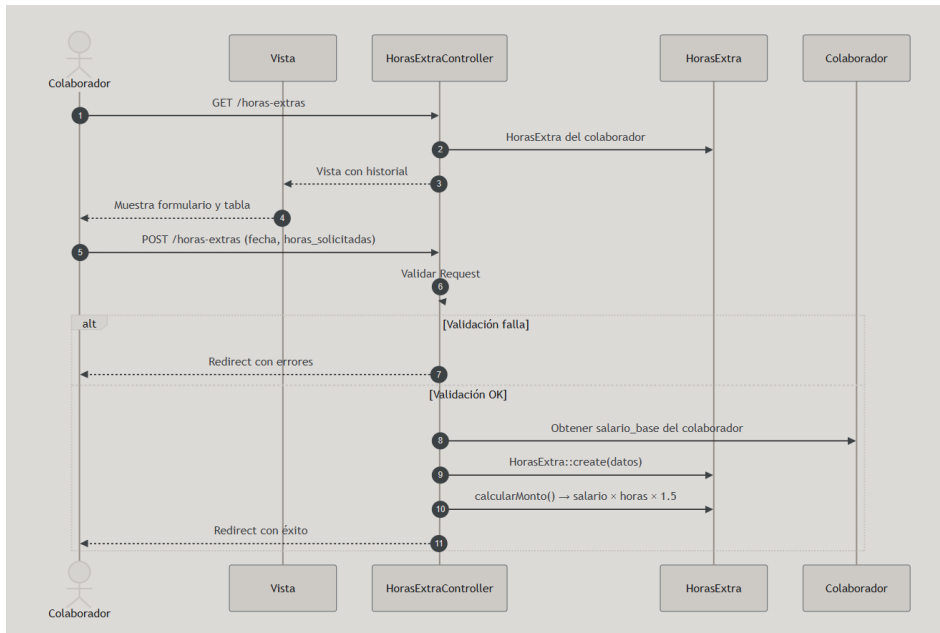


Fuente: Elaboración Propia (2026)

Aprobar o rechazar Vacaciones describe cómo la jefatura revisa una solicitud de vacaciones y decide aprobarla o rechazarla. Al aprobar, el sistema descuenta automáticamente los días del saldo del colaborador; al rechazar, el saldo no se ve afectado.

Figura 45

## Diagrama de Secuencia - Solicitar Horas Extra

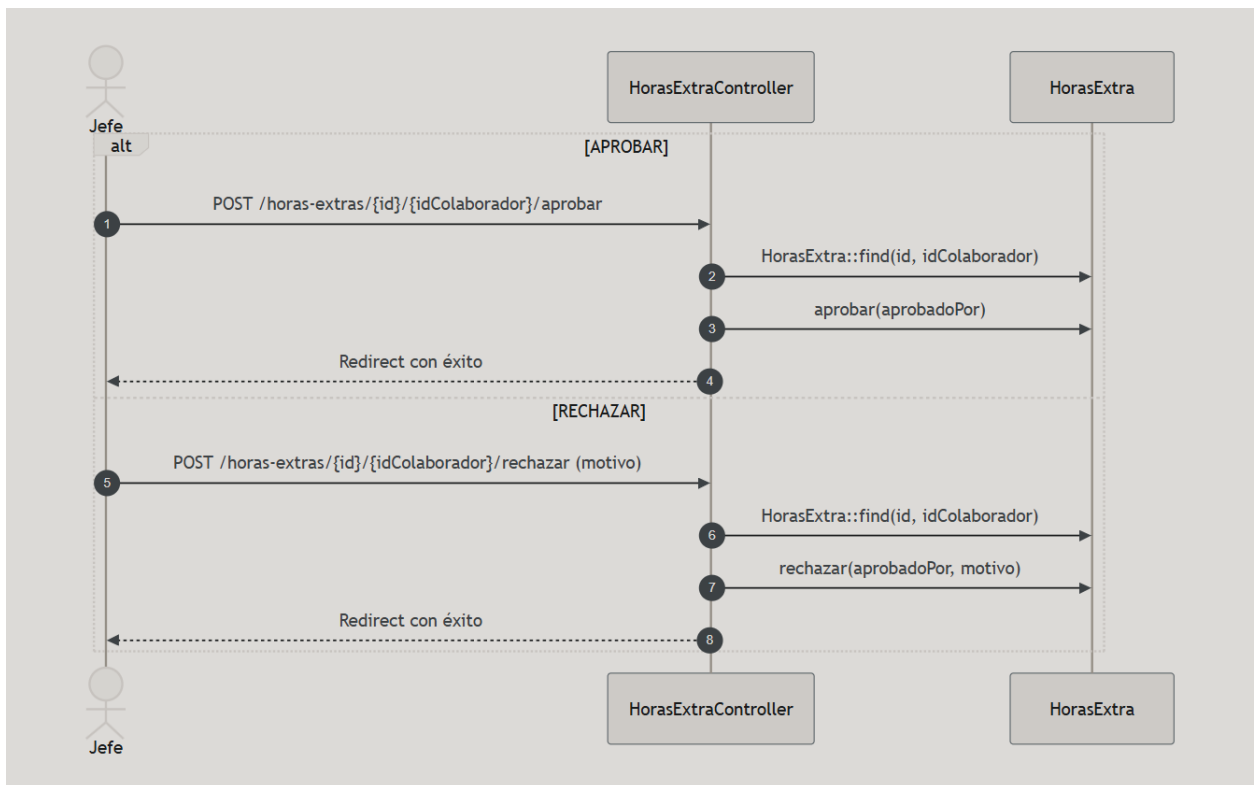


Fuente: Elaboración Propia (2026)

Solicitar Horas Extra explica cómo se registran las horas extra trabajadas por un colaborador, calculando automáticamente el monto a pagar según su salario base. El registro queda pendiente hasta que la jefatura lo apruebe o rechace.

Figura 46

Diagrama de Secuencia - Aprobar / Rechazar Horas Extra

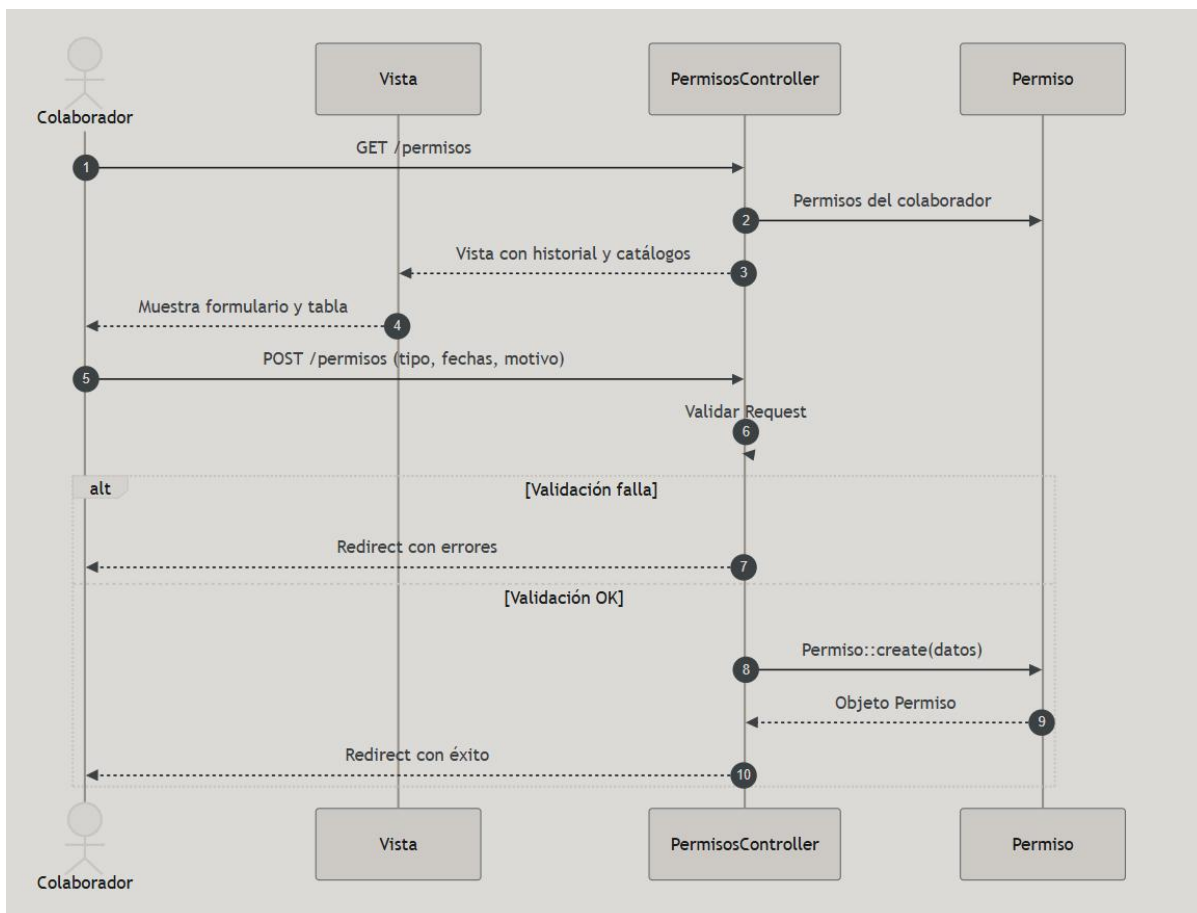


Fuente: Elaboración Propia (2026)

Aprobar o Rechazar Horas Extra muestra cómo la jefatura gestiona las solicitudes de horas extra pendientes. Al aprobar, el sistema registra quién autorizó y cuándo; al rechazar, guarda el motivo para dejar constancia de la decisión.

Figura 47

## Diagrama de Secuencia - Solicitar Permisos

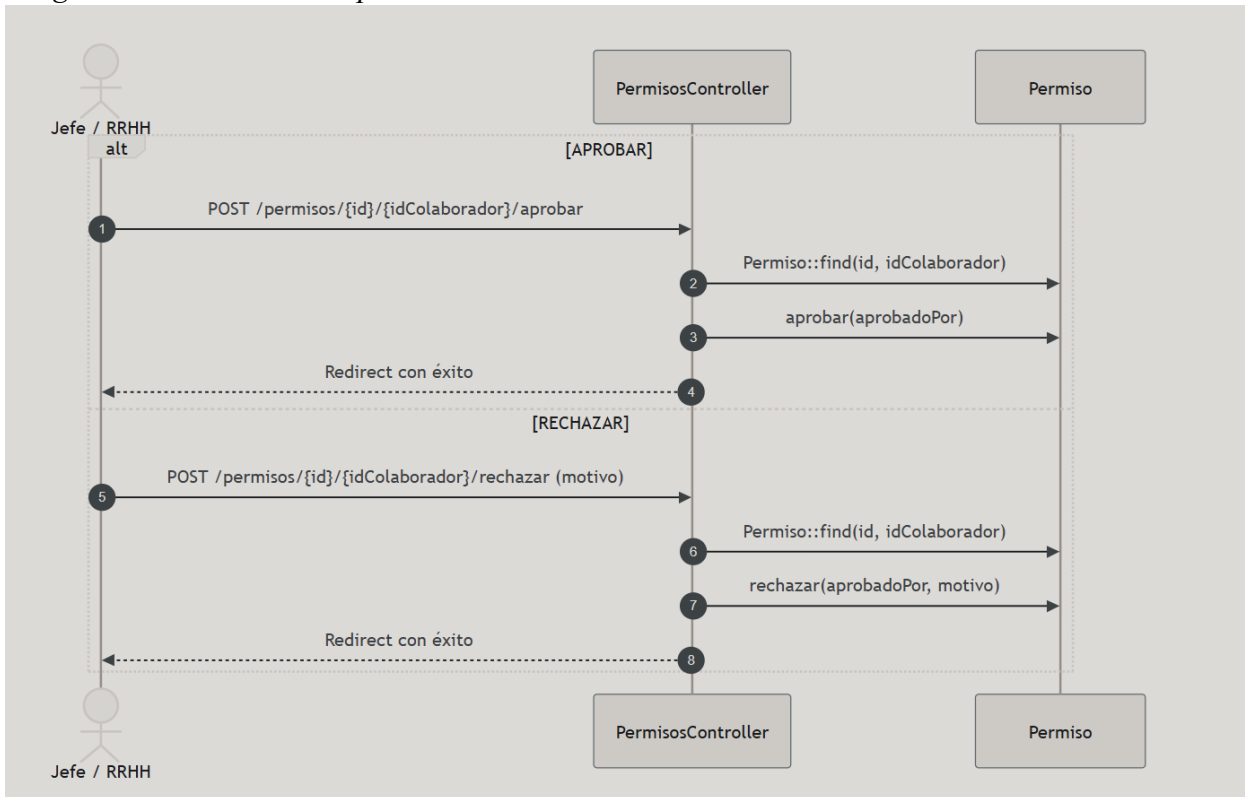


Fuente: Elaboración Propia (2026)

Solicitar permisos describe el proceso para registrar una solicitud de permiso laboral, indicando el tipo, las fechas y la justificación correspondiente. La solicitud queda en estado pendiente hasta que la jefatura tome una decisión

Figura 48

## Diagrama de Secuencia - Aprobar / Rechazar Permisos

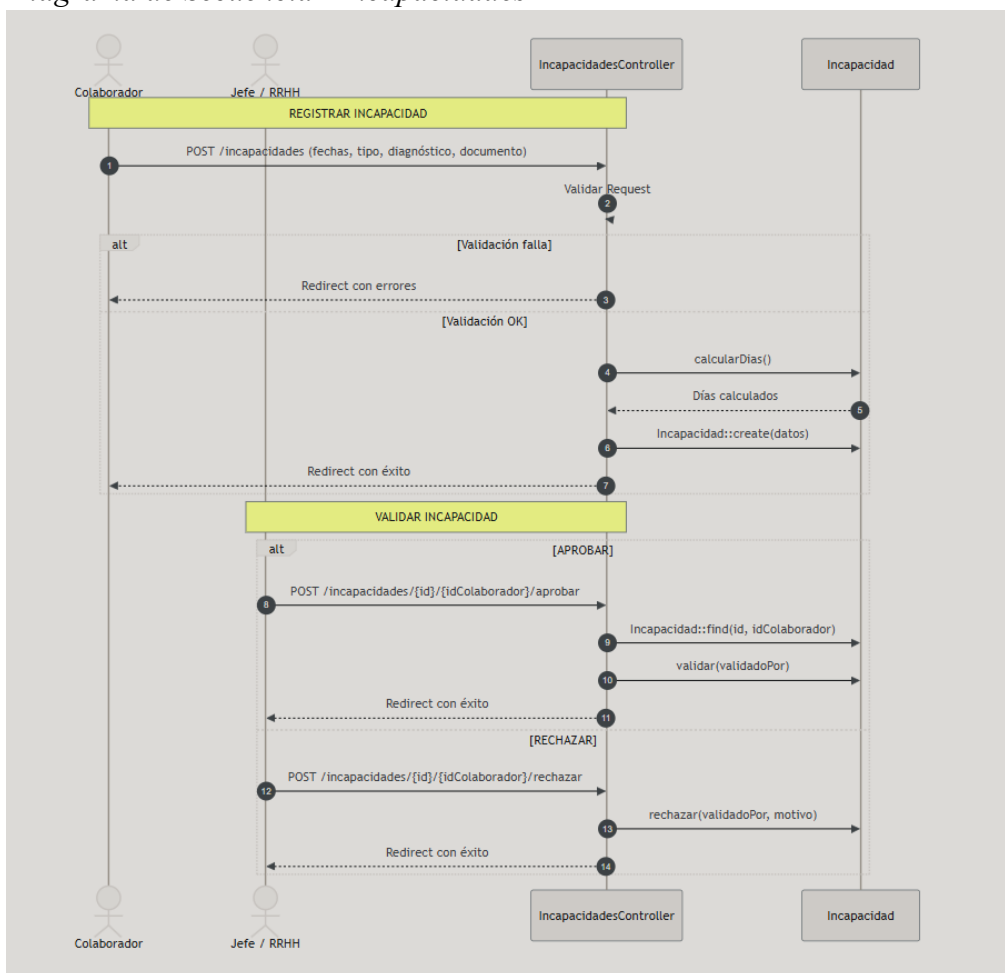


Fuente: Elaboración Propia (2026)

Aprobar o Rechazar permisos explica cómo la jefatura revisa y resuelve solicitudes de permiso. La aprobación o el rechazo quedan registrados con el responsable de la decisión y en caso de rechazo, también el motivo.

Figura 49

## Diagrama de Secuencia - Incapacidades

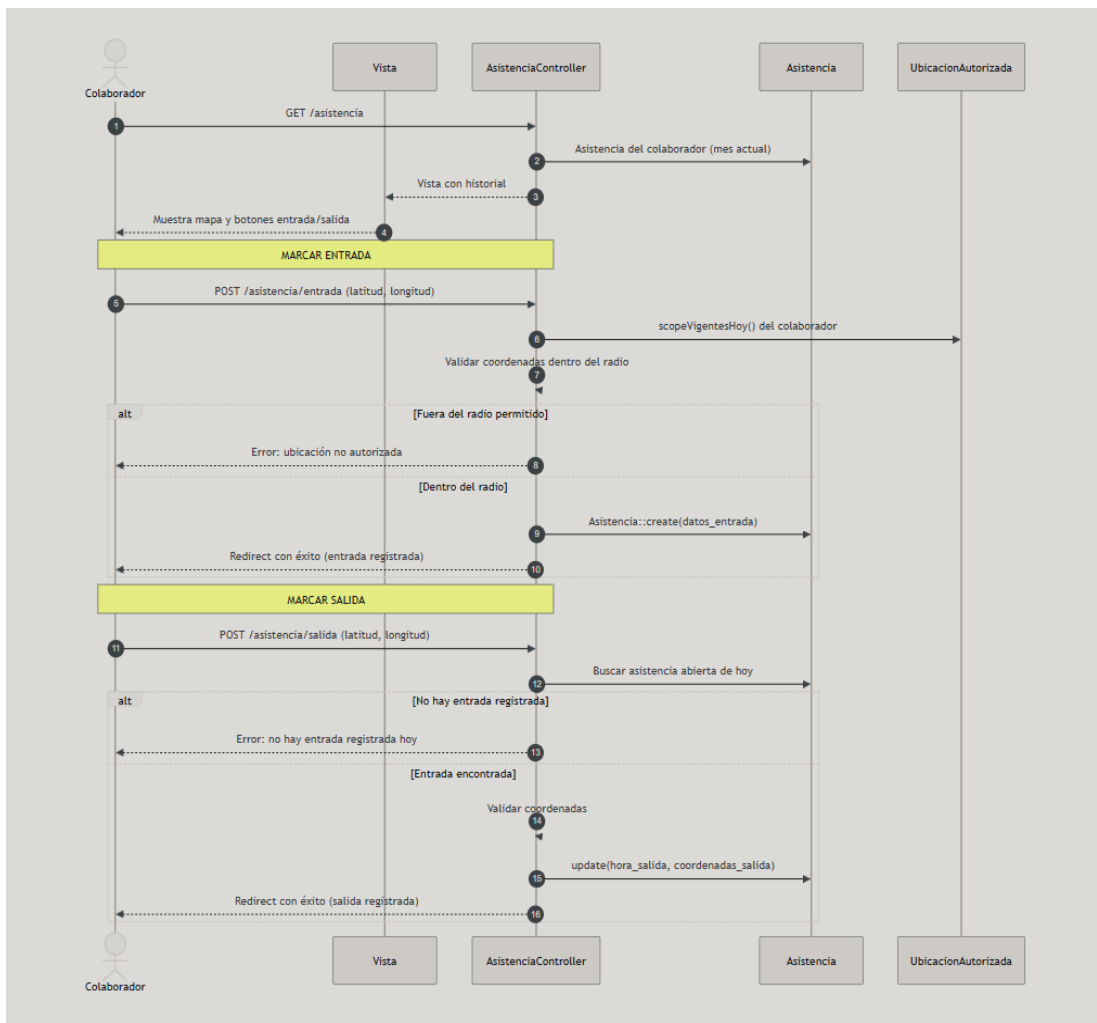


Fuente: Elaboración Propia (2026)

Registrar Asistencia (GPS) describe cómo un colaborador marca su entrada y salida usando la ubicación GPS del dispositivo. El sistema verifica que esté dentro de una zona autorizada antes de registrar la asistencia y calcula automáticamente las horas trabajadas al marcar la salida.

