

UNIVERSIDAD INTERNACIONAL DE LAS AMÉRICAS

FACULTAD DE INGENIERÍA INFORMÁTICA

TRABAJO FINAL DE GRADUACIÓN PARA OPTAR POR EL GRADO DE

BACHILLER EN INGENIERÍA DE INFORMÁTICA

Sistema web para la gestión y pago de planilla en AraWeb CR, ubicada en San José.

Emmanuel Cerdas Brenes

Tutor:

Fabián Rodríguez Sibaja

San José

Julio, 2026

CONTENIDO

CAPÍTULO I: INTRODUCCIÓN	29
Descripción del Problema.....	29
Problemáticas de la Empresa	29
Objetivos.....	30
Objetivo General.....	30
Objetivos Específicos.....	30
Justificación	31
Viabilidad Técnica	32
Viabilidad Operativa.....	34
Viabilidad Económica.....	35
Viabilidad Legal.....	37
Proyecciones.....	39
Alcance Funcional	40
Gestionar Nóminas	40
Gestionar Asistencia.....	40
Gestionar Horas Extras.....	40
Gestionar Permisos.....	40
Gestionar Incapacidades.....	40
Calcular Vacaciones	40
Calcular Aguinaldo.....	40
Gestionar Liquidación	40
Evaluar Personal.....	41
Mantenimientos	41
Consultas	41
Reportes.....	41
Seguridad.....	41
Alcance Metodológico	41
Ciclo de vida del Sistema.	42

Alcance Tecnológico	43
CAPÍTULO II: MARCO REFERENCIAL.....	45
Introducción al marco referencial.....	45
Sistemas de Información	46
Definición de sistema de información	47
Componentes de un sistema de información	48
Importancia de los sistemas de información en las organizaciones.....	50
Sistemas de información aplicados a la gestión de recursos humanos.....	51
Gestión de recursos humanos.....	53
Sistema de planilla como herramienta de gestión.....	54
Automatización de procesos de planilla	56
Desarrollo de software web	58
Desarrollo de sistemas web.....	58
Arquitectura cliente-servidor	59
Metodologías ágiles de desarrollo de software.....	59
Aspectos generales de la normativa laboral y tecnológica	60
Importancia del cumplimiento legal en sistemas informáticos.....	60
Protección de datos personales en sistemas de información.....	61
Relación entre tecnología y legislación laboral	61
Marco legal costarricense (enfoque general)	62
CAPITULO III: MARCO METODOLÓGICO	63
Enfoques de Investigación.....	63
Enfoque Cuantitativo	63
Enfoque Cualitativo	64
Enfoque Mixto	66
Enfoque de Investigación Seleccionado	67
Tipos de Investigación.....	67
Investigación Exploratoria	68
Investigación Descriptiva.....	69
Investigación Aplicada.....	70
Tipo de Investigación Seleccionado	71
Fuentes de Información	72

Fuentes de Información Primarias	72
Fuentes de Información Secundarias	73
Fuentes de Información Terciarias.....	73
Índices, bibliografías y catálogos	73
Variables.....	74
Variables de Investigación.....	74
Variables Conceptuales.....	74
Variables Operacionales	74
Variables Instrumentales.....	75
Población	77
Muestra	77
Cálculo de la muestra.....	78
Instrumentos de recolección de datos	79
La Observación	79
La Encuesta.....	80
Riesgos que conlleva la aplicación de cuestionarios	81
Documentos de archivo y fuentes gubernamentales.....	82
Documentos de archivo.....	82
Fuentes gubernamentales	83
Proceso para la Recolección y Análisis de Datos	84
CAPITULO IV: ANALISIS DE RESULTADOS	86
Cuestionario.....	86
Análisis interpretativo del cuestionario	87
Observación Directa	99
Análisis de la Observación Directa.....	99
CAPÍTULO V: PROPUESTA	101
Análisis	101
Análisis detallado del Software por desarrollar	101
Módulo Gestionar Nóminas	101
Módulo Gestionar Asistencia	101

Módulo Gestionar Horas Extras	102
Módulo Gestionar Permisos	102
Módulo Gestionar Incapacidades	102
Módulo Calcular Vacaciones	102
Módulo Calcular Aguinaldo	103
Módulo Gestionar Liquidación	103
Módulo Evaluar Personal	103
Módulo Mantenimientos	103
Módulo Consultas	103
Módulo Reportes	104
Módulo Seguridad	104
Análisis detallado del hardware requerido	104
Análisis detallado de los elementos relacionados con las telecomunicaciones	105
Descripción detallada de las herramientas técnicas utilizadas para el desarrollo	107
Descripción detallada del conocimiento básico que debe tener el recurso humano que operará el sistema.....	108
Diseño de casos de uso.	110
Diseño	132
Arquitectura del sistema	132
Arquitectura del software.....	134
Diseño de entradas	136
Diseño físico de la base de datos	141
Diccionario de Datos.....	143
Diseño de procesos	173
Diseño de salidas.....	180
Diagramas UML	185
Diagrama de Clases	185
Diagrama de Secuencia	187
Programación.....	192

Entradas.....	192
Recepción de credenciales de inicio de sesión (Módulo de Seguridad)	192
Registro de solicitud de permiso laboral (Módulo Gestionar Permisos).....	193
Salidas.....	194
Listado de sesiones activas del sistema (Módulo de seguridad)	194
Comprobante de nómina individual del colaborador (Módulo Gestionar Nóminas)	195
Procesos	196
Cálculo de aguinaldo según legislación costarricense (Módulo Calcular Aguinaldo)	196
Sincronización automática de marcaciones a registros de asistencia (Módulo Gestionar Asistencia).....	198
Validaciones.....	199
Reglas de negocio para el registro de marcaciones (Módulo Gestionar Asistencia).....	199
Validación de datos de incapacidad laboral (Módulo Gestionar Incapacidad)	200
Módulos señalados en el alcance	201
Gestionar Nóminas – Proceso masivo de cálculo y guardado de nómina.....	201
Gestionar Asistencia – Registro de marcación de entrada y salida	203
Gestionar Horas Extra – Aprobación de solicitudes por jefaturas.....	206
Gestionar Permisos – Aprobación de permiso con registro de historial.....	207
Gestionar Incapacidades – Impacto en registros de asistencia al aprobar.....	208
Calcular Vacaciones – Cálculo y presentación del saldo disponible	209
Calcular Aguinaldo – Servicio de cálculo proporcional por período	210
Gestionar Liquidación – Validaciones previas al cálculo de liquidación laboral.....	212
Evaluar Personal – Creación de evaluación con criterios y ponderación.....	213
Mantenimientos – Creación de horario semanal con días configurables	215
Consultas – Perfil 360 del empleado con información consolidada	216
Reportes – Generación de reportes de asistencia por empleado y período	217
Seguridad – Autenticación con hash PBKDF2 y emisión de ticket FormsAuth	218

Pruebas.....	221
CAPÍTULO VI: CONCLUSIONES Y RECOMENDACIONES.....	241
Conclusiones.....	241
Recomendaciones	244
Recomendación 1: Envío automático del comprobante de nómina por correo electrónico .	244
Recomendación 2: Generación automática de constancias y cartas de trabajo	245
Recomendación 3: Plan de capacitación al personal sobre el sistema	245
REFERENCIAS	247

TABLAS

Tabla 1 <i>Hardware y software requeridos para el desarrollo e implementación del sistema.</i>	33
Tabla 2 <i>Costos estimados del proceso de desarrollo del sistema.</i>	36
Tabla 3 <i>Desglose de costos estimado de licencias e infraestructura tecnológica.</i>	37
Tabla 4 <i>Cuadro de variables.</i>	76
Tabla 5 <i>Caso de Uso CU-01 Ingreso al Sistema.</i>	112
Tabla 6 <i>Caso de Uso CU-02 Administración de Usuarios.</i>	113
Tabla 7 <i>Caso de Uso CU-03 Administración de Empleados.</i>	114
Tabla 8 <i>Caso de Uso CU-04 Registro de Asistencia.</i>	115
Tabla 9 <i>Caso de Uso CU-05 Gestión de Solicitudes.</i>	116
Tabla 10 <i>Caso de Uso CU-06 Generación de Planilla.</i>	118
Tabla 11 <i>Caso de Uso CU-07 Registro de Incapacidades.</i>	119
Tabla 12 <i>Caso de Uso CU-08 Evaluar Personal.</i>	120
Tabla 13 <i>Caso de Uso CU-09 Cálculo Aguinaldo.</i>	121
Tabla 14 <i>Caso de Uso CU-10 Gestión de Liquidación Laboral.</i>	122
Tabla 15 <i>Caso de Uso CU-11 Gestionar Nómina.</i>	123
Tabla 16 <i>Caso de Uso CU-12 Consultar Asistencia.</i>	124
Tabla 17 <i>Caso de Uso CU-13 Consultar Comprobante Salarial.</i>	125
Tabla 18 <i>Caso de Uso CU-14 Consultar Historial Salarial.</i>	126
Tabla 19 <i>Caso de Uso CU-15 Gestionar Roles.</i>	127
Tabla 20 <i>Caso de Uso CU-16 Configurar Parámetros del Sistema.</i>	128
Tabla 21 <i>Caso de Uso CU-17 Administrar Seguridad.</i>	129
Tabla 22 <i>Caso de Uso CU-18 Auditoría del Sistema.</i>	130
Tabla 23 <i>Estructura de la tabla Aguinaldo</i>	145
Tabla 24 <i>Estructura de la tabla AportesSeguridadSocial</i>	145
Tabla 25 <i>Estructura de la tabla Asistencia</i>	146
Tabla 26 <i>Estructura de la tabla AuditoriaGeneral</i>	148
Tabla 27 <i>Estructura de la tabla Ausencia</i>	148
Tabla 28 <i>Estructura de la tabla Bitácora.</i>	149
Tabla 29 <i>Estructura de la tabla Bonificacion</i>	149
Tabla 30 <i>Estructura de la tabla Canton</i>	150
Tabla 31 <i>Estructura de la tabla ClaseHoraExtra</i>	150

Tabla 32 Estructura de la tabla Credencial.....	151
Tabla 33 Estructura de la tabla Departamento	151
Tabla 34 Estructura de la tabla DetalleAporte	152
Tabla 35 Estructura de la tabla DetalleNomina	152
Tabla 36 Estructura de la tabla DetalleRenta	154
Tabla 37 Estructura de la tabla Direccion.....	154
Tabla 38 Estructura de la tabla Distrito	155
Tabla 39 Estructura de la tabla Empleado	155
Tabla 40 Estructura de la tabla EmpleadoHorario	157
Tabla 41 Estructura de la tabla Feriado.....	157
Tabla 42 Estructura de la tabla FeriadoLaborado	158
Tabla 43 Estructura de la tabla HistorialBonificacion.....	158
Tabla 44 Estructura de la tabla HistorialHoraExtra.....	159
Tabla 45 Estructura de la tabla HistorialPermiso.....	159
Tabla 46 Estructura de la tabla HistorialVacaciones.....	160
Tabla 47 Estructura de la tabla HoraExtra	160
Tabla 48 Estructura de la tabla Horario	161
Tabla 49 Estructura de la tabla HorarioSemanal.....	161
Tabla 50 Estructura de la tabla Incapacidad.....	162
Tabla 51 Estructura de la tabla Jornada	163
Tabla 52 Estructura de la tabla Liquidacion	163
Tabla 53 Estructura de la tabla Nomina	164
Tabla 54 Estructura de la tabla Permiso	164
Tabla 55 Estructura de la tabla Persona	165
Tabla 56 Estructura de la tabla Provincia.....	166
Tabla 57 Estructura de la tabla Puesto.....	167
Tabla 58 Estructura de la tabla RegistroMarcacion	167
Tabla 59 Estructura de la tabla Rol	168
Tabla 60 Estructura de la tabla SalarioBrutoMensual.....	168
Tabla 61 Estructura de la tabla SesionUsuario	169
Tabla 62 Estructura de la tabla SolicitudHorasExtra	169
Tabla 63 Estructura de la tabla SolicitudVacaciones.....	170

Tabla 64 Estructura de la tabla Tardia.....	170
Tabla 65 Estructura de la tabla TipoCedula.....	171
Tabla 66 Estructura de la tabla TipoIncapacidad	171
Tabla 67 Estructura de la tabla TipoLiquidacion.....	171
Tabla 68 Estructura de la tabla TipoPermiso	172
Tabla 69 Estructura de la tabla TramosParametrosRenta	172
Tabla 70 Estructura de la tabla Vacaciones	173
Tabla 71 Resumen de Cobertura de Pruebas.....	221
Tabla 72 CU-01. Ingreso al Sistema. TC-01-01 - Autenticación exitosa con credenciales válidas	222
Tabla 73 CU-01. Ingreso al Sistema. TC-01-02 - Rechazo de credenciales incorrectas (validación del sistema).....	222
Tabla 74 CU-01. Ingreso al Sistema. TC-01-03 - Bloqueo de acceso a cuenta inactiva (validación del sistema).....	223
Tabla 75 CU-01. Ingreso al Sistema. TC-01-04 - Redirección obligatoria al cambio de contraseña en primer ingreso	223
Tabla 76 CU-02. Administración de Usuarios. TC-02-01 - Registro de nuevo usuario con datos completos.....	224
Tabla 77 CU-02. Administración de Usuarios. TC-02-02 - Control de acceso: módulo bloqueado sin rol Administrador	224
Tabla 78 CU-02. Administración de Usuarios. TC-02-03 - Validación de campos obligatorios en registro de usuario	225
Tabla 79 CU-02. Administración de Usuarios. TC-02-04 - Desactivación de usuario con confirmación.....	225
Tabla 80 CU-03. Administración de Empleados. TC-03-01 - Registro de empleado con información completa	226
Tabla 81 CU-03. Administración de Empleados. TC-03-02 - Validación de cédula duplicada en registro de empleado	226
Tabla 82 CU-03. Administración de Empleados. TC-03-03 - Modificación de datos de un empleado existente.....	227
Tabla 83 CU-03. Administración de Empleados. TC-03-04 - Inactivación de empleado conservando historial.....	227
Tabla 84 CU-04. Registro de Asistencia. TC-04-01 - Registro de entrada laboral.....	228
Tabla 85 CU-04. Registro de Asistencia. TC-04-02 - Registro de salida laboral	228
Tabla 86 CU-04. Registro de Asistencia. TC-04-03 - Prevención de registro duplicado de asistencia.....	229

Tabla 87 CU-05. <i>Gestión de Solicitudes. TC-05-01 - Creación de solicitud de horas extra.....</i>	229
Tabla 88 CU-05. <i>Gestión de Solicitudes. TC-05-02 - Creación de solicitud de vacaciones</i>	230
Tabla 89 CU-05. <i>Gestión de Solicitudes. TC-05-03 - Aprobación de solicitud por jefe inmediato</i>	230
Tabla 90 CU-05. <i>Gestión de Solicitudes. TC-05-04 - Rechazo de solicitud con motivo registrado</i>	231
Tabla 91 CU-05. <i>Gestión de Solicitudes. TC-05-05 - Validación de datos obligatorios en nueva solicitud</i>	231
Tabla 92 CU-06. <i>Generación de Planilla. TC-06-01 - Cálculo completo de nómina mensual con deducciones</i>	232
Tabla 93 CU-06. <i>Generación de Planilla. TC-06-02 - Validación de rango de mes en cálculo de nómina</i>	232
Tabla 94 CU-06. <i>Generación de Planilla. TC-06-03 - Descarga de comprobante salarial en PDF</i>	233
Tabla 95 CU-07. <i>Registro de Incapacidades. TC-07-01 - Registro de incapacidad con fechas y porcentaje válidos</i>	233
Tabla 96 CU-07. <i>Registro de Incapacidades. TC-07-02 - Validación de fecha fin posterior a fecha inicio</i>	234
Tabla 97 CU-07. <i>Registro de Incapacidades. TC-07-03 - Validación de porcentaje de pago dentro de rango</i>	234
Tabla 98 CU-07. <i>Registro de Incapacidades. TC-07-04 - Aprobación de incapacidad con impacto en asistencia</i>	235
Tabla 99 CU-08. <i>Evaluar Personal. TC-08-01 - Creación de evaluación de desempeño con notas válidas</i>	235
Tabla 100 CU-08. <i>Evaluar Personal. TC-08-02 - Validación de notas dentro del rango permitido (1-10)</i>	236
Tabla 101 CU-08. <i>Evaluar Personal. TC-08-03 - Validación de evaluación duplicada por período</i>	236
Tabla 102 CU-09. <i>Cálculo Aguinaldo. TC-09-01 - Cálculo y almacenamiento de aguinaldo anual</i>	237
Tabla 103 CU-09. <i>Cálculo Aguinaldo. TC-09-02 - Validación de rango de año en cálculo de aguinaldo</i>	237
Tabla 104 CU-10. <i>Gestión de Liquidación Laboral. TC-10-01 - Cálculo completo de liquidación laboral</i>	238
Tabla 105 CU-10. <i>Gestión de Liquidación Laboral. TC-10-02 - Validación de fecha no futura en liquidación</i>	238

Tabla 106 <i>CU-11. Gestionar Nómina. TC-11-01 - Cálculo y guardado de nómina mensual completa</i>	239
Tabla 107 <i>CU-11. Gestionar Nómina. TC-11-02 - Validación de rango de año en gestión de nómina</i>	239
Tabla 108 <i>CU-11. Gestionar Nómina. TC-11-03 - Verificación de trazabilidad en Bitácora al guardar nómina</i>	240

FIGURAS

Figura 1. <i>Ciclo de vida del sistema web de gestión y pago de planilla.</i>	43
Figura 2. <i>Cuestionario – Gráfico Pregunta 1.</i>	87
Figura 3. <i>Cuestionario – Gráfico Pregunta 2.</i>	88
Figura 4. <i>Cuestionario – Gráfico Pregunta 3.</i>	89
Figura 5. <i>Cuestionario – Gráfico Pregunta 4.</i>	90
Figura 6. <i>Cuestionario – Gráfico Pregunta 6</i>	91
Figura 7. <i>Cuestionario – Gráfico Pregunta 7.</i>	92
Figura 8. <i>Cuestionario – Gráfico Pregunta 8.</i>	93
Figura 9. <i>Cuestionario – Gráfico Pregunta 9.</i>	93
Figura 10. <i>Cuestionario – Gráfico Pregunta 10.</i>	94
Figura 11. <i>Cuestionario – Gráfico Pregunta 11.</i>	95
Figura 12. <i>Cuestionario – Gráfico Pregunta 12.</i>	96
Figura 13. <i>Cuestionario – Gráfico Pregunta 13.</i>	97
Figura 14. <i>Cuestionario – Gráfico Pregunta 14.</i>	98
Figura 15. <i>Cuestionario – Gráfico Pregunta 15.</i>	99
Figura 16. <i>Diagrama de Casos de uso.</i>	110
Figura 17. <i>Diagrama Arquitectura del sistema</i>	133
Figura 18 <i>Diagrama Arquitectura del software</i>	135
Figura 19 <i>Diseño de entrada - Pantalla Inicio de Sesión</i>	137
Figura 20 <i>Diseño de entrada - Pantalla Solicitar Vacaciones</i>	138
Figura 21 <i>Diseño de entrada - Nueva Evaluación de Desempeño</i>	139
Figura 22 <i>Diseño de entrada - Crear Persona</i>	140
Figura 23 <i>Diseño de entrada – Crear Rol</i>	141
Figura 24. <i>Diagrama del Diseño físico de la base de datos</i>	143
Figura 25 <i>Diagrama de flujo – Proceso de Autenticación e Inicio de Sesión.</i>	175
Figura 26 <i>Diagrama de flujo – Gestión de solicitudes</i>	177
Figura 27 <i>Diagrama de flujo – Generación de Planilla.</i>	179
Figura 28 <i>Diseño de salida – Detalle Nomina Mayo</i>	181
Figura 29 <i>Diseño de salida – Detalle Vacaciones</i>	182
Figura 30 <i>Diseño de salida – Reporte Aguinaldo</i>	183
Figura 31 <i>Diseño de salida – Reporte Asistencia</i>	184

Figura 32 <i>Diseño de salida – Resumen Mensual de Asistencia</i>	185
Figura 33 <i>Diagrama de Clases</i>	186
Figura 34 <i>Diagrama de Secuencia – Inicio de Sesión</i>	188
Figura 35 <i>Diagrama de Secuencia – Gestión de Solicitudes</i>	189
Figura 36 <i>Diagrama de Secuencia – Generación de Planilla</i>	191
Figura 37 <i>Recepción de credenciales de inicio de sesión</i>	192
Figura 38 <i>Registro de solicitud de permiso laboral</i>	193
Figura 39 <i>Listado de sesiones activas del sistema</i>	194
Figura 40 <i>Salida de comprobante de nómina individual</i>	196
Figura 41 <i>Proceso de cálculo del aguinaldo</i>	197
Figura 42 <i>Proceso de sincronización de marcaciones a asistencia</i>	198
Figura 43 <i>Validaciones de reglas de negocio para marcaciones de asistencia</i>	199
Figura 44 <i>Validaciones de datos en el registro de incapacidades</i>	200
Figura 45 <i>Proceso masivo de cálculo de nómina por período</i>	202
Figura 46 <i>Registro de marcación de asistencia de entrada y salida (Parte 1)</i>	203
Figura 47 <i>Registro de marcación de asistencia de entrada y salida (Parte 2)</i>	204
Figura 48 <i>Registro de marcación de asistencia de entrada y salida (Parte 3)</i>	205
Figura 49 <i>Aprobación de solicitudes de horas extra por jefatura</i>	206
Figura 50 <i>Aprobación de solicitud de permiso con registro en historial</i>	207
Figura 51 <i>Proceso de impacto de incapacidad en registros de asistencia</i>	208
Figura 52 <i>Cálculo y presentación del saldo de vacaciones disponible</i>	209
Figura 53 <i>Servicio de cálculo de aguinaldo proporcional</i>	211
Figura 54 <i>Validaciones en el cálculo de liquidación laboral</i>	212
Figura 55 <i>Creación de evaluación de personal con validación de criterios (Parte 1)</i>	213
Figura 56 <i>Creación de evaluación de personal con validación de criterios (Parte 2)</i>	214
Figura 57 <i>Creación de horario semanal con días y horas configurables</i>	215
Figura 58 <i>Consulta del perfil 360 del empleado</i>	217
Figura 59 <i>Formulario de generación de reportes de asistencia</i>	218
Figura 60 <i>Autenticación con PBKDF2 y emisión de ticket de seguridad (Parte 1)</i>	219
Figura 61 <i>Autenticación con PBKDF2 y emisión de ticket de seguridad (Parte 2)</i>	220

APÉNDICE

Apéndice A: Cuestionario.	251
Apéndice B: Guía de Observación.	252

DEDICATORIA

A mí madre, Guiselle María Brenes Quesada, quien con su amor incondicional ha sido el cimiento sobre el que he construido cada paso de mí vida. No existen palabras suficientes para expresar todo lo que representa para mí, pero este logro es, antes que cualquier otra cosa, tuyo. Gracias por cada sacrificio silencioso, por cada noche que velaste por mí sin que yo lo supiera, por nunca dejar de creer en este camino, aunque a veces yo mismo lo dudé. Eres la razón más profunda detrás de este esfuerzo, y espero que al ver estas páginas sientas reflejado todo el amor y la gratitud que llevo en el corazón.

A mí abuelita, Leidalia Quesada Campos, cuya ternura y sabiduría han iluminado mi camino desde siempre. Hay personas que con solo su presencia te enseñan más que cualquier libro, y usted es una de ellas. Me ha mostrado lo que significa la fortaleza tranquila, la bondad sin condiciones y el amor que no necesita demostración porque se siente en cada detalle. Este logro lleva en cada página el amor que me has dado sin pedir nada a cambio, cada momento a tu lado lo atesoro como uno de los más valiosos de mi vida.

A mi hermano, José Daniel Cerdas Brenes, compañero de vida, con quien he compartido los momentos que más importan y que más recuerdo. Crecer a tu lado me enseñó cosas que ninguna carrera universitaria podría haberme dado. Gracias por estar, por las conversaciones que me centraron cuando lo necesitaba, y por ser ese punto fijo en los momentos en que todo lo demás parecía moverse. Este título también tiene algo tuyo.

Y a esas personas que aparecieron en el momento en que mi camino parecía haberse apagado, y sin buscarlo, me devolvió la mirada que necesitaba para recordar el valor de lo que estaba construyendo. Este logro también es, en parte, gracias a ese instante.

A todos ustedes, con todo mi amor y mi gratitud más sincera.

Emmanuel Cerdas Brenes

AGRADECIMIENTOS

En primer lugar, agradezco a Dios, quien conoce cada detalle del camino recorrido, los tiempos que fueron necesarios y las razones detrás de cada experiencia vivida durante este proceso. No siempre entendí por qué las cosas tomaron el tiempo que tomaron, pero hoy, mirando hacia atrás, reconozco su mano en cada etapa. Su guía ha sido evidente no solo en los logros, sino también en los tropiezos y en los momentos de duda, cuando la motivación se había apagado y parecía difícil encontrar una razón para continuar. Fue en uno de esos momentos que apareció una persona que, sin pretenderlo, me devolvió la convicción de que este logro tenía un valor que yo había dejado de ver. Esos encuentros no son casualidad, y por ellos también le doy gracias a Él.

A mí madre, Guiselle María Brenes Quesada, quien fue mi sostén constante durante toda la carrera. Gracias por nunca dejar de apoyarme, por entender los momentos difíciles sin juzgarlos y por celebrar cada pequeño avance como si fuera el más grande. Tu presencia durante este proceso fue fundamental, y este título lleva tu nombre tan grabado como el mío.

A mí abuelita, Leidalía Quesada Campos, por su cariño incondicional y por ser una figura de fortaleza y amor en mi vida. Gracias por cada palabra de aliento, por cada una de sus oraciones y por estar presente en los momentos que más importaron.

A mí hermano, José Daniel Cerdas Brenes, por su compañía, su respaldo y por ser siempre una presencia constante en este camino.

Al Profesor Fabián Rodríguez Sibaja, tutor de este Trabajo Final de Graduación, por su orientación, disposición y guía a lo largo del desarrollo del proyecto. Su acompañamiento profesional y su retroalimentación oportuna fueron piezas clave para llevar esta propuesta a buen término. Le agradezco no solo el conocimiento compartido, sino también la paciencia y el compromiso demostrado durante todo el proceso.

A la Universidad Internacional de las Américas (UIA), por la formación académica brindada durante estos años, por los espacios de aprendizaje que me permitieron desarrollarme como profesional y por ser la institución que hace posible este logro. Agradezco a cada docente que aportó conocimiento, perspectiva y exigencia a lo largo de la carrera.

A los propietarios de AraWeb CR, por abrir las puertas de su empresa con disposición y confianza. Sin su colaboración y sin el contexto real que facilitaron, este proyecto no habría tenido la solidez que tiene. Su apertura fue un aporte invaluable para el desarrollo de esta investigación.

A los compañeros que en algún momento brindaron orientación, colaboración o apoyo durante la carrera. Aunque el entorno virtual limitó los lazos más allá de lo académico, cada intercambio fue un aporte genuino al proceso de aprendizaje.

Finalmente, a quienes cruzaron mi vida durante este tiempo y dejaron algo: una enseñanza, una motivación renovada, o simplemente la certeza de que vale la pena terminar lo que se empieza. No todos los aprendizajes vienen de un aula, y este proceso me lo confirmó.

Emmanuel Cerdas Brenes

2026

CAPÍTULO I: INTRODUCCIÓN

Descripción del Problema

El presente proyecto toma como referencia a la empresa AraWeb CR, ubicada en Pérez Zeledón, San José, Costa Rica. AraWeb CR es una organización costarricense dedicada al desarrollo web, especializada en la creación de páginas web personalizadas. Su enfoque se orienta a brindar soluciones digitales adaptadas a las necesidades específicas de cada cliente, priorizando la experiencia del usuario, el diseño funcional, la optimización para dispositivos móviles y el soporte técnico continuo. La empresa ofrece un servicio integral que abarca desde el diseño visual hasta la implementación técnica de sitios web, atendiendo tanto a pequeñas como a grandes empresas en el ámbito nacional.

A pesar de su orientación hacia la innovación tecnológica en los servicios que ofrece a sus clientes, AraWeb CR presenta limitaciones en la gestión interna de sus procesos administrativos relacionados con el recurso humano, particularmente en el manejo de la planilla salarial. Actualmente, dichos procesos se realizan de forma manual, lo cual genera diversas dificultades operativas y administrativas que impactan negativamente tanto a la organización como a sus colaboradores, evidenciando la necesidad de contar con una solución tecnológica que permita optimizar y automatizar estas tareas.

Problemáticas de la Empresa

El registro de la nómina salarial se realiza de forma manual, lo que provoca resultados contables erróneos y atrasos en los pagos. Esto ocasiona insatisfacción en los colaboradores y problemas administrativos en la empresa.

La asistencia de los colaboradores se lleva de forma manual, lo que genera errores en los reportes de tiempos laborados, retrasos en la verificación de jornadas y posibles conflictos con los empleados.

Las horas extraordinarias no se calculan ni se registran sistemáticamente, lo cual provoca inconformidades en los colaboradores por pagos incorrectos y atrasos en la autorización de dichas horas.

Los permisos de los colaboradores se gestionan de manera manual o verbal, lo que provoca desorden en la planificación, duplicidad de información y falta de transparencia en las decisiones.

Las incapacidades médicas se registran en físico, lo que ocasiona pérdida de información, retrasos en su control y problemas en el cálculo del salario en casos de sustituciones o ausencias prolongadas.

El registro y aprobación de vacaciones se realiza en papel, lo que provoca falta de control en los días acumulados, errores en el disfrute de los mismos y conflictos con los colaboradores.

El cálculo del aguinaldo se realiza manualmente, lo que genera errores en el pago a los colaboradores y posibles reclamos.

El cálculo de la liquidación laboral se realiza manualmente, lo que provoca errores, retrasos en el pago y conflictos legales con los colaboradores.

La empresa no cuenta con un proceso sistemático para evaluar el desempeño de los colaboradores, lo que dificulta la toma de decisiones relacionadas con reconocimientos, incentivos o medidas correctivas.

En conjunto, las situaciones descritas evidencian la existencia de debilidades significativas en la gestión de los procesos administrativos relacionados con la planilla salarial en la empresa AraWeb CR. La ausencia de un sistema informatizado que permita registrar, controlar y procesar de manera integrada la información del recurso humano incrementa el riesgo de errores, retrasa la toma de decisiones y genera inconformidades tanto en el ámbito administrativo como laboral. Ante este contexto, se identifica la necesidad de implementar una herramienta tecnológica que automatice dichos procesos, facilite el manejo adecuado de la información y contribuya a mejorar la eficiencia operativa, la transparencia y el cumplimiento de las obligaciones laborales de la organización.

Objetivos

Objetivo General

Desarrollar un sistema web para la gestión y pago de planilla en AraWeb CR mediante el uso de Visual Studio con el lenguaje de programación C# y la base de datos SQL Server.

Objetivos Específicos

- Analizar los requerimientos funcionales y no funcionales del sistema web.
- Construir el diseño de la base de datos y del sistema web mediante la herramienta Workbench y Visual Studio.

- Programar los diferentes módulos del sistema mediante la herramienta C#.
- Probar los diferentes módulos de forma independiente y también de forma integral lo que garantiza los resultados correctos del sistema.

Justificación

La presente investigación se justifica por la necesidad de mejorar la gestión de los procesos administrativos relacionados con la planilla salarial en la empresa AraWeb CR. Actualmente, dichos procesos se realizan de forma manual, lo cual genera errores en los cálculos, atrasos en los pagos y dificultades en el control de la información del recurso humano. Estas situaciones impactan negativamente tanto en el funcionamiento interno de la empresa como en la satisfacción, confianza y bienestar de sus colaboradores.

El desarrollo de un sistema web para la gestión y pago de planilla permite automatizar y centralizar la información asociada a salarios, asistencia, horas extraordinarias, permisos, incapacidades, vacaciones, aguinaldo y liquidaciones laborales. Con ello, se busca minimizar los errores derivados del manejo manual de los datos, optimizar los tiempos de procesamiento y proporcionar información confiable y oportuna que respalde la toma de decisiones administrativas.

Desde el punto de vista organizacional, el proyecto resulta relevante debido a que contribuye al fortalecimiento del control interno de los procesos de recursos humanos, mejora la eficiencia operativa y promueve una mayor transparencia en la gestión de la planilla salarial. Asimismo, la implementación del sistema web permite establecer procedimientos más ordenados y sistemáticos, reduce los conflictos administrativos y laborales asociados a pagos incorrectos o inconsistencias en la información.

En el ámbito académico y profesional, la investigación adquiere importancia al permitir la aplicación práctica de los conocimientos adquiridos durante la carrera de Ingeniería Informática, mediante el desarrollo de una solución tecnológica real. Para ello, se emplean herramientas y tecnologías como C#, SQL Server y el entorno de desarrollo Visual Studio, fortaleciendo competencias relacionadas con el análisis, diseño, desarrollo y validación de sistemas web empresariales.

En resumen, el proyecto se justifica porque propone una solución tecnológica orientada a optimizar la gestión de la planilla salarial en AraWeb CR, genera beneficios tanto para la organización como para sus colaboradores y contribuye al desarrollo profesional del estudiante mediante la aplicación integral de competencias propias de la disciplina informática.

Viabilidad Técnica

La viabilidad técnica del proyecto se considera alta, dado que la empresa AraWeb CR dispone de la infraestructura tecnológica mínima necesaria para el desarrollo e implementación de un sistema web orientado a la gestión y pago de planilla. La solución propuesta no requiere tecnologías complejas ni infraestructura especializada, ya que se basa en herramientas ampliamente utilizadas en entornos empresariales y académicos. Esta condición facilita el desarrollo del sistema y reduce las dificultades que pudieran surgir al adoptar el sistema.

En cuanto a la infraestructura de hardware, la empresa cuenta con computadoras de escritorio y portátiles utilizadas por el personal de jefaturas, contabilidad y recursos humanos. Estos equipos resultan suficientes para el acceso y uso del sistema web mediante un navegador, ya que la aplicación no demanda altos requerimientos de procesamiento ni componentes adicionales. Al tratarse de una solución basada en web, no es necesaria la adquisición de servidores dedicados ni dispositivos especializados, lo que disminuye la complejidad técnica y los costos asociados al proyecto.

Respecto a la infraestructura de software, AraWeb CR utiliza principalmente el sistema operativo Windows 11 y el navegador web Google Chrome, los cuales son plenamente compatibles con el sistema a desarrollar. Actualmente, la empresa no cuenta con un sistema informático para la gestión administrativa o de planilla, ni posee experiencia previa en el uso de sistemas web internos; sin embargo, esta situación no representa una limitación técnica. Por el contrario, constituye una oportunidad de mejora al permitir la incorporación de una solución tecnológica que modernice y estandarice los procesos administrativos relacionados con el recurso humano.

En relación con el hosting y el acceso al sistema, la empresa dispone de un servidor web externo que, en su estado actual, no es compatible con aplicaciones desarrolladas en Visual Studio bajo la arquitectura ASP.NET MVC. Debido a esta condición, y considerando las necesidades actuales de la organización, el sistema será ejecutado inicialmente en un entorno local. Este entorno resulta suficiente para el funcionamiento del prototipo y su uso interno, sin requerir disponibilidad continua las 24 horas ni acceso remoto en esta etapa del proyecto, lo cual simplifica su implementación técnica.

Con el fin de detallar de manera estructurada los recursos tecnológicos necesarios para el desarrollo y ejecución del sistema, en la Tabla 1 se presenta el hardware y software requerido, incluyendo sus respectivas versiones y características generales. Esta información permite

evidenciar que los requerimientos tecnológicos son accesibles y se encuentran alineados con las capacidades actuales de la empresa. Igualmente, el detalle de los recursos utilizados facilita la comprensión del entorno técnico en el cual será desarrollado el sistema y respalda la viabilidad de su implementación sin necesidad de inversiones adicionales significativas.

Tabla 1

Hardware y software requeridos para el desarrollo e implementación del sistema.

Categoría	Recurso	Especificación / Versión	Uso principal
Hardware	Computadora de escritorio o portátil	Procesador Intel i5 o equivalente, 8 GB RAM mínimo	Desarrollo y uso del sistema
Hardware	Almacenamiento	Disco duro o SSD, mínimo 256 GB	Almacenamiento de datos y aplicación
Software	Sistema operativo	Windows 11	Plataforma de ejecución
Software	Entorno de desarrollo	Visual Studio Community 2022	Desarrollo del sistema web
Software	Framework web	ASP.NET MVC (.NET Framework)	Estructura y arquitectura del sistema
Software	Lenguaje de programación	C#	Lógica de negocio
Software	Diagrama de base de datos	MySQL Workbench	Modelado entidad-relación
Software	Gestor de base de datos	SQL Server 2022	Gestión y persistencia de datos
Software	Navegador web	Google Chrome (versión actual)	Acceso al sistema

Fuente:” Elaboración propia, 2026”.

Nota: Los datos presentados en esta tabla fueron obtenidos mediante conversaciones directas con representantes de la empresa AraWeb CR, con el propósito de identificar la infraestructura tecnológica disponible y el software que la organización utiliza habitualmente en sus operaciones o desean empezar a utilizar. Esta información permitió comprender de manera clara las capacidades técnicas actuales de la

empresa y asegurar que los recursos propuestos para el desarrollo e implementación del sistema sean coherentes con su entorno tecnológico real. Del mismo modo, los datos reflejan un escenario ajustado a las condiciones operativas existentes, sin requerir adquisiciones adicionales significativas para la ejecución del proyecto.

Viabilidad Operativa

La viabilidad operativa del proyecto es favorable, ya que el sistema web propuesto será utilizado por áreas clave dentro de la empresa AraWeb CR, específicamente Jefaturas, Contabilidad y Recursos Humanos, las cuales son responsables directas de los procesos relacionados con la gestión y el pago de la planilla salarial. Estas áreas concentran las principales actividades administrativas vinculadas al control del personal, por lo que la implementación del sistema se alinea de forma natural con la estructura organizacional existente y con las funciones que actualmente desempeña cada departamento.

La empresa manifestó que los colaboradores muestran disposición para registrar su información mediante un sistema web, incluyendo procesos como asistencia, permisos y vacaciones. De igual manera, se indicó que el personal cuenta con conocimientos básicos en el uso de herramientas informáticas, tales como navegadores web y sistemas de registro digital, lo cual facilita la adopción del sistema propuesto. Esta condición reduce significativamente la necesidad de procesos extensos de capacitación, permitiendo una transición gradual y controlada desde los métodos manuales hacia un entorno digital más eficiente.

La implementación del sistema no implicará una reducción de personal, sino que contribuirá a optimizar los procesos administrativos existentes, disminuyendo la carga operativa manual, los errores en los cálculos y el tiempo invertido en tareas repetitivas. Además, el sistema representará un cambio positivo en la forma de trabajo de la organización, al transformar procedimientos manuales en procesos automatizados, más ordenados, trazables y con mayor control sobre la información generada.

Por eso, desde el punto de vista operativo, el sistema es viable, ya que se adapta adecuadamente a la dinámica de trabajo de AraWeb CR, cuenta con la aceptación de los usuarios finales y no requiere modificaciones significativas en la estructura organizacional ni en los roles actuales del personal. También, su implementación permitirá una transición ordenada hacia

procesos automatizados, facilitando la adopción del sistema y promoviendo un uso efectivo y sostenible dentro de la empresa a mediano y largo plazo.

Viabilidad Económica

La viabilidad económica del proyecto es alta, considerando que el desarrollo del sistema no representa un costo directo para la empresa, al tratarse de un proyecto de graduación. Por tanto, los costos asociados se limitan al valor estimado del desarrollo del sistema y a los requerimientos tecnológicos necesarios para su eventual implementación y operación. Este enfoque permite a la organización evaluar los beneficios del sistema sin asumir una inversión inicial significativa durante la fase de desarrollo.

Actualmente, la empresa no incurre en gastos relacionados con sistemas de planilla ni cuenta con un presupuesto fijo destinado a soluciones tecnológicas, ya que este se define de forma mensual según las necesidades del negocio. Para la etapa inicial, la empresa ha decidido que el sistema funcione en un entorno local, lo cual permite prescindir temporalmente de costos asociados a servicios de hosting, licencias de servidor o infraestructura en la nube, favoreciendo una implementación gradual y controlada.

En cuanto al software, la empresa no posee licencias específicas para este tipo de sistemas; sin embargo, manifestó su disposición a utilizar software gratuito o de uso comunitario, lo que contribuye a reducir significativamente los costos de implementación. Por su parte, no se requiere inversión en hardware adicional, ya que la infraestructura existente es suficiente para el funcionamiento del sistema, ni tampoco se necesitan adecuaciones físicas o mobiliario adicional para su puesta en marcha.

Finalmente, la empresa considera que la implementación del sistema podría generar ahorros a mediano y largo plazo, al disminuir errores en el cálculo de la planilla, reducir tiempos administrativos y mejorar el control y la trazabilidad de la información del recurso humano. Estos beneficios económicos indirectos refuerzan la conveniencia del proyecto, al aportar eficiencia operativa y optimización de recursos en los procesos internos de la organización.

Tabla 2

Costos estimados del proceso de desarrollo del sistema.

Etapas del proyecto	Actividad principal	Recurso	Jornadas	Costo unitario (C)	Costo total (C)
Análisis y planificación	Levantamiento de requerimientos	Programador	5	16.244,50	81.222,50
Diseño del sistema	Diseño de arquitectura y BD	Programador	6	16.244,50	97.467,00
Desarrollo por módulos	Programación del sistema	Programador	20	16.244,50	324.890,00
Pruebas e integración	Validación funcional	Programador	5	16.244,50	81.222,50
Implementación	Preparación del entorno	Programador	4	16.244,50	64.978,00
Total desarrollo					¢649.780,00

Fuente: “Salario Mínimos En Costa Rica, 2026”.

Nota: El cálculo del costo de desarrollo se basa en el salario mínimo oficial para el puesto de programador de computación sin título, establecido por el Ministerio de Trabajo y Seguridad Social y el Consejo Nacional de Salarios de Costa Rica.

Tabla 3

Desglose de costos estimado de licencias e infraestructura tecnológica.

Categoría	Recurso	Tipo	Modalidad	Costo mensual (USD)	Costo mensual estimado (₡)
Base de datos	SQL Server Standard (Azure)	Licencia	Suscripción mensual	\$100,00	₡52.000,00
Hosting web	Hostinger Business	Servicio	Suscripción mensual	\$20,00	₡10.400,00
Herramienta de desarrollo	Visual Studio Community	Software	Licencia gratuita	\$0,00	₡0,00
Total mensual estimado				\$120,00	₡62.400,00

Fuente:” Elaboración propia, 2026”.

Nota. Tipo de cambio estimado: ₡520 por dólar estadounidense. Los costos presentados corresponden a una estimación de las licencias y servicios tecnológicos necesarios para el lanzamiento y operación básica del sistema web. El costo de SQL Server Standard se obtuvo a partir del servicio de suscripción mensual ofrecido por Microsoft Azure, visualizado desde una cuenta registrada en el apartado de actualización de plan. El costo del servicio de hosting corresponde al plan Business de Hostinger, el cual presenta variaciones frecuentes debido a descuentos y promociones; por lo tanto, se estimó el valor mensual vigente al momento del análisis con base en la información publicada en su sitio oficial. La herramienta Visual Studio Community se incluye sin costo, al tratarse de una licencia gratuita destinada al desarrollo de aplicaciones. Estos montos reflejan un escenario realista y conservador para fines académicos y de defensa del proyecto.

Viabilidad Legal

Desde el punto de vista legal, el desarrollo del Sistema Web para la Gestión y Pago de Planilla para la empresa AraWeb CR resulta legalmente viable, ya que se concibe y ejecuta en apego a la normativa costarricense vigente relacionada con la protección de datos personales, la seguridad de la información, la propiedad intelectual y la prevención de delitos informáticos.

En primer lugar, el proyecto se enmarca dentro de lo establecido en la Ley N.º 8968, Ley de Protección de la Persona frente al Tratamiento de sus Datos Personales, la cual regula el tratamiento de datos personales y sensibles. El sistema gestionará información confidencial de los colaboradores, como datos personales, laborales y salariales, por lo que se diseñará considerando principios de confidencialidad, acceso restringido según roles, uso legítimo de la información y protección contra accesos no autorizados. La empresa indicó que actualmente maneja esta información de forma confidencial y que está dispuesta a incorporar políticas de seguridad, lo que garantiza el cumplimiento de dicha normativa.

En esa misma línea, el proyecto se rige por lo dispuesto en la Ley N.º 8148, que adiciona los artículos 196 BIS, 217 BIS y 229 BIS al Código Penal, relacionados con delitos informáticos como el acceso indebido a sistemas, la alteración de información y la violación de comunicaciones electrónicas. En atención a esta ley, el sistema incorporará mecanismos de autenticación, control de accesos, validación de usuarios y trazabilidad de acciones, con el fin de prevenir usos indebidos y proteger la integridad de la información almacenada.

De igual manera, se considera lo establecido en la Ley N.º 4573 para reprimir y sancionar los delitos informáticos, la cual refuerza la necesidad de proteger los sistemas de información frente a accesos no autorizados, manipulación de datos y uso fraudulento de tecnologías. El desarrollo del sistema se realizará aplicando buenas prácticas de programación segura y control de privilegios, contribuyendo a minimizar riesgos legales asociados al uso de tecnologías de información.

Por otra parte, el proyecto respeta lo dispuesto en la Ley N.º 6683, Ley de Derechos de Autor y Derechos Conexos, ya que el sistema será desarrollado como un proyecto original de graduación, utilizando herramientas de desarrollo y tecnologías debidamente licenciadas o de uso gratuito. El código fuente, la estructura del sistema y la base de datos corresponden a un desarrollo propio, evitando el uso no autorizado de software o material protegido por derechos de autor.

Finalmente, la empresa manifestó que cumple con la normativa laboral costarricense relacionada con salarios, aguinaldo, vacaciones e incapacidades, lo cual respalda la legalidad de los procesos que serán automatizados. El sistema no busca sustituir obligaciones legales, sino apoyar su correcta aplicación mediante una herramienta tecnológica que facilite el control, registro y cálculo de la información laboral conforme a la legislación vigente.

En conclusión, el proyecto es legalmente viable, ya que su desarrollo y futura implementación se encuentran alineados con las leyes costarricenses aplicables en materia de

protección de datos personales, seguridad de la información, delitos informáticos y derechos de autor. El análisis detallado de cada normativa y su aplicación específica al sistema será abordado en un apartado posterior del proyecto, conforme a los lineamientos establecidos.

Proyecciones

El desarrollo del prototipo del Sistema web para la gestión y pago de planilla en AraWeb CR tiene como propósito principal automatizar, centralizar y optimizar los procesos relacionados con la administración del recurso humano y el cálculo de la planilla salarial. Este sistema se diseñará considerando la normativa laboral vigente en Costa Rica, con el fin de garantizar el correcto manejo de la información y el cumplimiento de las obligaciones legales. Asimismo, se busca que la solución propuesta contribuya a mejorar la eficiencia operativa de la empresa, brindando mayor control, orden y confiabilidad en los procesos internos asociados a la gestión del personal.

Se proyecta la construcción de un sistema web funcional que permita a la organización gestionar de manera eficiente los procesos de asistencia, control de horas extra, permisos, incapacidades y vacaciones. De igual forma, el sistema contemplará el cálculo de aguinaldo, liquidaciones laborales y otros componentes salariales, reduciendo la probabilidad de errores humanos en los cálculos. Con la implementación de esta herramienta, se espera optimizar los tiempos operativos, disminuir la carga administrativa y reducir la dependencia de procesos manuales que actualmente limitan la productividad.

Igualmente, el sistema se concibe como un prototipo escalable, desarrollado con una arquitectura que permita su mejora y ampliación a futuro. Esta característica facilitará su adaptación a nuevas necesidades organizacionales, cambios en la normativa legal o la incorporación de funcionalidades adicionales. A mediano plazo, el prototipo podrá evolucionar hacia una solución definitiva para la empresa, contribuyendo al fortalecimiento de sus procesos internos y apoyando el crecimiento sostenido de AraWeb CR mediante el uso estratégico de la tecnología.

La organización se beneficia mediante:

- Automatización de procesos críticos de nómina y recursos humanos.
- Mayor exactitud en los cálculos salariales y cumplimiento normativo.
- Disponibilidad centralizada y segura de la información.

- Reducción de tiempos administrativos y reprocesos.
- Mejora en la toma de decisiones mediante reportes confiables.
- Transparencia en la gestión del personal y trazabilidad de las acciones.

Alcance Funcional

El sistema contará con las siguientes capacidades funcionales, organizadas en módulos:

- **Gestionar Nóminas:** Este módulo se encarga de realizar el procesamiento correcto de la nómina salarial, conforme a lo estipulado en la normativa laboral de Costa Rica.
- **Gestionar Asistencia:** Este módulo se encarga de registrar de forma digital la asistencia de los colaboradores para obtener una base de datos confiable de tiempos laborados.
- **Gestionar Horas Extras:** Este módulo se encarga de gestionar un adecuado registro de las horas extras laboradas por cada uno de los colaboradores. El colaborador realiza la solicitud vía sistema a su jefatura; si esta la aprueba, se notifica vía sistema a RRHH y posteriormente al colaborador. En caso de rechazo, el sistema notifica al colaborador de forma inmediata.
- **Gestionar Permisos:** Este módulo se encarga de gestionar de manera ordenada las solicitudes de permisos laborales. El colaborador registra su solicitud vía sistema, la cual es enviada automáticamente a su jefatura para revisión y notificación según la decisión tomada.
- **Gestionar Incapacidades:** Este módulo se encarga de registrar las incapacidades médicas, documentando las fechas de inicio y fin, así como la causa correspondiente.
- **Calcular Vacaciones:** Este módulo se encarga de gestionar las solicitudes de vacaciones de los colaboradores, actualizando automáticamente el saldo disponible según la aprobación o rechazo de la jefatura.
- **Calcular Aguinaldo:** Este módulo calcula el aguinaldo de cada colaborador en función del promedio de los salarios percibidos durante el año, conforme a la legislación costarricense.
- **Gestionar Liquidación:** Este módulo se encarga de calcular de forma precisa la liquidación laboral de los colaboradores, considerando los derechos adquiridos y la normativa vigente.

- ***Evaluar Personal:*** Este módulo se encarga de gestionar evaluaciones periódicas del desempeño del personal mediante indicadores previamente definidos.
- ***Mantenimientos:*** Este módulo permite el borrado, inserción, modificación y actualización de datos del sistema.
- ***Consultas:*** Este módulo se encarga de generar información proveniente de las diferentes tablas del sistema.
- ***Reportes:*** Este módulo genera reportes con formatos específicos, los cuales pueden ser visualizados en pantalla o impresos según las necesidades del usuario.
- ***Seguridad:*** Este módulo se encarga de la autenticación de usuarios, gestión de contraseñas, definición de perfiles y control de accesos al sistema.

Alcance Metodológico

Como enfoque metodológico para el desarrollo del sistema se adopta una metodología ágil de carácter simplificado, tomando como referencia el marco de trabajo Scrum. Esta elección se justifica debido a la naturaleza del proyecto, el cual corresponde a un proyecto de graduación y será desarrollado por un único responsable. Por esta razón, no se aplicarán todos los roles, eventos y artefactos formales establecidos por Scrum, sino únicamente sus principios fundamentales, tales como el desarrollo iterativo, la entrega incremental de funcionalidades y la adaptación continua a los cambios durante el proceso de desarrollo.

El desarrollo del sistema se lleva a cabo de forma iterativa e incremental, organizando el proyecto en módulos funcionales priorizados según su importancia dentro del sistema de gestión y pago de planilla. Cada módulo será sometido a un proceso gradual que incluye el análisis de requerimientos, el diseño lógico y físico, la implementación y las pruebas correspondientes. Este enfoque permite avanzar de manera ordenada y controlada, con lo que se asegura que cada componente cumpla con los requerimientos definidos antes de proceder al desarrollo del siguiente módulo.

Del mismo modo, la aplicación de una metodología ágil facilita la obtención de retroalimentación en etapas tempranas del proyecto, ya que al finalizar cada módulo se puede presentar su funcionamiento a la empresa o a su representante. En caso de identificarse observaciones, ajustes o mejoras, estas podrán incorporarse de manera oportuna durante el ciclo de desarrollo. De esta forma, se reduce el riesgo de realizar cambios significativos en fases avanzadas

del proyecto y se asegura una mayor alineación entre el sistema desarrollado y las necesidades reales de la organización.

Ciclo de vida del Sistema.

El ciclo de vida del sistema se estructura bajo un modelo iterativo e incremental, basado en un enfoque ágil adaptado al desarrollo de proyectos de software. Este modelo permite construir el sistema de manera progresiva, validando cada etapa antes de avanzar a la siguiente, lo que favorece el control de calidad y la detección temprana de errores. De igual manera, este enfoque facilita la incorporación de ajustes conforme se identifican nuevas necesidades durante el proceso de desarrollo, manteniendo una adecuada alineación con los objetivos del proyecto.

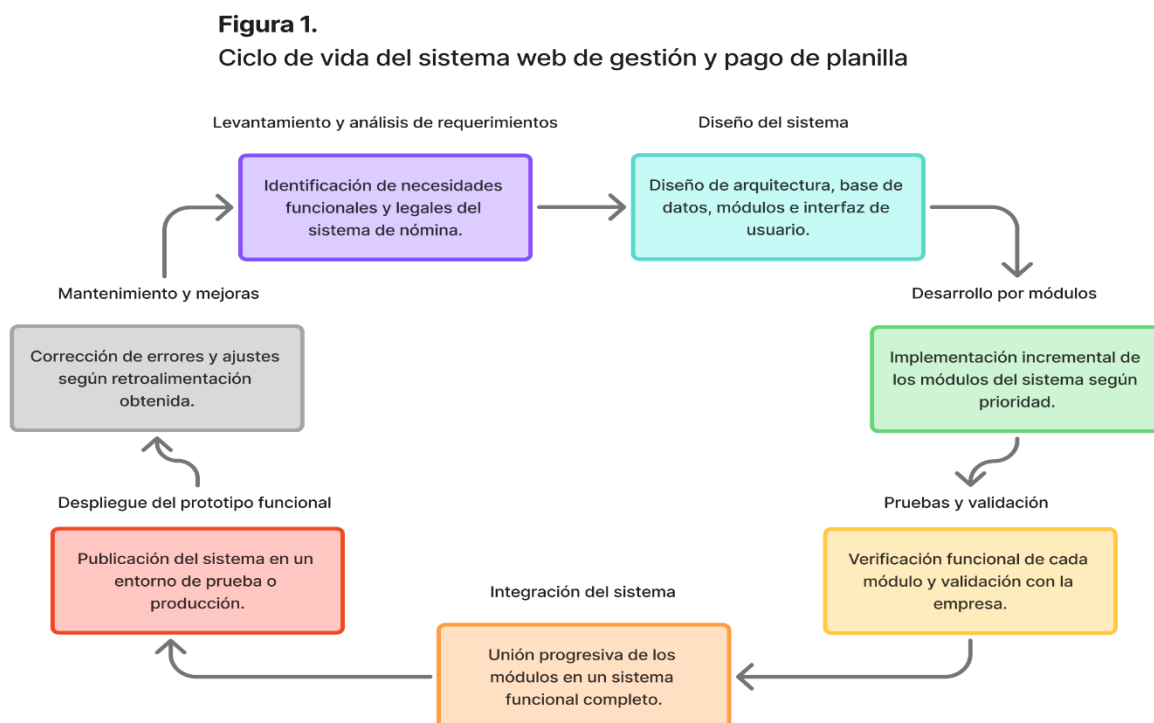
La primera etapa corresponde al levantamiento y análisis de requerimientos, en la cual se identifican y documentan las necesidades funcionales y no funcionales del sistema. Para ello, se consideran los procesos actuales de la empresa AraWeb CR, así como los requerimientos legales aplicables a la gestión de planilla en Costa Rica. Posteriormente, se desarrolla la fase de diseño del sistema, donde se define la arquitectura general, la estructura de la base de datos, la organización de los módulos y la interfaz de usuario, asegurando coherencia con las tecnologías seleccionadas y las buenas prácticas del desarrollo web.

A continuación, el sistema es implementado mediante un desarrollo modular e incremental, priorizando cada módulo según su relevancia dentro del proceso de gestión y pago de planilla. En esta fase se codifican las funcionalidades utilizando tecnologías como ASP.NET MVC, C# y SQL Server. Cada módulo desarrollado es sometido a pruebas funcionales y procesos de validación, con el fin de verificar su correcto funcionamiento y confirmar que cumple con los requerimientos establecidos antes de su integración al sistema general.

Finalmente, los módulos validados se integran progresivamente en un único sistema, lo que garantiza la correcta interacción entre ellos y la estabilidad de la aplicación. Una vez completada esta integración, el sistema se prepara para su despliegue como prototipo funcional en un entorno de prueba o de uso interno. Aunque el alcance principal del proyecto corresponde al desarrollo del prototipo, se contempla la posibilidad de realizar mantenimiento, correcciones y mejoras futuras, permitiendo una evolución controlada del sistema conforme a nuevas necesidades organizacionales.

Figura 1.

Ciclo de vida del sistema web de gestión y pago de planilla.



Fuente: “Elaboración propia, 2026”.

Alcance Tecnológico

El sistema propuesto es desarrollado bajo un enfoque web, lo que permite su acceso desde distintos dispositivos mediante un navegador web estándar, ya sea a través de una red local o mediante conexión a Internet. Esta característica facilita la disponibilidad del sistema para los usuarios autorizados, eliminando la necesidad de instalaciones locales en cada equipo. Además, este enfoque contribuye a una mayor centralización de la información y simplifica las tareas de mantenimiento y actualización del sistema.

Para el desarrollo de la aplicación se utiliza la plataforma ASP.NET MVC sobre .NET Framework, la cual proporciona una estructura robusta basada en el patrón Modelo–Vista–Controlador. Este enfoque permite una adecuada separación de responsabilidades entre la lógica de negocio, la presentación y el acceso a datos, lo que mejora la mantenibilidad y organización del código. El lenguaje de programación principal será C#, ampliamente utilizado en entornos empresariales y compatible con el desarrollo de soluciones web seguras y escalables.

Como gestor de base de datos se emplea SQL Server, debido a su capacidad para manejar grandes volúmenes de información de forma eficiente y segura. Esta tecnología permite almacenar y procesar datos relacionados con la nómina, asistencia, horas extra, permisos, vacaciones y demás módulos del sistema. También, garantiza la integridad, consistencia y confiabilidad de la información, aspectos fundamentales para la correcta gestión de datos sensibles del recurso humano.

La arquitectura del sistema es de tipo cliente-servidor, donde los usuarios acceden a la aplicación a través de un navegador web, mientras que la lógica del sistema y la base de datos se ejecuta en un servidor. Para efectos del desarrollo y validación del prototipo, el sistema puede alojarse en un servidor local, el cual resulta suficiente para las pruebas funcionales y la evaluación del proyecto. No obstante, la solución contempla la posibilidad de escalar hacia servicios en la nube o hosting externo, permitiendo su evolución futura hacia un entorno productivo conforme a las necesidades de la empresa.