Figura 50

## Diagrama de Secuencia - Asistencia

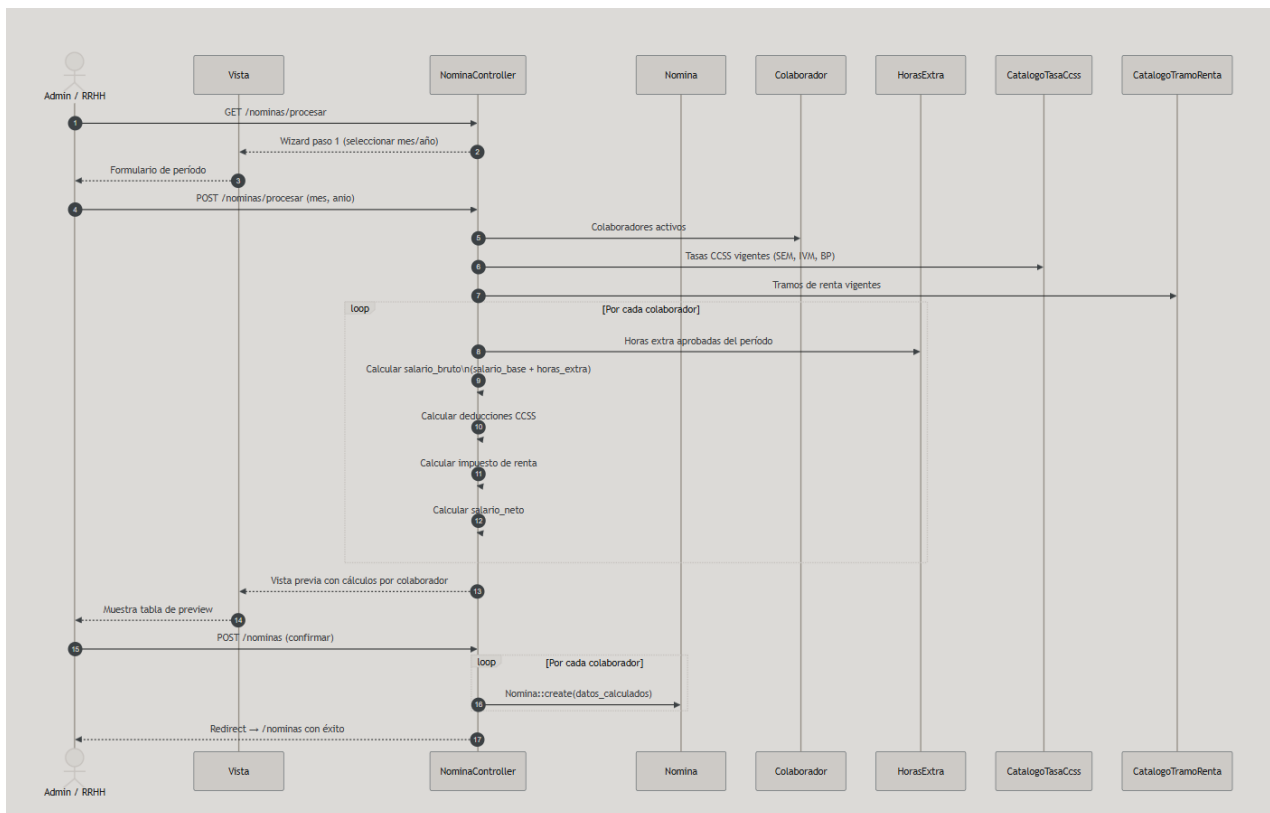


Fuente: Elaboración Propia (2026)

Registrar Asistencia (GPS) describe cómo un colaborador marca su entrada y salida usando la ubicación GPS del dispositivo. El sistema verifica que esté dentro de una zona autorizada antes de registrar la asistencia y calcula automáticamente las horas trabajadas al marcar salida.

Figura 51

## Diagrama de Secuencia - Procesar Nomina

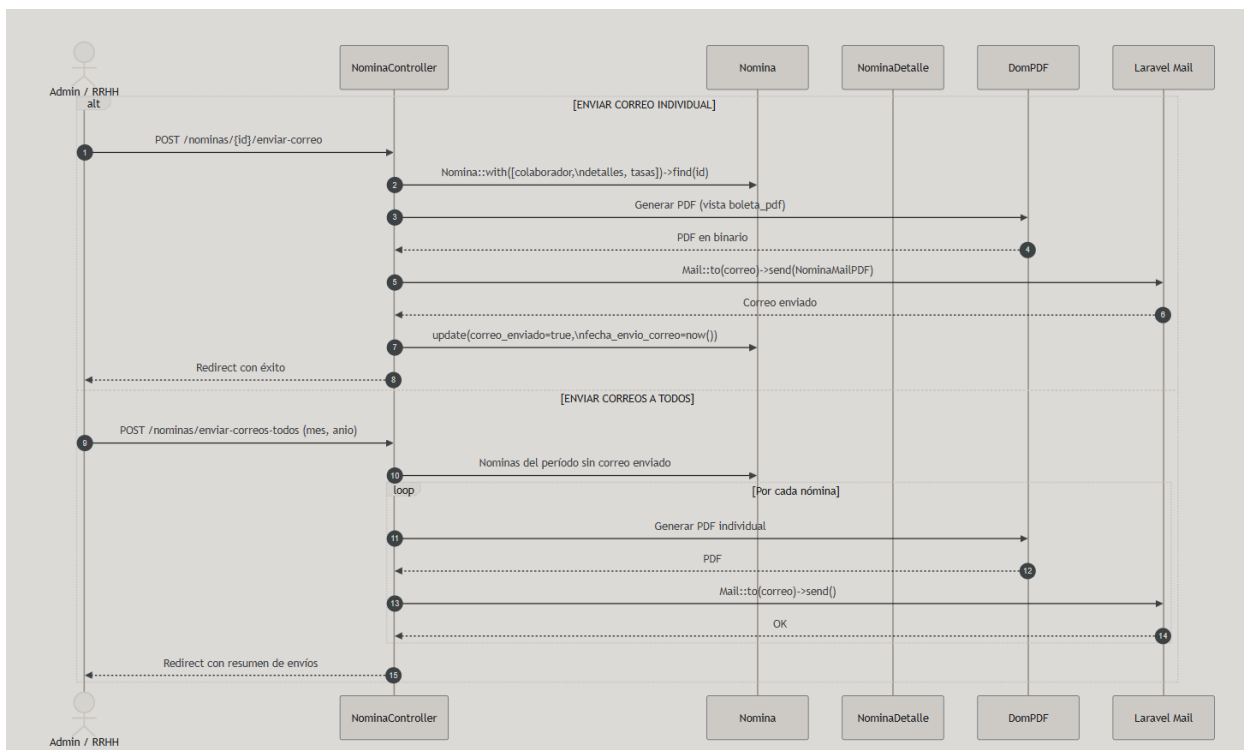


Fuente: Elaboración Propia (2026)

Procesar Nómina explica el proceso del cálculo de la planilla mensual, donde el sistema obtiene los datos de cada colaborador activo y calcula su salario considerando horas extra, deducciones de ley, e impuesto de renta. Antes de guardar, el administrador revisa un resumen y confirma el procesamiento.

Figura 52

## Diagrama de Secuencia - Correo Nomina

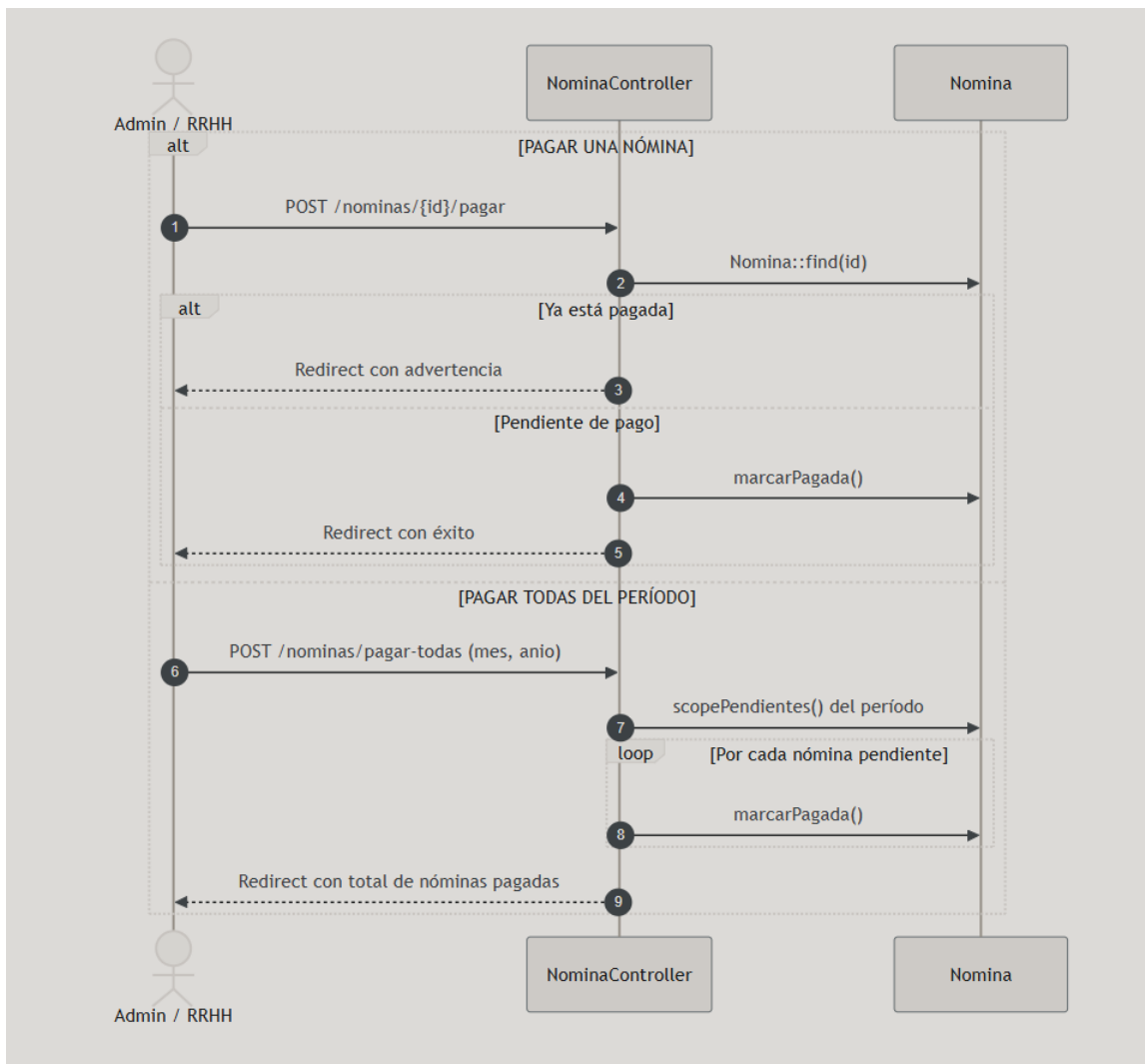


Fuente: Elaboración Propia (2026)

Correo Nómina muestra cómo el sistema genera la boleta de pago en PDF y la envía por correo electrónico al colaborador. Puede hacerse de forma individual o para todos los colaboradores de un periodo al mismo tiempo, registrando cada envío para evitar duplicados.

Figura 53

## Diagrama de Secuencia - Pagar Nomina

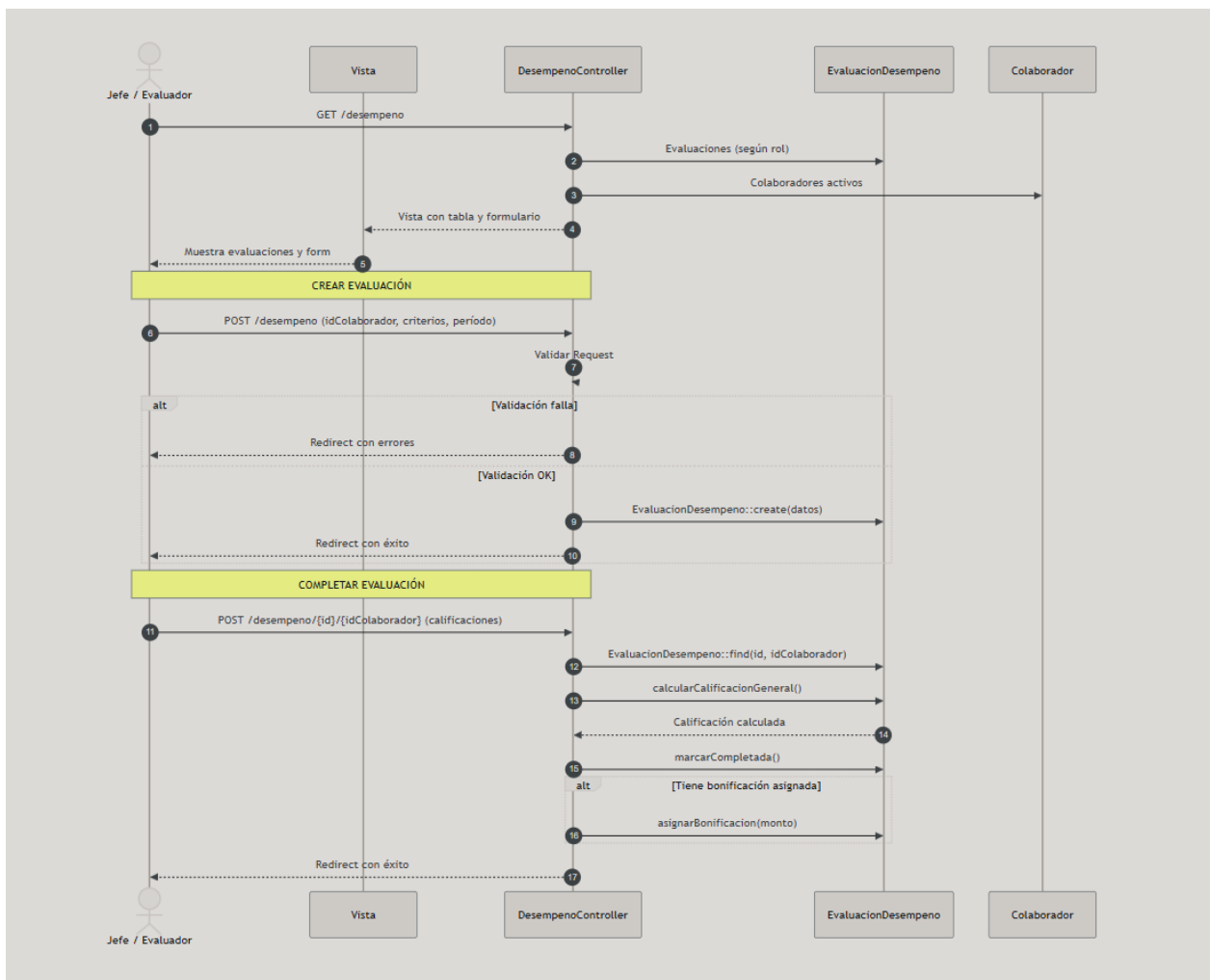


Fuente: Elaboración Propia (2026)

Pagar nomina describe cómo el administrador registra que una nómina ya fue pagada, ya sea de forma individual o para todo un período. El sistema valida que no haya sido marcada antes y registra la fecha del pago para efectos de auditoría.

Figura 54

## Diagrama de Secuencia - Evaluación Desempeño

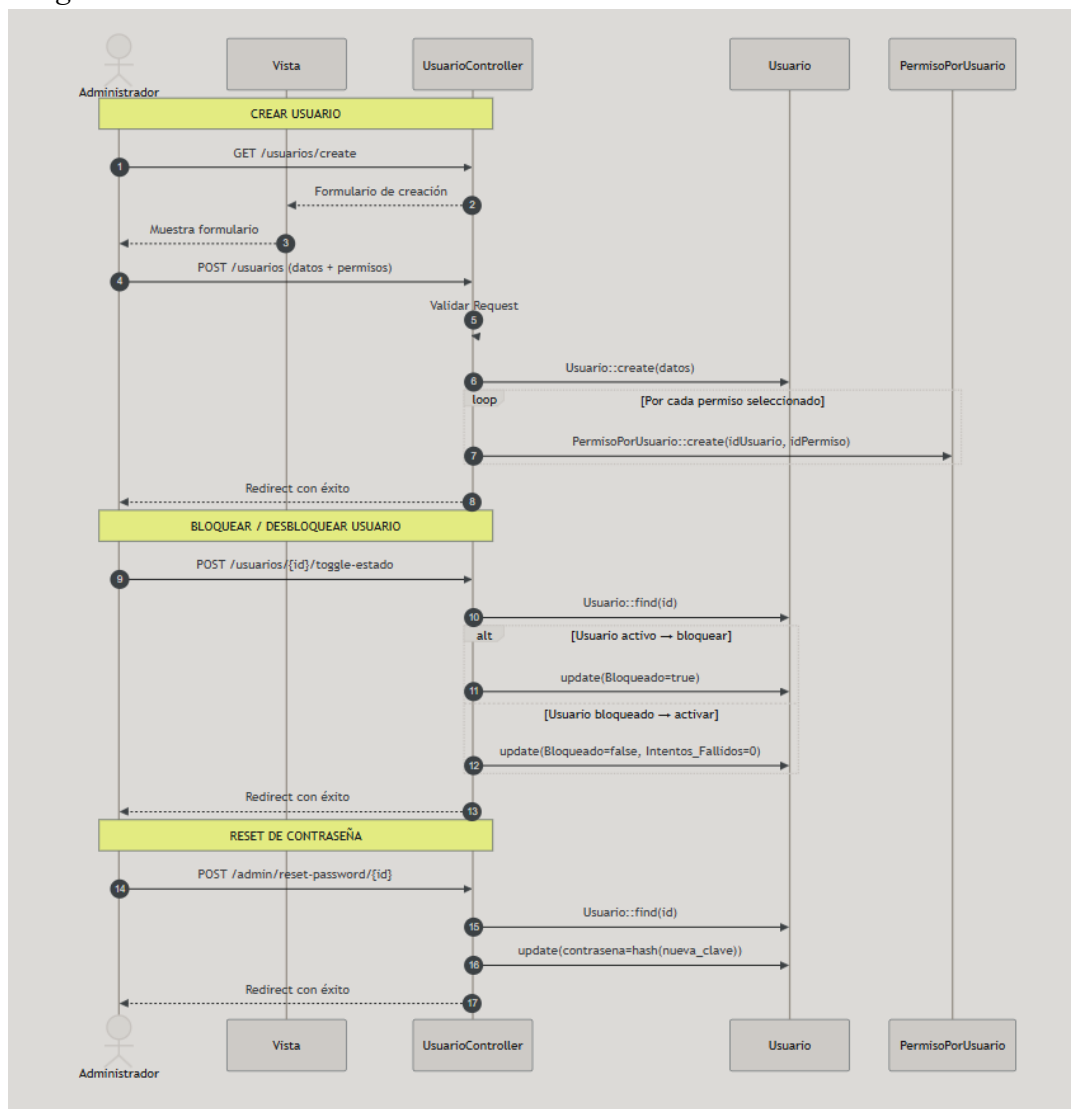


Fuente: Elaboración Propia (2026)

Evaluación Desempeño explica el ciclo de una evaluación: primero se crea asignando al colaborador y el período, y luego se complementa ingresando calificaciones y observaciones. Si el desempeño lo justifica, el sistema permite registrar además una bonificación económica asociada.