CAPÍTULO II: MARCO REFERENCIAL

Introducción al marco referencial

El marco referencial constituye uno de los pilares fundamentales de toda investigación académica, ya que permite sustentar teóricamente el problema de estudio a partir del conocimiento previamente desarrollado. Al respecto, Salazar (2021) indica que el marco referencial: “constituye el desarrollo de la perspectiva teórica, por lo tanto, es todo un proceso y un producto a la vez” (párr. 1) de la investigación. Esta doble naturaleza permite al investigador profundizar en los antecedentes existentes y, a la vez, consolidar los resultados del análisis documental realizado.

Desde esta perspectiva, el marco referencial posibilita la inmersión en el conocimiento previo relacionado con el tema de investigación, favoreciendo la comprensión del contexto en el cual se ubica el problema planteado. Al respecto, Salazar (2021) señala que dicha perspectiva teórica: “brinda una visión general sobre la ubicación del problema de investigación en el campo del conocimiento, proporcionando nuevas ideas que generen aprendizaje y valorando el aporte de otros investigadores” (párr. 2). Esto permite establecer conexiones entre distintos enfoques teóricos y valorar los aportes realizados por otros investigadores.

La construcción del marco referencial requiere un proceso sistemático de recopilación y análisis de información proveniente de fuentes confiables y pertinentes al objeto de estudio. En este proceso, resulta fundamental integrar la información seleccionada de manera organizada y coherente, evitando desviaciones temáticas que puedan afectar la claridad del estudio. Según Salazar (2021): “una vez extraída y recopilada la información que interesa de las referencias pertinentes para el problema de investigación, es posible empezar a elaborar el marco teórico, el cual se basará en la integración de la información recopilada” (párr. 4), en función del problema de investigación.

Por su parte, la revisión analítica de la literatura se convierte en una etapa clave para garantizar la solidez del marco referencial. Esta revisión implica detectar, obtener y consultar bibliografía y otros materiales que aporten información significativa para el estudio. Salazar (2021) enfatiza que: “la revisión debe ser selectiva (se requiere seleccionar solo las fuentes más importantes y recientes), para garantizar información actual y certera” (párr. 12), que respalde el desarrollo de la investigación.

En el contexto del presente proyecto, el marco referencial se orienta a sustentar el desarrollo de un sistema web para la gestión y pago de planilla. Para ello, se abordan conceptos relacionados con los sistemas de información, la gestión de recursos humanos, el desarrollo de software web y los aspectos normativos vinculados al tratamiento de la información laboral. De esta manera, el marco referencial establece una base teórica que permite justificar las decisiones técnicas y metodológicas adoptadas, manteniendo una relación directa y coherente con el problema de investigación planteado.

Sistemas de Información

Los sistemas de información constituyen un componente esencial dentro de las organizaciones modernas, ya que permiten gestionar de forma estructurada los datos generados por las actividades cotidianas y transformarlos en información útil. Sobre esto, el Equipo editorial Etecé (2025) define que un sistema de información es: “un conjunto ordenado de personas, procesos y herramientas cuyo fin es administrar datos e información, de manera que puedan ser recuperados y procesados fácil y rápidamente” (párr. 1). Esta definición resalta el carácter integrador de estos sistemas y su orientación hacia el manejo eficiente de la información.

La estructura de un sistema de información se basa en la interacción de diversos recursos organizados de acuerdo con un propósito informativo específico. Al respecto, el Equipo editorial Etecé (2025) señala que estos sistemas están conformados por: “recursos interconectados y en interacción, bien organizados y dispuestos del modo más conveniente” según el objetivo que se persiga (párr. 2). Esta organización permite que los datos sean recolectados, clasificados y procesados de manera sistemática, facilitando su comprensión y posterior utilización.

Dentro del contexto organizacional, los sistemas de información se utilizan para apoyar tanto las actividades operativas como las administrativas y estratégicas. Según el Equipo editorial Etecé (2025) indica que: “estos sistemas se emplean en organizaciones empresariales y gubernamentales, pero también en otros sitios o estructuras, como bibliotecas, universidades, museos, redes sociales o aplicaciones” (párr. 4), con el fin de organizar información y respaldar procesos de gestión y control. Así, contribuyen a mejorar la planificación, el seguimiento de las operaciones y la toma de decisiones.

Cabe señalar que un sistema de información no se limita únicamente al uso de herramientas tecnológicas. Aunque los recursos informáticos desempeñan un papel fundamental, el Equipo editorial Etecé (2025) aclara que estos sistemas también: “depende de su organización, de las

personas implicadas y de las actividades” (párr. 5), que garantizan su correcto funcionamiento. Esta perspectiva enfatiza que la efectividad de un sistema de información está directamente relacionada con su adecuada organización y con la participación activa de los usuarios.

Como resultado, los sistemas de información resultan instrumentos clave para mejorar la eficiencia organizacional, al permitir el almacenamiento, procesamiento y distribución de información relevante en los distintos niveles de la empresa. Su correcta implementación favorece el ordenamiento de los procesos internos y sienta las bases para el desarrollo de soluciones especializadas, como los sistemas de información aplicados a la gestión de recursos humanos,

Definición de sistema de información

La definición de sistema de información permite comprender con mayor precisión el alcance y la función de estos sistemas dentro de las organizaciones. Desde una perspectiva conceptual, el Equipo editorial Etecé (2025) establece que un sistema de información es: “un conjunto ordenado de personas, procesos y herramientas cuyo fin es administrar datos e información, de manera que puedan ser recuperados y procesados fácil y rápidamente” (párr. 1). Esta definición enfatiza que la gestión de la información no depende únicamente de la tecnología, sino de la correcta articulación entre los distintos elementos que conforman el sistema.

En esa misma línea, un sistema de información se caracteriza por la interrelación de recursos que actúan de forma coordinada para cumplir un propósito informativo específico. En esa línea, el Equipo editorial Etecé (2025) explica que estos sistemas están compuestos por: “recursos interconectados y en interacción, bien organizados y dispuestos del modo más conveniente en relación con un propósito informativo determinado” (párr. 2). Esta organización permite que los datos sean recolectados, clasificados y transformados en información comprensible y útil para los usuarios.

De manera complementaria, los sistemas de información presentan una serie de características que fortalecen su funcionalidad dentro de las organizaciones. De acuerdo con el Equipo editorial Etecé (2025), estos sistemas deben contar con una estructura organizada, integrar elementos informáticos y no informáticos, garantizar la seguridad de la información y poseer la capacidad de actualizarse conforme a nuevas necesidades y tecnologías. Estas características aseguran que la información gestionada sea confiable, accesible y protegida frente a usos no autorizados.

Además, la definición de sistema de información se vincula directamente con los distintos objetivos organizacionales que estos sistemas apoyan. Tal como señala el Equipo editorial Etecé (2025), los sistemas de información se emplean para almacenar información que se utiliza para tomar decisiones, dirigir las actividades de una organización, implementar cambios y realizar planificaciones. Esto evidencia que su función trasciende el almacenamiento de datos, convirtiéndose en un recurso estratégico para la gestión institucional.

En relación con el presente proyecto, la definición de sistema de información se ajusta al enfoque del desarrollo de un sistema web de planilla, ya que este tipo de solución integra personas, procedimientos y herramientas tecnológicas para administrar información laboral y salarial. Por ello, el sistema de planilla cumple con los principios fundamentales que definen a un sistema de información, al permitir el procesamiento, almacenamiento y distribución estructurada de datos relevantes para la gestión de recursos humanos.

Componentes de un sistema de información

La comprensión de los componentes de un sistema de información resulta esencial para analizar su funcionamiento dentro de una organización. Desde un enfoque general, estos sistemas no se limitan únicamente al uso de tecnología, sino que integran diversos elementos que trabajan de manera conjunta. En relación con esto, el Equipo editorial Etecé (2025) define que: “un sistema de información es un conjunto ordenado de personas, procesos y herramientas cuyo fin es administrar datos e información, de manera que puedan ser recuperados y procesados fácil y rápidamente” (párr. 1). Esta definición permite identificar que los componentes abarcan tanto recursos técnicos como humanos y organizativos, los cuales deben encontrarse debidamente articulados para garantizar el manejo adecuado de la información.

Desde una perspectiva más detallada, los sistemas de información se estructuran a partir de componentes tecnológicos claramente definidos. Al respecto, T. Pham et al. (2024) señalan que:

“los sistemas de información pueden considerarse como que tienen seis componentes principales: hardware, software, comunicaciones de red, datos, personas y procesos. Cada uno tiene un rol específico, y todos los roles deben trabajar juntos para tener un sistema de información que funcione” (párr. 2-3).

Esta clasificación permite comprender que la tecnología constituye una base fundamental del sistema, ya que soporta la captura, procesamiento, almacenamiento y transmisión de los datos necesarios para las operaciones organizacionales.

Dentro de estos componentes tecnológicos, el hardware y el software cumplen funciones complementarias que hacen posible el funcionamiento del sistema. Según T. Pham et al. (2024), “el hardware representa los componentes físicos de un sistema de información”, mientras que “el software es un conjunto de instrucciones que le indican al hardware qué hacer” (párr. 5–6). A estos elementos se suman los datos, entendidos como hechos brutos que, una vez organizados y procesados, se transforman en información útil para la toma de decisiones. Asimismo, la comunicación en red permite el intercambio de información entre dispositivos y usuarios, fortaleciendo la integración y accesibilidad del sistema.

No obstante, el valor real de un sistema de información no reside únicamente en la tecnología, sino también en el componente humano y en los procesos organizacionales. En palabras de T. Pham et al. (2024), “las personas y los procesos son los dos componentes que aportan valor a las organizaciones en la forma en que utilizan la colección de tecnologías para cumplir con los objetivos específicos de las organizaciones” (párr. 3). Los procesos, definidos como una serie de pasos orientados a alcanzar un objetivo, permiten que la tecnología se integre de manera efectiva a las actividades del negocio. En conjunto, la adecuada articulación entre tecnología, personas y procesos posibilita que los sistemas de información respalden funciones críticas, como ocurre en los sistemas de planilla utilizados para la gestión administrativa y laboral.

Importancia de los sistemas de información en las organizaciones

En el contexto organizacional actual, la información se ha consolidado como un recurso estratégico para el funcionamiento y la competitividad de las empresas. La creciente cantidad de datos generados por las actividades administrativas y operativas ha impulsado la adopción de soluciones tecnológicas que permitan su gestión eficiente. Sobre el tema, Gómez (2024) señala que: “la gran cantidad de información que se maneja en el ámbito organizacional, junto con la necesidad de procesarla y organizarla de manera eficiente, ha llevado a la implementación de sistemas informáticos que permitan automatizar y agilizar numerosos procesos” (párr. 1).

La implementación de sistemas de información incide directamente en la mejora de los procesos internos de las organizaciones. Estas herramientas tecnológicas permiten reducir la carga operativa del personal mediante la automatización de tareas repetitivas, lo cual contribuye a una mayor eficiencia institucional. Al respecto, Gómez (2024) afirma que: “estos sistemas permiten automatizar tareas repetitivas, reduciendo así la carga de trabajo de los empleados y minimizando la posibilidad de errores” (párr. 6). Este impacto resulta especialmente relevante en áreas que gestionan grandes volúmenes de información de forma constante.

Otro aporte fundamental de los sistemas de información se relaciona con la toma de decisiones organizacionales. El acceso oportuno a datos confiables fortalece los procesos de planificación, control y evaluación. En concordancia con lo anterior, Gómez (2024) indica que: “estos sistemas recopilan y analizan grandes cantidades de datos, proporcionando información relevante para la toma de decisiones informadas y basadas en datos concretos” (párr. 7). En consecuencia, la gestión administrativa se sustenta en información estructurada y verificable.

Igualmente, los sistemas de información desempeñan un papel clave en la comunicación y coordinación entre las distintas áreas de la organización. La integración de plataformas tecnológicas facilita el flujo de información y promueve el trabajo colaborativo tanto en el ámbito interno como externo. Según Gómez (2024), “los sistemas informáticos facilitan la comunicación entre los diferentes miembros de la organización, permitiendo una mayor colaboración y coordinación en los diferentes proyectos” (párr. 8). Esta capacidad fortalece la coherencia organizacional y la relación con clientes y proveedores.

En el ámbito administrativo, la relevancia de los sistemas de información se evidencia en la gestión de transacciones críticas relacionadas con el recurso humano. Procesos como el control de registros laborales, la administración de pagos y el manejo de planillas requieren altos niveles

de precisión y control. Gómez (2024) destaca que el Sistema de Procesamiento de Transacciones (TPS): “es el encargado de manejar las transacciones diarias que son críticas para las operaciones, tales como ventas, recibos, depósitos y nómina” (párr. 10). Este enfoque refuerza la importancia de los sistemas de información en la gestión de Recursos Humanos, aspecto que se vincula directamente con el análisis desarrollado en el siguiente apartado.

Sistemas de información aplicados a la gestión de recursos humanos

La gestión de los recursos humanos ha experimentado una transformación significativa a partir de la incorporación de sistemas de información orientados al manejo estructurado de los datos laborales. Estas soluciones tecnológicas surgen como respuesta a la necesidad de administrar de forma integral la información del personal y los procesos asociados a su ciclo de vida dentro de la organización. En este contexto, Oracle América Latina (2025) define que: “un sistema de información de recursos humanos (HRIS) es una solución de software que mantiene, gestiona y procesa información detallada de los empleados y políticas y procedimientos relacionados con los recursos humanos” (párr. 1).

Desde una perspectiva operativa y estratégica, los sistemas de información de recursos humanos permiten estandarizar las tareas propias del área, reduciendo la dependencia de procesos manuales y documentación física. Estas plataformas facilitan una gestión más ágil y ordenada de la información laboral. Oracle América Latina (2025) señala que: “al eliminar los procesos de RR. HH. manuales y en papel, los sistemas HRIS ofrecen interacciones más fluidas, optimizadas y eficientes entre los empleados y la empresa” (párr. 2), lo cual contribuye a mejorar la eficiencia organizacional.

Un elemento clave de estos sistemas es su capacidad para centralizar la información del personal en un único repositorio accesible y confiable. Esta centralización favorece el control, mantenimiento y actualización de los registros laborales. Al respecto, KBIBenefits (2022) indica que los sistemas de información en la gestión de recursos humanos ayudan a: “gestionar su capital humano y proporcionar una ubicación centralizada para almacenar los datos de los empleados” (párr. 10). Esta característica resulta fundamental para una administración ordenada del capital humano.

Del mismo modo, los sistemas de información aplicados a recursos humanos desempeñan un papel relevante en el cumplimiento de la normativa laboral y la reducción de riesgos legales. La correcta gestión de jornadas, horas extra, beneficios y pagos salariales exige precisión y

trazabilidad en la información. Al respecto, KBIBenefits (2022) destaca que: “los sistemas SIRH permiten a las organizaciones garantizar con mayor facilidad el cumplimiento de las leyes laborales, como las leyes de igualdad de oportunidades en el empleo (EEO), los requisitos de beneficios, las horas extras y la nómina” (párr. 3). Esta funcionalidad adquiere especial importancia en entornos regulados.

Desde el punto de vista estructural, los sistemas HRIS también contribuyen a una mejor organización interna mediante la definición clara de roles, jerarquías y responsabilidades. La disponibilidad de información estructurada permite optimizar la asignación del talento y fortalecer la planificación del personal. Oracle América Latina (2025) señala que: “desde una perspectiva estructural, los sistemas HRIS incluyen las jerarquías de roles, funciones e informes. Mediante la definición de la estructura organizativa, los sistemas HRIS proporcionan responsabilidades claras y una solución que respalda la funcionalidad y la coherencia operativas” (párr. 8). Esta visión integral favorece la toma de decisiones relacionadas con la gestión del recurso humano.

En el ámbito administrativo, los sistemas de información de recursos humanos se consolidan como herramientas clave para mejorar la eficiencia operativa de las organizaciones. La integración de funciones relacionadas con la administración del personal permite optimizar procesos y reducir la carga administrativa del área de recursos humanos. Sobre esto, KBIBenefits (2022) señala que: “el almacenamiento centralizado de los datos de empleo mejora la eficiencia en toda la organización. Integrar funciones clave en un solo sistema, como el seguimiento del tiempo y la gestión del rendimiento, también aumenta la eficiencia” (párr. 9). Esta centralización facilita el acceso a información confiable, fortalece el control de los procesos internos y contribuye a una gestión más ordenada y precisa del capital humano.

De ahí que los sistemas de información aplicados a la gestión de recursos humanos trascienden su función tecnológica y se convierten en un componente estratégico dentro de las organizaciones. Su implementación establece las bases para la automatización y el control de procesos críticos relacionados con el personal. Esta integración resulta especialmente relevante en la gestión de la planilla salarial, tema que se profundiza en las secciones siguientes.

Gestión de recursos humanos

La gestión de recursos humanos se ha consolidado como un eje estratégico dentro de las organizaciones modernas, al centrarse en el aprovechamiento del talento humano para alcanzar los objetivos empresariales. Desde esta perspectiva, IBM (2025) define que: “La gestión del capital humano es un conjunto de prácticas para atraer, reclutar, desarrollar y retener a los empleados para lograr los objetivos empresariales” (párr. 1). Esta concepción permite comprender que el desempeño organizacional depende directamente de la adecuada administración de las capacidades, habilidades y compromiso del personal.

Desde una visión organizacional, la gestión de recursos humanos trasciende las funciones administrativas tradicionales y se orienta a fortalecer las competencias del personal. En esa línea, IBM (2025) señala que: “Las organizaciones que dependen de los empleados para lograr sus objetivos asignan recursos para desarrollar las habilidades básicas que su personal necesita para obtener resultados” (párr. 2). Este enfoque facilita la identificación de brechas de capacidades y la alineación de los procesos de desarrollo con las necesidades estratégicas de la empresa.

De igual manera, la gestión del capital humano desempeña un papel relevante en la construcción de una cultura organizacional sólida y orientada al desarrollo profesional. De acuerdo con IBM (2025), “El uso de prácticas de HCM puede ayudar a crear una cultura organizacional sólida que promueva el desarrollo de los empleados y el compromiso con los objetivos de la empresa” (párr. 3). Este tipo de entornos fomenta la motivación, el sentido de pertenencia y la permanencia del talento dentro de la organización.

Dentro de este marco, la gestión de recursos humanos integra diversas áreas funcionales que resultan críticas para la operación empresarial. IBM (2025) indica que la HCM “proporciona un proceso para optimizar el rendimiento y la eficiencia del personal de una empresa, ofreciendo una estrategia global para guiar la atracción de talento, el reclutamiento, la incorporación y la formación.” (párr. 4). Estos procesos permiten asegurar que cada puesto sea ocupado por personal idóneo y alineado con los objetivos y valores institucionales.

Otro componente esencial es la incorporación y formación del personal, ya que facilita la adaptación de los nuevos colaboradores a la cultura organizacional y a las exigencias del puesto. Según IBM (2025), “La gestión del capital humano requiere la incorporación efectiva de nuevos empleados” (párr. 7). Este proceso contribuye a mejorar el desempeño individual y colectivo, al tiempo que promueve el aprendizaje continuo dentro de la empresa.

Adicionalmente, la gestión de recursos humanos incluye funciones operativas clave relacionadas con el control administrativo del personal. En relación con esto, IBM (2025) destaca que “La gestión de los horarios, las horas trabajadas y el tiempo libre es fundamental para la precisión y el cumplimiento de la nómina” (párr. 9). La correcta administración de estos elementos resulta determinante para garantizar la exactitud de la planilla y el cumplimiento de las disposiciones laborales.

Finalmente, la gestión de recursos humanos se apoya en la elaboración de informes y el análisis de datos como herramientas para la toma de decisiones. IBM (2025) expone que “La gestión del capital humano ofrece una base de datos de conocimientos procesable para ayudar a las organizaciones a agregar valor a sus empleados” (párr. 11). En este contexto, la integración de sistemas de información se convierte en un soporte clave para automatizar procesos y fortalecer la gestión del personal, preparando el terreno para el análisis del sistema de planilla como herramienta administrativa.

Sistema de planilla como herramienta de gestión

En el contexto actual de las organizaciones, la gestión de la planilla ha evolucionado de una función meramente operativa a un componente estratégico dentro de la administración del capital humano. Este cambio se evidencia cuando las empresas superan el uso de herramientas manuales, ya que, como indica Softland Costa Rica (2025), “las hojas de cálculo son una herramienta inicial, pero a medida que el negocio crece, se vuelven propensas a errores que consumen tiempo y obstaculizan la productividad” (párr. 1). Esta limitación impulsa a las organizaciones a adoptar soluciones tecnológicas especializadas para garantizar precisión y eficiencia administrativa.

Desde la perspectiva de los sistemas de información, el sistema de planilla se consolida como una herramienta tecnológica orientada a la automatización de procesos críticos. Sobre el tema, Softland Costa Rica (2025) señala que “la automatización de planillas ofrece una solución a estos problemas, proporcionando a las empresas una forma de automatizar tareas repetitivas, reducir los errores comunes al pagar la planilla en Costa Rica, y liberar tiempo valioso para tareas más estratégicas” (párr. 1). De esta manera, la planilla deja de ser un proceso aislado y se integra a la gestión organizacional.

El sistema de planilla también puede comprenderse como un subsistema que centraliza información laboral y administrativa para su procesamiento confiable. De acuerdo con Softland Costa Rica (2025), “un software para planillas es una herramienta digital diseñada para gestionar

el pago de los empleados, incluyendo el cálculo de salarios, deducciones, y otros beneficios” (párr. 6). Esta centralización de datos permite estructurar la información de forma ordenada y coherente, facilitando su uso para la toma de decisiones.

En contextos regulados como el costarricense, el sistema de planilla cumple un rol fundamental en el cumplimiento de la normativa laboral y fiscal. Según Softland Costa Rica (2025), “el software de nómina actualizado garantiza que las empresas cumplan con las normativas fiscales y laborales vigentes, lo cual es crucial para evitar sanciones” (párr. 9). Este aspecto refuerza la importancia del sistema de planilla como una herramienta de control y respaldo legal para la organización.

Desde el enfoque de la gestión del capital humano, el sistema de planilla aporta valor al integrar diversas áreas del ciclo laboral del colaborador. Al respecto, Admin (2024) afirma que “un Sistema de planilla es una herramienta diseñada para ayudarnos a mejorar nuestra gestión de capital humano” (párr. 1). Esta visión posiciona al sistema de planilla como un apoyo integral para la administración del personal y no únicamente como un mecanismo de pago.

También, estos sistemas abarcan múltiples funciones que fortalecen la gestión organizacional. Admin (2024) señala que “podemos simplificar muchas de las tareas relacionadas con la gestión de recursos humanos y centrarnos en otros aspectos importantes del negocio” (párr. 2). Al consolidar información sobre remuneraciones, asistencia, vacaciones y descansos médicos, se mejora la calidad y confiabilidad de los datos disponibles.

Adicionalmente, la integración del sistema de planilla con otras áreas organizacionales refuerza su papel como herramienta de gestión. De acuerdo con Admin (2024), “una herramienta digital nos ayuda a llevar un registro detallado de las remuneraciones de nuestros colaboradores y calcular el pago de nóminas de manera automática” (párr. 4). Esta integración contribuye a la coherencia entre la información contable, laboral y administrativa.

Finalmente, el sistema de planilla como herramienta de gestión facilita la generación de reportes y el análisis de información para la toma de decisiones. En este contexto, Softland Costa Rica (2025) destaca que “la capacidad de generar informes detallados y mantener un registro ordenado de todos los datos de nómina es esencial para la toma de decisiones empresariales” (párr. 10). De esta forma, el sistema de planilla trasciende la automatización del pago salarial y se consolida como un componente clave de los sistemas de información aplicados a la gestión del capital humano.

Automatización de procesos de planilla

La automatización de los procesos de planilla surge como una respuesta a las limitaciones que presentan los métodos manuales tradicionales dentro de la gestión de recursos humanos, especialmente en contextos donde el procesamiento salarial demanda una alta carga administrativa y es propenso a errores. Sobre esto, Finn y Downie (2025) explican que la automatización de los recursos humanos consiste en: “utilizar herramientas digitales para agilizar las tareas de RR. HH. que consumen mucho tiempo” (párr. 1), lo cual permite reducir el trabajo tedioso y mejorar la productividad del personal encargado de estos procesos. Esta perspectiva permite comprender que la automatización de la planilla no solo optimiza el cálculo salarial, sino que también contribuye a una gestión más eficiente del tiempo y los recursos organizacionales.

Desde un enfoque operativo, la automatización de la planilla implica el uso de software y algoritmos para gestionar actividades que anteriormente se realizaban de forma manual, tales como el procesamiento de datos, la gestión del tiempo laborado y la aplicación de deducciones correspondientes. De acuerdo con Finn y Downie (2025): “la automatización de RR. HH. utiliza software y algoritmos para gestionar actividades que antes realizaban manualmente los profesionales de RR. HH.” (párr. 2), lo que permite reducir errores, garantizar la coherencia de la información y mejorar la eficiencia de los procesos administrativos. Este enfoque resulta especialmente relevante para la planilla, dado que se trata de un proceso sensible que requiere precisión y consistencia en el manejo de la información laboral.

Por su parte, la automatización de los procesos de planilla se integra de manera natural con otros procesos de recursos humanos, tales como el control de asistencia, la gestión del tiempo libre y la administración de beneficios. Esta integración favorece un enfoque basado en datos actualizados y centralizados, lo cual fortalece la trazabilidad de la información del colaborador. En este contexto, Finn y Downie (2025) señalan que: “con la automatización los equipos de RR. HH. ahora cuentan con flujos de trabajo automatizados y la información necesaria para una toma de decisiones basada en datos” (párr. 8), lo que contribuye a una gestión más ordenada y coherente del ciclo laboral.

Otro aspecto relevante de la automatización de la planilla es su impacto en la toma de decisiones organizacionales, particularmente en los departamentos de recursos humanos y finanzas. Al generar información estructurada y casi en tiempo real, estos sistemas permiten analizar

tendencias relacionadas con costos laborales, horas trabajadas y uso de beneficios. Tal como indican Finn y Downie (2025):

La automatización de procesos de RR. HH. es algo más que ahorrar tiempo. El uso de la automatización de RR. HH., o incluso de la hiperautomatización que fusiona múltiples tecnologías y herramientas, proporciona a los profesionales de RR. HH. datos e información valiosos casi en tiempo real. Este método ayuda a los profesionales a tomar decisiones informadas en medio del cambiante panorama económico, tecnológico y laboral. (párr. 6).

Esta afirmación evidencia cómo la automatización de la planilla fortalece el rol estratégico de los departamentos de recursos humanos, al facilitar el acceso a información clara, estructurada y oportuna para el análisis organizacional. Al disponer de datos confiables sobre costos laborales, tiempos trabajados y comportamiento del personal, las áreas de RR. HH. pueden apoyar de manera más efectiva los procesos de planificación y control interno. En esa misma línea, esta disponibilidad de información contribuye a una mejor alineación entre las decisiones administrativas y los objetivos institucionales, promoviendo un crecimiento organizacional más ordenado y sostenible. De esta forma, la automatización trasciende la función operativa y se consolida como un apoyo clave para la gestión estratégica de la organización.

Desde la perspectiva del colaborador, la automatización de la planilla también impacta positivamente en la experiencia del empleado, especialmente mediante el uso de portales de autoservicio. Estas plataformas permiten consultar talones de pago, gestionar solicitudes de tiempo libre y actualizar información personal sin depender constantemente del personal administrativo. Finn y Downie (2025) destacan que los portales de autoservicio “permiten a los empleados acceder y actualizar la información personal, ver los talones de pago y solicitar tiempo libre y acceder a las políticas y recursos de RR. HH.” (párr. 29), lo que reduce la carga administrativa y mejora la transparencia de los procesos internos.

No obstante, la automatización de los procesos de planilla también plantea desafíos que deben ser considerados por las organizaciones. Entre ellos se encuentran la protección de los datos personales, la integración con los sistemas existentes y la necesidad de una implementación ética de las tecnologías utilizadas. En esa línea, Finn y Downie (2025) advierten que las herramientas de automatización deben implementarse “de forma ética y transparente, respetando la protección de

datos y minimizando los sesgos” (párr. 5), lo cual resulta fundamental en sistemas que gestionan información sensible de los colaboradores.

En resumen, la automatización de los procesos de planilla constituye un elemento clave en la modernización de la gestión de recursos humanos. Al reducir errores, optimizar tiempos, facilitar la toma de decisiones y mejorar la experiencia del colaborador, estos sistemas permiten a las organizaciones responder de manera más eficiente a sus necesidades operativas y estratégicas. Este enfoque sienta las bases para el desarrollo de soluciones tecnológicas más amplias, como los sistemas web integrados de gestión de planilla, los cuales se analizarán en los apartados siguientes del presente marco referencial.

Desarrollo de software web

Desarrollo de sistemas web

El desarrollo de sistemas web se inscribe dentro del campo general del desarrollo de software y responde a la necesidad de crear aplicaciones accesibles a través de navegadores, orientadas a satisfacer requerimientos específicos de los usuarios y de las organizaciones. En relación con esto, Quiroz Vázquez (2026) define el desarrollo de software como: “un conjunto de actividades informáticas dedicadas al proceso de creación, diseño, despliegue y soporte de software” (párr. 1). Esta concepción permite comprender que los sistemas web no se limitan a la programación, sino que abarcan un proceso integral que va desde la identificación de necesidades hasta el mantenimiento del producto final.

Desde una perspectiva organizacional, el desarrollo de software busca alinear la solución tecnológica con los objetivos estratégicos del negocio. Al respecto, Quiroz Vázquez (2026) señala que “el objetivo del desarrollo de software es crear un producto que satisfaga las necesidades del usuario y los objetivos comerciales de una manera eficiente, repetible y segura” (párr. 2). Esta orientación resulta especialmente relevante en sistemas web empresariales, como un sistema de planilla, donde la confiabilidad, la seguridad de la información y la eficiencia operativa son factores críticos.

El desarrollo de sistemas web modernos se apoya en un proceso estructurado conocido como ciclo de vida del desarrollo de software. Según Quiroz Vázquez (2026), “el ciclo de vida de desarrollo de software (SDLC) es un proceso paso a paso que los equipos de desarrollo utilizan para crear software de alta calidad, rentable y seguro” (párr. 19). Este enfoque metodológico

permite organizar el trabajo de manera lógica, reducir riesgos y asegurar que el sistema web cumpla con los requisitos funcionales y técnicos definidos desde la etapa de planificación.

Arquitectura cliente-servidor

La arquitectura cliente-servidor constituye la base técnica sobre la cual operan la mayoría de los sistemas web actuales. Este enfoque permite separar las responsabilidades de presentación, lógica de negocio y gestión de datos, facilitando la escalabilidad y el mantenimiento del sistema. En relación con este aspecto, Quiroz Vázquez (2026) explica que durante el desarrollo del diseño “los desarrolladores también establecen qué interfaces de programación de aplicaciones (API) conectarán la aplicación con otras aplicaciones, sistemas e interfaces de usuario” (párr. 24). Esta interacción entre componentes es característica esencial de la arquitectura cliente-servidor.

Dentro de esta arquitectura, el desarrollo web se divide comúnmente en dos áreas fundamentales: front-end y back-end. Quiroz Vázquez (2026) indica que “el desarrollo front-end es el desarrollo de la parte del software que ve el usuario” (párr. 55), mientras que “el desarrollo de back-end se ocupa de los aspectos que el usuario no ve, como la creación de la lógica y la infraestructura del lado del servidor que el software necesita para funcionar” (párr. 56). Esta separación permite construir sistemas web más robustos, donde la interfaz de usuario y el procesamiento de datos evolucionan de forma controlada y coherente.

La arquitectura cliente-servidor resulta especialmente adecuada para sistemas web empresariales, ya que facilita la integración con bases de datos, servicios externos y otros sistemas organizacionales. Esta característica es clave para aplicaciones como un sistema de planilla, donde la centralización de datos y el acceso seguro desde distintos dispositivos constituyen requisitos fundamentales para su correcto funcionamiento.

Metodologías ágiles de desarrollo de software

Las metodologías ágiles han adquirido un papel central en el desarrollo de software web debido a su capacidad de adaptación al cambio y su enfoque en la entrega incremental de valor. Quiroz Vázquez (2026) describe que el modelo ágil “divide los proyectos más grandes en “sprints” más pequeños o funciones consumibles y entrega rápidamente esas funciones a través del desarrollo incremental” (párr. 39). Este enfoque permite responder de manera efectiva a cambios en los requisitos y mejorar continuamente el producto.

En estrecha relación con las metodologías ágiles surge el modelo DevOps, el cual refuerza la colaboración entre desarrollo y operaciones. Según Quiroz Vázquez (2026), “DevOps combina el trabajo de los equipos de desarrollo y operaciones de IT y utiliza la automatización para optimizar la entrega de software de alta calidad” (párr. 40). Esta integración resulta especialmente útil en sistemas web, donde se requiere mantener el sistema en funcionamiento continuo mientras se implementan mejoras y correcciones.

La aplicación de metodologías ágiles y prácticas DevOps en el desarrollo de sistemas web favorece la calidad, la estabilidad y la evolución constante del software. En el contexto de un sistema web de planilla, este enfoque permite realizar ajustes conforme cambian los requerimientos organizacionales o normativos, con lo que se asegura que la solución tecnológica se mantenga alineada con las necesidades reales de la organización.

Aspectos generales de la normativa laboral y tecnológica

El uso intensivo de tecnologías de la información dentro de las organizaciones ha generado una mayor dependencia de los sistemas informáticos para la gestión de procesos críticos. Esta realidad implica nuevas responsabilidades legales relacionadas con la seguridad, la privacidad y el cumplimiento normativo. Al respecto, De Expertos En Jurídico (2025) señala que: “vivimos una época en la que el entorno digital está presente en todas las esferas de la vida cotidiana y se ha convertido en el motor fundamental del mundo empresarial” (párr. 1). Por eso, los sistemas de información deben diseñarse considerando no solo su eficiencia técnica, sino también su adecuación al marco normativo vigente.

Importancia del cumplimiento legal en sistemas informáticos

El cumplimiento legal constituye una obligación fundamental para las organizaciones, especialmente cuando emplean sistemas informáticos para gestionar información sensible y procesos estratégicos. En el ámbito corporativo, el respeto a las disposiciones legales garantiza que las operaciones se desarrollen dentro de un marco de seguridad jurídica. De Expertos En Jurídico (2025) define que: “el cumplimiento legal se refiere al deber de toda persona o entidad de actuar conforme a las leyes y regulaciones que le aplican” (párr. 5). Esta definición establece la base sobre la cual deben estructurarse los sistemas informáticos organizacionales.

El incumplimiento de las normas legales aplicables a los sistemas de información puede generar consecuencias significativas para las organizaciones. De Expertos En Jurídico (2025)

advierte: “El cumplimiento legal no es simplemente una buena práctica; es una obligación que, de no cumplirse, puede derivar en severas sanciones legales, pérdida de licencias, daños reputacionales y pérdida de confianza por parte de clientes e inversionistas” (párr. 6). Esta afirmación resalta que los sistemas informáticos deben concebirse como instrumentos que apoyen activamente el cumplimiento normativo, reduciendo riesgos legales y fortaleciendo la sostenibilidad institucional.

Protección de datos personales en sistemas de información

La protección de los datos personales se ha convertido en uno de los principales desafíos legales en el uso de sistemas de información, debido al volumen y la sensibilidad de la información procesada. En este contexto, De Expertos En Jurídico (2025) indica que: “muchas normativas legales contemplan específicamente la protección de datos personales, la ciberseguridad y la gestión de riesgos tecnológicos” (párr. 7). Esta situación obliga a que los sistemas informáticos incorporen medidas que garanticen un tratamiento adecuado de la información.

Las normativas relacionadas con la protección de datos establecen requisitos claros que deben ser cumplidos por las organizaciones que gestionan información digital. De Expertos En Jurídico (2025) señala: “En general, requieren que las organizaciones implementen medidas técnicas y organizativas para asegurar la confidencialidad, integridad y disponibilidad de los datos” (párr. 8). Estas exigencias refuerzan la necesidad de que los sistemas web de planilla cuenten con mecanismos de seguridad que protejan la información laboral y salarial de los colaboradores, minimizando el riesgo de accesos indebidos o pérdidas de datos.

Relación entre tecnología y legislación laboral

La relación entre tecnología y legislación laboral se ha fortalecido a medida que los sistemas informáticos asumen un rol central en la gestión del capital humano. Sobre el tema, De Expertos En Jurídico (2025) define que: “el compliance en TI es la aplicación del cumplimiento normativo al entorno de tecnologías de la información” (párr. 9). Esta definición evidencia que los sistemas utilizados para administrar información laboral deben alinearse con los requerimientos legales y éticos del entorno organizacional.

El cumplimiento normativo en tecnologías de la información abarca aspectos que van más allá de la seguridad técnica de los sistemas. De Expertos En Jurídico (2025) explica: “Aunque incluye aspectos relacionados con la seguridad, su alcance es más amplio e incorpora cuestiones

legales, éticas, de privacidad y de gestión del riesgo” (párr. 14). Por ello, los sistemas web de planilla deben diseñarse considerando tanto los aspectos tecnológicos como las obligaciones legales que regulan la relación laboral, lo que garantiza una gestión responsable y conforme a la normativa.

Marco legal costarricense (enfoque general)

El cumplimiento legal en el ámbito tecnológico actúa como un marco de referencia que orienta el uso adecuado de los sistemas de información dentro de las organizaciones. Este enfoque resulta especialmente relevante en contextos donde se gestionan datos laborales y salariales. De Expertos En Jurídico (2025) afirma:

En definitiva, el compliance TI actúa como un marco de gobernanza que guía las acciones de la empresa en el terreno tecnológico, asegurando que sus prácticas estén en conformidad con lo requerido por la ley y por los estándares internacionales. (párr. 11).

Desde esta perspectiva, los sistemas web de planilla utilizados en Costa Rica deben concebirse bajo un enfoque preventivo y normativo, el cual será profundizado posteriormente al abordar de manera específica la legislación laboral y tecnológica costarricense.

CAPITULO III: MARCO METODOLÓGICO

Enfoques de Investigación

Los enfoques de investigación constituyen las rutas metodológicas mediante las cuales se estudian los fenómenos científicos. Sampieri y Torres (2018) definen la investigación como un “conjunto de procesos sistemáticos, críticos y empíricos que se aplican al estudio de un fenómeno o problema con el resultado (o el objetivo) de ampliar su conocimiento” (p. 4). Esta concepción se aplica a los enfoques cuantitativo, cualitativo y mixto, los cuales representan distintas rutas para generar conocimiento. La elección de uno de ellos depende del problema de investigación, el contexto y los recursos disponibles. Por tanto, comprender sus características permite seleccionar la ruta más adecuada para abordar un fenómeno determinado.

Sampieri y Torres (2018) explican que existen “tres rutas fundamentales: la cuantitativa, la cualitativa y la mixta” (p. 4), las cuales no deben considerarse como caminos totalmente independientes. Los autores señalan que estas rutas “se entrelazan y comparten ciertas cuestiones, no son ‘caminos completamente independientes’” (p. 4). Esta afirmación evidencia que los enfoques pueden complementarse según las necesidades del estudio. Además, la selección de la ruta metodológica debe responder al planteamiento del problema y a las circunstancias particulares de la investigación. Por ello, ningún enfoque es intrínsecamente superior a otro.

Sampieri y Torres (2018) enfatizan que “ninguna es intrínsecamente mejor que la otra, solo constituyen diferentes aproximaciones al estudio de un fenómeno” (p. 16). Esta postura rompe con la antigua dicotomía entre métodos y propone una visión más pragmática. La decisión metodológica debe fundamentarse en el fenómeno que se desea estudiar y en los objetivos que se persiguen. Asimismo, el investigador debe considerar sus conocimientos y experiencia en cada ruta. Así, la investigación se orienta con mayor coherencia y rigor científico.

Enfoque Cuantitativo

El enfoque cuantitativo constituye una de las rutas fundamentales para la generación de conocimiento científico. Sampieri y Torres (2018) señalan que esta ruta “representa un conjunto de procesos organizado de manera secuencial para comprobar ciertas suposiciones” (pp. 5-6). Esto implica que cada fase del proceso investigativo sigue un orden riguroso y estructurado. La investigación inicia con la delimitación de una idea, de la cual se derivan objetivos, preguntas e

hipótesis que posteriormente serán sometidas a prueba. Por ello, el proceso cuantitativo se caracteriza por su planificación previa y control metodológico.

Igualmente, esta ruta se fundamenta en la medición objetiva de variables y el análisis estadístico de los datos obtenidos. Al respecto, Sampieri y Torres (2018) explican que “los datos se encuentran en forma de números (cantidades) y, por tanto, su recolección se fundamenta en la medición” (p. 6). Esto significa que la información recolectada debe ser cuantificable y susceptible de análisis estadístico. La precisión en la medición permite evaluar hipótesis y determinar relaciones causales entre variables. Por ello, el enfoque cuantitativo es ampliamente utilizado en investigaciones que buscan exactitud y generalización de resultados.

En cuanto a su propósito, el enfoque cuantitativo pretende describir, explicar y predecir fenómenos mediante la formulación y comprobación de hipótesis. Sampieri y Torres (2018) afirman que “la meta principal es la prueba de hipótesis y la formulación y demostración de teorías” (p. 7). Esta característica evidencia su orientación deductiva, ya que parte de marcos teóricos previos para contrastarlos con la realidad empírica. Además, busca minimizar el error y descartar explicaciones rivales mediante el control metodológico. Como resultado, su finalidad es consolidar conocimiento verificable y replicable.

Otra característica esencial de esta ruta es la búsqueda de objetividad durante todo el proceso investigativo. Los autores Sampieri y Torres (2018) destacan que existe una “búsqueda de la mayor objetividad posible en todo el proceso o ruta” (p. 7). Esto implica que el investigador debe evitar que sus creencias o percepciones influyan en los resultados. La estandarización de procedimientos fortalece la validez y confiabilidad de los datos obtenidos. Por tanto, el enfoque cuantitativo procura aproximarse a la realidad externa de manera controlada y sistemática.

Enfoque Cualitativo

El enfoque cualitativo se fundamenta en el estudio sistemático de los fenómenos desde su naturaleza y significado. Al respecto, Sampieri y Torres (2018) señalan que: “Con el enfoque cualitativo también se estudian fenómenos de manera sistemática” (p. 7). Del mismo modo, Sampieri y Torres (2018) explican que el investigador “comienza el proceso examinando los hechos en sí y revisado los estudios previos, ambas acciones de manera simultánea” (p. 7). De esta forma, el proceso no parte de la comprobación de teorías previas, sino de la observación directa de la realidad para generar explicaciones consistentes con lo que se estudia.

Una característica esencial de esta ruta es su dinamismo metodológico. Los autores Sampieri y Torres (2018) establecen que: “Las investigaciones cualitativas suelen producir preguntas antes, durante o después de la recolección y análisis de los datos” (p. 8). Además, indican que “la acción indagatoria se mueve de manera dinámica entre los hechos y su interpretación” (p. 8). Esto implica que el proceso no es lineal, sino flexible, permitiendo ajustes conforme avanza la investigación y se profundiza en el fenómeno estudiado.

El enfoque cualitativo se distingue por su carácter inductivo y reconstructivo. Sampieri y Torres (2018) afirman que “en la ruta cualitativa predomina la lógica o razonamiento inductivo, dirigiéndose de lo particular a lo general” (p. 9). De igual manera, Sampieri y Torres (2018) señalan que: “su propósito es “reconstruir” la realidad, tal como la observan los actores de un sistema social definido previamente” (p. 9). De ahí que se busca comprender las experiencias y significados desde la perspectiva de los participantes involucrados en el contexto investigado.

En relación con la generación de hipótesis, los autores Sampieri y Torres (2018) precisan que: “En la mayoría de los estudios cualitativos no se prueban hipótesis, sino que se generan durante el proceso y van refinándose conforme se recaban más datos” (p. 9). Además, Sampieri y Torres (2018) destacan que “la investigación cualitativa resulta interpretativa pues pretende encontrar sentido a los fenómenos y hechos en función de los significados que las personas les otorguen” (p. 9). Por tanto, el énfasis se encuentra en la comprensión profunda del fenómeno más que en la medición numérica.

Aplicado al proyecto Sistema web para la gestión y pago de planilla en AraWeb CR, ubicada en San José, este enfoque permite comprender las experiencias, percepciones y necesidades del personal involucrado en los procesos de planilla. Tal como señalan los autores Sampieri y Torres (2018), “la ruta cualitativa resulta conveniente para comprender fenómenos desde la perspectiva de quienes los viven” (p. 9). De este modo, se facilita la identificación de requerimientos reales que orienten el diseño y desarrollo del sistema propuesto.

Enfoque Mixto

El enfoque mixto se caracteriza por integrar de manera articulada las rutas cuantitativa y cualitativa dentro de un mismo proceso investigativo. Sampieri y Torres (2018) señalan que: “Esta tercera vía para realizar investigación entrelaza a las dos anteriores (cuantitativa y cualitativa) y las mezcla, pero es más que la suma de las dos anteriores e implica su interacción y potenciación” (p. 10). De este modo, no se trata solo de aplicar ambos métodos, sino de integrarlos estratégicamente para potenciar sus fortalezas.

También, los autores Sampieri y Torres (2018) establecen que: “Los métodos mixtos o híbridos representan un conjunto de procesos sistemáticos, empíricos y críticos de investigación e implican la recolección y el análisis de datos tanto cuantitativos como cualitativos, así como su integración y discusión conjunta” (p. 10). Esta integración permite generar “metainferencias” producto de toda la información recabada, lo que permite una comprensión más amplia del fenómeno bajo estudio.

En relación con su aplicación, Sampieri y Torres (2018) indican que: “Los métodos mixtos pueden implementarse de acuerdo a diversas secuencias” (p. 10). Además, Sampieri y Torres (2018), menciona que puede ocurrir: “lo cuantitativo precede a lo cualitativo, en otras ocasiones lo cualitativo es primero; también pueden desarrollarse de manera simultánea o en paralelo” (p. 10). Esto demuestra la flexibilidad estructural del enfoque mixto según las necesidades del estudio.

La comparación entre rutas también evidencia su carácter integrador. Sampieri y Torres (2018) explican que mientras la investigación cuantitativa utiliza el “zoom in” y la cualitativa el “zoom out”, “una investigación mixta las funciones de ‘zoom in’ (acercamiento) y ‘zoom out’ (alejamiento) se utilizan constantemente” (p. 12). Esto refleja la combinación continua entre precisión numérica y comprensión contextual.

En el proyecto Sistema web para la gestión y pago de planilla en AraWeb CR, ubicada en San José, este enfoque permite analizar datos numéricos relacionados con procesos de planilla y, simultáneamente, comprender las experiencias y necesidades del personal involucrado. Como señalan los autores, “las rutas cuantitativa y cualitativa pueden mezclarse en diversos grados, lo que nos conduce a una ruta mixta” (p. 16). Esto respalda metodológicamente la integración de ambas perspectivas en un mismo estudio.

Enfoque de Investigación Seleccionado

Para el presente estudio se selecciona el enfoque mixto, debido a que integra las fortalezas de las rutas cuantitativa y cualitativa. Sampieri y Torres (2018) señalan que: “las rutas cuantitativa y cualitativa pueden mezclarse en diversos grados, lo que nos conduce a una ruta mixta” (p. 16). Por su parte, Sampieri y Torres (2018) establecen que “ninguna es intrínsecamente mejor que la otra, solo constituyen diferentes aproximaciones al estudio de un fenómeno” (p. 16). Esto respalda la elección metodológica basada en la naturaleza del problema de investigación.

Además, Sampieri y Torres (2018) indican que “hoy en día se recomienda utilizar la ruta más adecuada de acuerdo a varios factores: el planteamiento del problema, los conocimientos que tengamos de cada método, las circunstancias particulares que rodean a la investigación y la propia experiencia” (p. 16). En el caso del proyecto Sistema web para la gestión y pago de planilla en AraWeb CR, ubicada en San José, se requiere analizar datos numéricos de planilla y, simultáneamente, comprender las necesidades y experiencias del personal involucrado.

La elección también se fundamenta en que “a veces un solo método no es suficiente” (p. 16). Por eso, la integración de datos cuantitativos y cualitativos permite una comprensión más amplia del proceso de gestión y pago de planilla, favoreciendo decisiones fundamentadas tanto en mediciones objetivas como en interpretaciones contextualizadas.

Tipos de Investigación

Los proyectos de investigación pueden clasificarse según diversos criterios metodológicos, lo cual permite delimitar con mayor precisión su alcance, propósito y orientación científica. Esta clasificación resulta fundamental para estructurar adecuadamente el diseño metodológico y garantizar coherencia entre el problema planteado y el enfoque seleccionado. Al respecto, Narváez & Villegas (2014) establecen una tipología amplia que contempla distintos elementos vinculados al propósito, los medios de obtención de datos, el nivel de profundidad del conocimiento y el método empleado. Su propuesta ofrece un marco de referencia integral que facilita la comprensión de la diversidad de enfoques investigativos existentes. Al respecto, los autores señalan que:

Los proyectos de investigación pueden ser clasificados con base en los siguientes criterios: por el propósito, la investigación puede ser básica o aplicada; según los medios usados para obtener los datos, puede ser documental, de campo o experimental; atendiendo al nivel de

conocimientos que se adquieren, podrá ser exploratoria, descriptiva o explicativa; dependiendo del campo de conocimientos en que se realiza, es científica o filosófica; conforme al tipo de razonamiento empleado, es espontánea, racional o empírico-racional; acorde con el método utilizado, es analítica, sintética, deductiva, inductiva, histórica, comparativa, etc.; y conforme al número de investigadores que la realizan, es individual o colectiva. (párr. 1).

De igual manera, la investigación implica un proceso sistemático orientado a la solución de problemas y generación de conocimiento. Al respecto, Morganelli (2024) indica que “La investigación es mucho más que buscar palabras clave en un motor de búsqueda” (párr. 4). En esa misma línea, Morganelli (2024) señala que “cualquier tipo de investigación académica debería centrarse primero en identificar un problema y luego idear soluciones a él” (párr. 4).

Por tanto, los tipos de investigación pueden clasificarse atendiendo a múltiples criterios, tales como su propósito, nivel de profundidad, método empleado o campo de aplicación, lo que evidencia la amplitud y diversidad del campo metodológico. Esta clasificación permite organizar de manera sistemática los estudios científicos, facilitando la selección del enfoque más pertinente según la naturaleza del problema planteado. Asimismo, posibilita estructurar el proceso investigativo con mayor rigor, coherencia y claridad conceptual. En consecuencia, la correcta identificación del tipo de investigación constituye un elemento fundamental para garantizar la solidez metodológica del proyecto desarrollado.)

Investigación Exploratoria

La investigación exploratoria se caracteriza por abordar problemáticas que no han sido estudiadas en profundidad. Según Tegan (2023), “Investigación exploratoria es un enfoque metodológico que investiga preguntas de investigación que no han sido previamente estudiados en profundidad” (párr. 1). Este tipo de estudio permite aproximarse a fenómenos poco analizados, generando una base inicial para investigaciones posteriores más estructuradas. Sobre esto, su finalidad no es ofrecer conclusiones definitivas, sino ampliar la comprensión preliminar del objeto de estudio. De esta manera, facilita la identificación de variables relevantes, posibles relaciones entre ellas y aspectos que requieren mayor delimitación conceptual dentro del proceso investigativo.

Igualmente, Tegan (2023) establece que “La investigación exploratoria se utiliza a menudo cuando el tema que estás estudiando es nuevo o cuando proceso de recolección de datos es un desafío por alguna razón” (párr. 4). Esto indica que su aplicación resulta pertinente cuando no existe un marco previo sólido que oriente el análisis del problema a solucionar. Como resultado, este tipo de investigación permite al investigador adaptarse a contextos inciertos o con escasa información disponible. Además, favorece la formulación progresiva de hipótesis o preguntas más específicas que podrán desarrollarse en estudios posteriores. Por ello, se considera una etapa inicial clave dentro de procesos investigativos más amplios.

En cuanto a su naturaleza metodológica, Tegan (2023) señala que: “La investigación exploratoria es frecuentemente cualitativa y de naturaleza primaria” (párr. 2). Además, Tegan (2023) destaca su flexibilidad al indicar que “es muy flexible, rentable y abierto. Eres libre de proceder como creas que es mejor” (párr. 40). Estas características permiten adaptar el proceso investigativo a contextos donde la información disponible es limitada. Del mismo modo, la predominancia del enfoque cualitativo posibilita obtener información detallada mediante técnicas como entrevistas, observación o revisión documental. Esta apertura metodológica contribuye a comprender el fenómeno desde distintas perspectivas antes de establecer procedimientos más rígidos o cuantificables.

Aplicado al proyecto Sistema web para la gestión y pago de planilla en AraWeb CR, ubicada en San José, este tipo de investigación resulta pertinente porque permite examinar inicialmente la situación actual del proceso de planilla, identificar problemáticas y comprender las necesidades de la organización antes de diseñar una solución tecnológica estructurada. De igual manera, facilita la detección de debilidades operativas, posibles riesgos y oportunidades de mejora dentro del sistema actual. Esta aproximación preliminar proporciona insumos relevantes para definir con mayor precisión los requerimientos funcionales del sistema web propuesto. De ahí que la investigación exploratoria constituye una base metodológica adecuada para orientar el desarrollo del proyecto con mayor coherencia y fundamentación técnica.

Investigación Descriptiva

La investigación descriptiva se orienta a detallar y caracterizar una realidad concreta sin establecer relaciones causales. En esa línea, Narváez & Villegas (2014) indican que “Este tipo de investigación se efectúa cuando se desea describir, en todos sus componentes principales, una realidad” (párr. 9). De igual manera, Narváez & Villegas (2014) “Mediante este tipo de

investigación, que utiliza el método de análisis, se logra caracterizar un objeto de estudio o una situación concreta, señalar sus características y propiedades” (párr. 9).

De igual forma, Rholetter et al. (2023) establecen que “La investigación descriptiva es un enfoque metodológico que busca detallar comportamientos, situaciones, eventos y resultados sin profundizar en predicciones teóricas ni relaciones de causa y efecto” (párr. 1). Este tipo de investigación puede desarrollarse mediante diversos métodos, ya que como indica Rholetter et al. (2023), “Puede llevarse a cabo mediante diversos medios, como la observación, los estudios de caso y las encuestas” (párr. 1).

En cuanto a su alcance, Rholetter et al. (2023) señalan que: “El propósito de la investigación descriptiva, que puede ser cualitativa o cuantitativa, es describir comportamientos, situaciones, eventos y resultados. No pretende hacer predicciones teóricas ni explicar causa y efecto” (párr. 3). Además, Narváez & Villegas (2014) reconocen que: “Combinada con ciertos criterios de clasificación sirve para ordenar, agrupar o sistematizar los objetos involucrados en el trabajo indagatorio” (Narváez & Villegas, 2014, párr. 9).

Aplicado al proyecto Sistema web para la gestión y pago de planilla en AraWeb CR, ubicada en San José, la investigación descriptiva permite detallar el funcionamiento actual del proceso de planilla, identificar sus componentes, registrar sus características operativas y sistematizar la información relevante. Así, proporciona una base estructurada para comprender la realidad organizacional antes de implementar la solución tecnológica propuesta.

Investigación Aplicada

La investigación aplicada se orienta a la solución de problemas concretos del entorno real. En relación con esto, Research (2024), define que “La investigación aplicada es una exploración sistemática y organizada que se centra en resolver problemas específicos del mundo real o en mejorar prácticas, productos y servicios” (párr. 3). También, Research (2024), establece que “A diferencia de la investigación básica, que busca ampliar el conocimiento en general, la investigación aplicada toma conocimientos existentes y los emplea para encontrar soluciones prácticas a retos concretos” (párr. 3).

En cuanto a su finalidad, Research (2024) señala que “El propósito principal de la investigación aplicada es generar ideas y soluciones que tengan un impacto directo y tangible” (párr. 4). Además, Research (2024) enfatiza en que: “Se enfoca en cerrar la brecha entre teoría y

práctica, aplicando conocimientos en contextos reales para abordar desafíos específicos, tomar decisiones informadas y fomentar la innovación en distintos campos” (párr. 4).

Respecto a su relevancia, Research (2024) indica que la investigación aplicada aporta “Resolución de problemas: aporta soluciones prácticas a retos reales, mejorando procesos, productos y servicios” (Research, 2024, párr. 6). De igual manera, Research (2024) menciona: “En las empresas, la investigación aplicada contribuye a productos mejorados, mayor eficiencia y una ventaja en el mercado” (párr. 6).

Aplicado al proyecto Sistema web para la gestión y pago de planilla en AraWeb CR, ubicada en San José, este tipo de investigación resulta pertinente, ya que el estudio no se limita a describir una realidad, sino que busca desarrollar una solución tecnológica concreta que optimice el proceso de planilla, mejore la eficiencia operativa y responda a una necesidad organizacional específica. Por ello, la investigación aplicada permite transformar el análisis previo en una propuesta funcional con impacto directo en el entorno empresarial.

Tipo de Investigación Seleccionado

La investigación seleccionada para el desarrollo del proyecto es la investigación aplicada, debido a que el propósito central del estudio consiste en diseñar e implementar una solución tecnológica concreta para mejorar el proceso de gestión y pago de planilla en la empresa. Este tipo de investigación se caracteriza por centrarse en la resolución de problemas específicos del entorno real, utilizando conocimientos existentes para generar soluciones prácticas con impacto directo en la organización. Su orientación está dirigida a producir resultados funcionales que puedan ser utilizados de manera inmediata en el contexto donde surge la necesidad. Por eso, no se limita únicamente al análisis teórico, sino que busca intervenir de manera directa en la realidad estudiada para optimizar procesos y mejorar la eficiencia operativa.

A diferencia de la investigación exploratoria, que se orienta al primer acercamiento al problema, o la descriptiva, que se limita a detallar una realidad sin intervenir en ella, la investigación aplicada permite transformar el diagnóstico previo en una propuesta funcional y operativa. Sobre el tema, su enfoque resulta el más pertinente, ya que el proyecto no solo analiza la situación actual, sino que propone un sistema web como respuesta concreta a una necesidad organizacional identificada. En consecuencia, el estudio trasciende la etapa de análisis para materializar una herramienta tecnológica que aporte mejoras tangibles en la gestión administrativa.

Por lo tanto, la investigación aplicada constituye el marco metodológico adecuado para alcanzar los objetivos planteados en el proyecto.

Fuentes de Información

En toda investigación académica, las fuentes de información constituyen el sustento que permite respaldar teóricamente el estudio y otorgarle validez científica. Estas representan los medios a través de los cuales el investigador accede al conocimiento existente, analiza antecedentes y construye el marco conceptual que orienta el desarrollo del proyecto. Según explica Editorial Etecé (2025), “Las fuentes de información son documentos que se consultan para obtener datos de un tema sobre el que se quiere saber más” (párr. 2). Por su parte, Editorial Etecé (2025) señala que en una investigación “conforman el soporte que se usa para producir un escrito u otro tipo de trabajo” (párr. 2). En complemento, Maranto y González (2015) establecen que “Una fuente de información es todo aquello que nos proporciona datos para reconstruir hechos y las bases del conocimiento” (p. 2). En el desarrollo del proyecto Sistema web para la gestión y pago de planilla en AraWeb CR, ubicada en San José, las fuentes de información constituyen el fundamento teórico y metodológico que respalda el análisis y la propuesta tecnológica planteada.