Figura 55

## Diagrama de Secuencia - Gestión de Usuarios

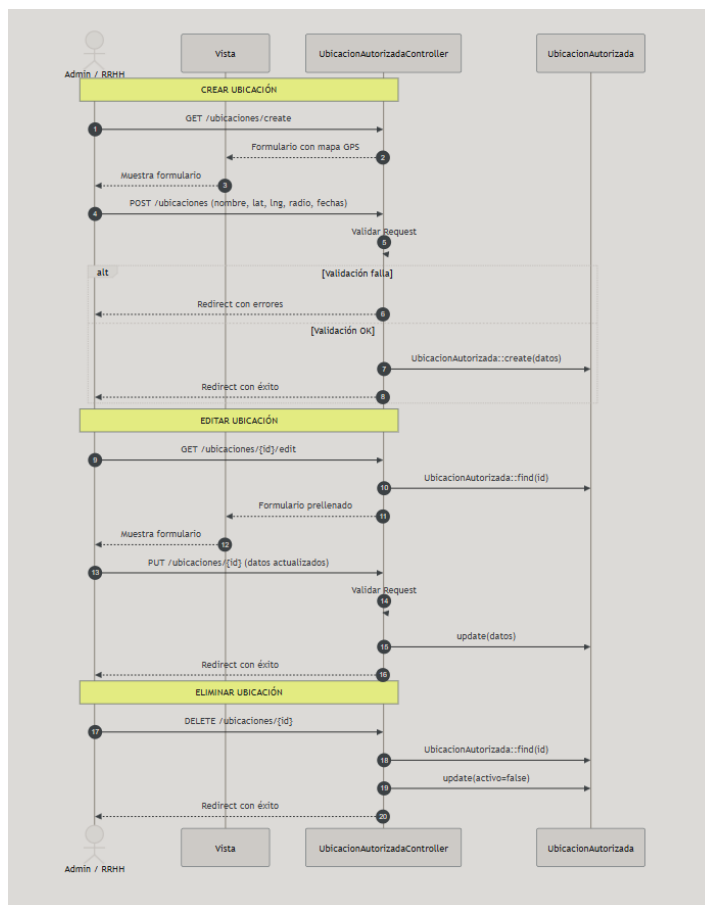


Fuente: Elaboración Propia (2026)

Gestión de Usuarios muestra tres acciones administrativas sobre los usuarios del sistema: crear uno nuevo con sus permisos, bloquearlo o desbloquearlo, y restablecer su contraseña. Todas las operaciones quedan registradas y los cambios aplican desde la próxima sesión del usuario.

Figura 56

## Diagrama de Secuencia - Ubicaciones Autorizadas

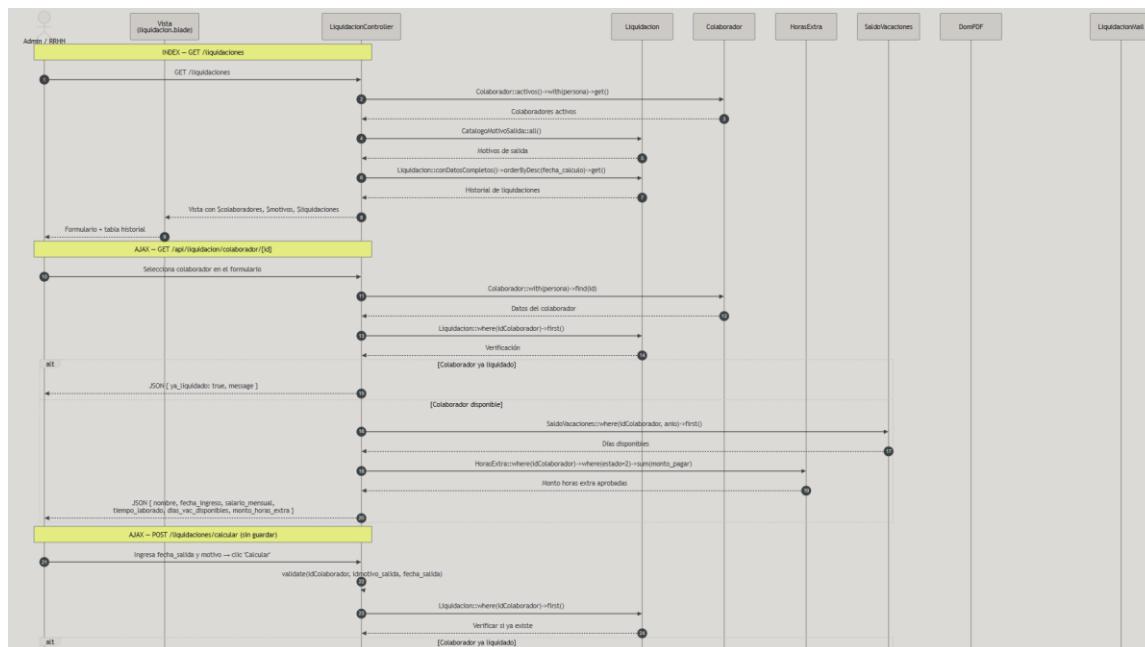


Fuente: Elaboración Propia (2026)

Ubicaciones Autorizadas describe cómo el administrador gestiona los puntos GPS desde los cuales los colaboradores pueden registrar su asistencia, definiendo coordenadas, radio de validación y período de vigencia. Las ubicaciones eliminadas se desactivan en lugar de borrarse, para conservar el historial de asistencias asociadas.

Figura 57

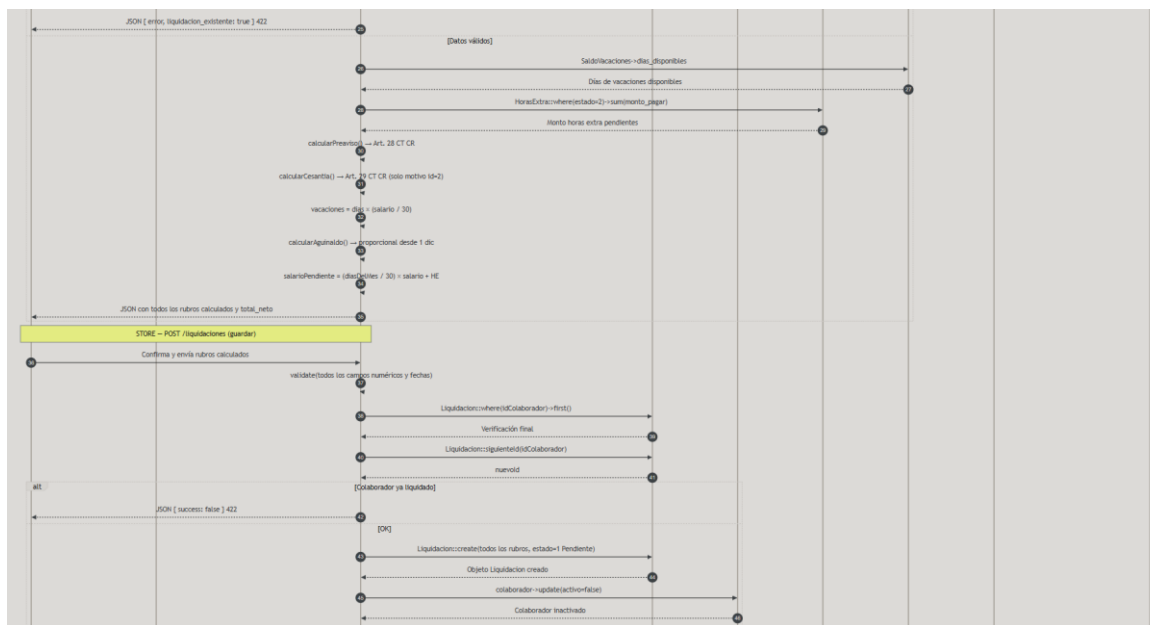
Diagrama de Secuencia - Procesar Liquidación



Fuente: Elaboración Propia (2026)

Figura 58

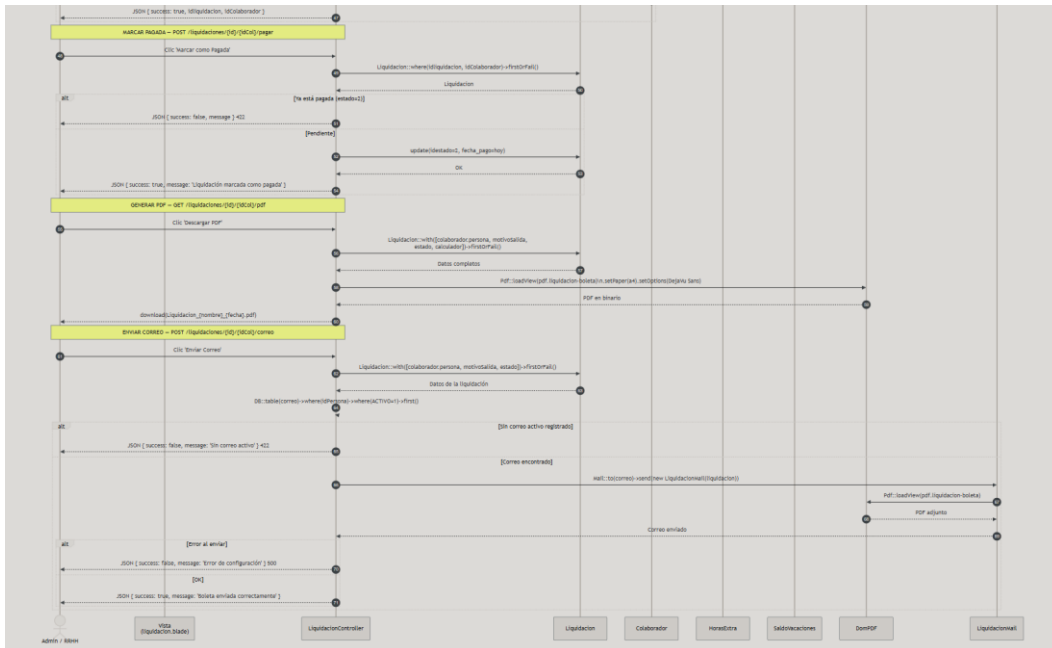
Diagrama de Secuencia - Procesar Liquidación



Fuente: Elaboración Propia (2026)

**Figura 59**

*Diagrama de Secuencia - Procesar Liquidación*

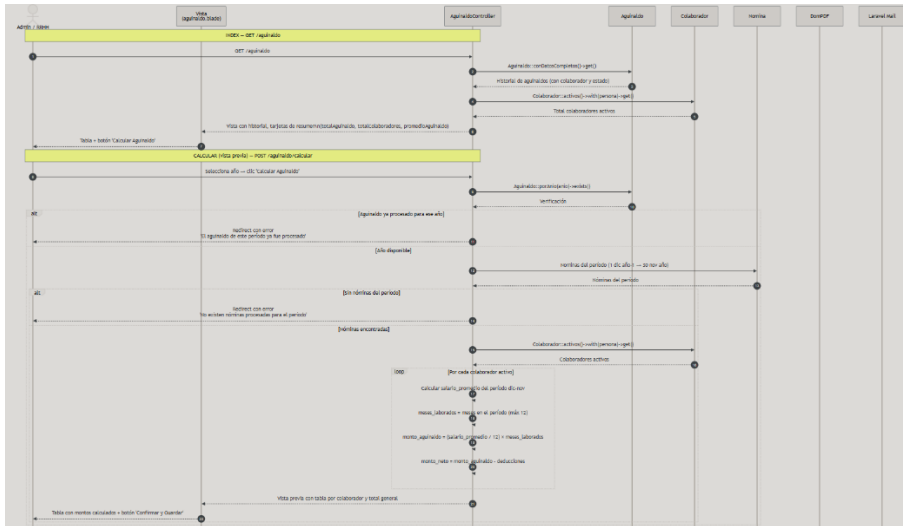


**Fuente: Elaboración Propia (2026)**

Procesar liquidación muestra el proceso completo de liquidación de un colaborador al terminar su relación laboral, calculando rubros como preaviso, cesantía, vacaciones no disfrutadas y aguinaldo proporcional. Al confirmar, el colaborador queda inactivo en el sistema y se genera su boleta de liquidación disponible para descarga y envío por correo.

**Figura 60**

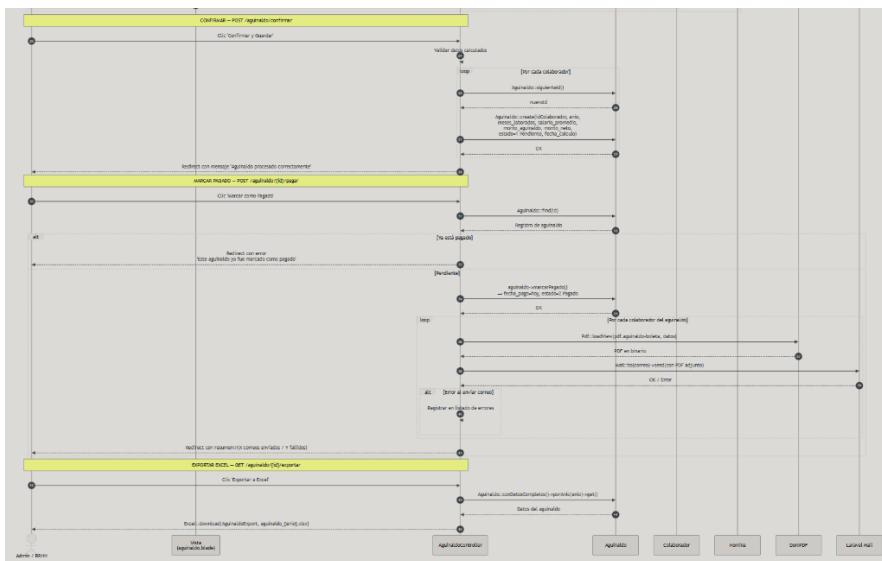
*Diagrama de Secuencia - Procesar Aguinaldo*



Fuente: Elaboración Propia (2026)

**Figura 61**

*Diagrama de Secuencia - Procesar Aguinaldo*



Fuente: Elaboración Propia (2026)

Procesar Aguinaldo explica el cálculo y pago del aguinaldo anual, donde el sistema determina el monto proporcional de cada colaborador según meses laborados en el período diciembre-noviembre. Al marcarlo como pagado, se generan y envían las boletas individuales por correo.

## Programación

En las siguientes tablas se muestra toda la parte programada en lo que son entradas, procesos, salidas y validaciones.

### Gestión Colaboradores

#### Figura 62

##### Lista de Colaboradores

```

public function index(Request $request)
{
    // Consulta base usando el scope personalizado del modelo
    $query = Colaborador::conDatosCompletos();

    // Filtro por nombre o apellidos si se recibió texto de búsqueda
    if ($request->filled('search')) {
        $search = strtolower($request->search);
        $query->whereHas('persona', function ($q) use ($search) {
            $q->whereRaw('LOWER(Nombre) LIKE ?', ["%{$search}%"])
                ->orWhereRaw('LOWER(Apellido1) LIKE ?', ["%{$search}%"])
                ->orWhereRaw('LOWER(Apellido2) LIKE ?', ["%{$search}%"]);
        });
    }

    // Filtro por estado activo/inactivo si se recibió el parámetro
    if ($request->filled('status') && $request->status !== 'all') {
        $request->status === 'Active' ? $query->activos() : $query->inactivos();
    }

    // Resultados paginados (15 por página)
    $colaboradores = $query->paginate(15);

    // Contadores para los indicadores de la vista
    $stats = [
        'total' => Colaborador::count(),
        'activos' => Colaborador::activos()->count(),
        'inactivos' => Colaborador::inactivos()->count(),
    ];

    return view('colaboradores', compact('colaboradores', 'stats'));
}

```

Fuente: Elaboración Propia (2026)

La imagen anterior es un método llamado `index` donde gestiona la lista de colaboradores del sistema, donde obtiene los colaboradores desde la base de datos donde carga sus datos completos y permite filtrarlos por nombre o apellidos mediante una búsqueda de texto. Además, calcula estadísticas generales que se muestran como indicadores en la pantalla.

**Figura 63***Formulario de Registro*

```
public function create()
{
    // Catálogos para los selectores del formulario
    $tiposIdentificacion = CatalogoTipoIdentificacion::where('activo', true)->orderBy('descripcion')->get();
    $generos = CatalogoGenero::where('Activo', true)->orderBy('Descripcion_Genero')->get();
    $países = CatalogoPais::where('activo', true)->orderBy('descripcion')->get();
    $ocupaciones = CatalogoOcupacion::where('Activo', true)->orderBy('Descripcion')->get();
    $jefes = Colaborador::with('persona')->where('activo', true)->get();
    $provincias = CatalogoProvincia::orderBy('Nombre_Provincia')->get();
    $tiposDireccion = CatalogoTipoDireccion::all();
    $tiposCorreo = CatalogoTipoCorreo::all();
    $tiposTelefono = CatalogoTipoTelefono::all();

    return view('colaboradores-agregar', compact(
        'tiposIdentificacion',
        'generos',
        'países',
        'ocupaciones',
        'jefes',
        'tiposCorreo',
        'tiposTelefono',
        'tiposDireccion',
        'provincias'
    ));
}
```

Fuente: Elaboración Propia (2026)

La imagen anterior es un método llamado `create` donde prepara un formulario para agregar un nuevo colaborador, cargando todos los catálogos necesarios desde la base de datos como tipos de identificación, géneros, países, ocupaciones, provincias, entre otros. Todos estos datos se envían a la vista, para poblar los selectores del formulario.