Fuentes de Información Primarias

Las fuentes primarias son esenciales porque proporcionan información directa y original sobre el fenómeno que se estudia. Permiten al investigador acercarse a los hechos sin intermediaciones interpretativas, lo cual fortalece la precisión y autenticidad de los datos recopilados. De acuerdo con Maranto y González (2015), “Este tipo de fuentes contienen información original es decir son de primera mano, son el resultado de ideas, conceptos, teorías y resultados de investigaciones” (p. 3). Además, Maranto y González (2015) precisan que “Contienen información directa antes de ser interpretada, o evaluado por otra persona” (p. 3). Por su parte, Editorial Etecé (2025) indica que “Son las más cercanas al evento que se investiga, es decir, con la menor cantidad posible de intermediaciones” (párr. 7). En el presente proyecto, las fuentes primarias comprenden información directa obtenida de la organización, lo que permite sustentar técnicamente el diseño del sistema web propuesto.

Fuentes de Información Secundarias

Las fuentes secundarias cumplen una función analítica dentro de la investigación, ya que permiten interpretar, organizar y profundizar en información previamente generada. Estas aportan una visión estructurada del conocimiento existente y facilitan la comprensión teórica del problema de estudio. Como lo señalan Maranto y González (2015), “Este tipo de fuentes son las que ya han procesado información de una fuente primaria” (p. 3). En esa misma línea, Maranto y González (2015) explican que dicho proceso puede darse mediante “una interpretación, un análisis, así como la extracción y reorganización de la información de la fuente primaria” (p. 3). En concordancia, Editorial Etecé (2025) afirma que “Son las que se basan en las primarias y las sintetizan, analizan, interpretan o evalúan” (párr. 8). Para el proyecto, estas fuentes permiten fundamentar conceptualmente el sistema de gestión de planilla a partir de estudios, libros y documentos especializados.

Fuentes de Información Terciarias

Las fuentes terciarias facilitan la localización y organización del conocimiento disponible, sirviendo como herramientas de consulta general que orientan al investigador hacia documentos más específicos. Son particularmente útiles en las etapas iniciales de estructuración del marco referencial. Según comentan Maranto y González (2015), “Este tipo de fuentes son las que recopilan fuentes de información primarias o secundarias” (p. 3). Para comprender su uso, Maranto y González (2015) añaden que “Estas fuentes son utilizadas para buscar datos o para obtener una idea general sobre algún tema” (p. 3). De igual manera, Editorial Etecé (2025) define que “Son las que recopilan y comentan las fuentes primarias y/o secundarias, y por lo tanto, son una lectura mixta de testimonios e interpretaciones” (párr. 9). En el marco del proyecto, este tipo de fuentes facilita la localización organizada de información relevante para estructurar el marco referencial.

Índices, bibliografías y catálogos

Dentro del proceso investigativo, los instrumentos de búsqueda documental permiten identificar fuentes pertinentes y confiables, optimizando el acceso a información especializada. Estos recursos organizan referencias y orientan al investigador en la selección adecuada del material bibliográfico. En relación con los instrumentos de búsqueda documental, Maranto y González (2015) señalan que dentro de las fuentes terciarias se encuentran “bibliografías,

almacenes, directorios, donde se encuentran la referencia de otros documentos, que contienen nombres, títulos de revistas y otras publicaciones” (p. 3). Asimismo, Editorial Etecé (2025) menciona que son fuentes de información “los listados bibliográficos, los índices” (párr. 11). Estos recursos resultan fundamentales en el proyecto, ya que permiten identificar literatura pertinente y confiable para sustentar la investigación y garantizar rigurosidad académica.

Variables

Variables de Investigación

Las variables de investigación constituyen elementos esenciales dentro del diseño metodológico, ya que permiten estructurar el estudio, delimitar su alcance y facilitar la medición de los fenómenos analizados. En el contexto del proyecto Sistema web para la gestión y pago de planilla en AraWeb CR, ubicada en San José, las variables orientan la definición de los componentes técnicos y funcionales que serán evaluados. Según expone Stewart (2025), “Las variables son componentes fundamentales de la investigación que permiten medir y analizar datos” (párr. 1). Igualmente, Stewart (2025) define que “Una variable representa cualquier característica, número o cantidad que puede medirse o cuantificarse” (párr. 3). Estas definiciones permiten comprender que las variables no solo estructuran el análisis, sino que también hacen posible formular hipótesis e interpretar resultados dentro del proceso investigativo.

Variables Conceptuales

En cuanto a la definición conceptual, SalusPlay (s. f.) explica que “sería la definición real. Se trata de definiciones de diccionarios o de libros especializados que describen la esencia o las características de una variable, objeto o fenómeno” (párr. 9). Del mismo modo, SalusPlay (s. f.) aclara que dichas definiciones “son necesarias pero insuficientes para definir las variables de la investigación, porque no nos vinculan directamente con ‘la realidad’” (párr. 10). En el proyecto del sistema web para la gestión y pago de planilla en AraWeb CR, una variable conceptual podría ser la eficiencia del proceso de planilla, entendida teóricamente como la capacidad del sistema para optimizar tiempos, reducir errores y mejorar la organización administrativa.

Variables Operacionales

La definición operacional permite trasladar la variable desde un plano abstracto hacia uno empírico y medible. SalusPlay (s. f.) indica que la definición operacional “especifica qué

actividades u operaciones deben realizarse para medir una variable” (párr. 10). Esto implica que toda variable debe transformarse en acciones concretas que permitan la recolección de datos verificables.

Además, SalusPlay (s. f.) establece que “la definición conceptual es de índole teórica, mientras que la operacional nos da las bases de medición y la definición de los indicadores” (párr. 11). El proceso mediante el cual se lleva una variable de un nivel abstracto a un plano práctico se denomina operacionalización, según SalusPlay (s. f.) cuya función es “precisar o concretar al máximo el significado o alcance que se otorga a una variable en un determinado estudio” (párr. 14).

En el caso del sistema web propuesto, la eficiencia del proceso de planilla podría operacionalizarse mediante indicadores como el tiempo promedio de procesamiento, la cantidad de errores detectados antes y después de la implementación del sistema, o el nivel de satisfacción de los usuarios internos. De esta manera, la variable conceptual adquiere un carácter medible y verificable dentro del contexto del estudio.

Variables Instrumentales

El concepto de variable instrumental se enmarca dentro de los estudios que buscan establecer relaciones causales en contextos observacionales. El análisis de variables instrumentales según Iwashyna y Kennedy (2013), “es una técnica comúnmente utilizada en las ciencias sociales para proporcionar evidencia de que un tratamiento causa un resultado, en contraste con la evidencia de que un tratamiento está meramente asociado con diferencias en un resultado” (párr. 1). Esto implica que el propósito central de las variables instrumentales es fortalecer la inferencia causal cuando no existe un experimento controlado aleatorizado.

Iwashyna y Kennedy (2013) explican que esta técnica “explota situaciones en las que cierto grado de aleatoriedad afecta la forma en que se seleccionan los pacientes para un tratamiento” (párr. 1), lo cual permite aproximarse a condiciones de asignación aleatoria dentro de datos observacionales. Al respecto, Iwashyna y Kennedy (2013) definen que “Una variable instrumental es una característica del mundo que lleva a que algunas personas tengan más probabilidades de recibir el tratamiento específico que queremos estudiar, pero no cambia de otro modo los resultados de esos pacientes” (párr. 1).

Tabla 4
Cuadro de variables.

Objetivo específico	Variable	Variable conceptual	Variable operacional	Variable instrumental
Analizar los requerimientos funcionales y no funcionales del sistema web.	Requerimientos del sistema.	Según Sommerville (2011), “Los requerimientos para un sistema son descripciones de lo que el sistema debe hacer: el servicio que ofrece y las restricciones en su operación” (p. 83).	Cuestionarios. Observación.	Encuesta. Guía de observación.
Construir el diseño de la base de datos y del sistema web mediante la herramienta Workbench y Visual Studio.	Diseño de base de datos y arquitectura.	Según IBM Documentation (2026), “Al diseñar una base de datos, está modelando un sistema empresarial real que contiene un conjunto de entidades y sus características, o atributos, y las reglas o relaciones entre dichas entidades” (párr. 1).	Casos de Uso, Documentos de análisis. Diagrama entidad relación.	MySQL Workbench (modelado entidad-relación).
Programar los diferentes módulos del sistema mediante la herramienta C#.	Desarrollo de módulos del sistema.	Según Open University (2018) indican que “En el desarrollo de software, existe una larga tradición de descomposición de un sistema en módulos más pequeños. Esta modularización es la técnica estándar para gestionar sistemas grandes y complejos” (párr. 1).	Documentos de diseño. Diagramas de Clases Gestor de base de datos.	Visual Studio Community 2022. SQL Server 2022.
Probar los diferentes módulos de forma independiente y también de forma integral lo que garantiza los resultados correctos del sistema.	Pruebas del sistema.	Según Powell y Smalley (2025), las pruebas del sistema “se pueden clasificar en uno de tres grupos principales. La primera, las pruebas funcionales, se ocupa del rendimiento del sistema, pero no busca una respuesta más profunda más allá de saber si el software funciona como se prometió” (párr. 13).	Pruebas unitarias. Pruebas integrales.	Visual Studio Community 2022. Casos de prueba.

Fuente:” Elaboración propia, 2026”.

Población

En el desarrollo de una investigación, resulta fundamental delimitar adecuadamente el grupo de estudio, partiendo primero de la identificación del universo. Según comenta Salazar (2021), “Universo: es la totalidad de individuos o elementos que presentan una característica susceptible para ser estudiada” (p. 5). De igual manera, el autor señala que este puede ser finito o infinito, por lo que metodológicamente se requiere seleccionar únicamente una parte que sea viable para el análisis. En el proyecto Sistema web para la gestión y pago de planilla en AraWeb CR, ubicada en San José, el universo estaría conformado por todos los colaboradores y elementos vinculados al proceso de gestión de planilla dentro de la organización.

Por otra parte, como lo explica Salazar (2021), “Población: es la parte del universo con la cual se va a realizar el estudio, según las características de la investigación” (p. 5). Esto implica que la población no abarca necesariamente la totalidad del universo, sino el subconjunto que cumple con los criterios definidos en el planteamiento metodológico. En el caso del presente proyecto, la población estará constituida por los funcionarios administrativos y personal encargado del manejo de planillas en AraWeb CR, ya que son quienes interactúan directamente con el proceso que será optimizado mediante la implementación del sistema web propuesto.

Muestra

En el proceso metodológico, después de definir la población, corresponde determinar la muestra cuando el tamaño del grupo de estudio lo requiere. Según expresa Salazar (2021), “Muestra: es la parte o subconjunto de la población, con la que se realiza el estudio y para seleccionarla se deben delimitar las características de la población” (p. 5). Esto implica que la muestra constituye una fracción representativa de la población total, seleccionada bajo criterios previamente establecidos, con el fin de garantizar que los resultados obtenidos sean válidos y pertinentes dentro del contexto de la investigación. La adecuada delimitación de estas características resulta esencial para asegurar coherencia entre los objetivos del estudio y los sujetos analizados.

No obstante, en el caso del proyecto Sistema web para la gestión y pago de planilla en AraWeb CR, ubicada en San José, la empresa cuenta con una población menor a diez empleados. Debido a este tamaño reducido, no se requiere realizar un cálculo muestral, ya que se trabajará con la totalidad de la población existente. Por tanto, el estudio adoptará un enfoque censal,

incorporando a todos los colaboradores vinculados al proceso de gestión de planilla, lo que permitirá obtener información completa y directa sin necesidad de seleccionar un subconjunto representativo.

Cálculo de la muestra

El tamaño de la muestra constituye un elemento determinante en el diseño metodológico de una investigación, especialmente cuando se aplican instrumentos como encuestas. Según señala QuestionPro (2026), “El tamaño de la muestra de una encuesta es muy importante para poder realizar una investigación de manera correcta, por lo que hay que tener en cuenta los objetivos y las circunstancias en que se desarrolle la investigación” (párr. 1). Esta consideración implica que la definición del tamaño muestral no es arbitraria, sino que debe responder directamente a la finalidad del estudio y al contexto en el cual se desarrolla.

También, QuestionPro (2026) enfatiza que “Una muestra demasiado grande dará lugar a la pérdida de valiosos recursos como tiempo y dinero, mientras que una muestra pequeña puede no proporcionar información confiable” (párr. 3). Sobre esto, determinar el tamaño adecuado requiere equilibrar la precisión de los resultados con la eficiencia en el uso de recursos. Tal como plantea QuestionPro (2026), “Esto sin duda depende de qué tan exactos necesites que sean los datos obtenidos en tu encuesta, que tan cercanos quieres que sean a los de la población total” (párr. 4), lo cual refuerza la importancia de definir previamente el nivel de exactitud esperado en el estudio.

Por otra parte, QuestionPro (2026) indica que el tamaño de la muestra puede ser “Representativa: Hace referencia a que todos los miembros de un grupo de personas tengan las mismas oportunidades de participar en la investigación” (párr. 5), así como “Adecuada: Se refiere a que el tamaño de la muestra debe de ser obtenido mediante un análisis que permite resultados como disminuir el margen de error” (párr. 6). En el contexto del proyecto Sistema web para la gestión y pago de planilla en AraWeb CR, ubicada en San José, estos criterios refuerzan la necesidad de que cualquier procedimiento muestral garantice equidad en la participación y confiabilidad en los resultados, aun cuando, debido al tamaño reducido de la empresa, se haya determinado trabajar con la totalidad de la población.

Con base en lo anterior, para determinar de manera técnica y objetiva el número de participantes requeridos en una investigación, se recurre a la fórmula estadística para el cálculo del tamaño de la muestra, la cual permite establecer la cantidad mínima necesaria de sujetos según el nivel de confianza, el margen de error y la variabilidad esperada. Esta fórmula garantiza que los

resultados obtenidos posean validez y precisión dentro de parámetros previamente definidos. A continuación, se presenta la fórmula correspondiente para el cálculo del tamaño de la muestra.

n = tamaño de la muestra.

N = tamaño de la población.

K = nivel de confianza.

p = proporción esperada.

q = probabilidad de fracaso.

e = precisión (margen de error).

$$n = \frac{k^2 qpN}{e^2(N - 1) + k^2 pq}$$

Fuente: "QuestionPro, 2026".

Instrumentos de recolección de datos

La Observación

La observación constituye uno de los instrumentos fundamentales dentro del proceso investigativo, al permitir el contacto directo con la realidad objeto de estudio. Según comenta U San Marcos (2020), "La observación es un elemento fundamental para todo proceso de investigación; en ella se apoya el investigador para poder recolectar la mayor cantidad de datos" (p. 2). Por su parte, el mismo texto de U San Marcos (2020) señala que "Gran parte del centenar de conocimientos que forman lo que es la ciencia ha sido logrado mediante la técnica de observación" (p. 2). En el contexto del proyecto Sistema web para la gestión y pago de planilla en AraWeb CR, ubicada en San José, esta técnica permitirá examinar directamente los procedimientos actuales relacionados con la gestión de planilla dentro de la empresa.

Desde otra perspectiva, Lifeder (2021) define que "La observación es una técnica que consiste precisamente en observar el desarrollo del fenómeno que se desea analizar" (párr. 19). En esa misma línea, Lifeder (2021) indica que "Al aplicar esta técnica con enfoque cualitativo, es necesario organizar las observaciones en categorías temáticas para poder darle un orden al análisis" (párr. 23). En el marco del proyecto, esta organización temática facilitará la identificación de patrones, errores recurrentes y oportunidades de mejora en el sistema manual o semi-digital

actualmente utilizado por la empresa, contribuyendo así a fundamentar el diseño del sistema web propuesto.

En relación con su clasificación, U San Marcos (2020) indica que “Existen dos clases de observación establecidas: la primera es la observación no científica y la segunda es la observación científica” (p. 3). Como lo explica U San Marcos (2020), “Observar científicamente significa observar con un objetivo claro, definido y preciso” (p. 3), lo que implica una planificación previa y la elaboración de una guía estructurada. Por el contrario, U San Marcos (2020), menciona que “Observar de una manera no científica, por el contrario, significa realizar una observación sin intención, sin objetivo definido y por tanto, sin preparación previa o guía de la misma” (p. 3). Para efectos de esta investigación, se adoptará una observación científica, orientada específicamente a identificar debilidades y necesidades en el proceso actual de pago de planilla.

Respecto a sus modalidades, U San Marcos (2020) expone que “La observación científica puede darse de diferentes modalidades como lo son: directa o indirecta, participante o no participante, estructurada o no estructurada, de campo o de laboratorio, así como, individual o de equipo” (p. 5). En cuanto a la observación directa, U San Marcos (2020) establece que “La observación es directa cuando el investigador se pone en contacto personalmente con el hecho o fenómeno que trata de investigar” (p. 5), U San Marcos (2020) también establece que “es indirecta cuando el investigador entra en conocimiento del hecho o fenómeno observando a través de las observaciones realizadas anteriormente por otra persona” (p. 5). En el desarrollo del presente proyecto, se empleará principalmente la modalidad indirecta, al analizar los procedimientos administrativos de la empresa e información brindada por los colaboradores.

La Encuesta

La encuesta se configura como una de las técnicas más utilizadas dentro del enfoque cuantitativo, debido a su capacidad para recopilar información estructurada de manera sistemática. Según señala Medina Romero et al. (2023), “La encuesta es una técnica de investigación que se utiliza para recopilar información de un gran número de personas” (p. 23). Asimismo, Medina Romero et al. (2023) indican que “Se trata de una herramienta versátil y accesible que permite a los investigadores obtener información sobre comportamientos, actitudes, opiniones y demografía de una población objetivo” (p. 23). En el proyecto Sistema web para la gestión y pago de planilla en AraWeb CR, ubicada en San José, esta técnica permitirá conocer la percepción de los colaboradores respecto al proceso actual de gestión de planilla.

En cuanto a sus ventajas, Medina Romero et al. (2023) destacan que “Una de las principales ventajas de la encuesta es su capacidad para recopilar información de un gran número de personas en un período corto de tiempo” (p. 24). Del mismo modo, Medina Romero et al. (2023) afirman que “Las encuestas también son una herramienta eficiente y económica, ya que permiten a los investigadores recopilar información a través de un medio de comunicación más económico que otras técnicas, como las entrevistas” (p. 24).

Respecto a los criterios para su elección, Medina Romero et al. (2023) establecen que “Para elegir la encuesta como técnica desde el enfoque cuantitativo, se deben tener los siguientes criterios: Establecer que el estudio va a tener solo datos numéricos y por tanto habrá una prueba de hipótesis” (p. 25). Igualmente, Medina Romero et al. (2023) señalan que “El instrumento debe tener criterios de confiabilidad y validez para ser aplicado” (p. 25). Estos lineamientos serán considerados en la aplicación de la encuesta dentro del proyecto, lo que garantiza la consistencia y el rigor metodológico en el análisis de los datos recopilados.

Riesgos que conlleva la aplicación de cuestionarios

El cuestionario constituye un instrumento ampliamente utilizado en los procesos de investigación, pero su aplicación implica riesgos que deben ser considerados con rigor metodológico. Según comenta Medina Romero et al. (2023), “Un cuestionario bien diseñado es crucial para obtener resultados precisos y confiables. Debe tener preguntas claras y concisas, que permitan al participante proporcionar información precisa y relevante” (p. 39). En el contexto del proyecto Sistema web para la gestión y pago de planilla en AraWeb CR, ubicada en San José, un diseño inadecuado podría generar datos imprecisos que afecten la toma de decisiones para el desarrollo del sistema.

Como lo explican Medina Romero et al. (2023), “es importante asegurarse de que el cuestionario no contenga preguntas sesgadas o tendenciosas, ya que esto puede afectar la validez de los resultados” (p. 39). De igual manera, Medina Romero et al. (2023), señalan que “También es importante garantizar la privacidad y la confidencialidad de los participantes, especialmente si se están recopilando datos sensibles o confidenciales” (p. 39). Estos aspectos representan riesgos metodológicos y éticos que deben prevenirse mediante un diseño responsable, sobre todo cuando se recaban datos relacionados con procesos administrativos internos de la empresa.

En relación con el diseño técnico del instrumento, Medina Romero et al. (2023) establecen que “El diseño de un cuestionario de investigación debe ser cuidadoso y riguroso, para asegurarse

de que las preguntas sean claras, objetivas y no sesgadas” (p. 40). Además, Medina Romero et al. (2023), indican que “debe asegurarse de que el cuestionario sea apropiado para el grupo de personas que se está investigando y para el propósito de la investigación” (p. 40). Si no se consideran estas directrices, existe el riesgo de que los resultados no reflejen adecuadamente la realidad del proceso de gestión de planilla en AraWeb CR.

Por otra parte, en cuanto a su aplicación, Medina Romero et al. (2023) advierten que “Es importante asegurarse de que los participantes entiendan las preguntas y tengan suficiente tiempo para responderlas” (p. 40). Así mismo Medina Romero et al. (2023), reiteran que “También es importante garantizar la privacidad y la confidencialidad de los participantes” (p. 40). Una aplicación inadecuada puede provocar respuestas incompletas o erróneas, afectando la calidad de los datos obtenidos para fundamentar el desarrollo del sistema web propuesto.

Finalmente, en términos de estructura y tipos de respuesta, Medina Romero et al. (2023) indican que “Se puede aplicar un cuestionario dicotómico (con dos opciones de respuestas), o un cuestionario politómico (de tres a más posibles respuestas)” (p. 39). Una selección inadecuada del tipo de cuestionario puede constituir un riesgo metodológico, ya que podría limitar la información necesaria para analizar adecuadamente las necesidades del proceso de pago de planilla en la organización.

En este apartado se mencionó constantemente a Medina Romero et al. (2023), debido a que la información de esta fuente ha sido precisa y completa para llevar a cabo la investigación y redacción de esta sección del marco metodológico. La forma en que los autores desarrollan los conceptos relacionados con el cuestionario y la encuesta permitió fundamentar teóricamente los elementos abordados, manteniendo coherencia con el enfoque de la investigación. Del mismo modo, los criterios expuestos en la obra facilitaron la comprensión de los aspectos técnicos que deben considerarse en la aplicación de instrumentos de recolección de datos. En esa línea, la consulta de dicha fuente contribuyó como referencia conceptual para sustentar adecuadamente este apartado dentro del trabajo de investigación.

Documentos de archivo y fuentes gubernamentales

Documentos de archivo

Los documentos de archivo constituyen una fuente primaria de información dentro de los procesos organizacionales, ya que surgen directamente de las actividades propias de una entidad.

Según explica Villuendas (2025), “Los documentos de archivo son aquellos que son generados por las organizaciones durante el desarrollo de sus actividades, como actas, resoluciones, acuerdos, expedientes, nóminas, libros de contabilidad y un lugar etcétera” (párr. 1). Esta definición permite comprender que dichos documentos forman parte del funcionamiento cotidiano de una organización, lo cual resulta pertinente para el análisis del proceso de gestión y pago de planilla en AraWeb CR, ubicada en San José.

En cuanto a sus características formales, Villuendas (2025) señala que “Se pueden presentar en múltiples soportes, desde textuales a fotográficos, audiovisuales o electrónicos” (párr. 1). De igual manera, Villuendas (2025) indica que “Pueden presentarse en distintos formatos de archivo de documento (textual, gráfico, audiovisual o digital) y se conservan como evidencia de una actividad” (párr. 2). Esta diversidad de formatos evidencia la necesidad de contar con sistemas que permitan organizar y gestionar adecuadamente la información, especialmente cuando se pretende implementar un sistema web orientado a la administración de planillas.

Fuentes gubernamentales

Las fuentes gubernamentales representan un insumo relevante dentro de los procesos de investigación, debido a su carácter oficial y a la naturaleza pública de la información que contienen. Según indica MC Library Montgomery College (s. f.), “Toda información pública producida por los gobiernos federales, estatales, locales e internacionales se considera una fuente primaria útil para cualquier proyecto de investigación” (párr. 1). Esta característica las convierte en referentes formales para respaldar estudios que requieran datos normativos, estadísticos o administrativos, como ocurre en el proyecto Sistema web para la gestión y pago de planilla en AraWeb CR, ubicada en San José.

En cuanto a su disponibilidad y acceso, MC Library Montgomery College (s. f.), señala que “Un documento gubernamental puede estar disponible en diversos formatos, como impreso, en CD-ROM y en línea” (párr. 1). Esta diversidad de formatos facilita la consulta y verificación de información oficial, lo cual resulta fundamental cuando se requiere sustentar procedimientos vinculados al cumplimiento de obligaciones legales o administrativas relacionadas con la gestión de planillas.

También, MC Library Montgomery College (s. f.), establece que estas fuentes “Constituyen una fuente confiable de información detallada sobre datos y estadísticas sobre un tema específico” (párr. 1). En relación con esto, la utilización de fuentes gubernamentales dentro de la presente

investigación permite reforzar la validez de la información empleada, especialmente en aquellos aspectos vinculados con regulaciones laborales, disposiciones fiscales o lineamientos oficiales que inciden directamente en el diseño y funcionamiento del sistema web propuesto para la empresa AraWeb CR.

Proceso para la Recolección y Análisis de Datos

El proceso de recolección y análisis de datos constituye un componente esencial dentro de la gestión de proyectos, ya que permite sustentar la toma de decisiones de manera objetiva. Según comenta Admin (2023), “Los Métodos para Recopilación y Análisis de Datos son herramientas esenciales en la gestión de proyectos” (párr. 1). Por su parte, Admin (2023), señala que “se utilizan para recolectar, valorar y evaluar datos e información con el objetivo de obtener una comprensión más profunda de una situación en particular, lo cual ayuda a la toma de decisiones estratégicas en un proyecto” (párr. 1). En el marco del proyecto Sistema web para la gestión y pago de planilla en AraWeb CR, ubicada en San José, este proceso permitirá fundamentar técnicamente el diseño y desarrollo del sistema propuesto.

En relación con los enfoques metodológicos, Admin (2023) explica que “Estos métodos incluyen una variedad de técnicas cualitativas y cuantitativas” (párr. 2). Además, Admin (2023) detalla que “Las técnicas cualitativas suelen ser subjetivas y se basan en la interpretación y el juicio del gestor del proyecto o del equipo” (párr. 2), mientras que “las técnicas cuantitativas son objetivas y se basan en datos numéricos que pueden ser analizados de forma estadística” (párr. 2). Esta diferenciación resulta pertinente para definir las herramientas que se emplearán en la investigación, considerando las características y necesidades de la empresa.

En cuanto a la planificación del proceso, Admin (2023) indica que “Una de las primeras etapas en la recopilación de datos es definir qué tipo de información se necesita” (párr. 3). Igualmente, Admin (2023) establece que “También es crucial establecer el marco temporal para la recopilación de datos y la cantidad de datos que se necesita” (párr. 3). Estas directrices permiten estructurar adecuadamente la fase inicial del estudio, con lo que se asegura que la información obtenida responda directamente a los objetivos planteados para la mejora del proceso de gestión de planilla.

Posteriormente, Admin (2023) manifiesta que “Una vez que se han definido estos parámetros, se puede seleccionar el método más adecuado para la recopilación de datos. Algunas técnicas pueden ser más efectivas que otras en determinadas situaciones.” (párr. 4). En el caso de

AraWeb CR, la selección del método responderá a la necesidad de identificar tanto percepciones generales como aspectos específicos del proceso actual.

Respecto al análisis de la información, Admin (2023), indica que “Después de recopilar los datos, el siguiente paso es analizarlos. Este proceso puede implicar la identificación de patrones, la comparación de datos, la interpretación de los resultados y la formulación de conclusiones.” (párr. 5). En esa misma línea, Admin (2023), advierte que “El análisis de datos puede ser un proceso complejo y a menudo requiere de habilidades y conocimientos especializados” (párr. 5). Este procedimiento será fundamental para transformar los datos recolectados en información útil que oriente el desarrollo del sistema web.

Finalmente, Admin (2023), destaca que “Es importante tener en cuenta que la recopilación y el análisis de datos deben ser un proceso continuo durante toda la vida del proyecto” (párr. 6). Del mismo modo, Admin (2023), menciona que este proceso “se lleva a cabo a lo largo de todo el ciclo de vida del proyecto, desde la fase de inicio hasta la de cierre, y ayuda a informar y guiar la toma de decisiones en todas las etapas del proyecto” (párr. 7). Como resultado, dentro del proyecto propuesto, la recolección y análisis de datos no se limitarán a una etapa específica, sino que formarán parte integral del seguimiento y mejora continua del sistema implementado hasta su conclusión.

CAPITULO IV: ANALISIS DE RESULTADOS

Este capítulo presenta la interpretación de la información obtenida durante el proceso investigativo, con el propósito de evidenciar los hallazgos derivados de la aplicación de los instrumentos definidos en el marco metodológico. Los datos fueron recolectados mediante la aplicación de cuestionarios y la guía de observación desarrollada en la empresa AraWeb CR, ubicada en San José, permitiendo conocer la situación actual relacionada con la gestión y pago de planilla.

La investigación se llevó a cabo durante el mes de abril del año 2026, periodo en el cual se recopiló información relevante proveniente de los colaboradores y del análisis directo de los procesos administrativos existentes. Los resultados obtenidos sirvieron como base para identificar necesidades, problemáticas y oportunidades de mejora dentro de la organización. Así, el análisis realizado permite fundamentar técnicamente el diseño del sistema web propuesto, alineando los resultados obtenidos con los objetivos planteados en el estudio.

Cuestionario

El presente apartado expone el análisis de los resultados obtenidos mediante la aplicación del cuestionario dirigido a los colaboradores de la empresa AraWeb CR, ubicada en San José, como parte del proceso investigativo del proyecto Sistema web para la gestión y pago de planilla. La encuesta fue aplicada durante el mes de abril del año 2026 con el propósito de recopilar información directa sobre la percepción de los empleados respecto al proceso actual de gestión y pago de planilla. A partir de las respuestas obtenidas, se identificaron fortalezas, debilidades y necesidades asociadas al manejo manual de la información salarial.

Los datos recolectados permitieron evaluar el nivel de eficiencia, confiabilidad y organización del proceso existente, así como determinar el grado de aceptación hacia la implementación de una solución tecnológica. De ahí que el análisis presentado a continuación interpreta cada resultado obtenido, relacionándolo con la viabilidad y pertinencia del desarrollo de un sistema web orientado a optimizar la gestión administrativa y mejorar la experiencia de los colaboradores dentro de la organización.

Análisis interpretativo del cuestionario

Pregunta 1. ¿El proceso actual de gestión de planilla en la empresa se realiza de manera eficiente?

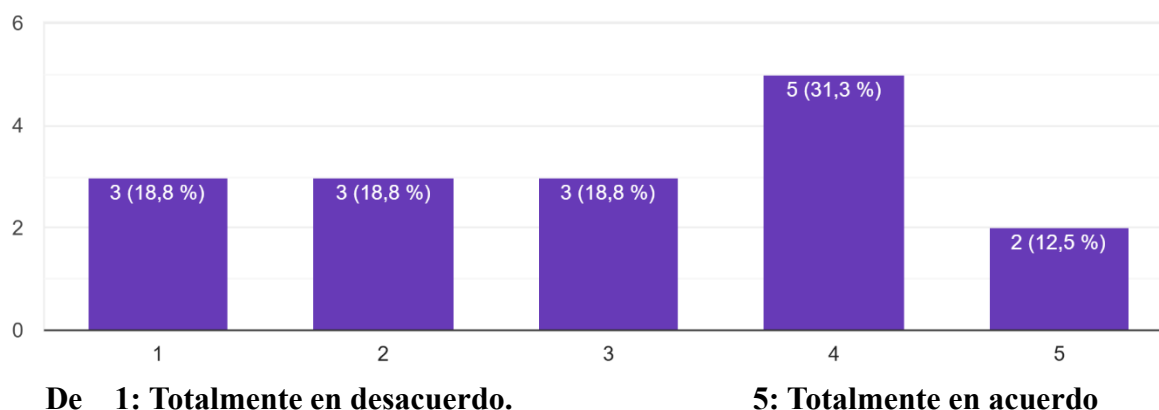
Los resultados obtenidos reflejan opiniones divididas entre los colaboradores, predominando respuestas neutrales y niveles de desacuerdo moderado. Esto evidencia que el proceso actual no es percibido como completamente eficiente, lo cual confirma la existencia de limitaciones operativas asociadas al manejo manual de la planilla. La dispersión de respuestas indica inconsistencias en la experiencia de los usuarios y falta de estandarización en los procedimientos administrativos. Por eso, se identifica una oportunidad clara de mejora mediante la automatización del proceso. Este resultado respalda la necesidad de implementar una solución tecnológica que optimice la gestión salarial.

Figura 2.

Cuestionario – Gráfico Pregunta 1.

1. ¿El proceso actual de gestión de planilla en la empresa se realiza de manera eficiente?

Número de colaboradores



Fuente: Respuestas del cuestionario creado en Google Forms.

Pregunta 2. ¿Considera que el proceso actual de pago de planilla presenta retrasos frecuentes?

La mayoría de respuestas se concentra entre posiciones neutrales y de acuerdo, lo que sugiere que los retrasos constituyen una situación recurrente dentro de la organización. Este comportamiento confirma que los procesos manuales generan tiempos de ejecución prolongados y dependencia de tareas administrativas repetitivas. La percepción de retraso impacta directamente

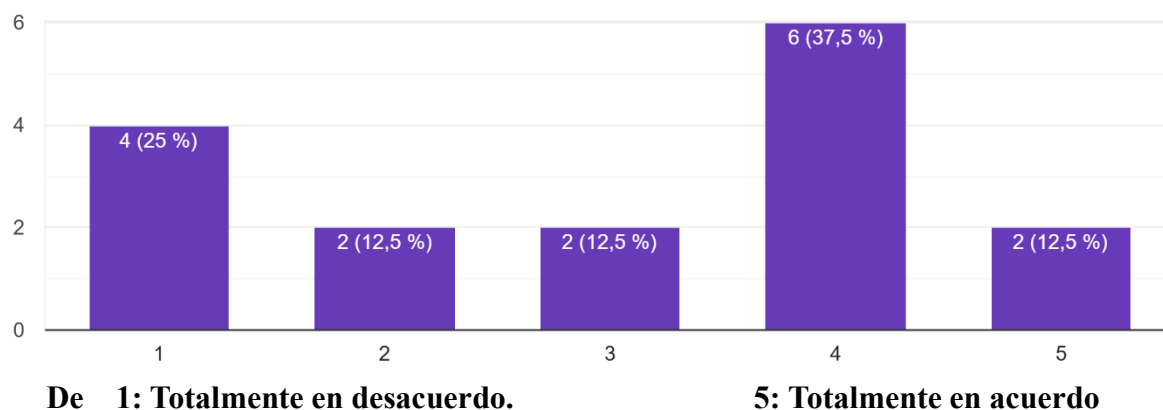
la satisfacción laboral y la confianza en el sistema de pago. Por tanto, los resultados evidencian la necesidad de reducir tiempos operativos mediante herramientas digitales que automaticen cálculos y validaciones. La implementación de un sistema web se perfila como una alternativa viable para mitigar esta problemática.

Figura 3.

Cuestionario – Gráfico Pregunta 2.

2. ¿Considera que el proceso actual de pago de planilla presenta retrasos frecuentes?

Número de colaboradores



Fuente: Respuestas del cuestionario creado en Google Forms.

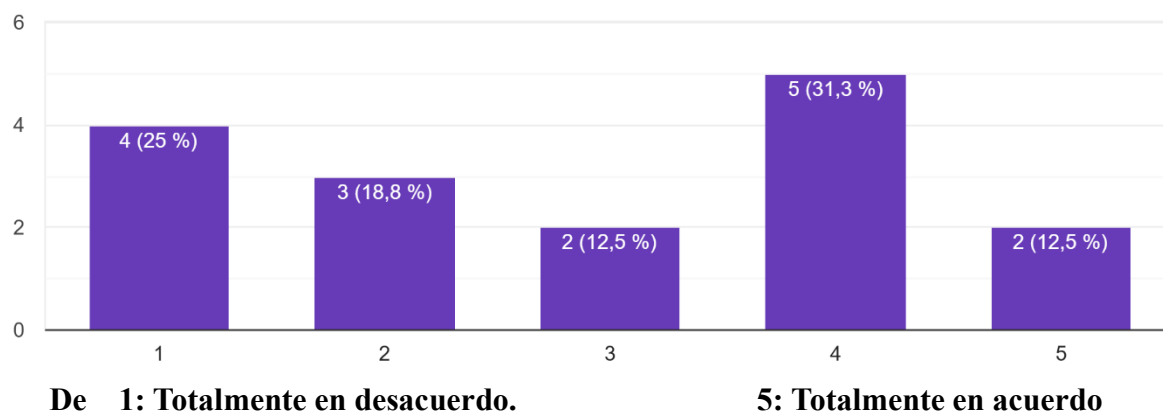
Pregunta 3. ¿Con qué frecuencia detecta errores en su cálculo de salario?

Las respuestas obtenidas muestran una tendencia hacia valores intermedios, indicando que los errores no son constantes, pero sí ocurren con cierta frecuencia. Este hallazgo resulta relevante, ya que incluso errores ocasionales afectan la credibilidad del proceso de planilla. La existencia de cálculos manuales incrementa la probabilidad de inconsistencias humanas y reprocesos administrativos. Desde una perspectiva organizacional, la reducción de errores constituye un factor crítico para mejorar la eficiencia financiera. Por tanto, se reafirma la importancia de incorporar mecanismos automatizados de cálculo dentro del sistema propuesto.

Figura 4.*Cuestionario – Gráfico Pregunta 3.*

3. ¿Con qué frecuencia detecta errores en su cálculo de salario?

Número de colaboradores



Fuente: Respuestas del cuestionario creado en Google Forms.

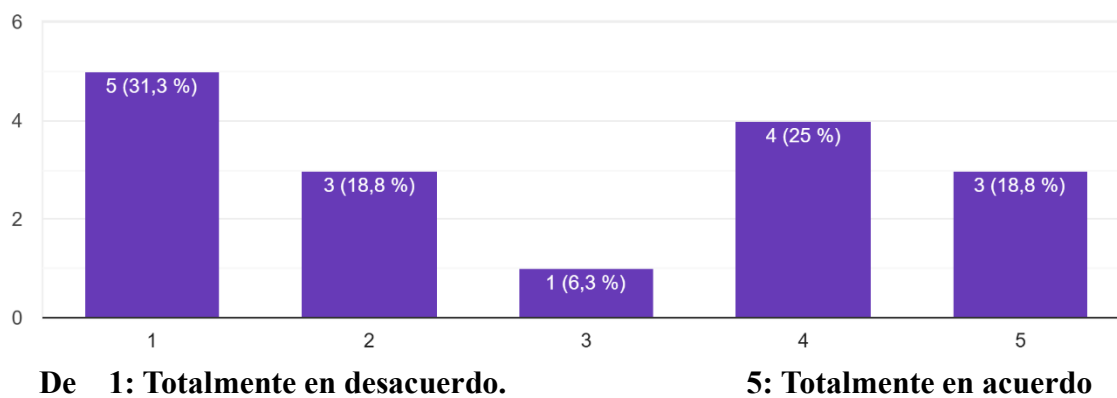
Pregunta 4. ¿El sistema actual (manual) facilita el acceso rápido a la información de planilla?

Los resultados evidencian una percepción predominantemente negativa respecto al acceso a la información salarial. Los colaboradores consideran que la consulta de datos requiere tiempo adicional y depende de intermediación administrativa. Esta situación limita la transparencia y dificulta la toma de decisiones oportunas tanto para empleados como para responsables administrativos. La falta de disponibilidad inmediata de información confirma las debilidades propias de los registros manuales. Por ello, un sistema web con acceso en línea permitiría mejorar significativamente la disponibilidad y consulta de la información.

Figura 5.*Cuestionario – Gráfico Pregunta 4.*

4. ¿El sistema actual (manual) facilita el acceso rápido a la información de planilla?

Número de colaboradores



Fuente: Respuestas del cuestionario creado en Google Forms.

Pregunta 5. ¿La información relacionada con su salario es clara y comprensible?

Considerando la tendencia general observada en el resto de las preguntas, las respuestas presentan una valoración intermedia con inclinación positiva. Esto indica que, aunque los colaboradores logran comprender su información salarial, existen oportunidades para mejorar la presentación y organización de los datos. La claridad de la información depende actualmente de explicaciones adicionales por parte del personal administrativo. Un sistema digital permitiría estructurar los datos de manera más visual, ordenada y accesible. Como resultado, la digitalización contribuiría a fortalecer la comprensión y transparencia del proceso de planilla.

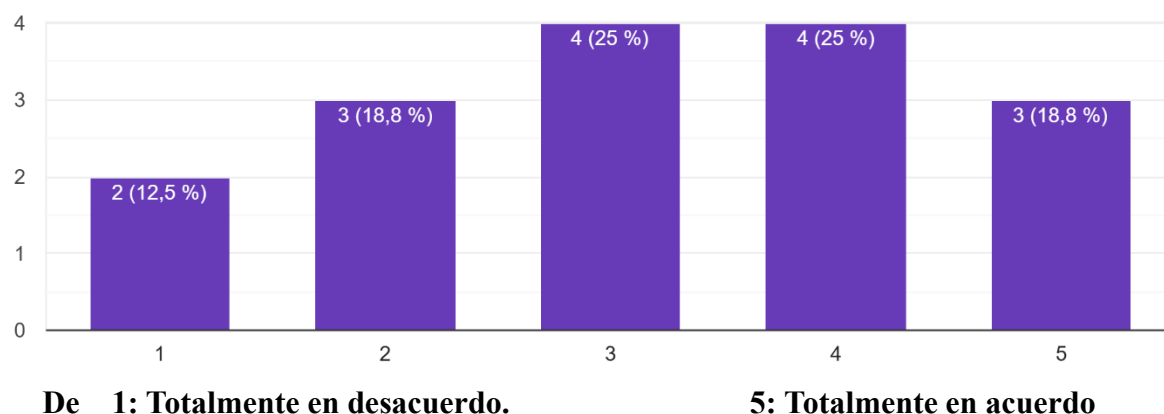
Pregunta 6. ¿Considera que el proceso actual garantiza la confidencialidad de la información salarial?

Las respuestas muestran percepciones mixtas respecto a la seguridad de la información, evidenciando que el manejo manual genera dudas sobre la confidencialidad de los datos salariales. El almacenamiento físico o informal incrementa el riesgo de acceso no autorizado y pérdida de información. Este resultado resalta la importancia de incorporar mecanismos de seguridad informática, autenticación y control de accesos. La implementación de un sistema web permitiría aplicar políticas de seguridad alineadas con buenas prácticas tecnológicas. Por tanto, la protección de datos se convierte en un argumento clave para la modernización del proceso.

Figura 6.*Cuestionario – Gráfico Pregunta 6*

6. ¿Considera que el proceso actual garantiza la confidencialidad de la información salarial?

Número de colaboradores



Fuente: Respuestas del cuestionario creado en Google Forms.

Pregunta 7. ¿Cree que la implementación de un sistema web mejoraría la organización de la información de planilla?

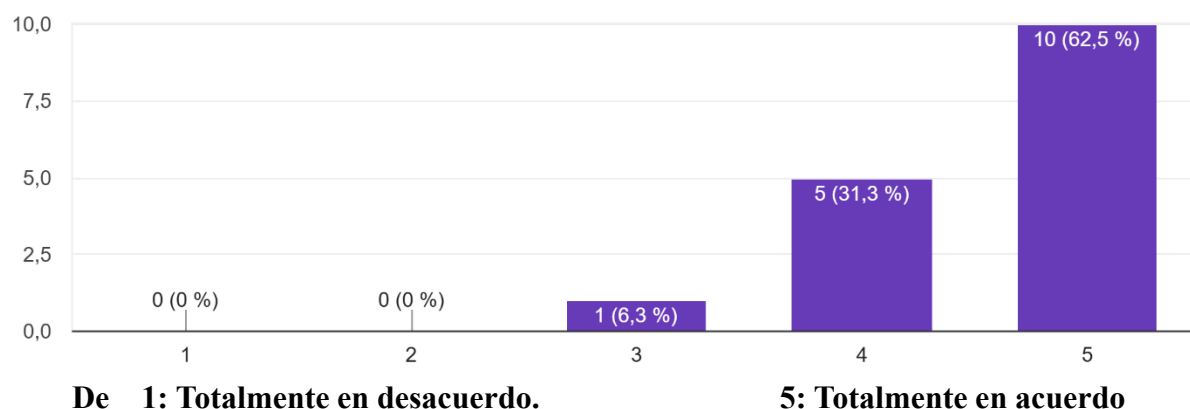
Las respuestas presentan un alto nivel de acuerdo, evidenciando una aceptación generalizada hacia la digitalización del proceso. Los colaboradores reconocen que un sistema web facilitaría la organización estructurada de registros laborales y salariales. Este resultado demuestra apertura al cambio tecnológico dentro de la empresa. Además, confirma que la problemática actual es percibida colectivamente. De ahí que la implementación del sistema propuesto se alinea con las expectativas de los usuarios finales.

Figura 7.

Cuestionario – Gráfico Pregunta 7.

7. ¿Cree que la implementación de un sistema web mejoraría la organización de la información de planilla?

Número de colaboradores



Fuente: Respuestas del cuestionario creado en Google Forms.

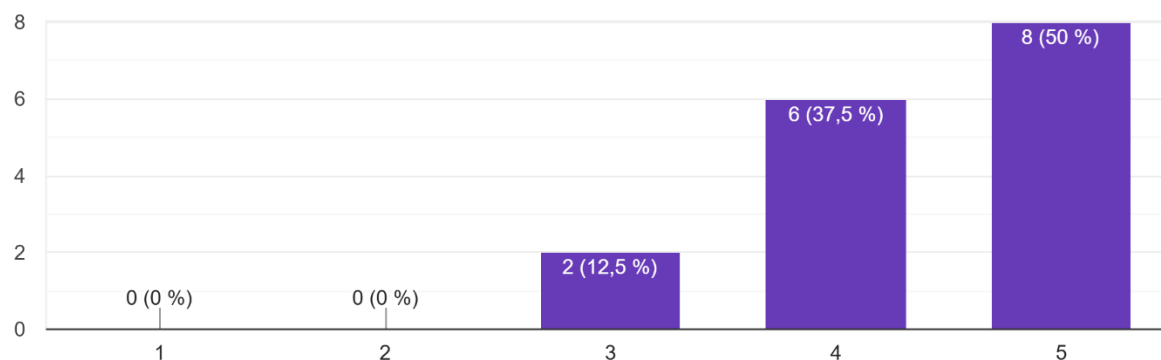
Pregunta 8. ¿Un sistema web permitiría reducir errores en el cálculo de pagos y deducciones?

Los participantes manifiestan un consenso positivo respecto a la capacidad del sistema para disminuir errores operativos. La automatización de cálculos es reconocida como un mecanismo confiable para mejorar la precisión de los pagos. Este resultado valida directamente el objetivo principal del proyecto tecnológico. La percepción favorable fortalece la justificación técnica del desarrollo del sistema web. Asimismo, evidencia que los colaboradores asocian la tecnología con mayor confiabilidad administrativa.

Figura 8.*Cuestionario – Gráfico Pregunta 8.*

8. ¿Un sistema web permitiría reducir errores en el cálculo de pagos y deducciones?

Número de colaboradores



De 1: Totalmente en desacuerdo.

5: Totalmente en acuerdo

Fuente: Respuestas del cuestionario creado en Google Forms.

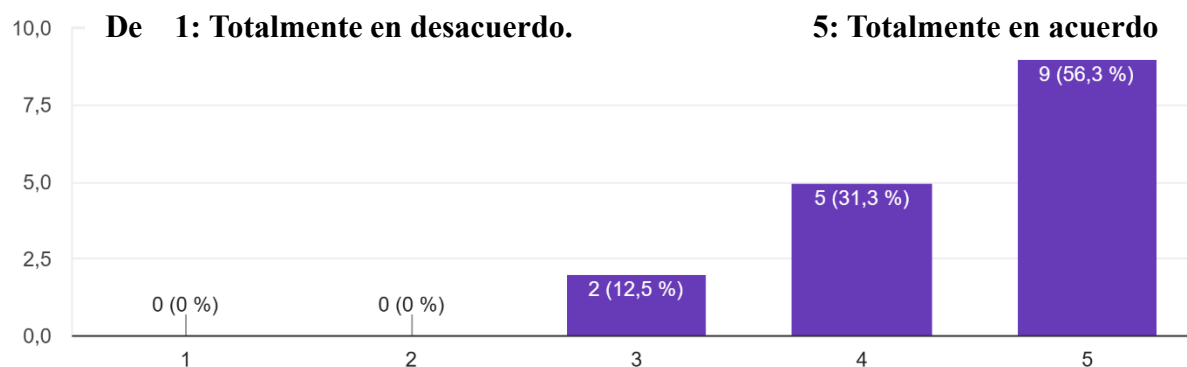
Pregunta 9. ¿Considera que un sistema automatizado agilizaría el proceso de pago de planilla?

La mayoría de respuestas se ubica en niveles altos de acuerdo, reflejando una expectativa clara de mejora en la eficiencia operativa. Los colaboradores perciben que la automatización reduciría tiempos de procesamiento y validación de información. Este resultado confirma la relación directa entre digitalización y optimización administrativa. La agilidad en los pagos representa un beneficio tanto organizacional como laboral. Por ello, la automatización se posiciona como un elemento estratégico dentro del proyecto.

Figura 9.*Cuestionario – Gráfico Pregunta 9.*

9. ¿Considera que un sistema automatizado agilizaría el proceso de pago de planilla?

Número de colaboradores



De 1: Totalmente en desacuerdo.

5: Totalmente en acuerdo

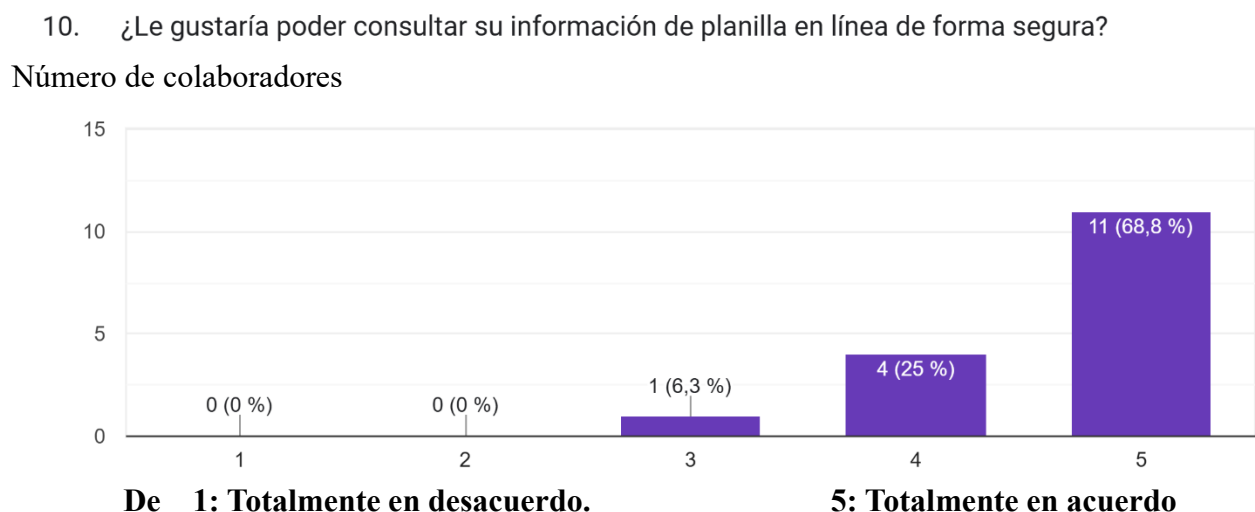
Fuente: Respuestas del cuestionario creado en Google Forms.

Pregunta 10. ¿Le gustaría poder consultar su información de planilla en línea de forma segura?

Las respuestas evidencian un alto interés por disponer de acceso digital seguro a la información personal de planilla. Este comportamiento refleja necesidades actuales asociadas a la transformación digital y autoservicio del empleado. El acceso en línea mejora la autonomía del colaborador y reduce la carga administrativa del área encargada. Además, fortalece la transparencia institucional. Este resultado respalda la incorporación de funcionalidades de consulta web dentro del sistema propuesto.

Figura 10.

Cuestionario – Gráfico Pregunta 10.



Fuente: Respuestas del cuestionario creado en Google Forms.

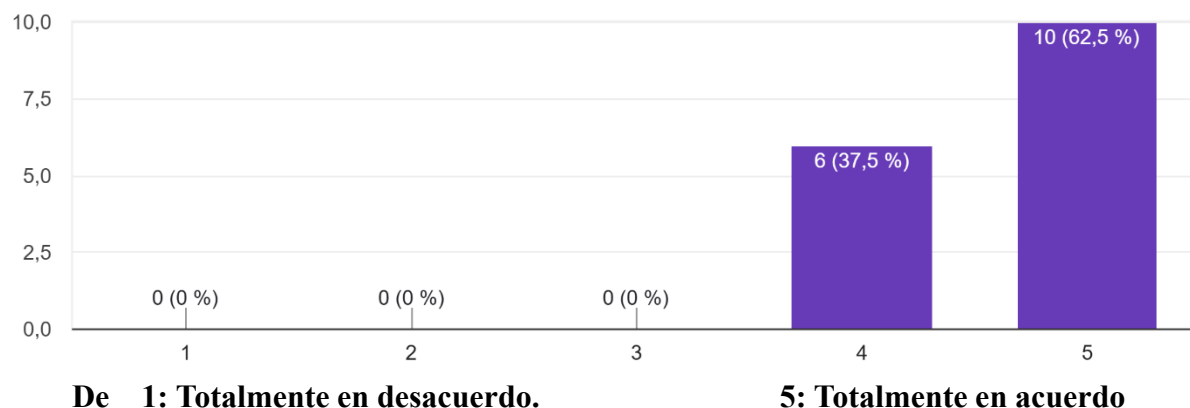
Pregunta 11. ¿Considera que la empresa necesita modernizar el proceso actual de gestión de planilla?

Los resultados muestran consenso casi total hacia la necesidad de modernización. Los colaboradores identifican claramente limitaciones del modelo manual actual. Esta percepción colectiva confirma el diagnóstico realizado durante las etapas previas del estudio. La modernización se reconoce como un paso necesario para mejorar la eficiencia organizacional. Por eso, el desarrollo del sistema web responde directamente a una necesidad institucional validada por los usuarios.

Figura 11.*Cuestionario – Gráfico Pregunta 11.*

11. ¿Considera que la empresa necesita modernizar el proceso actual de gestión de planilla?

Número de colaboradores



Fuente: Respuestas del cuestionario creado en Google Forms.

Pregunta 12. ¿La implementación de un sistema web contribuiría a una mayor transparencia en los pagos?

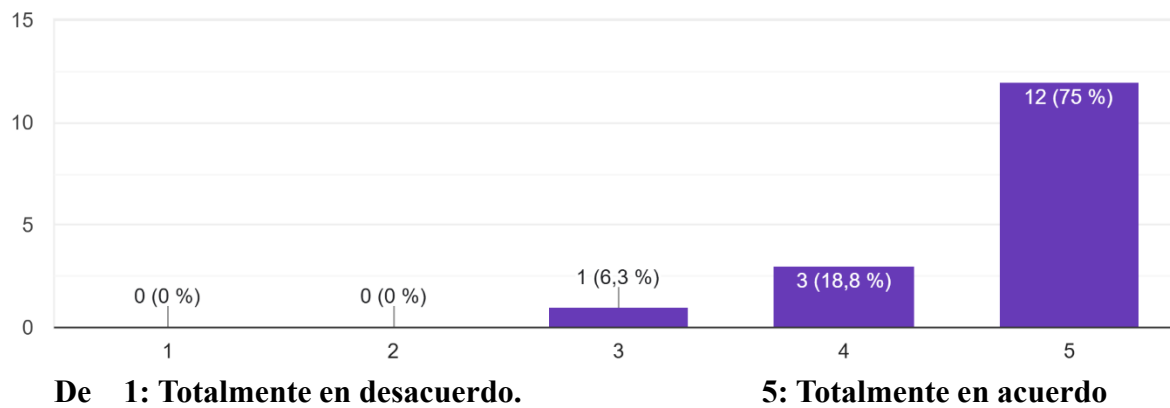
Las respuestas reflejan una valoración altamente positiva respecto al impacto del sistema en la transparencia organizacional. Los colaboradores asocian la digitalización con mayor claridad en cálculos, deducciones y registros históricos. Este resultado fortalece la dimensión ética y administrativa del proyecto. La transparencia contribuye a reducir conflictos laborales y aumentar la confianza interna. Por tanto, el sistema web se visualiza como un mecanismo de control y trazabilidad de la información.

Figura 12.

Cuestionario – Gráfico Pregunta 12.

12. ¿La implementación de un sistema web contribuiría a una mayor transparencia en los pagos?

Número de colaboradores



Fuente: Respuestas del cuestionario creado en Google Forms.

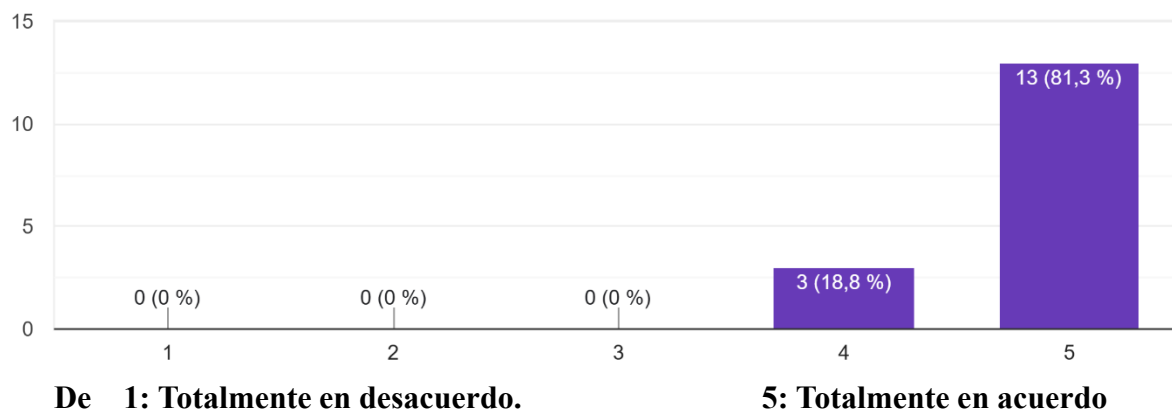
Pregunta 13. ¿Estaría dispuesto(a) a recibir capacitación para utilizar un nuevo sistema web de planilla?

Los participantes manifiestan una disposición favorable hacia la capacitación tecnológica, lo que representa un factor crítico de éxito para la implementación del sistema. La aceptación del aprendizaje reduce la resistencia al cambio organizacional. Este resultado evidencia una cultura abierta a la innovación dentro de la empresa. Además, facilita la adopción progresiva del sistema web. La capacitación se confirma como parte esencial del proceso de implementación.

Figura 13.*Cuestionario – Gráfico Pregunta 13.*

13. ¿Estaría dispuesto(a) a recibir capacitación para utilizar un nuevo sistema web de planilla?

Número de colaboradores



Fuente: Respuestas del cuestionario creado en Google Forms.