Figura 64

## Guardar Nuevo Colaborador

```

public function store(Request $request)
{
    // Validación de campos obligatorios
    $request->validate([
        'nombre' => 'required|string|max:45',
        'apellido1' => 'required|string|max:45',
        'apellido2' => 'required|string|max:45',
        'numero_identificacion' => 'required|string|max:20|unique:persona,numero_identificacion',
        'fecha_ingreso' => 'required|date',
        'salario_base' => 'required|numeric|min:0',
        'idCatalogo_Ocupacion' => 'required|exists:catalogo_ocupacion,idCatalogo_Ocupacion',
        'Jefatura_idColaborador' => 'required|exists:colaborador,idColaborador',
    ]);

    // Verificar sesión activa antes de operar
    if (!Auth::check()) {
        return back()->with('error', 'Sesion expirada o usuario no autenticado. No se puede guardar.');
```

```

    }

    DB::beginTransaction();

    try {
        // -----
        // BLOQUE 1: PERSONA
        // -----
        $nuevoIdPersona = (Persona::max('idPersona') ?? 0) + 1;

        $persona = new Persona();
        $persona->idPersona = $nuevoIdPersona;
        $persona->Nombre = $request->nombre;
        $persona->Apellido1 = $request->apellido1;
        $persona->Apellido2 = $request->apellido2;
        $persona->numero_identificacion = $request->numero_identificacion;
        $persona->Fecha_Nacimiento = $request->fecha_ingreso;
        $persona->idCatalogo_tipo_identificacion = $request->idCatalogo_tipo_identificacion ?? 1;
        $persona->idCatalogo_Genero = $request->idCatalogo_Genero ?? 1;
        $persona->idCatalogo_pais = $request->idCatalogo_pais ?? 1;
        $persona->save();

        // -----
        // BLOQUE 2: COLABORADOR
        // -----
        $nuevoIdColaborador = (Colaborador::max('idColaborador') ?? 0) + 1;

        $colaborador = new Colaborador();
        $colaborador->idColaborador = $nuevoIdColaborador;
        $colaborador->Persona_idPersona = $persona->idPersona;
        $colaborador->idUsuario = Auth::id();
        $colaborador->Fecha_Ingreso = $request->fecha_ingreso;
        $colaborador->salario_base = $request->salario_base;
        $colaborador->idCatalogo_Ocupacion = $request->idCatalogo_Ocupacion;
        $colaborador->Jefatura_idColaborador = $request->Jefatura_idColaborador;
        $colaborador->activo = 1;
        $colaborador->save();
    }
}

```

Fuente: Elaboración Propia (2026)

En la figura anterior se muestra un método llamado store donde gestiona el registro de un nuevo colaborador en el sistema, validando primero que todos los campos obligatorios sean correctos y que la sesión del usuario esté activa. Luego, dentro de una transacción en la base de datos, crea dos registros relacionados: primero la persona con los datos personales y después el colaborador vinculado a esa persona, asignándole su ocupación, jefatura y usuario autentico.

Figura 65

## Guardar Nuevo Colaborador

```

// BLOQUE 3: CORREO (solo si fue proporcionado)
if ($request->filled('correo')) {
    $correo = new Correo();
    $correo->Direccion_Correo = $request->correo;
    $correo->idPersona = $persona->idPersona;
    $correo->idCatalogo_Tipo_Correo = $request->idCatalogo_Tipo_Correo ?? 1;
    $correo->Fecha_Ingreso = now();
    $correo->ACTIVO = 1;
    $correo->save();
}

// BLOQUE 4: TELÉFONO (solo si fue proporcionado)
if ($request->filled('telefono')) {
    $telefono = new Telefono();
    $telefono->Numero_Telefono = preg_replace('/[^0-9]/', '', $request->telefono);
    $telefono->idPersona = $persona->idPersona;
    $telefono->idCatalogo_Tipo_Telefono = $request->idCatalogo_Tipo_Telefono ?? 1;
    $telefono->Fecha_Ingreso = now();
    $telefono->Activo = 1;
    $telefono->save();
}

// BLOQUE 5: DIRECCIÓN (solo si se seleccionó un distrito)
if ($request->filled('idCatalogo_Distrito')) {
    $nuevoIdDireccion = (Direccion::max('idDireccion') ?? 0) + 1;

    $direccion = new Direccion();
    $direccion->idDireccion = $nuevoIdDireccion;
    $direccion->Persona_idPersona = $persona->idPersona;
    $direccion->idCatalogo_Provincia = $request->idCatalogo_Provincia;
    $direccion->idCatalogo_Canton = $request->idCatalogo_Canton;
    $direccion->idCatalogo_Distrito = $request->idCatalogo_Distrito;
    $direccion->idCatalogo_Tipo_Direccion = $request->idCatalogo_Tipo_Direccion ?? 1;
    $direccion->Detalle = $request->direccion_detalle ?? '';
    $direccion->save();
}

DB::commit();

return redirect()
    ->route('colaboradores.index')
    ->with('success', 'Colaborador guardado exitosamente');
} catch (\Exception $e) {
    DB::rollback();

    Log::error('Fallo en store de Colaborador: ' . $e->getMessage());
    Log::error('Trace: ' . $e->getTraceAsString());

    return back()
        ->withInput()
        ->with('error', 'Error al guardar los datos en la base de datos: ' . $e->getMessage());
}
}

```

Fuente: Elaboración Propia (2026)

Esta es la continuación del método store, donde de forma opcional registra el correo, teléfono y dirección del colaborador solo si fueron proporcionados en el formulario. Una vez guardados todos los datos, confirma la transacción en la base de datos y redirige al listado de colaboradores con un mensaje de éxito. En caso de cualquier error, revierte todos los cambios con un rollback y devuelve al usuario con un mensaje de error.

## Figura 66

### Detalle Colaborador

```
public function show($id)
{
    // Carga el colaborador con todas sus relaciones anidadas
    $colaborador = Colaborador::with([
        'persona.genero',
        'persona.tipoIdentificacion',
        'persona.pais',
        'persona.correos.tipoCorreo',
        'persona.telefonos.tipoTelefono',
        'persona.direcciones.provincia',
        'persona.direcciones.canton',
        'persona.direcciones.districto',
        'persona.direcciones.tipoDireccion',
        'ocupacion',
        'jefe.persona',
        'subordinados.persona',
        'usuario'
    ]->findOrFail($id);

    return view('colaboradores-detalle', compact('colaborador'));
}
```

Fuente: Elaboración Propia (2026)

La imagen anterior es un método llamado show donde carga el detalle completo de un colaborador específico, trayendo de una sola vez todas sus relaciones anidadas como datos personales, correos, teléfonos, dirección, ocupación, jefatura y subordinados. Si el colaborador no existe, lanza un error automáticamente. Luego envía toda esa información a la vista colaboradores detalle para su presentación.

Figura 67

## Actualizar Colaborador

```

public function update(Request $request, $id)
{
    // Validación de campos obligatorios
    $request->validate([
        'Nombre' => 'required|string|max:45',
        'Apellido1' => 'required|string|max:45',
        'Fecha_Nacimiento' => 'required|date',
        'idCatalogo_Genero' => 'required|exists:catalogo_genero,idCatalogo_Genero',
        'idcatalogo_pais' => 'required|exists:catalogo_pais,idcatalogo_pais',
        'idcatalogo_tipo_identificacion' => 'required|exists:catalogo_tipo_identificacion,idcatalogo_tipo_identificacion',
        'salario_base' => 'required|numeric|min:0',
        'idCatalogo_Ocupacion' => 'required|exists:catalogo_ocupacion,idCatalogo_Ocupacion',
        'fecha_ingreso' => 'required|date',
    ]);

    // Carga el colaborador con sus relaciones de contacto
    $colaborador = Colaborador::with(['persona.direcciones', 'persona.correos', 'persona.telefonos'])
        ->findOrFail($id);

    DB::beginTransaction();

    try {
        // =====
        // 1. ACTUALIZAR PERSONA
        // =====
        $colaborador->persona->update([
            'Nombre' => $request->Nombre,
            'Apellido1' => $request->Apellido1,
            'Apellido2' => $request->Apellido2,
            'Fecha_Nacimiento' => $request->Fecha_Nacimiento,
            'idCatalogo_Genero' => $request->idCatalogo_Genero,
            'idcatalogo_pais' => $request->idcatalogo_pais,
            'idcatalogo_tipo_identificacion' => $request->idcatalogo_tipo_identificacion,
        ]);

        // =====
        // 2. ACTUALIZAR COLABORADOR
        // =====
        $colaborador->update([
            'salario_base' => $request->salario_base,
            'idCatalogo_Ocupacion' => $request->idCatalogo_Ocupacion,
            'Jefatura_idColaborador' => $request->Jefatura_idColaborador ? $id,
            'Fecha_Ingreso' => $request->fecha_ingreso,
            'activo' => $request->has('activo') ? 1 : 0,
        ]);
    }
}

```

Fuente: Elaboración Propia (2026)

En la figura anterior es un método llamado update donde gestión donde gestiona la actualización de los datos de un colaborador existente, validando primero que todos los campos obligatorios sean correctos. Luego, dentro de una transacción, actualiza en dos pasos: primero los datos personales de la persona como nombre, apellidos y fecha de nacimiento, y después los datos propios del colaborador como salario, ocupación, jefatura y estado activo.

Figura 68

## Actualizar Colaborador

```

// =====
// 3. ACTUALIZAR O CREAR CORREO
// Se usa query builder directo por la llave primaria compuesta de la tabla.
// =====
if ($request->filled('correo')) {
    $idPersona = $colaborador->persona->idPersona;

    $correoExistente = DB::table('correo')
        ->where('idPersona', $idPersona)
        ->first();

    if ($correoExistente) {
        DB::table('correo')
            ->where('Direccion_Correo', $correoExistente->Direccion_Correo)
            ->where('idPersona', $idPersona)
            ->update([
                'Direccion_Correo' => $request->correo,
                'idCatalogo_Tipo_Correo' => $request->idCatalogo_Tipo_Correo ?? 1,
            ]);

        Log::info('Correo actualizado', [
            'anterior' => $correoExistente->Direccion_Correo,
            'nuevo' => $request->correo
        ]);
    } else {
        DB::table('correo')->insert([
            'Direccion_Correo' => $request->correo,
            'idPersona' => $idPersona,
            'idCatalogo_Tipo_Correo' => $request->idCatalogo_Tipo_Correo ?? 1,
            'Fecha_Ingreso' => now(),
            'ACTIVO' => 1,
        ]);

        Log::info('Correo creado', ['correo' => $request->correo]);
    }
}

```

Fuente: Elaboración Propia (2026)

Esta es una continuación del método update, donde gestiona el correo del colaborador de forma inteligente: si ya existe un correo registrado lo actualiza con el nuevo valor, y si no existe lo crea desde cero. En ambos casos registra el resultado en el log del sistema para mantener trazabilidad de los cambios realizados.

Figura 69

## Actualizar Colaborador

```

// =====
// 4. ACTUALIZAR O CREAR TELÉFONO
// Se usa query builder directo por la llave primaria compuesta de la tabla.
// =====
if ($request->filled('telefono')) {
    $idPersona = $colaborador->persona->idPersona;
    $numeroLimpio = preg_replace('/[0-9]/', '', $request->telefono);

    $telefonoExistente = DB::table('telefono')
        ->where('idPersona', $idPersona)
        ->first();

    if ($telefonoExistente) {
        DB::table('telefono')
            ->where('Numero_Telefono', $telefonoExistente->Numero_Telefono)
            ->where('idPersona', $idPersona)
            ->update([
                'Numero_Telefono' => $numeroLimpio,
                'idCatalogo_Tipo_Telefono' => $request->idCatalogo_Tipo_Telefono ?? 1,
            ]);

        Log::info('Teléfono actualizado', [
            'anterior' => $telefonoExistente->Numero_Telefono,
            'nuevo' => $numeroLimpio
        ]);
    } else {
        DB::table('telefono')->insert([
            'Numero_Telefono' => $numeroLimpio,
            'idPersona' => $idPersona,
            'idCatalogo_Tipo_Telefono' => $request->idCatalogo_Tipo_Telefono ?? 1,
            'Fecha_Ingreso' => now(),
            'Activo' => 1,
        ]);

        Log::info('Teléfono creado', ['numero' => $numeroLimpio]);
    }
}

```

Fuente: Elaboración Propia (2026)

Esta es una continuación del método update, donde gestiona el teléfono del colaborador de forma similar al correo: primero limpia el número eliminando cualquier carácter que no sea dígito, luego verifica si ya existe un teléfono registrado para actualizarlo, o lo crea si no existe. En ambos casos registra el resultado en el log del sistema para mantener trazabilidad de los cambios.

**Figura 70**

### *Actualizar Colaborador*

```

// =====
// 3. Actualiza o crea dirección
// Se usa query builder directo por la llave primaria compuesta de la tabla.
// =====
if ($request->filled('idCatalogo_Distrito')) {
    $idPersona = $colaborador->persona->idPersona;

    $direccionExistente = DB::table('direccion')
        ->where('Persona_idPersona', $idPersona)
        ->first();

    if ($direccionExistente) {
        DB::table('direccion')
            ->where('idDireccion', $direccionExistente->idDireccion)
            ->where('Persona_idPersona', $idPersona)
            ->update([
                'idCatalogo_Provincia' => $request->idCatalogo_Provincia,
                'idCatalogo_Canton' => $request->idCatalogo_Canton,
                'idCatalogo_Distrito' => $request->idCatalogo_Distrito,
                'idCatalogo_Tipo_Direccion' => $request->idCatalogo_Tipo_Direccion ?? 1,
                'Detalle' => $request->direccion_detalle ?? ''
            ]);

        Log::info('Dirección actualizada', [
            'idDireccion' => $direccionExistente->idDireccion
        ]);
    } else {
        $nuevoIdDireccion = DB::table('direccion')->insert([
            'idDireccion' => $nuevoIdDireccion,
            'Persona_idPersona' => $idPersona,
            'idCatalogo_Provincia' => $request->idCatalogo_Provincia,
            'idCatalogo_Canton' => $request->idCatalogo_Canton,
            'idCatalogo_Distrito' => $request->idCatalogo_Distrito,
            'idCatalogo_Tipo_Direccion' => $request->idCatalogo_Tipo_Direccion ?? 1,
            'Detalle' => $request->direccion_detalle ?? ''
        ]);

        Log::info('Dirección creada', ['idDireccion' => $nuevoIdDireccion]);
    }

    DB::commit();

    return redirect()
        ->route('colaboradores.index')
        ->with('success', 'Colaborador actualizado exitosamente');
} catch (\Exception $e) {
    DB::rollback();

    Log::error('Error al actualizar colaborador', [
        'mensaje' => $e->getMessage(),
        'línea' => $e->getLine(),
        'trace' => $e->getTraceAsString()
    ]);

    return back()
        ->withInput()
        ->with('error', 'Error al actualizar: ' . $e->getMessage());
}
}

```

Fuente: Elaboración Propia (2026)

Esta es la continuación y cierre del método update, donde gestiona la dirección del colaborador siguiendo la misma lógica: si ya existe una dirección registrada la actualización con provincia, cantón, distrito y detalle, y si no existe la crea. Una vez guardados todos los cambios confirma la transacción y redirige al listado con un mensaje de éxito. En caso de error, revierte todos los cambios con un rollback y devuelve al usuario con un mensaje de error.

## Figura 71

### Borrado Lógico y AJAX

```
/**
 * Desactiva un colaborador (baja lógica, no elimina el registro).
 */
public function destroy($id)
{
    try {
        $colaborador = Colaborador::findOrFail($id);
        $colaborador->update(['activo' => 0]);

        // Respuesta JSON para integrarse con SweetAlert en el frontend
        return response()->json(['success' => true]);
    } catch (\Exception $e) {
        return response()->json(['success' => false, 'message' => $e->getMessage()], 500);
    }
}

/**
 * Devuelve los cantones de una provincia en formato JSON (llamada AJAX).
 */
public function getCantones($idProvincia)
{
    $data = DB::table('catalogo_canton')
        ->where('idCatalogo_Provincia', $idProvincia)
        ->orderBy('Nombre_Canton')
        ->get();

    return response()->json($data);
}

/**
 * Devuelve los distritos de un cantón en formato JSON (llamada AJAX).
 */
public function getDistritos($idProvincia, $idCanton)
{
    $data = DB::table('catalogo_distrito')
        ->where('idCatalogo_Provincia', $idProvincia)
        ->where('idCatalogo_Canton', $idCanton)
        ->orderBy('Nombre_Distrito')
        ->get();

    return response()->json($data);
}
```

Fuente: Elaboración Propia (2026)

La figura anterior muestra tres métodos adicionales del controlador. El método `destroy` realiza una baja lógica del colaborador, es decir, no elimina el registro, sino que simplemente lo marca como inactivo y devuelve una respuesta JSON para integrarse con SweetAlert en el frontend. Los métodos `getCantones` y `getDistritos` son endpoints AJAX que devuelven en formato JSON los cantones de una provincia y los distritos de un cantón respectivamente, utilizados para cargar dinámicamente los selectores de dirección en el formulario.

## Gestión Nómina

Figura 72

### Lista Nómina

```

public function index(Request $request)
{
    /** @var \App\Models\Usuario $usuario */
    $usuario = Auth::user();

    // Mes y año actual como valores por defecto si no vienen en la URL
    $mes = $request->get('mes', date("m"));
    $anio = $request->get('anio', date("Y"));

    // Admin/RRHH ve todas las nóminas del período; empleado solo las propias
    if ($usuario->hasPermission("Procesar Nómina")) {
        $query = Nomina::with([
            'colaborador.persona',
            'estado',
        ])
            ->where('mes', $mes)
            ->where('anio', $anio);

        if ($request->filled('colaborador')) {
            $query->where('idColaborador', $request->colaborador);
        }

        $nominas = $query->orderBy('idColaborador')->get();

        $totalBruto = $nominas->sum('total_ingresos');
        $totalDeducciones = $nominas->sum('total_deducciones');
        $totalNeto = $nominas->sum('salario_netos');
        $totalColaboradores = $nominas->count();
    } else {
        $idColaborador = $usuario->colaborador->idColaborador ?? null;

        if ($idColaborador) {
            $nominas = collect();
        } else {
            $nominas = Nomina::with(['estado'])
                ->where('idColaborador', $idColaborador)
                ->where('anio', $anio)
                ->orderBy('mes', 'desc')
                ->get();
        }

        $totalBruto = null;
        $totalDeducciones = null;
        $totalNeto = null;
        $totalColaboradores = null;
    }

    $meses = [
        1 => 'Enero', 2 => 'Febrero', 3 => 'Marzo', 4 => 'Abril', 5 => 'Mayo', 6 => 'Junio', 7 => 'Julio', 8 => 'Agosto', 9 => 'Setiembre', 10 => 'Octubre', 11 => 'Noviembre', 12 => 'Diciembre'
    ];

    $anios = range(2025, date("Y"));

    return view('nominas', compact(
        'nominas', 'totalBruto', 'totalDeducciones', 'totalNeto', 'totalColaboradores', 'mes', 'anio', 'meses', 'anios'
    ));
}

```

Fuente: Elaboración Propia (2026)

Este método se llama index donde gestiona la lista de nóminas del sistema, adaptando los resultados según el rol del usuario: si tiene permisos de administrador muestra todas las nóminas con opción de filtrar por mes, año y colaborador calculando totales de ingresos, deducciones y salario neto, mientras que si es un colaborador común únicamente muestra sus propias nóminas. Finalmente envía todos los datos a la vista junto con los catálogos de meses y años para los filtros.