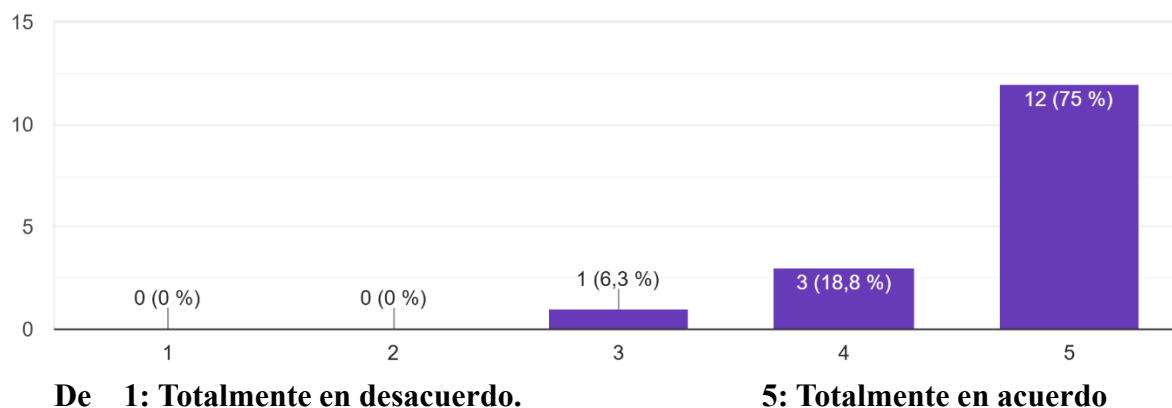
Pregunta 14. ¿Considera que la digitalización del proceso de planilla mejoraría la eficiencia administrativa de la empresa?

Las respuestas presentan niveles elevados de acuerdo, indicando que los colaboradores reconocen el impacto positivo de la digitalización en la gestión administrativa. La eficiencia operativa es percibida como uno de los principales beneficios esperados. Este hallazgo se alinea con los objetivos estratégicos del proyecto. La reducción de tareas manuales permitiría optimizar recursos y tiempos de trabajo. Por tanto, la digitalización se consolida como una solución pertinente para la organización.

Figura 14.*Cuestionario – Gráfico Pregunta 14.*

14. ¿Considera que la digitalización del proceso de planilla mejoraría la eficiencia administrativa de la empresa?

Número de colaboradores



Fuente: Respuestas del cuestionario creado en Google Forms.

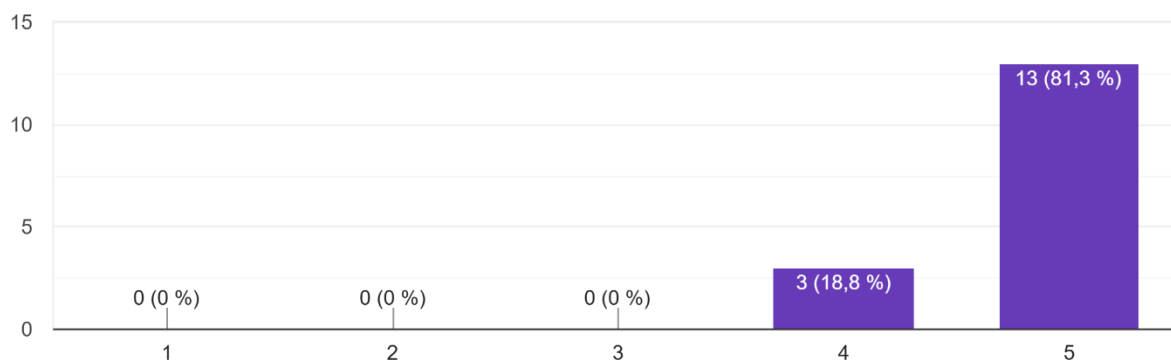
Pregunta 15. ¿Recomendaría la implementación de un sistema web para la gestión y pago de planilla en AraWeb CR?

Los resultados evidencian un respaldo generalizado hacia la implementación del sistema web propuesto. La recomendación directa por parte de los colaboradores confirma la aceptación del proyecto desde la perspectiva del usuario final. Este consenso valida la viabilidad funcional y organizacional del desarrollo tecnológico. Igualmente, demuestra que la solución responde a necesidades reales identificadas dentro de la empresa. En resumen, los resultados del cuestionario respaldan plenamente la pertinencia del sistema web para mejorar la gestión y pago de planilla en AraWeb CR.

Figura 15.*Cuestionario – Gráfico Pregunta 15.*

15. ¿Recomendaría la implementación de un sistema web para la gestión y pago de planilla en AraWeb CR?

Número de colaboradores



De 1: Totalmente en desacuerdo.

5: Totalmente en acuerdo

Fuente: Respuestas del cuestionario creado en Google Forms.

Observación Directa

Análisis de la Observación Directa

La observación directa realizada en AraWeb CR permitió identificar de manera objetiva las condiciones reales bajo las cuales se ejecutan los procesos administrativos relacionados con la gestión y pago de planilla. Los resultados evidencian una alta dependencia de procedimientos manuales para el registro, cálculo y control de información laboral, situación que coincide con las percepciones obtenidas mediante el cuestionario aplicado a los colaboradores.

Se determinó que la mayoría de los procesos críticos, tales como cálculo salarial, control de asistencia, manejo de vacaciones, incapacidades y generación de reportes, no se encuentran soportados por herramientas tecnológicas integradas. Esta condición incrementa la probabilidad de errores humanos, retrabajos administrativos y atrasos en la ejecución de tareas operativas. Del mismo modo, la ausencia de mecanismos de autenticación y control de accesos limita la confidencialidad y seguridad de la información salarial.

La observación también permitió evidenciar oportunidades claras de mejora relacionadas con la estandarización de procesos, centralización de datos y automatización de cálculos conforme a la normativa laboral vigente. Aunque algunas prácticas administrativas se realizan de manera

funcional, estas dependen principalmente de la experiencia del personal y no de un sistema estructurado que garantice continuidad operativa.

Como resultado, los hallazgos obtenidos confirman la necesidad de implementar un Sistema web para la gestión y pago de planilla en AraWeb CR, ya que la digitalización permitiría optimizar tiempos, reducir errores, fortalecer la seguridad de la información y mejorar la eficiencia administrativa general de la organización.

El análisis integral de los resultados obtenidos mediante la aplicación del cuestionario y la guía de observación directa permitió evaluar de manera objetiva la situación actual del proceso de gestión y pago de planilla en la empresa AraWeb CR, ubicada en San José. Los datos recopilados evidencian que los procedimientos administrativos se desarrollan principalmente de forma manual, lo cual genera retrasos operativos, posibilidad de errores en los cálculos salariales y limitaciones en el acceso oportuno a la información.

De igual manera, los colaboradores manifestaron una alta aceptación hacia la implementación de herramientas tecnológicas que permitan modernizar dichos procesos y mejorar la organización administrativa. La observación directa confirmó estas percepciones al identificar ausencia de automatización, controles de acceso y generación estructurada de reportes. De ahí que la triangulación de resultados valida el cumplimiento de los objetivos planteados en la investigación y demuestra la viabilidad y pertinencia del desarrollo de un Sistema web para la gestión y pago de planilla en AraWeb CR como solución tecnológica orientada a optimizar la eficiencia, seguridad y transparencia institucional.

CAPÍTULO V: PROPUESTA

El presente capítulo tiene como propósito presentar la propuesta tecnológica desarrollada a partir del diagnóstico organizacional y del análisis de resultados obtenidos en el capítulo anterior. Con base en la información recopilada mediante el cuestionario aplicado y la observación directa realizada en la empresa AraWeb CR, se identificaron debilidades operativas asociadas al manejo manual de la planilla y a la gestión administrativa del recurso humano. En respuesta a dichas necesidades, se plantea el diseño de un Sistema Web para la gestión y pago de planilla, orientado a optimizar los procesos internos, reducir errores operativos y fortalecer el control de la información institucional.

El objetivo principal de este capítulo consiste en describir de forma técnica y estructurada la solución propuesta, evidenciando su viabilidad funcional dentro del entorno organizacional. Para ello, la propuesta se desarrolla mediante distintos análisis que abarcan el software por implementar, los requerimientos de hardware, las telecomunicaciones necesarias, las herramientas tecnológicas utilizadas y las competencias requeridas del recurso humano encargado de operar el sistema.

Análisis

Análisis detallado del Software por desarrollar

Módulo Gestionar Nóminas

El módulo Gestionar Nóminas constituye el núcleo operativo del sistema propuesto, ya que permite automatizar el procesamiento del cálculo salarial de los colaboradores conforme a la normativa laboral vigente en Costa Rica. Este componente integra información proveniente de asistencia, horas extras, incapacidades y deducciones legales, lo que garantiza exactitud en los pagos y reducción de errores humanos. Además, facilita la generación automática de registros históricos de planilla, fortaleciendo la trazabilidad financiera y administrativa de la organización.

Módulo Gestionar Asistencia

El módulo Gestionar Asistencia tiene como finalidad registrar digitalmente la jornada laboral de los colaboradores mediante un sistema centralizado de control de tiempos laborados. Su implementación permite eliminar registros manuales susceptibles a inconsistencias, generando una

base de datos confiable para el cálculo de salarios y control organizacional. También, contribuye a mejorar la supervisión administrativa y la transparencia en la gestión del recurso humano.

Módulo Gestionar Horas Extras

El módulo Gestionar Horas Extras permite administrar de forma estructurada el registro y aprobación de jornadas extraordinarias. El colaborador realiza la solicitud directamente desde el sistema, la cual será evaluada por la jefatura correspondiente antes de su validación final por Recursos Humanos. Este flujo automatizado garantiza control documental, evita pagos incorrectos y mantiene comunicación directa entre las partes involucradas mediante notificaciones automáticas del sistema.

Módulo Gestionar Permisos

El módulo Gestionar Permisos organiza digitalmente el proceso de solicitud y aprobación de permisos laborales, eliminando prácticas informales que generan desorden administrativo. A través del sistema, el colaborador registrará su solicitud, la cual será enviada automáticamente a la jefatura para su análisis. El sistema notificará la resolución adoptada, asegurando registro formal, trazabilidad del proceso y transparencia en la toma de decisiones administrativas.

Módulo Gestionar Incapacidades

El módulo Gestionar Incapacidades permite registrar y administrar incapacidades médicas mediante el almacenamiento estructurado de fechas, motivos y documentación asociada. Esta funcionalidad evita pérdida de información y facilita el control de ausencias laborales que impactan directamente el cálculo salarial. Además, contribuye a mantener respaldo documental necesario para efectos administrativos y legales.

Módulo Calcular Vacaciones

El módulo Calcular Vacaciones automatiza la administración del disfrute y acumulación de días vacacionales de los colaboradores. El sistema permite registrar solicitudes, gestionar aprobaciones por parte de la jefatura y actualizar automáticamente los saldos disponibles. Con ello se elimina el control manual en papel, reduciendo conflictos administrativos y garantizando precisión en el manejo del beneficio laboral.

Módulo Calcular Aguinaldo

El módulo Calcular Aguinaldo realiza el cálculo automático del aguinaldo considerando el promedio salarial anual de cada colaborador, conforme a la legislación costarricense. Esta automatización asegura exactitud en el cálculo del beneficio, disminuye riesgos de reclamos laborales y reduce significativamente el tiempo requerido para su procesamiento administrativo.

Módulo Gestionar Liquidación

El módulo Gestionar Liquidación permite calcular de manera precisa las liquidaciones laborales cuando finaliza la relación contractual de un colaborador. El sistema considera antigüedad, prestaciones acumuladas, vacaciones pendientes y demás obligaciones legales aplicables. Su implementación minimiza errores manuales y contribuye a evitar conflictos legales derivados de cálculos incorrectos.

Módulo Evaluar Personal

El módulo Evaluar Personal incorpora un mecanismo estructurado para la realización de evaluaciones del desempeño laboral. Mediante indicadores previamente definidos, la empresa puede generar métricas objetivas que respalden decisiones relacionadas con incentivos, capacitaciones o mejoras organizacionales. Este componente fortalece la gestión estratégica del talento humano dentro de la organización.

Módulo Mantenimientos

El módulo Mantenimientos permite la administración general de los datos del sistema mediante operaciones de inserción, modificación, actualización y eliminación de registros. Este módulo funciona como soporte transversal para el resto de funcionalidades, con lo que se garantiza integridad, consistencia y actualización permanente de la información almacenada.

Módulo Consultas

El módulo Consultas facilita el acceso rápido a la información almacenada en las diferentes tablas del sistema, permitiendo realizar búsquedas y visualización de datos en tiempo real. Su propósito es apoyar la toma de decisiones administrativas mediante acceso inmediato a información confiable y organizada.

Módulo Reportes

El módulo Reportes permite generar documentos formales derivados de los procesos de planilla, asistencia y gestión administrativa. Los reportes pueden visualizarse en pantalla o imprimirse según las necesidades del usuario, presentando la información en formatos estructurados que facilitan auditorías internas y control organizacional.

Módulo Seguridad

El módulo Seguridad garantiza la protección de la información mediante mecanismos de autenticación de usuarios, control de contraseñas y asignación de perfiles de acceso. Este componente establece niveles de permisos según roles organizacionales, asegurando confidencialidad, integridad y disponibilidad de los datos sensibles gestionados por el sistema.

Análisis detallado del hardware requerido

El análisis del hardware requerido para el desarrollo del sistema web de gestión y pago de planilla considera, en primera instancia, los recursos necesarios durante la fase de programación del prototipo. Para esta etapa se establece el uso de estaciones de trabajo destinadas al desarrollo de software, las cuales deben contar con un procesador mínimo Intel Core i5 o equivalente, 16 GB de memoria RAM y unidad de almacenamiento sólido (SSD) de al menos 512 GB. Estas características permiten ejecutar simultáneamente entornos de desarrollo integrados, servidores locales, herramientas de control de versiones y sistemas gestores de bases de datos sin afectar el desempeño general del equipo. Por su parte, el uso de hardware de gama media garantiza estabilidad durante las pruebas funcionales, la depuración del sistema y la validación del prototipo previo a su implementación definitiva.

En relación con el ambiente de producción, el sistema propuesto está diseñado bajo una arquitectura web que permite su ejecución mediante un servidor central accesible a través de internet o red local institucional. Se establece la utilización de una infraestructura compuesta por dos servidores independientes: uno destinado exclusivamente al motor de base de datos y otro encargado de alojar la capa aplicativa y la capa de presentación desarrolladas en ASP.NET MVC sobre .NET Framework. Esta separación responde a buenas prácticas de arquitectura empresarial orientadas a fortalecer la seguridad, optimizar el rendimiento y mejorar la administración del sistema, evitando la exposición directa del servidor de datos.

El servidor de base de datos tendrá como función principal el almacenamiento, procesamiento y resguardo de la información crítica del sistema de planilla, restringiendo su acceso únicamente a conexiones internas controladas desde el servidor de aplicaciones. Por su parte, el servidor aplicativo gestionará la lógica de negocio, los procesos de autenticación y la publicación del sistema web hacia los clientes finales mediante protocolos seguros. Esta distribución contribuye a reducir riesgos asociados a accesos no autorizados y permite implementar políticas diferenciadas de respaldo, monitoreo y control del tráfico de información.

Desde el punto de vista operativo, se recomienda que ambos servidores cuenten con procesadores multinúcleo, al menos 16 GB de memoria RAM y almacenamiento SSD superior a 1 TB, garantizando tiempos de respuesta adecuados, alta disponibilidad y capacidad de escalabilidad futura. Adicionalmente, esta arquitectura facilita la ejecución de mantenimientos independientes, actualizaciones controladas y una mejor gestión del crecimiento del sistema, alineándose con estándares actuales de implementación en soluciones empresariales basadas en tecnologías web.

La implementación puede realizarse mediante infraestructura física dentro de la organización o a través de servicios de computación en la nube, alternativa que permite reducir costos iniciales de adquisición y mantenimiento. Este enfoque favorece la flexibilidad operativa y facilita la expansión del sistema conforme aumente la cantidad de colaboradores o procesos administrativos gestionados por la empresa.

Por otra parte, los equipos utilizados por el personal administrativo y los colaboradores no requieren especificaciones avanzadas, debido a que el acceso al sistema se realizará mediante un navegador web. Se consideran suficientes computadoras de oficina con procesador Intel Core i3 o superior, 8 GB de memoria RAM y conexión estable a internet, lo cual permite una adopción tecnológica sin necesidad de reemplazar la infraestructura existente. Esta característica representa una ventaja económica significativa para AraWeb CR, ya que disminuye la inversión en hardware adicional y simplifica el proceso de implementación dentro del entorno organizacional. En conjunto, la propuesta tecnológica demuestra ser viable desde el punto de vista técnico y financiero, alineándose con las necesidades reales identificadas durante el diagnóstico institucional.

Análisis detallado de los elementos relacionados con las telecomunicaciones

El funcionamiento del sistema web para la gestión y pago de planilla requiere una infraestructura de telecomunicaciones básica, acorde con las capacidades tecnológicas actuales de la empresa AraWeb CR y con el enfoque de implementación local definido en el estudio de

viabilidad técnica. El sistema se ejecuta dentro de una red interna organizacional, permitiendo que los usuarios autorizados accedan mediante computadoras conectadas a la misma red local utilizando un navegador web. Este esquema reduce la complejidad técnica y elimina la necesidad de configuraciones avanzadas de comunicación externa durante la etapa inicial del proyecto.

El tipo de red requerido corresponde a una red local cableada o inalámbrica existente dentro de la organización, utilizada actualmente para las operaciones administrativas y acceso general a internet. El sistema aprovecha esta infraestructura ya disponible sin requerir modificaciones estructurales ni la adquisición de equipos especializados adicionales. El acceso al sistema se realiza mediante protocolos web estándar, lo que permite la comunicación entre los equipos cliente y el servidor local donde se ejecuta la aplicación, lo que garantiza un funcionamiento estable dentro del entorno interno de la empresa.

En cuanto al uso de internet, su participación no es indispensable para la operación básica del prototipo, ya que el sistema está diseñado para funcionar en un entorno local controlado. No obstante, la conectividad a internet puede emplearse de forma complementaria para actualizaciones de software, respaldo de información o futuras etapas de despliegue en servidores externos. Este enfoque mantiene coherencia con la decisión organizacional de iniciar la implementación sin requerir servicios de hosting ni disponibilidad permanente en línea.

Respecto a la configuración técnica, el sistema utiliza puertos estándar asociados a aplicaciones web, evitando configuraciones complejas de red o aperturas especiales que puedan comprometer la seguridad informática institucional. En esa misma línea, no se requieren dispositivos de telecomunicaciones especializados, ya que routers y puntos de acceso existentes resultan suficientes para garantizar la comunicación entre los usuarios y el sistema. Esta característica refuerza la viabilidad técnica del proyecto al minimizar dependencias tecnológicas adicionales.

Desde una perspectiva económica y operativa, la infraestructura de telecomunicaciones necesaria no implica costos adicionales significativos, debido a que el sistema aprovecha los recursos de red ya disponibles en la empresa. Esta condición mantiene coherencia con el análisis de viabilidad económica previamente desarrollado, donde se establece que la implementación inicial del sistema busca optimizar los procesos administrativos sin generar inversiones tecnológicas relevantes. Por eso, la solución propuesta se integra de manera natural al entorno tecnológico existente, facilitando su adopción y sostenibilidad operativa.

Descripción detallada de las herramientas técnicas utilizadas para el desarrollo

El desarrollo del sistema web para la gestión y pago de planilla se fundamenta en el uso de herramientas tecnológicas ampliamente adoptadas en el ámbito profesional del desarrollo de software empresarial, garantizando estabilidad, compatibilidad y facilidad de mantenimiento. Para la construcción del prototipo se emplea el entorno de desarrollo integrado Visual Studio Community, el cual permite la programación, depuración y validación del sistema dentro de un mismo entorno de trabajo. Esta herramienta facilita la aplicación de buenas prácticas de ingeniería de software, promoviendo una estructura organizada del código y una adecuada gestión del ciclo de desarrollo del sistema.

La arquitectura del sistema se implementa mediante el framework ASP.NET MVC bajo la plataforma .NET, utilizando el lenguaje de programación C# para la construcción de la lógica de negocio y la interacción con los distintos módulos funcionales. Este enfoque permite separar adecuadamente las capas de presentación, control y acceso a datos, favoreciendo la escalabilidad y el mantenimiento futuro de la aplicación. Asimismo, el uso de tecnologías web estándar posibilita que el sistema sea accedido desde navegadores comunes sin requerir instalaciones adicionales en los equipos de los usuarios, alineándose con el objetivo de implementar una solución tecnológica accesible y adaptable al entorno operativo de la empresa.

En relación con la persistencia de la información, el sistema utiliza SQL Server como motor de base de datos relacional, encargado de almacenar, organizar y administrar la información asociada a colaboradores, asistencia, pagos y procesos administrativos. Este gestor de base de datos ofrece mecanismos integrados de seguridad, control de integridad y administración de transacciones, elementos necesarios para garantizar la confiabilidad de los datos procesados por el sistema. La selección de esta tecnología responde a su integración directa con el entorno de desarrollo utilizado y a su reconocimiento como estándar en soluciones empresariales orientadas a la gestión administrativa.

Desde el punto de vista del licenciamiento y costos tecnológicos, el proyecto prioriza el uso de herramientas con modelos de licencia gratuitos o de suscripción accesible, permitiendo reducir la inversión inicial requerida por la organización. Visual Studio Community se utiliza bajo licencia gratuita para desarrollo académico y profesional, mientras que el motor de base de datos y los servicios de hospedaje pueden operar bajo esquemas de suscripción escalables según las necesidades futuras de la empresa. Este enfoque asegura coherencia con el análisis de viabilidad

económica previamente establecido, permitiendo que la implementación del sistema evolucione progresivamente sin generar dependencias tecnológicas complejas ni costos elevados en etapas iniciales.

Descripción detallada del conocimiento básico que debe tener el recurso humano que operará el sistema.

El recurso humano encargado de operar el sistema web para la gestión y pago de planilla debe poseer conocimientos básicos en el uso de herramientas informáticas orientadas a entornos administrativos. Se requiere familiaridad con sistemas operativos de uso común, navegación en aplicaciones web y manejo general de plataformas digitales mediante interfaces gráficas. Estas competencias permiten interactuar con los módulos del sistema sin necesidad de conocimientos avanzados en programación o administración técnica. El enfoque del sistema prioriza la usabilidad, facilitando que personal administrativo pueda ejecutar las tareas operativas diarias de forma eficiente y con una curva de aprendizaje reducida.

Adicionalmente, el usuario operador debe comprender conceptos fundamentales relacionados con la gestión de recursos humanos y procesos de planilla, tales como registro de asistencia, control de horas extraordinarias, administración de permisos laborales, cálculo de deducciones y generación de reportes administrativos. Este conocimiento funcional resulta necesario para interpretar correctamente la información registrada en el sistema y validar los resultados generados automáticamente. La correcta operación depende más del entendimiento de los procesos organizacionales que del dominio técnico especializado, lo cual fortalece la viabilidad operativa del proyecto dentro del contexto empresarial analizado.

Desde la perspectiva de seguridad y manejo de la información, el personal debe poseer criterios básicos de buenas prácticas digitales, incluyendo el uso responsable de credenciales de acceso, protección de datos sensibles y verificación de información antes de su registro definitivo. Igualmente, se espera capacidad para seguir procedimientos establecidos y aplicar controles internos definidos por la organización. Estas competencias contribuyen a garantizar la integridad, confidencialidad y disponibilidad de la información gestionada por el sistema, asegurando una operación alineada con los principios de control administrativo y continuidad operativa definidos en el proyecto.

Adicionalmente, el correcto funcionamiento y sostenibilidad del sistema requiere la participación de recurso humano técnico especializado encargado del mantenimiento evolutivo, correctivo y preventivo de la aplicación y de la base de datos. Este personal no forma parte de la operación diaria administrativa, sino que cumple un rol estratégico orientado a garantizar la estabilidad tecnológica, seguridad informática y continuidad del servicio.

Para el mantenimiento del aplicativo se recomienda contar con un desarrollador de software, preferiblemente con formación en Ingeniería Informática o áreas afines, con conocimientos en desarrollo web bajo el framework ASP.NET MVC utilizando .NET Framework. Dicho profesional debe poseer experiencia en programación orientada a objetos, arquitectura en capas, consumo de servicios web, control de versiones mediante herramientas como Git, así como habilidades para análisis de requerimientos, corrección de incidencias, optimización de código y desarrollo de nuevas funcionalidades conforme evolucione el sistema. Dependiendo del tamaño organizacional, estas funciones pueden ser asumidas por un desarrollador senior o junior bajo supervisión técnica.

En relación con la administración de datos, se recomienda la participación de un Administrador de Bases de Datos (DBA) o profesional con conocimientos equivalentes en gestión de bases de datos relacionales. Este recurso debe dominar el manejo de SQL Server, incluyendo diseño lógico y físico de bases de datos, optimización de consultas SQL, creación de procedimientos almacenados, gestión de respaldos y restauraciones, monitoreo del rendimiento y aplicación de políticas de seguridad sobre la información. Del mismo modo, se consideran necesarios conocimientos en versiones actuales del motor SQL Server, administración de usuarios, control de accesos y estrategias de recuperación ante fallos.

De manera complementaria, el personal técnico responsable del mantenimiento deberá poseer conocimientos básicos en administración de servidores web, configuración de entornos IIS, despliegue de aplicaciones, gestión de ambientes de desarrollo, pruebas y producción, así como nociones de seguridad informática orientadas a aplicaciones web, tales como control de vulnerabilidades, manejo de certificados digitales y actualización periódica de componentes tecnológicos.

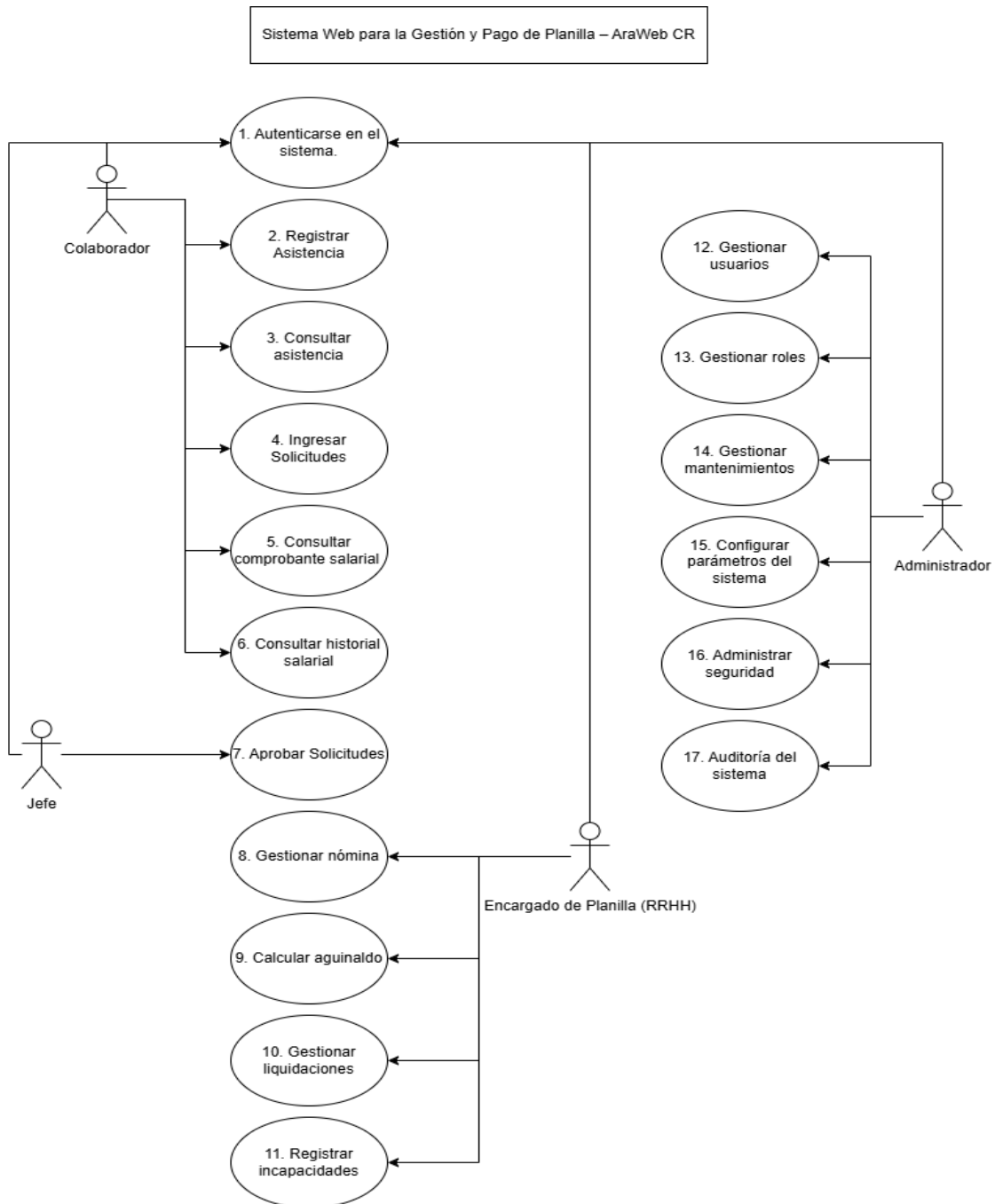
La incorporación de estos perfiles técnicos responde a una recomendación profesional orientada a asegurar la sostenibilidad del sistema a largo plazo, permitiendo realizar mejoras continuas, atender incidentes operativos y mantener la integridad de la información institucional.

De esta forma, el proyecto no solo contempla la operación funcional del sistema, sino también su mantenimiento tecnológico y evolución futura dentro del entorno organizacional de AraWeb CR.

Diseño de casos de uso.

Figura 16.

Diagrama de Casos de uso.



Fuente: Elaboración propia.

El diagrama de casos de uso del sistema web para la gestión y pago de planilla presenta la representación funcional de las principales interacciones entre los usuarios y la plataforma desarrollada. Este modelo permite identificar de manera clara los procesos que el sistema ofrece y cómo cada actor participa dentro de las operaciones administrativas definidas en el proyecto. Su utilización facilita la comprensión del alcance funcional del sistema, estableciendo los límites de interacción entre los usuarios y las funcionalidades disponibles antes de profundizar en aspectos técnicos de implementación.