Figura 73

*Boleta de Pago*

```

public function show($id)
{
    $nomina = Nomina::with([
        'colaborador.persona',
        'estado',
        'tramoRenta',
        'tasaSes',
        'tasaIes',
        'tasaIps',
    ])->findOrFail($id);

    // Empleado no puede ver boletas de otros colaboradores
    /** @var \App\Models\User $usuario */
    $usuario = Auth::user();

    if (!$usuario->hasPermission('Procesar Nómina')) {
        $idColaborador = $usuario->colaborador->idColaborador ?? null;

        if ($nomina->idColaborador != $idColaborador) {
            abort(403, 'No tienes permiso para ver esta boleta.');
```

Fuente: Elaboración Propia (2026)

La imagen anterior en un método llamado Show donde carga el detalle completo de una nómina específica, verificado primero que el colaborador tenga permiso para verla, ya que un empleado no puede acceder a boletas ajenas. Luego separa los conceptos en ingresos y deducciones, calcula los totales y el salario neto resultante, enviando toda esa información a la vista para su presentación.

Figura 74

## Generar Nómina del Mes

```

public function store(Request $request)
{
    $request->validate([
        'mes' => 'required|integer|min:1|max:12',
        'anio' => 'required|integer|min:2023|max:2030',
    ]);

    $mes = (int) $request->mes;
    $anio = (int) $request->anio;

    $meses = [
        1 => 'ENERO',
        2 => 'FEBRERO',
        3 => 'MARZO',
        4 => 'ABRIL',
        5 => 'MAYO',
        6 => 'JUNIO',
        7 => 'JULIO',
        8 => 'AGOSTO',
        9 => 'SEPTIEMBRE',
        10 => 'OCTUBRE',
        11 => 'NOVIEMBRE',
        12 => 'DICIEMBRE'
    ];

    // Verificar que no existan ya nóminas para ese periodo
    $yaExiste = $nominas->where('mes', $mes)
        ->where('anio', $anio)
        ->exists();

    if ($yaExiste) {
        return redirect()
            ->route('nominas.index', ['mes' => $mes, 'anio' => $anio])
            ->with('error', "Ya existen nóminas para {$meses[$mes]} {$anio}.");
    }

    // Catálogos cargados una sola vez fuera del foreach
    $tasaSem = CatalogoTasaCcss::find(1); // SRH 5.50%
    $tasaIvm = CatalogoTasaCcss::find(2); // IVM 4.33%
    $tasaBp = CatalogoTasaCcss::find(3); // BP 3.00%

    $tramos = CatalogoTramoRenta::orderBy('tramo_desde')->get();

    $ultimoNomina = $nominas->max('idnomina') ?? 0;
    $ultimoDetalle = $nominadetalles->max('idnomina_detalle') ?? 0;

    $colaboradores = $colaboradores->with([
        'persona',
        'creditofiscales.catalogoCreditoFiscal',
    ])
        ->where('activo', 1)
        ->get();
}

```

Fuente: Elaboración Propia (2026)

La figura anterior es un método llamado store donde inicia el proceso de generación de nóminas, validando primero que el mes y año sean correctos y verificando que no existan ya nóminas generadas para ese período. Luego carga los catálogos necesarios como las tasas CCSS, IVM, BP, los tramos de renta y obtiene todos los colaboradores activos con sus créditos fiscales para procesar cada uno en los pasos siguientes.

## Figura 75

### Generar Nómina del Mes

```

// DB::transaction garantiza que si algo falla, nada se guarda
DB::transaction(function () use (
    $colaboradores,
    $mes,
    $anio,
    $meses,
    $tasaSem,
    $tasaIvm,
    $tasaBp,
    $ramo,
    $ultimoIDNomina,
    $ultimoIDDetalle
) {
    foreach ($colaboradores as $colaborador) {
        $salarioBase = (float) $colaborador->salario_base;

        // Horas extra aprobadas del mes (estado 2 = Aprobada)
        $totalHE = (float) DB::table('horas_extra')
            ->where('idColaborador', $colaborador->idColaborador)
            ->where('idcatalogo_estado_solicitud', 2)
            ->whereMonth('fecha_trabajo', $mes)
            ->whereYear('fecha_trabajo', $anio)
            ->sum('monto_pagar');

        $totalIngresos = $salarioBase + $totalHE;

        $deduccionSem = round($totalIngresos * ($tasaSem->porcentaje_trabajador / 100), 2);
        $deduccionIvm = round($totalIngresos * ($tasaIvm->porcentaje_trabajador / 100), 2);
        $deduccionBp = round($totalIngresos * ($tasaBp->porcentaje_trabajador / 100), 2);

        // En Costa Rica: base renta = Ingresos - SEM - IVM
        $baseRenta = $totalIngresos - $deduccionSem - $deduccionIvm;

        // Crédito fiscal: monto_mensual * cantidad por cada dependiente registrado
        // Ejemplo: 2 hijos * €1,710 * 2 = €3,420 de reducción al impuesto
        $creditoFiscal = 0;
        if ($colaborador->creditosFiscales) {
            foreach ($colaborador->creditosFiscales as $credito) {
                if ($credito->catalogoCreditoFiscal) {
                    $creditoFiscal += (float) $credito->catalogoCreditoFiscal->monto_mensual
                        * (int) $credito->cantidad;
                }
            }
        }
    }
}

```

Fuente: Elaboración Propia (2026)

Este es una continuación del método store, donde dentro de una transacción recorre cada colaborador activo para calcular su nómina individual: toma su salario base, suma las horas extra aprobadas del período y calcula las deducciones obligatorias de CCSS, IVM y Banco Popular. Luego determina la base gravable para la renta y calcula el crédito fiscal según la cantidad de dependientes registrados del colaborador.

Figura 76

## Generar Nómina del Mes

```

// Cálculo marginal: cada tramo aplica solo sobre la parte del salario en ese rango
$impuestoRenta = 0;
$tramoAplicado = 1;
$baseRestante = $baseRenta;

foreach ($tramos as $tramo) {
    if ($baseRenta <= (float) $tramo->tramo_desde) {
        break;
    }

    $montoEnTramo = min(
        $baseRestante,
        (float) $tramo->tramo_hasta - (float) $tramo->tramo_desde
    );
    $montoEnTramo = max(0, $montoEnTramo);
    $impuestoRenta += round($montoEnTramo * ($tramo->porcentaje / 100), 2);
    $tramoAplicado = $tramo->idcatalogo_tramo_renta;
    $baseRestante = max(0, $baseRenta - (float) $tramo->tramo_hasta);
}

// Aplicar crédito fiscal; resultado no puede ser negativo
$impuestoRenta = max(0, round($impuestoRenta - $creditoFiscal, 2));

$totalDeducciones = $deduccionSem + $deduccionIvm + $deduccionBp + $impuestoRenta;
$salarioNeto = $totalIngresos - $totalDeducciones;

$ultimoIdNomina++;
$idNomina = $ultimoIdNomina;

$nomina::create([
    'idnomina' => $idNomina,
    'idcolaborador' => $colaborador->idcolaborador,
    'periodo' => $meses[$mes] . ' ' . $anio,
    'anio' => $anio,
    'mes' => $mes,
    'salario_base' => $salarioBase,
    'total_horas_extra' => $totalHE,
    'total_bonificaciones' => 0,
    'total_ingresos' => $totalIngresos,
    'total_deducciones' => $totalDeducciones,
    'salario_netto' => $salarioNeto,
    'idcatalogo_tasa_ccss_sem' => 1,
    'idcatalogo_tasa_ccss_ivm' => 2,
    'idcatalogo_tasa_ccss_bp' => 3,
    'idcatalogo_tramo_renta' => $tramoAplicado,
    'idcatalogo_estado_solicitud' => 1, // 1 = Pendiente
    // Fecha proyectada al último día del mes; se actualiza al marcar como pagada
    'fecha_pago' => \Carbon\Carbon::create($anio, $mes, 1)->endOfMonth()->dateString(),
    'pagado_por' => '',
]);

```

Fuente: Elaboración Propia (2026)

Esta es una continuación del método store, donde aplica el cálculo marginal del impuesto de renta recorriendo cada tramo y aplicando el porcentaje correspondiente solo sobre la parte del salario que cae en ese rango. Luego resta el crédito fiscal al impuesto calculado, suma todas las deducciones para obtener el salario neto, y finalmente crea el registro de la nómina en la base de datos con todos los datos calculados, estableciendo la fecha de pago proyectada al último día del mes.

Figura 77

## Generar Nómina del Mes

```

$lineas = [];

// Salario base (tipo 1 = Ingreso)
$ultimoIdDetalle++;
$lineas[] = [
    'idnomina_detalle' => $ultimoIdDetalle,
    'idnomina' => $idNomina,
    'idColaborador' => $colaborador->idColaborador,
    'concepto' => 'Salario Base',
    'monto' => $salarioBase,
    'idcatalogo_tipo_concepto' => 1,
    'fecha_creacion' => now(),
    'fecha_actualizacion' => now(),
];

// Horas extra (tipo 1 = Ingreso, solo si tiene)
if ($totalHE > 0) {
    $ultimoIdDetalle++;
    $lineas[] = [
        'idnomina_detalle' => $ultimoIdDetalle,
        'idnomina' => $idNomina,
        'idColaborador' => $colaborador->idColaborador,
        'concepto' => 'Horas Extra',
        'monto' => $totalHE,
        'idcatalogo_tipo_concepto' => 1,
        'fecha_creacion' => now(),
        'fecha_actualizacion' => now(),
    ];
}

// CCSS SEM (tipo 3 = Dedución CCSS)
$ultimoIdDetalle++;
$lineas[] = [
    'idnomina_detalle' => $ultimoIdDetalle,
    'idnomina' => $idNomina,
    'idColaborador' => $colaborador->idColaborador,
    'concepto' => 'CCSS SEM ' . $tasaSem->porcentaje_trabajador . '%',
    'monto' => $deduccionSem,
    'idcatalogo_tipo_concepto' => 3,
    'fecha_creacion' => now(),
    'fecha_actualizacion' => now(),
];

// CCSS IVM (tipo 3 = Dedución CCSS)
$ultimoIdDetalle++;
$lineas[] = [
    'idnomina_detalle' => $ultimoIdDetalle,
    'idnomina' => $idNomina,
    'idColaborador' => $colaborador->idColaborador,
    'concepto' => 'CCSS IVM ' . $tasaIvm->porcentaje_trabajador . '%',
    'monto' => $deduccionIvm,
    'idcatalogo_tipo_concepto' => 3,
    'fecha_creacion' => now(),
    'fecha_actualizacion' => now(),
];

```

Fuente: Elaboración Propia (2026)

Esta es la continuación del método store, donde construye las líneas de detalle de la nómina registrando cada concepto por separado: primero el salario base como ingreso, luego las horas extra si las hubiera, y después las deducciones obligatorias del CCSS SEM e IVM, cada una con su tipo de concepto, monto y fecha correspondiente para conformar el desglose completo de la boleta de pago.

Figura 78

*Generar Nomina del Mes*

```

// Banco Popular (tipo 5 = Deducción BP)
$ultimoIdDetalle++;
$lineas[] = [
    'idnomina_detalle' => $ultimoIdDetalle,
    'idnomina' => $idNomina,
    'idColaborador' => $colaborador->idColaborador,
    'concepto' => 'Banco Popular ' . $tasaBP . 'porcentaje_trabajador . '%',
    'monto' => $deduccionBP,
    'idcatalogo_tipo_concepto' => 5,
    'fecha_creacion' => now(),
    'fecha_actualizacion' => now(),
];

// Impuesto de renta (tipo 4, solo si aplica)
if ($impuestoRenta > 0) {
    $ultimoIdDetalle++;
    $lineas[] = [
        'idnomina_detalle' => $ultimoIdDetalle,
        'idnomina' => $idNomina,
        'idColaborador' => $colaborador->idColaborador,
        'concepto' => 'Impuesto de Renta',
        'monto' => $impuestoRenta,
        'idcatalogo_tipo_concepto' => 4,
        'fecha_creacion' => now(),
        'fecha_actualizacion' => now(),
    ];
}

// INSERT masivo: todas las líneas en una sola query
NominaDetalle::insert($lineas);

// Vincular las horas extra del mes a esta nómina
DB::table('horas_extra')
->where('idColaborador', $colaborador->idColaborador)
->where('idcatalogo_estado_solicitud', 2)
->whereMonth('fecha_trabajo', $mes)
->whereYear('fecha_trabajo', $anio)
->update(['idnomina' => $idNomina]);

} // fin foreach colaboradores
}); // fin DB::transaction

return redirect()
->route('nominas.index', ['mes' => $mes, 'anio' => $anio])
->with('success', 'Nóminas de {$meses}{$mes}} {$anio} generadas correctamente.');
```

Fuente: Elaboración Propia (2026)

Esta es la continuación y cierre del método store, donde agrega las deducciones del Banco Popular y el impuesto de renta solo si aplica. Una vez construidas todas las líneas, las inserta en la base de datos en una sola operación masiva para mayor eficiente y vincula las horas extra del período a la nómina generada. Finalmente cierra la transacción y redirige al listado de nóminas con un mensaje de éxito indicando el mes y año procesados.

Figura 79

*Control de Pago - Marcar Pagada*

```

public function marcarPagada($id)
{
    $nomina = Nomina::findOrFail($id);

    if ($nomina->idcatalogo_estado_solicitud == 9) {
        return redirect()
            ->route('nominas.index', ['mes' => $nomina->mes, 'anio' => $nomina->anio])
            ->with('warning', 'Esta nómina ya fue marcada como pagada anteriormente.');
```

Fuente: Elaboración Propia (2026)

La figura anterior es un método llamado `marcarPagada` donde gestiona el cambio de estado de una nómina a pagada, verificando primero que no haya sido marcada anteriormente para evitar duplicados. Si todo es correcto, actualiza su estado a pagada registrando la fecha de pago y el usuario que realizó la acción y redirige al listado con un mensaje de éxito.

**Figura 80**

### *Control de Pago - Marcar Todas Pagadas*

```

public function marcarTodasPagadas(Request $request)
{
    $request->validate([
        'mes' => 'required|integer|min:1|max:12',
        'anio' => 'required|integer|min:2025|max:2030',
    ]);

    $mes = (int) $request->mes;
    $anio = (int) $request->anio;

    $meses = [
        1 => 'Enero', 2 => 'Febrero', 3 => 'Marzo', 4 => 'Abril', 5 => 'Mayo', 6 => 'Junio', 7 => 'Julio', 8 => 'Agosto', 9 => 'Setiembre', 10 => 'Octubre', 11 => 'Noviembre', 12 => 'Diciembre'
    ];

    // Solo actualiza los que aún no están pagados (estado != 9)
    $cantidad = Nomina::where('mes', $mes)
        ->where('anio', $anio)
        ->where('idcatalogo_estado_solicitud', '!=', 9)
        ->update([
            'idcatalogo_estado_solicitud' => 9,
            'fecha_pago' => now(),
            'pagado_por' => Auth::user()->idUsuario,
        ]);

    if ($cantidad == 0) {
        return redirect()
            ->route('nominas.index', ['mes' => $mes, 'anio' => $anio])
            ->with('warning', "Todas las nóminas de {$meses[$mes]} {$anio} ya estaban pagadas.");
    }

    return redirect()
        ->route('nominas.index', ['mes' => $mes, 'anio' => $anio])
        ->with('success', "{$cantidad} nóminas de {$meses[$mes]} {$anio} marcadas como pagadas.");
}

```

Fuente: Elaboración Propia (2026)

La imagen anterior es un método llamado `marcarTodasPagadas` donde gestiona el pago masivo de nóminas de un período específico, validando el mes y año recibidos. Actualiza en una sola operación todas las nóminas que aún no estén marcadas como pagadas, registrando la fecha y el usuario que realizó la acción. Si todas ya estaban pagadas previamente muestra una advertencia, de lo contrario redirige con un mensaje indicando la cantidad de nóminas procesadas.

**Figura 81****Distribución de Boletas - Enviar Correo**

```

public function enviarCorreo($id)
{
    set_time_limit(120);
    $nomina = Nomina::with([
        'colaborador.persona',
        'estado',
    ])->findOrFail($id);

    if (!$nomina->estaPagada()) {
        return redirect()
            ->route('nominas.index', ['mes' => $nomina->mes, 'anio' => $nomina->anio])
            ->with('error', 'No se puede enviar la boleta porque la nómina aún no ha sido pagada.');
```

Fuente: Elaboración Propia (2026)

El método mostrado anteriormente es llamado `enviarCorreo` donde gestiona el envío de la boleta de pago por correo al colaborador, verificando primero que la nómina esté pagada y que el colaborador tenga un correo registrado. Si ambas condiciones se cumplen, envía la boleta usando la clase `BoletaPagoMail` y actualiza la nómina marcándola como enviada con la fecha correspondiente, redirigiendo finalmente con un mensaje de éxito indicando el correo al que fue enviada.

## Figura 82

### Distribución de Boletas - Enviar Todos los Correos

```

public function enviarCorreosTodos(Request $request)
{
    set_time_limit(120);
    $request->validate([
        'mes' => 'required|integer|min:1|max:12',
        'anio' => 'required|integer|min:2025|max:2030',
    ]);

    $mes = (int) $request->mes;
    $anio = (int) $request->anio;

    $meses = [
        1 => 'Enero', 2 => 'Febrero', 3 => 'Marzo', 4 => 'Abril', 5 => 'Mayo', 6 => 'Junio', 7 => 'Julio', 8 => 'Agosto', 9 => 'Setiembre', 10 => 'Octubre', 11 => 'Noviembre', 12 => 'Diciembre'
    ];

    $nominas = $nomina->with(['colaborador.persona', 'estado']);
    $nominas->where('mes', $mes)
        ->where('anio', $anio)
        ->get();

    if ($nominas->isEmpty()) {
        return redirect()
            ->route('nominas.index', ['mes' => $mes, 'anio' => $anio])
            ->with('error', "No hay nóminas para {$meses[$mes]} {$anio}.");
    }

    $enviados = 0;
    $sinCorreo = 0;

    foreach ($nominas as $nomina) {
        if ($nomina->estaPagada) {
            continue;
        }

        $idPersona = $nomina->colaborador->Persona_idPersona;

        $correoColaborador = DB::table('correo')
            ->where('idPersona', $idPersona)
            ->where('ACTIVO', 1)
            ->orderBy('idCatalogo_Tipo_Correo')
            ->value('Direccion_Correo');

        if ($correoColaborador) {
            $sinCorreo++;
            continue;
        }

        $url::to($correoColaborador)->send(new BoletaPagoMail($nomina));

        $nomina->update([
            'correo_enviado' => 1,
            'fecha_envio_correo' => now(),
        ]);

        $enviados++;
    }

    $mensaje = "{$enviados} boletas enviadas correctamente para {$meses[$mes]} {$anio}.";
    if ($sinCorreo > 0) {
        $mensaje .= " ({$sinCorreo} colaboradores sin correo registrado)";
    }

    return redirect()
        ->route('nominas.index', ['mes' => $mes, 'anio' => $anio])
        ->with('success', $mensaje);
}

```

Fuente: Elaboración Propia (2026)

La figura anterior es un método llamado `enviarCorreosTodos` donde gestiona el envío masivo de boletas de pago para todas las nóminas de un período específico, recorriendo cada una y omitiendo las que no estén pagadas o el colaborador no tenga el correo registrado. Envía cada boleta usando `BoletaPagoMail` y lleva contadores de enviados y colaboradores sin correo, mostrando al final un mensaje resumido con los resultados del proceso.

**Figura 83***Exportar a Excel*

```

public function exportarExcel(Request $request)
{
    $mes = (int) $request->get('mes', date('n'));
    $anio = (int) $request->get('anio', date('Y'));

    $meses = [
        1 => 'Enero',
        2 => 'Febrero',
        3 => 'Marzo',
        4 => 'Abril',
        5 => 'Mayo',
        6 => 'Junio',
        7 => 'Julio',
        8 => 'Agosto',
        9 => 'Septiembre',
        10 => 'Octubre',
        11 => 'Noviembre',
        12 => 'Diciembre'
    ];

    $nombreMes = strtoupper($meses[$mes] ?? 'DESCONOCIDO');

    $nominas = Nomina::with([
        'colaborador.persona',
        'estado',
    ])
    ->where('mes', $mes)
    ->where('anio', $anio)
    ->orderBy('idColaborador')
    ->get();

    if ($nominas->isEmpty()) {
        return redirect()
        ->route('nominas.index', ['mes' => $mes, 'anio' => $anio])
        ->with('error', "No existen nóminas para {$meses[$mes]} ({$anio}.");
    }

    $idsNominas = $nominas->pluck('idnomina');
}

```

Fuente: Elaboración Propia (2026)

La imagen anterior es un método llamado `exportarExcel` donde inicia el proceso de exportación de nóminas a Excel, obteniendo el mes y año del request y cargando todas las nóminas del período con sus datos de colaborador y estado. Si no existen nóminas para ese período redirige con un mensaje de error, de lo contrario recolecta todos los IDs de las nóminas para usarlos en la generación del archivo.

## Figura 84

### Exportar a Excel

```

// Detalles agrupados por idnomina para evitar N+1 queries
$todosDetalles = NominaDetalle::whereIn('idnomina', $idsNominas)
->orderBy('idnomina')
->orderBy('idcatalogo_tipo_concepto')
->get()
->groupBy('idnomina');

$spreadsheet = new Spreadsheet();
$spreadsheet->getProperties()
->setCreator('Sistema RRHH Tendenza')
->setTitle('Nómina ($nombre) ($anio)')
->setSubject('Reporte de Nóminas')
->setDescription('Generado automáticamente por Sistema RRHH Tendenza');

$ws1 = $spreadsheet->getActiveSheet();
$ws1->setTitle('Resumen Nómina');

// Estilos reutilizables para las dos hojas
$estiloTitulo = [
    'font' => ['bold' => true, 'color' => ['rgb' => 'FFFFFF'], 'size' => 14, 'name' => 'Arial'],
    'fill' => ['fillType' => Fill::FILL_SOLID, 'startColor' => ['rgb' => 'A3A668']],
    'alignment' => ['horizontal' => Alignment::HORIZONTAL_CENTER, 'vertical' => Alignment::VERTICAL_CENTER],
];

$estiloHeader = [
    'font' => ['bold' => true, 'color' => ['rgb' => '000000'], 'size' => 10, 'name' => 'Arial'],
    'fill' => ['fillType' => Fill::FILL_SOLID, 'startColor' => ['rgb' => 'C8D4E8']],
    'alignment' => [
        'horizontal' => Alignment::HORIZONTAL_CENTER,
        'vertical' => Alignment::VERTICAL_CENTER,
        'wrapText' => true
    ],
    'borders' => ['allBorders' => ['borderStyle' => Border::BORDER_THIN, 'color' => ['rgb' => 'AAAAAA']]],
];

$estiloTotal = [
    'font' => ['bold' => true, 'color' => ['rgb' => 'FFFFFF'], 'size' => 10, 'name' => 'Arial'],
    'fill' => ['fillType' => Fill::FILL_SOLID, 'startColor' => ['rgb' => 'A3A668']],
    'alignment' => ['horizontal' => Alignment::HORIZONTAL_RIGHT, 'vertical' => Alignment::VERTICAL_CENTER],
    'borders' => ['allBorders' => ['borderStyle' => Border::BORDER_THIN, 'color' => ['rgb' => 'AAAAAA']]],
];

$bordeThin = [
    'borders' => ['allBorders' => ['borderStyle' => Border::BORDER_THIN, 'color' => ['rgb' => 'AAAAAA']]],
];

```

Fuente: Elaboración Propia (2026)

Esta es la continuación del método `exportarExcel`, donde carga todos los detalles de las nóminas agrupados para evitar consultas innecesarias, crea el archivo `spreadsheet` con su título y descripción, y define los estilos reutilizables para las celdas como títulos, encabezados, totales y bordes que se aplicarán a lo largo de las hojas del reporte.

Figura 85

## Exportar a Excel

```

$ws1->mergeCells('A1:L1');
$ws1->setCellValue('A1', "TENDENZA 8 PUERTALLUM | REPORTE DE NÓMINA | ($nombreMes) ($anio)");
$ws1->getStyle('A1')->applyFromArray($estiloTitulo);
$ws1->getRowDimension(1)->setRowHeight(32);

$fechaHoy = now()->format('d/m/Y');
$ws1->mergeCells('A2:L2');
$ws1->setCellValue(
    'A2',
    "Período: {$meses}{$mes} {$anio} | Generado: {$fechaHoy} | . .
    Tasas CCSS 2026: SEM 5.50% | IVM 4.33% | BP 1.00%";
);
$ws1->getStyle('A2')->applyFromArray([
    'font' => ['italic' => true, 'color' => ['rgb' => '333333'], 'size' => 9, 'name' => 'Arial'],
    'fill' => ['fillType' => Fill::FILL_SOLID, 'startColor' => ['rgb' => 'D00808']],
    'alignment' => ['horizontal' => Alignment::HORIZONTAL_CENTER, 'vertical' => Alignment::VERTICAL_CENTER],
]);
$ws1->getRowDimension(2)->setRowHeight(18);
$ws1->getRowDimension(3)->setRowHeight(6);

$headers = [
    'A' => 'No.',
    'B' => 'Colaborador',
    'C' => 'Cédula',
    'D' => 'Salario Base (€)',
    'E' => 'Horas Extra (€)',
    'F' => 'Total Ingresos (€)',
    'G' => 'CCSS SEM (€)',
    'H' => 'CCSS IVM (€)',
    'I' => 'Banco Psp. (€)',
    'J' => 'Imp. Renta (€)',
    'K' => 'Total Deduc. (€)',
    'L' => 'Neto a Pagar (€)',
];

foreach ($headers as $col => $texto) {
    $ws1->setCellValue("${col}4", $texto);
    $ws1->getStyle("${col}4")->applyFromArray($estiloHeader);
}
$ws1->getRowDimension(4)->setRowHeight(36);

$anchos = ['A' => 6, 'B' => 30, 'C' => 14, 'D' => 16, 'E' => 16, 'F' => 18, 'G' => 14, 'H' => 14, 'I' => 14, 'J' => 14, 'K' => 18, 'L' => 18];
foreach ($anchos as $col => $ancho) {
    $ws1->getColumnDimension($col)->setWidth($ancho);
}

```

Fuente: Elaboración Propia (2026)

Aquí se construye el encabezado de la hoja de resumen de la hoja de resumen con el título de la empresa, el período y las tasas aplicadas. Luego define y aplica los encabezados de columnas como número, colaborador, cédula, salario base, horas extra, deducciones y neto a pagar, ajustando además el ancho de cada columna para una presentación ordenada del reporte.

Figura 86

## Exportar a Excel

```

$fila = 5;
$contador = 1;
$colores = ['F9F9F9', 'FFFFFF']; // filas alternadas
$formatoMoneda = '{#,##0.00;(#,##0.00);@.00}'; // cero visible como 0.00

foreach ($nominas as $nominas) {
    $nombre = $nominas->colaborador->persona->Nombre;
    $nominas->colaborador->persona->Apellido1;
    $nominas->colaborador->persona->Apellido2;
    $cedula = $nominas->colaborador->persona->numero_identificacion ?? 'N/A';
    $bgColor = $colores[$contador % 2];

    $ws1->setCellValue("A{$fila}", $contador);
    $ws1->setCellValue("B{$fila}", trim($nombre));
    $ws1->setCellValue("C{$fila}", $cedula);
    $ws1->setCellValue("D{$fila}", (float) $nominas->salario_base);
    $ws1->setCellValue("E{$fila}", (float) $nominas->total_horas_extra);
    $ws1->setCellValue("F{$fila}", "=SUM($fila:$K{$fila})");
    $ws1->setCellValue("K{$fila}", (float) $nominas->total_deducciones);
    $ws1->setCellValue("L{$fila}", "=F{$fila}-K{$fila}");

    $detalles = $todosDetalles[$nominas->idnominas] ?? collect();
    $sum = $detalles->where('idcatalogo_tipo_concepto', 3)->filter(fn($d) => str_contains($d->concepto, 'SEM'))->sum('monto');
    $ivm = $detalles->where('idcatalogo_tipo_concepto', 3)->filter(fn($d) => str_contains($d->concepto, 'IVM'))->sum('monto');
    $bp = $detalles->where('idcatalogo_tipo_concepto', 5)->sum('monto');
    $ren = $detalles->where('idcatalogo_tipo_concepto', 4)->sum('monto');

    $ws1->setCellValue("G{$fila}", (float) $sum);
    $ws1->setCellValue("H{$fila}", (float) $ivm);
    $ws1->setCellValue("I{$fila}", (float) $bp);
    $ws1->setCellValue("J{$fila}", (float) $ren);

    foreach (['D', 'E', 'F', 'G', 'H', 'I', 'J', 'K', 'L'] as $col) {
        $ws1->getStyle("$col{$fila}")->getNumberFormat()->setFormatCode($formatoMoneda);
    }

    $estiloFila = [
        'font' => ['name' => 'Arial', 'size' => 10],
        'fill' => ['fillType' => Fill::FILL_SOLID, 'startColor' => [$bgColor]],
        'borders' => ['allBorders' => ['borderStyle' => Border::BORDER_THIN, 'color' => 'AAAAAA']],
    ];
    $ws1->getStyle("A{$fila}:L{$fila}")->applyFromArray($estiloFila);
    $ws1->getStyle("A{$fila}")->getAlignment()->setHorizontal(Alignment::HORIZONTAL_CENTER);
    $ws1->getStyle("L{$fila}")->getFont()->setBold(true);
    $ws1->getStyle("L{$fila}")->getColor()->setColor('1A663A');
    $ws1->getRowDimension($fila)->setRowHeight(18);

    $fila++;
    $contador++;
}

```

Fuente: Elaboración Propia (2026)

Esta continuación recorre cada nómina para escribir sus datos en una fila del reporte, incluyendo nombre, cédula, salario base, horas extra, deducciones individuales de CCSS, IVM y Banco Popular, y el neto a pagar. Aplica formato de moneda a las columnas numéricas y alterna el color entre filas para mejorar la legibilidad visual del reporte.

## Figura 87

### Exportar a Excel

```

$filaTotal = $fila;
$ws1->mergeCells("A:$filaTotal:C:$filaTotal");
$ws1->setCellValue("A:$filaTotal", "TOTALES PERIODO");
$ws1->getStyle("A:$filaTotal:L:$filaTotal")->applyFromArray($estiloTotal);
$ws1->getStyle("A:$filaTotal")->getAlignment()->setHorizontal(Alignment::HORIZONTAL_CENTER);
$ws1->getRowDimension($filaTotal)->setRowHeight(22);

$filaIni = 5;
$filaFin = $filaTotal - 1;
foreach ($D, $E, $F, $G, $H, $I, $J, $K, $L) as $col {
    $ws1->setCellValue("$col:$filaTotal", "=SUM($col:$filaIni:$col:$filaFin)");
    $ws1->getStyle("$col:$filaTotal")->getNumberFormat()->setFormatCode($formatoMoneda);
}

$filaNota = $filaTotal - 2;
$ws1->mergeCells("A:$filaNota:L:$filaNota");
$ws1->setCellValue(
    "A:$filaNota",
    "Nota: Este reporte es de carácter confidencial." .
    "Tasas CGS vigentes 2026: SEM 5.50% | IWM 4.33% | BP 1.00% | " .
    "Tramos de renta según resolución DGT-R-033-2023."
);
$ws1->getStyle("A:$filaNota")->applyFromArray([
    'font' => ['italic' => true, 'color' => ['rgb' => '555555'], 'size' => 8, 'name' => 'Arial'],
    'alignment' => ['horizontal' => Alignment::HORIZONTAL_LEFT, 'wrapText' => true],
]);
$ws1->getRowDimension($filaNota)->setRowHeight(24);

$ws1->freezePane('A5');

$ws2 = $spreadsheet->createSheet();
$ws2->setTitle('Desglose por concepto');

$ws2->mergeCells("A1:E1");
$ws2->setCellValue("A1", "DESGLOSE DE CONCEPTOS - {$nombreRes} {$anio}");
$ws2->getStyle("A1")->applyFromArray($estiloTitulo);
$ws2->getRowDimension(1)->setRowHeight(20);

$headers2 = ['A' => 'Colaborador', 'B' => 'Concepto', 'C' => 'Tipo', 'D' => 'Monto (€)', 'E' => 'Estado Nómina'];
foreach ($headers2 as $col => $texto) {
    $ws2->setCellValue("$col", $texto);
    $ws2->getStyle("$col")->applyFromArray($estiloHeader);
}

$ws2->getRowDimension(3)->setRowHeight(28);
$ws2->getColumnDimension('A')->setWidth(32);
$ws2->getColumnDimension('B')->setWidth(26);
$ws2->getColumnDimension('C')->setWidth(18);
$ws2->getColumnDimension('D')->setWidth(16);
$ws2->getColumnDimension('E')->setWidth(16);

```

Fuente: Elaboración Propia (2026)

Aquí se agrega la fila de totales del período con sumas automáticas de todas las columnas numéricas y una nota al pie con la información confidencial sobre las tasas y tramos de renta aplicados. Luego crea una segunda hoja llamada desglose por concepto, con sus propios encabezados de colaborador, concepto, tipo, monto y estado de nómina, para mostrar el detalle individual de cada línea de pago.

Figura 88

*Exportar a Excel*

```

// Colores por tipo de concepto para la hoja de desglose
$coloresTipo = [
    1 => 'E8E8E8', // Ingreso      + verde claro
    3 => 'FFF3E0', // Deducción CCSS + naranja claro
    4 => 'FFE8E8', // Renta          + rojo claro
    5 => 'E8E8F3', // Banco Popular  + morado claro
];

$fila2 = 4;
foreach ($nominas as $nomina) {
    $nombre = "Incl $nomina->colaborador->persona->Nombre . ' ' .
    $nomina->colaborador->persona->Apellido0 . ' ' .
    $nomina->colaborador->persona->Apellido2);
    $estado = $nomina->estado->description ?? 'Pendiente';
    $detalles = $todosDetalles[$nomina->idnomina] ?? collect();

    foreach ($detalles as $detalle) {
        $tipo = $detalle->idcatalogo_tipo_concepto;
        $bgDet = $coloresTipo[$tipo] ?? 'FFFFFF';

        $ws2->setCellValue("A{$fila2}", $nombre);
        $ws2->setCellValue("B{$fila2}", $detalle->concepto);
        $ws2->setCellValue("C{$fila2}", $detalle->tipoConcepto->description ?? 'N/A');
        $ws2->setCellValue("D{$fila2}", $detalle->monto);
        $ws2->setCellValue("E{$fila2}", $estado);

        $ws2->getStyle("D{$fila2}")->getNumberFormat()->setFormatCode($formatoMoneda);
        $ws2->getStyle("A{$fila2}:E{$fila2}")->applyFromArray([
            'font' => ['name' => 'Arial', 'size' => 10],
            'fill' => ['fillType' => Fill::FILL_SOLID, 'startColor' => ['rgb' => $bgDet]],
            'borders' => ['allBorders' => ['borderStyle' => Border::BORDER_THIN, 'color' => ['rgb' => 'AAAAA']]],
        ]);
        $ws2->getRowDimension($fila2)->setRowHeight(17);
        $fila2++;
    }
}

$ws2->freezePane('A4');
$spreadsheet->setActiveSheetIndex(0);

// Enviar como descarga directa sin guardar en el servidor
$nombreArchivo = "Nomina_{$nombreMes}_{$anio}.xlsx";
header('Content-Type: application/vnd.openxmlformats-officedocument.spreadsheetml.sheet');
header('Content-Disposition: attachment; filename={$nombreArchivo}');
header('Cache-Control: max-age=0');

$writer = new Xlsx($spreadsheet);
$writer->save('php://output');
exit;

```

Fuente: Elaboración Propia (2026)

Este es el cierre del método `exportarExcel`, donde llena la hoja de desglose recorriendo cada nómina y sus líneas de detalle, aplicando un color diferente por tipo de concepto como verde para ingresos, naranja para CCSS, rojo para renta y morado para Banco Popular. Finalmente configura la hoja activa, genera el nombre del archivo con el mes y año correspondiente y lo envía directamente al navegador como descarga sin guardarlo en el servidor.

Figura 89

*Wizard - Procesar*

```

public function procesar()
{
    $meses = [
        1 => 'Enero', 2 => 'Febrero', 3 => 'Marzo', 4 => 'Abril', 5 => 'Mayo', 6 => 'Junio', 7 => 'Julio', 8 => 'Agosto', 9 => 'Setiembre', 10 => 'Octubre', 11 => 'Noviembre', 12 => 'Diciembre'
    ];

    $anios = range(2025, date('Y'));

    // La vista detecta que no viene $preview y muestra el paso 1
    return view('nominas.nomina-procesar', compact('meses', 'anios'));
}

```

Fuente: Elaboración Propia (2026)

En la figura anterior se muestra un método llamado `procesar` donde simplemente prepara la vista para el formulario de generación de nómina, cargando el catálogo de meses

y un rango de años desde el año que uno desee, pero en este caso 2025 hasta el año actual y los envía a la vista para que el usuario seleccione el periodo a procesar.

## Figura 90

### *Wizard - Vista Previa*

```
public function preview(Request $request)
{
    $request->validate([
        'mes' => 'required|integer|min:1|max:12',
        'anio' => 'required|integer|min:2025|max:2030',
    ]);

    $mes = (int) $request->mes;
    $anio = (int) $request->anio;

    $meses = [
        1 => 'Enero', 2 => 'Febrero', 3 => 'Marzo', 4 => 'Abril', 5 => 'Mayo', 6 => 'Junio', 7 => 'Julio', 8 => 'Agosto', 9 => 'Setiembre', 10 => 'Octubre', 11 => 'Noviembre', 12 => 'Diciembre'
    ];

    $anios = range(2025, date('Y'));

    // Si ya existe nómina para ese periodo la vista bloquea el botón de confirmar
    $yaExiste = $nomina->where('mes', $mes)
        ->where('anio', $anio)
        ->exists();
}
```

Fuente: Elaboración Propia (2026)

La imagen anterior es un método llamado `preview` donde valida que el mes y año seleccionados sean correctos, carga el catálogo de meses y el rango de años disponibles, y verifica si ya existen nóminas generadas para ese período. Esta última verificación se envía a la vista para bloquear el botón de confirmación en caso de que el período ya haya sido procesado anteriormente.

## Figura 91

### Wizard - Vista Previa

```

// Catálogos cargados una sola vez fuera del foreach
$tasaSem = CatalogoTasaCcss::find(1); // SEM 5.50%
$tasaIvm = CatalogoTasaCcss::find(2); // IVM 4.33%
$tasaBp = CatalogoTasaCcss::find(3); // BP 1.00%

$tramos = CatalogoTramoRenta::orderBy('tramo_desde')->get();

$colaboradores = Colaborador::with([
    'persona',
    'creditosFiscales.catalogoCreditoFiscal',
])
->where('activo', 1)
->get();

// Mismos cálculos que store() pero sin insertar nada
$preview = [];
$totalBruto = 0;
$totalDeducciones = 0;
$totalNeto = 0;

foreach ($colaboradores as $colaborador) {
    $salarioBase = (float) $colaborador->salario_base;

    // Horas extra aprobadas del mes (estado 2 = Aprobada)
    $totalHE = (float) DB::table('horas_extra')
->where('idColaborador', $colaborador->idColaborador)
->where('idcatalogo_estado_solicitud', 2)
->whereMonth('fecha_trabajo', $mes)
->whereYear('fecha_trabajo', $anio)
->sum('monto_pagar');

    $totalIngresos = $salarioBase + $totalHE;

    $deduccionSem = round($totalIngresos * ($tasaSem->porcentaje_trabajador / 100), 2);
    $deduccionIvm = round($totalIngresos * ($tasaIvm->porcentaje_trabajador / 100), 2);
    $deduccionBp = round($totalIngresos * ($tasaBp->porcentaje_trabajador / 100), 2);

    // En Costa Rica: base renta = ingresos - SEM - IVM
    $baseRenta = $totalIngresos - $deduccionSem - $deduccionIvm;

    // Crédito fiscal: monto_mensual * cantidad por cada dependiente registrado
    // Ejemplo: 3 hijos + (1,710 * 3 = €5,130 de crédito fiscal
    $creditoFiscal = 0;
    foreach ($colaborador->creditosFiscales as $credito) {
        if ($credito->catalogoCreditoFiscal) {
            $creditoFiscal += (float) $credito->catalogoCreditoFiscal->monto_mensual
                * (int) $credito->cantidad;
        }
    }
}

```

Fuente: Elaboración Propia (2026)

Esto es una continuación del método preview, donde realiza exactamente los mismos cálculos que el método store, pero sin insertar nada en la base de datos. Carga las tasas del CCSS, IVM y Banco Popular, los tramos de renta y los colaboradores activos, y por cada uno calcula su salario base, horas extra, deducciones y crédito fiscal, acumulando los totales para mostrarlos como previsualización antes de confirmar la generación de nóminas.

Figura 92

## Wizard - Vista Previa

```

// Cálculo marginal: cada tramo aplica solo sobre la parte del salario en ese rango
$impuestoRenta = 0;
$baseRestante = $baseRenta;

foreach ($tramos as $tramo) {
    if ($baseRenta <= (float) $tramo->tramo_desde) break;

    $montoEnTramo = max(0, min($baseRestante, (float)$tramo->tramo_hasta - (float)$tramo->tramo_desde));
    $impuestoRenta += round($montoEnTramo * ($tramo->porcentaje / 100), 2);
    $baseRestante = max(0, $baseRenta - (float)$tramo->tramo_hasta);
}

// Aplicar crédito fiscal; resultado no puede ser negativo
$impuestoRenta = max(0, round($impuestoRenta - $creditoFiscal, 2));

$deducciones = $deduccionSem + $deduccionIvm + $deduccionBp + $impuestoRenta;
$neto = $totalIngresos - $deducciones;

$preview[] = [
    'idColaborador' => $colaborador->idColaborador,
    'nombre' => $colaborador->persona->Nombre . ' '
        . $colaborador->persona->Apellido1 . ' '
        . $colaborador->persona->Apellido2,
    'salario_base' => $salarioBase,
    'horas_extra' => $totalHE,
    'total_ingresos' => $totalIngresos,
    'deducciones' => $deducciones,
    'neto' => $neto,
];

$totalBruto += $totalIngresos;
$totalDeducciones += $deducciones;
$totalNeto += $neto;
} // fin foreach colaboradores

return view('nominas.nomina-procesar', compact(
    'meses', // arreglo de meses para el selector (1 => 'Enero', ...)
    'años', // lista de años disponibles (2025 hasta año actual)
    'mes', // mes seleccionado por el usuario
    'año', // año seleccionado por el usuario
    'preview', // montos calculados por colaborador (salario, deducciones, neto)
    'yaExiste', // true si ya hay nómina generada para ese mes/año
    'totalBruto', // suma de ingresos de todos los colaboradores
    'totalDeducciones', // suma de deducciones de todos los colaboradores
    'totalNeto' // suma de neto a pagar de todos los colaboradores
));
}

```

Fuente: Elaboración Propia (2026)

Esta es la continuación y cierre del método preview, donde aplica el cálculo marginal del impuesto de renta por tramos, resta el crédito fiscal y obtiene el neto de cada colaborador. Todos estos datos se acumulan en un arreglo de previsualización junto con los totales generales de ingresos, deducciones y neto, y se envían a la vista para que el usuario revise los montos antes de confirmar la generación definitiva de las nóminas.

## Pruebas

El proceso de pruebas aplicadas al sistema tiene como finalidad verificar que cada uno de los módulos desarrollados funcione de manera correcta y cumpla con los

requerimientos establecidos durante la etapa de análisis y diseño del sistema. A través de la ejecución de distintos casos de prueba se evalúan los diferentes escenarios que pueden presentarse durante el uso del sistema por parte de los usuarios.

Durante estas pruebas se analizan situaciones como el ingreso de datos válidos, datos incorrectos y la verificación de campos obligatorios, con el propósito de comprobar que el sistema responda adecuadamente ante cada uno de estos casos. Asimismo, se valida que el sistema muestre los mensajes correspondientes cuando se detecten errores y que permita la correcta ejecución de los procesos cuando la información ingresada sea válida.

De esta manera, el proceso de pruebas permite garantizar la confiabilidad del sistema, el correcto funcionamiento de sus módulos y la integración de la información almacenada en la base de datos, contribuyendo a asegurar la calidad del sistema desarrollado para la gestión de recursos humanos.

### Tabla 93

#### *Caso de prueba - Inicio de sesión*

RRHH Pruebas	
Código	CP – 01
Nombre del caso de Prueba	Iniciar Sesión
Fecha de elaboración	10/7/2025
Autor	Miguel Aguilar
Precondiciones:	
El usuario debe estar previamente registrado en el sistema y tener su cuenta activa.	
Prioridad:	Alta
Descripción:	
Este caso de prueba verifica la funcionalidad del módulo de autenticación de usuarios para asegurar que el inicio de sesión se realice correctamente con credenciales válidas y que el sistema responda adecuadamente en caso de errores.	
Detalle de resultados	

Datos de entrada	Resultado esperado	Resultado Actual	Estado
Ingresar usuario valido y contraseña correcta	El sistema valida las credenciales y redirige al panel principal	Resultado esperado obtenido	Prueba exitosa
Usuario válido y contraseña correcta	Mostrar mensaje de error	Resultado esperado obtenido	Prueba exitosa
Campos vacíos	Mostrar mensaje de error	Resultado esperado obtenido	Prueba exitosa
Usuario desactivado	Mostrar mensaje de cuenta deshabilitada	Resultado esperado obtenido	Prueba exitosa

Fuente: Elaboración Propia (2026)

Este caso de prueba tiene como objetivo verificar que el sistema permita el acceso únicamente cuando las credenciales ingresadas sean correctas. Además, se evalúa que el sistema muestre mensajes de error cuando la contraseña sea incorrecta, los campos estén vacíos o la cuenta del usuario se encuentre desactivada.

#### Tabla 94

##### *Caso de prueba - Registro de vacaciones*

RRHH Pruebas	
Código	CP – 02
Nombre del caso de Prueba	Registro de vacaciones
Fecha de elaboración	10/7/2025
Autor	Miguel Aguilar
Precondiciones:	
El colaborador debe estar registrado en el sistema y contar con días disponibles de vacaciones	
Prioridad:	Alta

Descripción:			
Este caso de pruebas verifica que el sistema registre correctamente una solicitud de vacaciones y valide que el colaborador tenga saldo disponible.			
Detalle de resultados			
Datos de entrada	Resultado esperado	Resultado Actual	Estado
Ingresar fechas validas de vacaciones.	Registrar solicitud correctamente	Resultado esperado	Prueba exitosa
No seleccionar fecha	Mostrar error de campo obligatorio	Resultado esperado	Prueba exitosa
Solicitar más días de los disponibles	Mostrar alerta de saldo insuficiente	Resultado esperado obtenido	Prueba exitosa
Fechas duplicadas	Mostrar error de registro existente	Resultado esperado obtenido	Prueba exitosa

Fuente: Elaboración Propia (2026)

Este caso de prueba tiene como objetivo verificar que el sistema permita registrar correctamente las solicitudes de vacaciones de los colaboradores. Asimismo, se valida que el sistema controle los campos obligatorios y muestre advertencias cuando la información ingresada sea incorrecta o incompleta.

### Tabla 95

*Caso de prueba - Registro de permisos*

RRHH Pruebas	
Código	CP - 03
Nombre del caso de Prueba	Registro de permisos
Fecha de elaboración	10/7/2025
Autor	Miguel Aguilar
Precondiciones:	
El colaborador debe estar registrado y activo en el sistema	

Prioridad:	Media		
Descripción			
Este caso de prueba verifica el registro de solicitudes de permisos laborales y valida los campos requeridos.			
Detalle de resultados			
Datos de entrada	Resultado esperado	Resultado Actual	Estado
Registrar permiso con datos validos	Guardar solicitud correctamente	Resultado esperado obtenido	Prueba exitosa
Campos vacíos	Mostrar evidencia	Resultado esperado obtenido	Prueba exitosa
Seleccionar fechas inválidas	Mostrar error	Resultado esperado obtenido	Prueba exitosa

Fuente: Elaboración Propia (2026)

Este caso de prueba tiene como objetivo comprobar que el sistema registre correctamente las solicitudes de permisos de los colaboradores. También se verifica que el sistema valide los datos ingresados y muestre mensajes de advertencia cuando existan campos obligatorios sin completar.

## Tabla 96

### *Caso de prueba - Registro de marca*

RRHH Pruebas	
Código	CP - 04
Nombre del caso de Prueba	Registro de marca
Fecha de elaboración	10/7/2025
Autor	Miguel Aguilar
Precondiciones:	
El colaborador debe estar registrado y autenticado en el sistema	
Prioridad:	Alta

Descripción:			
Este caso de prueba verifica el registro correcto de la marca de entrada y salida del colaborador			
Detalle de resultados			
Datos de entrada	Resultado esperado	Resultado Actual	Estado
Registrar marca de entrada	Guardar hora de entrada	Resultado esperado obtenido	Prueba exitosa
Registrar marca de salida	Guardar hora de salida	Resultado esperado obtenido	Prueba exitosa
Intentar marcar dos veces seguidas	Mostrar alerta	Resultado esperado obtenido	Prueba exitosa

Fuente: Elaboración Propia (2026)

Este caso de prueba tiene como objetivo verificar que el sistema registre correctamente las marcas de entrada y salida de los colaboradores. Además, se evalúa que el sistema controle posibles errores o registros duplicados.

### Tabla 97

#### *Caso de prueba - Cálculo de aguinaldo*

RRHH Pruebas	
Código	CP - 05
Nombre del caso de Prueba	Cálculo de aguinaldo
Fecha de elaboración	10/7/2025
Autor	Miguel Aguilar
Precondiciones:	
El colaborador debe tener registros salariales durante el periodo correspondiente	
Prioridad:	Alta
Descripción:	

Este caso de prueba verifica que el sistema realice correctamente el cálculo de aguinaldo conforme a los salarios registrados durante el periodo anual			
Detalle de resultados			
Datos de entrada	Resultado esperado	Resultado Actual	Estado
Seleccionar periodo correcto	Calcular aguinaldo correctamente	Resultado esperado obtenido	Prueba exitosa
Periodo incompleto	Mostrar advertencia	Resultado esperado obtenido	Prueba exitosa

Fuente: Elaboración Propia (2026)

Este caso de prueba tiene como objetivo verificar que el sistema realice correctamente el cálculo del aguinaldo de los colaboradores con base en los salarios registrados durante el periodo correspondiente.

### Tabla 98

#### *Caso de prueba - Procesamiento de nómina*

RRHH Pruebas	
Código	CP - 06
Nombre del caso de Prueba	Procesar nómina
Fecha de elaboración	10/7/2025
Autor	Miguel Aguilar
Precondiciones:	
Los colaboradores deben estar registrados y con asistencia registrada	
Prioridad:	Alta
Descripción:	
Este caso de prueba verifica que el sistema calcule correctamente la nómina de los colaboradores considerando deducciones y salarios	
Detalle de resultados	

Datos de entrada	Resultado esperado	Resultado Actual	Estado
Generar nómina con datos válidos	Calcular salario correctamente	Resultado esperado obtenido	Prueba exitosa
Error en cálculo	Mostrar alerta	Resultado esperado obtenido	Prueba exitosa

Fuente: Elaboración Propia (2026)

Este caso de prueba tiene como objetivo comprobar que el sistema procese correctamente la nómina de los colaboradores, realizando los cálculos correspondientes de salarios, deducciones y otros pagos.

### Tabla 99

#### *Caso de prueba - Liquidación*

RRHH Pruebas			
Código	CP - 07		
Nombre del caso de Prueba	Cálculo de liquidación		
Fecha de elaboración	10/7/2025		
Autor	Miguel Aguilar		
Precondiciones:			
El colaborador debe estar registrado y tener fechas de salida			
Prioridad:	Alta		
Descripción:			
Este caso de prueba verifica el cálculo automático de la liquidación laboral considerando preaviso, cesantía y vacaciones pendientes			
Detalle de resultados			
Datos de entrada	Resultado esperado	Resultado Actual	Estado

Ingresar datos correctos	Calcular liquidación correctamente	Resultado esperado obtenido	Prueba exitosa
Campos incompletos	Mostrar evidencia	Resultado esperado obtenido	Prueba exitosa

Fuente: Elaboración Propia (2026)

Este caso de prueba tiene como objetivo verificar que el sistema realice correctamente el cálculo de la liquidación de un colaborador al finalizar su relación laboral, considerando los diferentes montos correspondientes.

## CAPITULO VI: CONCLUSIONES Y RECOMENDACIONES

El presente capítulo tiene como propósito exponer las conclusiones y recomendaciones derivadas del proceso de análisis, diseño, desarrollo y validación del sistema web de gestión del pago de planilla implementado para la empresa Tendenzza & puerTalum. A partir del cumplimiento del objetivo general y de los objetivos específicos planteados durante el desarrollo del proyecto, se presentan los principales resultados obtenidos, así como los aportes del sistema al proceso de gestión de Recursos Humanos dentro de la organización.

Asimismo, se analizan los beneficios que aporta la implementación del sistema en términos de automatización de procesos, mejora en la eficiencia administrativa y reducción de errores en el manejo de información del personal. Finalmente, se plantean una serie de recomendaciones orientadas a fortalecer el uso del sistema, garantizar su mantenimiento y facilitar su evolución futura dentro de la empresa.

### **Conclusiones**

El desarrollo del sistema web de gestión del pago de planilla para la empresa Tendenzza & puerTalum permitió evidenciar la importancia de la implementación de herramientas tecnológicas en los procesos administrativos relacionados con la gestión de Recursos Humanos. A través del desarrollo del prototipo se logró cumplir con el objetivo general planteado, el cual consistía en desarrollar un sistema web mediante el uso de PHP, JavaScript, Laravel, jQuery y MySQL, capaz de optimizar el control y la administración de procesos como asistencia, permiso, vacaciones, horas extra, nómina, incapacidades, liquidaciones, y cálculo de aguinaldo.

En primer lugar, el análisis de los requerimientos funcionales y no funcionales permitió identificar las principales problemáticas que enfrentaba la empresa en la gestión manual de horas extra, el cálculo de nómina y retrasos en los procesos administrativos. A

partir de este análisis fue posible definir los módulos necesarios para el desarrollo del sistema, alineando la solución tecnológica con las necesidades reales de la organización.

En relación con el diseño del sistema, se estableció una arquitectura basada en el patrón Modelo-Vista-Controlador (MVC), lo que permitió organizar de forma estructurada los componentes del sistema y facilitar su mantenimiento y escalabilidad. Asimismo, se diseñó una base de datos relacional utilizando MySQL, en la cual se definieron las tablas, relaciones y restricciones necesarias para garantizar la integridad de los datos almacenados. Este diseño permitió estructurar adecuadamente la información relacionada con los colaboradores, las asistencias, permisos, vacaciones y procesos de planilla.

Durante la fase de desarrollo se implementaron los distintos módulos del sistema utilizando el framework Laravel, integrando herramientas de desarrollo frontend como Blade, jQuery, Bootstrap, y la plantilla Velzon, lo que permitió construir una interfaz moderna, responsiva e intuitiva para los usuarios del sistema. La implementación de estas tecnologías permitió automatizar procesos que anteriormente se realizaban de manera manual, reduciendo el riesgo de errores humanos y mejorando la eficiencia en la gestión administrativa.

Finalmente, la fase de pruebas permitió verificar el correcto funcionamiento de los módulos desarrollados, mediante la ejecución de casos de prueba orientados a validar la integración entre el backend desarrollado en Laravel y la interfaz frontend basada en Blade. Los resultados obtenidos demostraron que el sistema responde adecuadamente ante diferentes escenarios de uso, garantizando la correcta ejecución de los procesos y la confiabilidad de la información almacenada.

En consecuencia, se concluye que el sistema desarrollado representa una solución tecnológica viable para Tendenzza & puerTalum, ya que contribuye a la automatización de procesos clave de Recursos Humanos, mejora la eficiencia operativa y fortalece el control de la información administrativa dentro de la empresa.

## Recomendaciones

Una vez finalizado el desarrollo del sistema web para la empresa Tendenzza & puerTalum, se plantean las siguientes recomendaciones orientadas a garantizar el uso adecuado del sistema y facilitar su evaluación futura de la organización.

En primer lugar, se recomienda realizar un proceso de capacitación dirigido al personal del área administrativa y de Recursos Humanos, con el objetivo de garantizar el uso correcto de las funcionalidades del sistema. Esta capacitación debería ser impartida por un profesional con conocimientos en desarrollo web con PHP, manejo del framework Laravel, administración de bases de datos MySQL y gestión de sistemas empresariales. Se estima que esta capacitación tenga una duración aproximada de dos semanas, tiempo suficiente para explicar el funcionamiento general del sistema, los módulos disponibles y las buenas prácticas de uso. El costo estimado para este proceso podría oscilar entre los ¢600 000 y ¢800 000 colones, considerando la contratación temporal de un profesional especializado.

En segundo lugar, se recomienda implementar un proceso de mantenimiento técnico periódico del sistema, con el fin de garantizar su correcto funcionamiento, realizar actualizaciones de seguridad y resolver posibles errores que puedan surgir durante su uso. Este mantenimiento debería ser realizado por un desarrollador web con experiencia en Laravel, PHP y administración de base de datos MySQL. Se proyecta que estas tareas de mantenimiento se realicen de manera semestral, con un costo aproximado de ¢200 000 colones por cada revisión técnica.

Asimismo, se recomienda considerar la expansión futura del sistema, incorporando un módulo de gestión de clientes y proyectos, que permita registrar clientes, llevar el control de los proyectos activos y establecer fechas de inicio y finalización para cada uno. Esta funcionalidad ayudaría a mejorar la organización y el seguimiento de las actividades de la empresa. Su desarrollo podría ser realizado por un desarrollador full stack con experiencia en Laravel y MySQL, en un tiempo estimado de dos a tres meses, con un aspecto aproximado

entre ₡1 800 000 y ₡2 200 000 colones, dependiendo de las funcionalidades que se implementen.

Finalmente, se recomienda que el administrador del sistema, en coordinación con el departamento de TI de Tendenzza & Puertalum, establezca políticas internas de respaldo y seguridad de la información. Se sugiere realizar copias de seguridad de la base de datos de forma semanal, o posterior a cada cierre de nómina mensual, y almacenarlas en el servidor (VPS) contratado por la empresa. Asimismo, se debe garantizar la protección de los datos personales de los colaboradores conforme a la normativa vigente. Estas medidas permitirán prevenir pérdidas de información y asegurar la continuidad operativa del sistema ante cualquier eventualidad técnica.

## Referencias

- Admin. (2025, 23 de febrero). *El futuro del desarrollo de software: 15 tendencias clave para 2025*. INFORMÁTICA y TECNOLOGÍA.  
<https://www.codigosinformaticos.com/futuro-del-desarrollo-de-software/>
- Alexandrea, J. (2025, 28 de abril). What is bootstrap? Hostinger Tutorials.  
<https://www.hostinger.com/tutorials/what-is-bootstrap>
- Ali, I. (2025). Seguridad en Laravel: Buenas prácticas para proteger tu aplicación. Cloudways. Recuperado de <https://www.cloudways.com/es/blog/seguridad-laravel/>
- A, J. (2024, 26 de julio). *Qué es la Arquitectura Web | Componentes y Tipos*. Niauxus - Blog de Desarrollo Web. <https://niaxus.com/2024/07/26/que-es-la-arquitectura-web-componentes-tipos/>
- Astera. (2024, 16 agosto). Integridad de datos en una base de datos: ¿por qué es importante? <https://www.astera.com/knowledge-center/data-reliability/>
- Auth0. (2022, 14 septiembre). Adding salt to hashing: A better way to store passwords. Auth0. <https://auth0.com/blog/adding-salt-to-hashing-a-better-way-to-store-passwords/>
- AWS. (s. f.). What is a framework in programming and engineering? Amazon Web Services. <https://aws.amazon.com/what-is-framework/>
- AWS. (s.f.). Seguridad y frameworks. Recuperado de <https://aws.amazon.com/compare/the-difference-between-monolithic-and-microservices-architecture/>
- Babu, R. (2019). *SQL Server Stored Procedures*. SQL Server Tutorial. <https://www.sqlservertutorial.net/sql-server-stored-procedures>
- Boada, D., & Boada, D. (2025, 23 de abril). *Guía para principiantes: ¿Qué es el desarrollo de aplicaciones web?* ES Tutoriales. <https://www.hostinger.com/es/tutoriales/que-es-el-desarrollo-de-aplicaciones-web>

- Bootstrap: Qué es, cómo usarlo y ventajas en 2025. (2025, 18 de febrero). *Blog HostingTG*.  
<https://www.hostingtg.com/blog/bootstrap-que-es/>
- B, G., & B, G. (2025, 13 de marzo). *¿Qué es MySQL? Explicación detallada para principiantes*. ES Tutoriales. <https://www.hostinger.com/es/tutoriales/que-es-mysql>
- Cifrar Online. (2023). Ejemplo de encriptación de contraseñas. <https://cifraronline.com>
- Computer Weekly. (2023). Database normalization in MySQL: Four quick and easy steps. <https://www.computerweekly.com/tutorial/Database-normalization-in-MySQL-Four-quick-and-easy-steps>
- ClasificaciónDe.org. (2022, 9 de mayo). Tipos de fuentes: ¿Cómo se clasifican? ClasificaciónDe.org. <https://www.clasificacionde.org/tipos-de-fuentes/>
- Cordero, F. (2023). Sistemas inteligentes para la gestión administrativa. *Revista de Tecnología Aplicada*, 15(1), 28–34. Recuperado de <https://www.studocu.com/es-mx/document/centro-de-estudios-docentes-e-investigacion/tecnologia-de-la-informacion/ensayo-sistemas-inteligentes-2023/74612076>
- Contributors, M. o. J. T. A. B. (s. f.). Get started with Bootstrap. <https://getbootstrap.com/docs/5.3/getting-started/introduction/>
- Datasunrise. (2023). Data integrity in MySQL. <https://www.datasunrise.com/knowledge-center/data-integrity-in-mysql/>
- Dupuis, A. (2024, 17 octubre). Tipos de variables en una investigación. *Técnicas de Investigación*. <https://tecnicasdeinvestigacion.com/tipos-de-variables/>
- Establishing a secure connection . . . (s. f.). SciELO. <https://www.scielo.br/j/cenf/a/bQ7Rtq5WRLKMj8RvX8fMXkD/?format=html&lang=pt>
- FreeCodeCamp. (2023). *What is a software framework?* FreeCodeCamp News. <https://www.freecodecamp.org/news/what-is-a-framework-software-frameworks-definition/>
- Gómez, R. (2023). *Sistemas integrados para RRHH*. Editorial Técnica Corporativa. Recuperado de <https://costaricacrm.com/21-mejores-software-de-recursos-humanos-con-integracion-de-crm/>
- González Rivera, P. L. (2024). Criterios actualizados sobre la metodología de la investigación educativa: Una aproximación bibliográfica. *Revista Mendive*, 22(1).

Recuperado de: [http://scielo.sld.cu/scielo.php?script=sci\\_arttext&pid=S1815-76962024000100031](http://scielo.sld.cu/scielo.php?script=sci_arttext&pid=S1815-76962024000100031)

- Harsh, K. (2025, 17 de enero). *¿Qué es la Arquitectura de las Aplicaciones Web? Desglosando una Aplicación Web*. Kinsta®. <https://kinsta.com/es/blog/arquitectura-aplicaciones-web/>
- Herrera, M. (2024). Arquitectura de software para la expansión organizacional. Editorial Digital Avanzada. Recuperado de <https://multiplica.com/conocimiento/articulos/lar-arquitectura-de-software-como-motor-del-crecimiento-empresarial/>
- Hernández-Sampieri, R., & Mendoza, C. (2018). *Metodología de la investigación: Las rutas cuantitativa, cualitativa y mixta* (6.ª ed.). McGraw-Hill Education. Recuperado de: <https://virtual.cuautitlan.unam.mx/rudics/?p=2612>
- Hostinger. (2025). What is Laravel? Hostinger Tutorials. <https://www.hostinger.com/tutorials/what-is-laravel>
- Hostinger. (s.f.). ¿Qué es una aplicación web?. Recuperado de <https://www.hostinger.es/tutoriales/aplicacion-web>
- IBM. (2023). *¿Qué es la automatización?*. IBM Cloud Learn Hub. <https://www.ibm.com/cloud/learn/automation>
- Introduction | laravel-permission. (s. f.). Spatie. <https://spatie.be/docs/laravel-permission/v5/introduction>
- Kinsta. (2023, 11 de diciembre). Alojamiento Laravel Rápido con Despliegues Sencillos y Soporte 24/7. Kinsta®. <https://kinsta.com/es/alojamiento-de-aplicaciones/laravel/>
- Kinsta. (s. f.). What is Laravel and why do I use it? Kinsta Knowledge Base. <https://kinsta.com/knowledgebase/what-is-laravel/>
- Laravel. (s. f.). Hashing. Laravel. <https://laravel.com/docs/12.x/ hashing>
- Laravel News. (2023). Laravel Ecosystem Overview. Recuperado de <https://laravel-news.com/laravel-ecosystem-video-series>
- Lacorzana, V. (2021, 11 de agosto). Fuentes de información, tipos y sus características. 5poder.org. . <https://5poder.org/2021/08/fuentes-de-informacion-tipos-y-sus-caracteristicas/>
- Lacruz, M. (2022). *Recursos humanos: definición, concepto y funciones*. OpenHR Blog. <https://www.openhr.cloud/blog/recursos-humanos-funciones-dentro-de-la-empresa>

López-Sisniega, C. L., Gutiérrez-Diez, M. del C., & Bordas-Beltrán, J. L. (2021). *Calidad en el desarrollo de software en economías emergentes versus clase mundial: caso Chihuahua, México*. Información Tecnológica, 32(6).

[https://www.scielo.cl/scielo.php?script=sci\\_arttext&pid=S0718-07642021000600191](https://www.scielo.cl/scielo.php?script=sci_arttext&pid=S0718-07642021000600191)

Lowry, C. (2023, 15 de marzo). *Fuentes primarias, secundarias y terciarias*. LibreTexts español.

[https://espanol.libretexts.org/Bookshelves/Humanidades/Investigacion\\_y\\_Alphabetizacion\\_Informacional/Elegir\\_y\\_usar\\_fuentes\\_-\\_Una\\_guia\\_para\\_la\\_investigacion\\_academica\(Lowry\)/02%3A\\_Tipos\\_de\\_Fuentes/2.05%3A\\_Fuentes\\_Primarias\\_Secundarias\\_y\\_Terciarias](https://espanol.libretexts.org/Bookshelves/Humanidades/Investigacion_y_Alphabetizacion_Informacional/Elegir_y_usar_fuentes_-_Una_guia_para_la_investigacion_academica(Lowry)/02%3A_Tipos_de_Fuentes/2.05%3A_Fuentes_Primarias_Secundarias_y_Terciarias)

Ministerio de Trabajo y Seguridad Social. (2023). Lista de salarios mínimos año 2023.

<https://www.mtss.go.cr/temas-laborales/salarios/LISTA%20DE%20SALARIOS%20MINIMOS%20%20ANO%202023.pdf>

Miro. (2025, 6 marzo). Prototipado rápido: Qué es y cómo puede ayudarte.

<https://help.miro.com/hc/es/articles/26654269713682-Miro-Prototypes-Disponibilidad-limitada>

Molina, G. (2024). Gestión por indicadores en entornos digitales. Editorial Empresarial Avanzada. Recuperado de

[https://lamolinavirtual.org/capacitaciones/capacitaciones3\\_view.php?page=view&editid1=12788](https://lamolinavirtual.org/capacitaciones/capacitaciones3_view.php?page=view&editid1=12788)

Muñoz, A., & Muñoz, A. (2022, 18 de julio). *¿Qué es frontend y backend?* Asociación Conexión El Salvador. <https://conexion.sv/que-es-frontend-y-backend/>

Omatech. (2023). Ventajas de las aplicaciones web. Recuperado de

<https://www.omatech.com/blog/2022/08/31/que-es-una-web-spa/>

Omatech. (2023, 30 marzo). Ventajas de los frameworks. <https://abamobile.com/web/what-is-a-framework-types-and-advantages/>

OpenJS Foundation. (s. f.). *jQuery*. <https://jquery.com/>

Pérez, L. (2024). Transformación digital en la gestión del talento. Revista de Innovación Empresarial, 13(2), 28–35. Recuperado de [https://www.g-](https://www.g-talent.net/blogs/articulos/recursos-humanos-y-la-transformacion-digital-en-2024-un-paso-hacia-el-futuro)

[talent.net/blogs/articulos/recursos-humanos-y-la-transformacion-digital-en-2024-un-paso-hacia-el-futuro](https://www.g-talent.net/blogs/articulos/recursos-humanos-y-la-transformacion-digital-en-2024-un-paso-hacia-el-futuro)

- PhpLightDoc. (s. f.). *Free Open-Source API Documentation Generator for Laravel*.  
[https://phplightdoc.ru/en?utm\\_source=laravelnews&utm\\_medium=link&utm\\_campaign=laravelnews](https://phplightdoc.ru/en?utm_source=laravelnews&utm_medium=link&utm_campaign=laravelnews)
- Piña-Ferrer, L. S. (2023). *El enfoque cualitativo: Una alternativa compleja dentro del mundo de la investigación*. Revista Koinonía, 8(15). Recuperado de:  
[https://ve.scielo.org/scielo.php?script=sci\\_arttext&pid=S2542-30882023000100001](https://ve.scielo.org/scielo.php?script=sci_arttext&pid=S2542-30882023000100001)
- Programación.net. (s. f.). *Cómo encriptar datos sensibles en bases de datos*.  
[https://www.programacion.net/articulo/como\\_encriptar\\_datos\\_sensibles\\_en\\_bases\\_de\\_datos](https://www.programacion.net/articulo/como_encriptar_datos_sensibles_en_bases_de_datos)
- Rodríguez, A., & Méndez, C. (2022). Estrategia de pruebas para organizaciones desarrolladoras de portales web. Revista Cubana de Ciencias Informáticas, 16(3).  
[http://scielo.sld.cu/scielo.php?script=sci\\_arttext&pid=S2227-18992022000300118](http://scielo.sld.cu/scielo.php?script=sci_arttext&pid=S2227-18992022000300118)
- Rodríguez, O. (2025). El futuro del diseño web responsivo. UXVerse Studio. Recuperado de <https://www.uxversestudio.com/es/blog/the-future-of-responsive-web-design-in-2025>
- Salazar, D. (2023). Integración de sistemas en entornos corporativos. Revista de Tecnología Empresarial, 14(1), 14–20. Recuperado de  
<https://www.sydle.com/es/blog/integracion-de-sistemas-6140d39a84679b13bf127a93>
- Santos, M. (2022). Técnicas de recolección de datos en investigación aplicada. Revista de Ciencias Sociales y Tecnología, 18(3), 33–40.
- Santos, D. (2025, 10 de abril). Cómo recolectar datos: métodos, ejemplos y herramientas clave. HubSpot. <https://blog.hubspot.es/marketing/recoleccion-de-datos>
- Sommerville, I. (2020). *Software Engineering* (10.<sup>a</sup> ed.). Pearson Education. [Resumen de requerimientos de ingeniería de software - Sommerville - Studocu](#)
- Suárez, E. (2023). *Tipos de investigación y su clasificación: Guía completa*. Experto Universitario. Recuperado de: <https://expertouniversitario.es/blog/tipos-de-investigacion/>
- Snyk. (2023, 15 junio). Password storage best practices. Snyk.  
<https://snyk.io/articles/password-storage-best-practices/>

- Tejera-Martínez, F., Aguilera, D., & Vílchez-González, J. M. (2020). Lenguajes de programación y desarrollo de competencias clave: Revisión sistemática. *Revista Electrónica de Investigación Educativa*, 22.  
[https://www.scielo.org.mx/scielo.php?script=sci\\_arttext&pid=S1607-40412020000100127](https://www.scielo.org.mx/scielo.php?script=sci_arttext&pid=S1607-40412020000100127)
- TecnoDigital. (2025, 10 de agosto). *Qué es jQuery y por qué sigue siendo clave en desarrollo web*. *Informática y Tecnología Digital*.  
<https://informatecdigital.com/que-es-jquery-y-por-que-sigue-siendo-clave-en-desarrollo-web/>
- ThemeForest. (s. f.). *Velzon – ASP.NET Core admin & dashboard template*. ThemeForest.  
<https://themeforest.net/item/velzon-aspnet-core-admin-dashboard-template/36077495>
- Universidad San Marcos. (2020). *Investigación, enfoques y el problema investigativo. Parte I* [PDF]. Recuperado de:  
<https://repositorio.usam.ac.cr/xmlui/bitstream/handle/11506/1391/LEC%20MET%200011%202020.pdf?sequence=1>
- W3C. (2021). *Web design and accessibility guidelines*. World Wide Web Consortium.  
<https://www.w3.org/>

## Apéndice

### Apéndice A: Guía de Entrevista Aplicada al Área de Recursos Humanos

#### GUÍA DE ENTREVISTA

**Organización: Tendenzza & puertAlum**

**Nombre del Entrevistado: Jaqueline Rivera Zúñiga**

**Cargo: Recursos Humanos**

#### **Preguntas:**

- ¿Cómo se realiza actualmente el registro de nuevos colaboradores?
- ¿Qué tipo de información se almacena sobre cada empleado?
- ¿Existe algún sistema digital para gestionar expedientes laborales?
- ¿Cuál es el procedimiento para solicitar permisos laborales?
- ¿Quién valida o aprueba estas solicitudes?
- ¿Se lleva un registro histórico de permisos otorgados?
- ¿Cómo se gestionan las solicitudes de vacaciones?
- ¿Existen algún calendario o sistema que controle los días acumulados por cada empleado?
- ¿Cómo se realiza el cálculo del aguinaldo actualmente?
- ¿Qué herramienta se utiliza para este proceso (Excel, software,)?
- ¿Qué aspectos considera que podrían mejorar si se automatizan estos procesos?
- ¿Ha tenido experiencia previa con sistemas automatizados en otras áreas?
- ¿Cómo se lleva el control de las incapacidades reportadas por los colaboradores?
- ¿Cuál es el procedimiento actual para generar la nómina semanal o mensual?
- ¿Se mantiene algún registro histórico de horas extra, ausencias o maracas realizadas por los colaboradores?
- ¿Qué tipo de errores son más frecuentes durante el cálculo de la planilla en salarios, deducciones, bonificaciones entre otros?
- ¿Qué problemas suelen presentarse cuando un colaborador solicita vacaciones?

¿Con que frecuencia se detectan pagos incorrectos en horas extra y cómo se corrigen actualmente?

¿Qué problemas se generan cuando se intenta consultar el historial de permisos, incapacidades o ausencias de un colaborador?

## Apéndice B: Guía de Entrevista Aplicada al Área de Recursos Humanos

### GUÍA DE OBSERVACIÓN

**Nombre de la Empresa:** Tendenzza & puertAlum

**Actividad de la Empresa:** Soluciones en vidrio de alta gama y automatización de espacios comerciales, residenciales y corporativos.

**OBJETIVO:** Observar y evaluar las actividades relacionadas con el proyecto de automatización de procesos en el área de Recursos Humanos de la empresa Tendenzza & puertAlum, específicamente en lo referente a la gestión de permisos, vacaciones, cálculo de aguinaldos, control de asistencia

No	Aspectos por observar	Cumple	No Cumple	Oportunidad de mejora	Detalle de Observación
1	Registro de permisos laborales.	X			Se realizan mediante formularios físicos entregados al jefe de área.
2	Validación de solicitudes por jefatura.	X			El jefe revisa manualmente cada solicitud y firma en papel.
3	Cálculo de aguinaldos.		X	X	Se realiza en Exel sin fórmulas automatizadas, y hay riesgo de error humano.

No	Aspectos por observar	Cumple	No Cumple	Oportunidad de mejora	Detalle de Observación
4	Uso de herramientas digitales.		X	X	No se utiliza ningún sistema para gestionar Recursos Humanos, ya que todo es manual.
5	Registro histórico de vacaciones.	X			Se lleva en una hoja de cálculo compartida entre jefaturas.
6	Comunicación entre áreas sobre permisos.		X	X	No hay trazabilidad ni confirmación digital ya que se pierde la información.

<b>No</b>	<b>Aspectos por observar</b>	<b>Cumple</b>	<b>No Cumple</b>	<b>Oportunidad de mejora</b>	<b>Detalle de Observación</b>
7	Seguridad de la información.		X	X	Los archivos están en carpetas locales sin respaldo ni cifrado.