Dentro del diagrama se contemplan los actores organizacionales involucrados en la operación del sistema, tales como colaborador, jefe inmediato, encargado de planilla y administrador, cada uno asociado a un conjunto específico de acciones según sus responsabilidades laborales. A través de los casos de uso se describen procesos como el ingreso a la aplicación mediante autenticación por roles, el registro de asistencia, la aprobación de solicitudes, la ejecución automatizada de la planilla y la administración de parámetros del sistema. Esta estructuración evidencia la correcta separación de funciones y el control de acceso basado en perfiles de usuario.

El modelo contribuye además a validar que los requerimientos funcionales identificados durante el análisis se encuentren correctamente representados dentro del diseño del sistema. El diagrama sirve como apoyo para la comunicación entre el contexto organizacional y el desarrollo tecnológico, permitiendo visualizar de forma integrada los procesos que serán soportados por la solución propuesta. Por ello, se fortalece la coherencia entre las necesidades institucionales detectadas y las funcionalidades implementadas en la aplicación web.

Tabla 5*Caso de Uso CU-01 Ingreso al Sistema.*

Plantilla para la elaboración de Caso de Uso	
Número Caso de Uso: CU-01	Nombre del Caso de Uso: Ingreso al Sistema
Fecha elaboración:	04/04/2026
Descripción Caso de Uso:	Este caso de uso describe el proceso mediante el cual los usuarios autorizados acceden al sistema web mediante autenticación. El sistema valida las credenciales. Al autenticarse correctamente, se registra la sesión en SesionUsuario y el evento LOGIN en Bitácora. Si es acceso por primera vez, el sistema redirige obligatoriamente al cambio de contraseña.
Autor caso de uso:	Emmanuel Cerdas Brenes
Actores relacionados:	Todos los roles: Administrador, RRHH, Jefe, Colaborador
Precondiciones:	1. El usuario debe existir en la tabla Credencial con cuenta activa. 2. El usuario debe estar asociado a un registro de Empleado.
Flujo Básico del caso de uso	
Detallar el paso a paso del Flujo Básico	
1.1.1 El actor accede a la página principal del sistema web. 1.1.2 El sistema muestra el formulario de autenticación. 1.1.3 El actor ingresa usuario y contraseña. 1.1.4 El actor selecciona la opción Ingresar. 1.1.5 El sistema valida el formato de las credenciales ingresadas. 1.1.6 El sistema verifica las credenciales contra la base de datos (FA-1). 1.1.7 El sistema valida que el usuario se encuentre activo. 1.1.8 El sistema identifica el rol asociado al usuario. 1.1.9 El sistema registra el evento de acceso en la Bitácora. 1.1.10 El sistema redirige al menú principal según permisos asignados.	
Sub Flujos	
Nombre	Detalle
N/A	N/A
Flujos Alternos	
FA-1 Credenciales inválidas	3.1.1 El sistema muestra mensaje de autenticación incorrecta. 3.1.2 El sistema registra intento fallido. 3.1.3 El actor puede intentar nuevamente.
Requerimientos especiales	
La contraseña debe viajar cifrada. El sistema debe registrar auditoría del acceso.	
Post-Condiciones	
Usuario autenticado correctamente dentro del sistema.	

El caso de uso Ingreso al Sistema describe el proceso mediante el cual los usuarios autorizados acceden al sistema web utilizando credenciales previamente registradas. Este proceso constituye el punto inicial de interacción entre el usuario y la plataforma, con lo que se asegura que únicamente actores válidos puedan utilizar las funcionalidades disponibles. Durante la autenticación, el sistema valida la información ingresada, identifica el rol asignado y habilita el acceso conforme a los permisos establecidos. Además, se registra el evento de acceso dentro del módulo de auditoría para mantener trazabilidad operativa. Este mecanismo asegura control de seguridad, protección de la información y correcta gestión de accesos dentro del entorno del sistema.

Tabla 6

Caso de Uso CU-02 Administración de Usuarios.

Plantilla para la elaboración de Caso de Uso	
Número Caso de Uso: CU-02	Nombre del Caso de Uso: Administración de Usuarios
Fecha elaboración:	04/04/2026
Descripción Caso de Uso:	Permite al administrador registrar, consultar, modificar y desactivar usuarios del sistema.
Autor caso de uso:	Emmanuel Cerdas Brenes
Actores relacionados:	Administrador.
Precondiciones:	Usuario autenticado. Usuario con rol administrador.
Flujo Básico del caso de uso	
Detallar el paso a paso del Flujo Básico	
1.1.1 Administrador accede al módulo Mantenimiento, Usuarios. 1.1.2 El sistema valida permisos administrativos (FA-1). 1.1.3 El sistema carga listado de usuarios registrados. 1.1.4 El administrador visualiza información general del usuario. 1.1.5 El administrador selecciona acción a ejecutar. 1.1.6 Registrar Usuario ir a SF1. 1.1.7 Modificar Usuario ir a SF2. 1.1.8 Desactivar Usuario ir a SF-3. 1.1.9 El sistema ejecuta el subflujo correspondiente.	
Sub Flujos	
SF-1 Registrar Usuario	2.1.1 Actor ingresa datos del usuario. 2.1.2 Selecciona rol. 2.1.3 Presiona Guardar. 2.1.4 Sistema valida información (FA-2). 2.1.5 Sistema registra usuario.
SF-2 Modificar Usuario	2.1.1 Actor selecciona usuario. 2.1.2 Modifica información. 2.1.3 Sistema valida datos. 2.1.4 Sistema actualiza registro.
SF-3 Desactivar Usuario	2.1.1 Actor selecciona usuario. 2.1.2 Sistema solicita confirmación (FA-3). 2.1.3 Sistema cambia estado a inactivo.
Flujos Alternos	
FA-1 Validar permisos	El sistema impide acceso si el usuario no posee rol administrador.
FA-2 Validar Información	El sistema solicita completar campos obligatorios.

FA-3 Confirmar eliminación	El sistema solicita confirmación antes de desactivar el usuario.
Requerimientos especiales	
Control de auditoría obligatorio. Contraseñas almacenadas cifradas.	
Post-Condiciones	
Usuario registrado, actualizado o desactivado.	

El caso de uso Administración de Usuarios permite gestionar el ciclo de vida de las cuentas del sistema mediante operaciones de registro, modificación y desactivación de usuarios. Este proceso es ejecutado exclusivamente por el administrador, quien posee privilegios para controlar el acceso al sistema según los roles definidos. El sistema valida permisos antes de permitir cualquier operación administrativa, garantizando la integridad del control de seguridad. De igual manera, se aplican validaciones sobre la información ingresada y se mantiene registro histórico de cambios realizados. Este caso de uso resulta fundamental para asegurar la correcta asignación de responsabilidades y el mantenimiento del esquema de control de acceso institucional.

Tabla 7

Caso de Uso CU-03 Administración de Empleados.

Plantilla para la elaboración de Caso de Uso	
Número Caso de Uso: CU-03	Nombre del Caso de Uso: Administración de Empleados
Fecha elaboración:	04/04/2026
Descripción Caso de Uso:	Permite gestionar la información laboral de los colaboradores de la empresa.
Autor caso de uso:	Emmanuel Cerdas Brenes
Actores relacionados:	Administrador, Encargado de Planilla.
Precondiciones:	Usuario autenticado. Permisos administrativos.
Flujo Básico del caso de uso	
Detallar el paso a paso del Flujo Básico	
1.1.1 Actor ingresa al módulo Empleados. 1.1.2 Sistema valida permisos (FA-1). 1.1.3 Sistema muestra opciones de mantenimiento. 1.1.4 Actor selecciona acción a ejecutar. 1.1.5 Registrar Empleado ir a SF-1. 1.1.6 Modificar Empleado ir a SF-2. 1.1.7 Inactivar Empleado ir a SF-3. 1.1.8 El sistema ejecuta el subflujo correspondiente.	
Sub Flujos	
SF-1 Registrar Empleado	2.1.1 Actor selecciona Nuevo Empleado. 2.1.2 Ingresar datos personales. 2.1.3 Ingresar información contractual y salarial. 2.1.4 Asigna jefe inmediato. 2.1.5 Sistema valida cédula única (FA-2). 2.1.6 Sistema guarda registro.
SF-2 Modificar Empleado	2.2.1 Actor selecciona empleado. 2.2.2 Sistema muestra información actual. 2.2.3 Actor modifica datos. 2.2.4 Sistema valida consistencia.

	2.2.5 Sistema actualiza información.
SF-3 Inactivar Empleado	2.3.1 Actor selecciona inactivar. 2.3.2 Sistema solicita confirmación. 2.3.3 Sistema cambia estado laboral. 2.3.4 Sistema mantiene historial.
Flujos Alternos	
FA-1 Validación de permisos.	El sistema valida los permisos correspondientes.
FA-2 Datos incompletos.	El sistema informa que faltan datos por agregar.
FA-3 Datos inválidos.	El sistema notifica los datos inválidos.
Requerimientos especiales	
Validación de cédula única. Historial de cambios almacenado.	
Post-Condiciones	
Información del empleado actualizada en el sistema.	

El caso de uso Administración de Empleados gestiona la información laboral necesaria para el funcionamiento del sistema de planilla. A través de este proceso se registran datos personales, contractuales y salariales de los colaboradores, los cuales sirven como base para cálculos posteriores relacionados con pagos y beneficios. El sistema valida la consistencia de los datos ingresados y mantiene un historial que permite conservar la trazabilidad administrativa. Además, contempla la actualización o inactivación del colaborador sin eliminar información histórica relevante. Este caso de uso garantiza que la información laboral se mantenga centralizada, actualizada y disponible para los procesos operativos del sistema.

Tabla 8

Caso de Uso CU-04 Registro de Asistencia.

Plantilla para la elaboración de Caso de Uso	
Número Caso de Uso: CU-04	Nombre del Caso de Uso: Registro de Asistencia
Fecha elaboración:	04/04/2026
Descripción Caso de Uso:	Permite al colaborador registrar su entrada y salida laboral.
Autor caso de uso:	Emmanuel Cerdas Brenes
Actores relacionados:	Administrador, Jefe, Colaborador.
Precondiciones:	Usuario autenticado. Empleado activo.
Flujo Básico del caso de uso	
Detallar el paso a paso del Flujo Básico	
1.1.1 Colaborador accede a Marcar Entrada / Salida. 1.1.2 El sistema consulta el último registro de marcación del empleado en sesión. 1.1.3 Si no hay registros previos o el último es Salida, habilita Entrada. 1.1.4 Si el último registro es Entrada, habilita Salida. 1.1.5 El colaborador opcionalmente ingresa una observación. 1.1.6 Sistema registra fecha y hora automáticamente. 1.1.7 Sistema guarda registro (FA-1).	
Sub Flujos	

Nombre	Detalle
N/A	N/A
Flujos Alternos	
<i>Corresponde a lo que debe realizar el sistema ante posibles errores</i>	
FA-1 Registro duplicado.	Sistema indica que ya existe un registro para ese periodo.
Requerimientos especiales	
Registro automático con hora del servidor.	
Post-Condiciones	
Asistencia almacenada correctamente.	

El caso de uso Registro de Asistencia permite al colaborador registrar su jornada laboral mediante el control de entradas y salidas diarias. El sistema captura automáticamente la fecha y hora desde el servidor, evitando manipulaciones manuales de la información registrada. Este proceso constituye una fuente primaria de datos utilizada posteriormente en el cálculo automático de planilla. También, se aplican validaciones para prevenir registros duplicados o inconsistentes dentro del mismo periodo laboral. La funcionalidad contribuye al control operativo del tiempo trabajado y facilita la generación de información confiable para la gestión salarial.

Tabla 9

Caso de Uso CU-05 Gestión de Solicitudes.

Plantilla para la elaboración de Caso de Uso	
Número Caso de Uso: CU-05	Nombre del Caso de Uso: Aprobar Solicitudes
Fecha elaboración:	04/04/2026
Descripción Caso de Uso:	Permite al colaborador registrar solicitudes laborales y al jefe revisarlas, aprobarlas o rechazarlas según el tipo de solicitud presentada.
Autor caso de uso:	Emmanuel Cerdas Brenes
Actores relacionados:	Jefe Inmediato.
Precondiciones:	Usuario autenticado.
Flujo Básico del caso de uso	
Detallar el paso a paso del Flujo Básico	
1.1.1 El colaborador ingresa al módulo Solicitudes. 1.1.2 El sistema muestra las solicitudes agrupadas por tipo de solicitud y por estado "Pendientes de Aprobar" o "Aprobadas". 1.1.3 El colaborador selecciona Registrar nueva solicitud. 1.1.4 El sistema muestra los tipos disponibles. 1.1.5 El colaborador selecciona tipo de solicitud. 1.1.6 Solicitud de Horas Extra ir a SF-1. 1.1.7 Solicitud de Vacaciones ir a SF-2. 1.1.8 Registro de Incapacidad ir a SF-3. 1.1.9 Solicitud de Permiso ir a SF-4. 1.1.10 El sistema registra la solicitud con estado Pendiente. 1.1.11 El sistema notifica al jefe inmediato. 1.1.12 El jefe inmediato accede al módulo de solicitudes pendientes. 1.1.13 El jefe revisa la solicitud seleccionada. 1.1.14 El jefe aprueba o rechaza la solicitud (SF-5). 1.1.15 El sistema actualiza estado final y registra auditoría.	
Sub Flujos	

SF-1 Gestión de Horas Extra	2.1.1 El sistema muestra formulario de horas extra. 2.1.2 El colaborador ingresa fecha, cantidad de horas y justificación. 2.1.3 El sistema valida información. 2.1.4 El sistema registra solicitud pendiente. 2.1.8 El sistema registra la solicitud con estado Pendiente. 2.1.9 El sistema notifica automáticamente al jefe inmediato.
SF-2 Gestión de Vacaciones	2.2.1 El sistema consulta saldo disponible. 2.2.2 El colaborador ingresa fechas solicitadas. 2.2.3 El sistema valida disponibilidad. 2.2.4 El sistema registra solicitud pendiente.
SF-3 Registro de Incapacidades	2.3.1 El sistema solicita datos médicos y fechas. 2.3.2 El colaborador adjunta información requerida. 2.3.3 El sistema valida fechas y duplicidad. 2.3.4 El sistema registra incapacidad pendiente.
SF-4 Gestión de Permisos	2.4.1 El sistema muestra formulario de permiso laboral. 2.4.2 El colaborador ingresa fechas y motivo. 2.4.3 El sistema valida datos obligatorios. 2.4.4 El sistema registra solicitud pendiente.
SF-5 Resolución de Solicitud	2.5.1 El jefe inmediato revisa la información completa. 2.5.2 Selecciona Aprobar o Rechazar. 2.5.3 El sistema actualiza el estado. 2.5.4 El sistema notifica al colaborador.
Flujos Alternos	
FA-1 Datos incompletos	3.1.1 El sistema detecta campos obligatorios sin completar. 3.1.2 El sistema resalta los campos faltantes. 3.1.3 Se muestra mensaje indicando que la información debe completarse antes de enviar la solicitud. 3.1.4 El flujo retorna al paso 2.1.3 del subflujo SF-1.
FA-4 Solicitud ya procesada	3.4.1 Si el jefe intenta gestionar una solicitud previamente aprobada o rechazada, el sistema impide la acción. 3.4.2 Se muestra mensaje indicando que la solicitud ya posee estado final.
Requerimientos especiales	
Notificación automática al colaborador.	
Post-Condiciones	
Solicitud actualizada con estado final.	

El caso de uso Aprobar Solicitudes centraliza la gestión de solicitudes laborales dentro del sistema. Todas las solicitudes realizadas por los colaboradores —horas extra, vacaciones, incapacidades y permisos— siguen un flujo unificado que permite su registro, revisión y resolución por parte del jefe inmediato. El sistema organiza automáticamente las solicitudes según su tipo y estado, facilitando el control administrativo y la trazabilidad de decisiones. Cada solicitud ejecuta un sub flujo específico dependiendo de su naturaleza, garantizando validaciones particulares sin perder la integración dentro de un único proceso de aprobación.

Tabla 10*Caso de Uso CU-06 Generación de Planilla.*

Plantilla para la elaboración de Caso de Uso	
Número Caso de Uso: CU-06	Nombre del Caso de Uso: Generación de Planilla
Fecha elaboración:	04/04/2026
Descripción Caso de Uso:	Permite calcular automáticamente los salarios considerando asistencia, horas extra, deducciones y beneficios.
Autor caso de uso:	Emmanuel Cerdas Brenes
Actores relacionados:	Encargado de Planilla.
Precondiciones:	Información laboral registrada. Periodo de pago definido.
Flujo Básico del caso de uso	
Detallar el paso a paso del Flujo Básico	
1.1.1 Encargado accede al módulo Planilla. 1.1.2 Selecciona periodo. 1.1.3 Validar Información de Planilla ir a SF-1 1.1.4 Ejecutar Cálculo Automático ir a SF-2 1.1.5 Sistema genera planilla preliminar. 1.1.6 Confirmar Generación ir a SF-3 1.1.7 Generar Comprobantes Salariales ir a SF-4	
Sub Flujos	
SF-1 Validar Información de Planilla	2.1.1 El sistema consulta registros de asistencia del periodo seleccionado. 2.1.2 El sistema consulta horas extra aprobadas. 2.1.3 El sistema consulta solicitudes aprobadas (vacaciones, permisos, incapacidades). 2.1.4 El sistema verifica que todos los colaboradores activos tengan información completa. 2.1.5 El sistema identifica inconsistencias laborales (FA-1). 2.1.6 Si no existen errores, el proceso continúa al cálculo automático.
SF-2 Ejecutar Cálculo Automático	2.2.1 El sistema calcula salario base por colaborador. 2.2.2 El sistema calcula horas extra según normativa laboral. 2.2.3 El sistema aplica deducciones legales (CCSS, renta u otras). 2.2.4 El sistema aplica beneficios o incentivos configurados. 2.2.5 El sistema calcula salario bruto y salario neto. 2.2.6 El sistema genera registros individuales de pago.
SF-3 Confirmar Generación de Planilla	2.3.1 El encargado de planilla revisa resumen general. 2.3.2 El sistema muestra totales salariales y deducciones. 2.3.3 El encargado selecciona Confirmar Planilla.
SF-4 Generar Comprobantes Salariales	2.4.1 El sistema genera comprobante salarial individual en formato PDF. 2.4.2 El sistema almacena el documento en el historial del colaborador. 2.4.3 El sistema habilita descarga del comprobante.
Flujos Alternos	
FA-1 Datos faltantes para cálculo	3.1.1 El sistema detecta ausencia de asistencia, salarios o configuraciones necesarias. 3.1.2 El sistema muestra listado de inconsistencias. 3.1.3 El sistema impide ejecutar cálculo hasta corregir la información. 3.1.4 El flujo retorna al paso 1.1.2 del flujo básico.
Requerimientos especiales	
Cálculos según normativa laboral costarricense. Registro histórico de planillas.	
Post-Condiciones	
Planilla generada y almacenada.	

El caso de uso Generación de Planilla representa el proceso central del sistema, encargado de calcular automáticamente los salarios de los colaboradores para un periodo determinado. El sistema integra información proveniente de asistencia, horas extra, solicitudes aprobadas y configuraciones laborales previamente registradas. Antes del cálculo, se ejecutan validaciones para detectar inconsistencias que puedan afectar los resultados finales. Posteriormente, se generan los registros salariales individuales y los comprobantes de pago en formato digital. Este proceso automatizado reduce errores manuales, mejora la eficiencia operativa y garantiza consistencia en la administración salarial.

Tabla 11

Caso de Uso CU-07 Registro de Incapacidades.

Plantilla para la elaboración de Caso de Uso	
Número Caso de Uso: CU-07	Nombre del Caso de Uso: Registro de Incapacidades
Fecha elaboración:	04/04/2026
Descripción Caso de Uso:	Permite registrar incapacidades médicas de los colaboradores para su control administrativo y consideración dentro del cálculo de planilla.
Autor caso de uso:	Emmanuel Cerdas Brenes
Actores relacionados:	Colaborador, Jefe, Encargado de Planilla, Administrador.
Precondiciones:	Usuario autenticado. Colaborador registrado en el sistema.
Flujo Básico del caso de uso	
Detallar el paso a paso del Flujo Básico	
1.1.1 Usuario accede al módulo Incapacidades. 1.1.2 Selecciona Registrar Incapacidad. 1.1.3 El sistema muestra formulario de registro. 1.1.4 Usuario selecciona colaborador. 1.1.5 El usuario selecciona tipo, fechas de inicio y fin, porcentaje de pago y observaciones. 1.1.6 El sistema valida que fecha_fin >= fecha_inicio (FA-1). 1.1.7 El sistema valida que el porcentaje esté entre 0 y 1 (FA-2). 1.1.8 El sistema asigna origen según rol (Colaborador, RRHH, Jefatura, Administrador). 1.1.9 El sistema calcula días Incapacidad y guarda con estado Pendiente.	
Sub Flujos	
SF-1: Aprobar	2.1.1 El actor accede a IncapacidadesActivas y aprueba. El sistema ejecuta AplicarImpactoAsistencia para cada día del período y actualiza la tabla Asistencia.
SF-2: Rechazar	2.2.1 El actor rechaza e ingresa observaciones. El sistema guarda estado Rechazada y observaciones_rrhh.
SF-3: MisIncapacidades	2.3.1 El empleado visualiza únicamente sus propias incapacidades.
Flujos Alternos	
FA-1 Fecha fin menor a fecha inicio	El sistema muestra error.
FA-2 Porcentaje fuera de rango	El sistema muestra error de validación.
FA-3 Incapacidad no en estado Pendiente	El sistema impide la acción y redirige.
Requerimientos especiales	
El porcentaje de pago debe estar entre 0 y 1 (ejemplo: 0.60 = 60%). La aprobación de una incapacidad impacta automáticamente los registros de Asistencia del período.	
Post-Condiciones	
La incapacidad queda registrada con estado Pendiente, Aprobada o Rechazada. Al aprobar, los registros de Asistencia del período quedan actualizados con estado Incapacidad.	

El caso de uso Registro de Incapacidades permite documentar las ausencias laborales justificadas por motivos médicos dentro del sistema. Mediante este proceso se registran periodos de incapacidad asociados a un colaborador, asegurando que dicha información sea considerada durante el cálculo de planilla. El sistema valida fechas y evita registros inconsistentes que puedan afectar la gestión salarial. Además, se conserva un historial administrativo que facilita consultas posteriores y control organizacional. Este caso de uso contribuye a mantener una gestión adecuada de ausencias laborales justificadas y su impacto dentro de los procesos de pago.

Tabla 12

Caso de Uso CU-08 Evaluar Personal.

Plantilla para la elaboración de Caso de Uso	
Número Caso de Uso: CU-08	Nombre del Caso de Uso: Evaluar Personal
Fecha elaboración:	04/04/2026
Descripción Caso de Uso:	Permite registrar evaluaciones de desempeño de los colaboradores.
Autor caso de uso:	Emmanuel Cerdas Brenes
Actores relacionados:	Administrador, RRHH, Jefe.
Precondiciones:	Deben existir empleados activos registrados. El usuario debe estar autenticado con rol Administrador o RRHH.
Flujo Básico del caso de uso	
Detallar el paso a paso del Flujo Básico	
1.1.1 El actor accede a Evaluacion/Crear. 1.1.2 El sistema muestra combo de empleados activos, año actual y semestre actual por defecto (FA-1). 1.1.3 El actor ingresa nombre del evaluador, las cinco notas, comentarios y fecha de evaluación. 1.1.4 El sistema valida que todas las notas estén entre 1 y 10 (FA-2). 1.1.5 El sistema verifica que no exista evaluación previa para ese empleado, año y semestre (FA-3 y FA-4). 1.1.6 El sistema calcula la nota_final (promedio) y asigna la calificación descriptiva. 1.1.7 El sistema guarda la evaluación y registra en Bitácora CREAR EVALUACION (FA-5). 1.1.8 El sistema redirige al listado de evaluaciones.	
Sub Flujos	
SF-1: Ver detalle	El actor accede a Evaluacion/Detalle para ver el desglose completo de una evaluación específica.
Flujos Alternos	
FA-1 Empleado no seleccionado	El sistema muestra error.
FA-2 Nota fuera del rango 1-10	El sistema muestra error y no guarda.
FA-3 Evaluación duplicada para el período	El sistema muestra error de duplicado.
FA-4 Fecha de evaluación inválida	El sistema muestra error de fecha.
FA-5 Evaluador no especificado	El sistema muestra error de campo obligatorio.
Requerimientos especiales	

N/A
Post-Condiciones
Evaluación almacenada en historial.

El caso de uso Evaluar Personal permite registrar evaluaciones básicas del desempeño de los colaboradores como apoyo a la gestión administrativa. El proceso consiste en la asignación de criterios previamente definidos que permiten valorar aspectos generales del desempeño laboral. El sistema almacena los resultados asociados a cada colaborador, generando un historial que puede ser consultado posteriormente. Aunque el módulo presenta una implementación sencilla, proporciona información útil para seguimiento interno y futuras decisiones organizacionales. Este caso de uso introduce un mecanismo inicial de evaluación dentro del sistema sin agregar complejidad operativa innecesaria.

Tabla 13

Caso de Uso CU-09 Cálculo Aguinaldo.

Plantilla para la elaboración de Caso de Uso	
Número Caso de Uso: CU-09	Nombre del Caso de Uso: Cálculo Aguinaldo
Fecha elaboración:	04/04/2026
Descripción Caso de Uso:	Permite calcular automáticamente el aguinaldo anual de los colaboradores según salarios registrados.
Autor caso de uso:	Emmanuel Cerdas Brenes
Actores relacionados:	Encargado de Planilla.
Precondiciones:	Deben existir empleados activos con historial de nómina en el año seleccionado. El usuario debe estar autenticado con rol Administrador o RRHH.
Flujo Básico del caso de uso	
Detallar el paso a paso del Flujo Básico	
1.1.1. El actor accede a Aguinaldo/Calcular. 1.1.2. El sistema muestra combo de empleados activos y campo de año (por defecto año actual) (FA-1). 1.1.3. El actor selecciona empleado y año y envía el formulario (FA-2). 1.1.4. El sistema valida el año (entre 2000 y año actual + 1). 1.1.5. El sistema invoca AguinaldoService.CalcularAguinaldo. 1.1.6. El sistema muestra DetalleAguinaldo con el resultado. 1.1.7. El actor confirma y el sistema invoca GuardarAguinaldo (FA-3). 1.1.8. El sistema registra en Bitácora la acción GUARDAR AGUINALDO con empleado, año y monto.	
Sub Flujos	
SF-1: Ver historial	2.1.1 El actor accede a Aguinaldo/Index para ver todos los aguinaldos guardados.
Flujos Alternos	

FA-1 Empleado no seleccionado	Sistema impide cálculo hasta completar información.
FA-2 Año inválido	El sistema muestra error.
FA-3 Error en el cálculo	El sistema muestra el mensaje de excepción del servicio.
Requerimientos especiales	
El cálculo de aguinaldo debe basarse en el historial de nóminas del período diciembre-noviembre.	
Post-Condiciones	
El aguinaldo del empleado para el año queda guardado en la tabla Aguinaldo. La acción queda registrada en Bitácora.	

El caso de uso Cálculo de Aguinaldo permite realizar automáticamente el cálculo del aguinaldo correspondiente a cada colaborador conforme a la normativa laboral costarricense. El sistema recopila la información salarial acumulada durante el periodo anual y ejecuta el cálculo utilizando los registros históricos de planilla. Antes de generar el resultado, se validan los datos disponibles para evitar inconsistencias que puedan afectar el monto final. Posteriormente, el sistema almacena el cálculo realizado y lo deja disponible para consulta o generación de reportes. Esta funcionalidad automatiza un proceso tradicionalmente manual y asegura uniformidad en la aplicación del beneficio laboral.

Tabla 14

Caso de Uso CU-10 Gestión de Liquidación Laboral.

Plantilla para la elaboración de Caso de Uso	
Número Caso de Uso: CU-10	Nombre del Caso de Uso: Gestión de Liquidación Laboral
Fecha elaboración:	04/04/2026
Descripción Caso de Uso:	Permite calcular la liquidación laboral de un colaborador al finalizar la relación laboral.
Autor caso de uso:	Emmanuel Cerdas Brenes
Actores relacionados:	Administrador, Encargado de Planilla.
Precondiciones:	Usuario autenticado.
Flujo Básico del caso de uso	
Detallar el paso a paso del Flujo Básico	
1.1.1. El actor accede a Liquidacion/Calcular. 1.1.2. El sistema muestra combos de empleados activos, tipos de liquidación y campo de fecha. 1.1.3. El actor selecciona empleado, tipo de liquidación, fecha y observaciones opcionales (FA-2). 1.1.4. El sistema valida que la fecha no sea futura (FA-2). 1.1.5 El sistema invoca LiquidacionService.CalcularLiquidacion: 1.1.5.1 Sistema calcula salarios pendientes. 1.1.5.2 Sistema calcula vacaciones no disfrutadas. 1.1.5.3 Sistema calcula aguinaldo proporcional. 1.1.5.4 Sistema calcula beneficios correspondientes. 1.1.5.5 Sistema genera monto final. 1.1.6 El sistema muestra DetalleLiquidacion con el desglose completo (FA-3). 1.1.7 El actor confirma y el sistema invoca GuardarLiquidacion. 1.1.8 El sistema registra en Bitácora GUARDAR LIQUIDACION con nombre, tipo y neto.	
Sub Flujos	
SF-1: Ver historial	El actor accede a Liquidacion/Index para ver todas las liquidaciones guardadas.

SF-2: Gestión de tipos	El Administrador gestiona TiposLiquidacion (codigo, nombre, aplica_cesantia, aplica_preaviso) vía TipoLiquidacionController.
Flujos Alternos	
FA-1: Fecha de liquidación futura	El sistema muestra error.
FA-2: Datos incompletos	El sistema muestra error y retorna al formulario.
FA-3: Error en el cálculo	El sistema muestra el mensaje de excepción del servicio.
Requerimientos especiales	
El tipo de liquidación determina si aplica cesantía y preaviso (Código de Trabajo CR).	
Post-Condiciones	
La liquidación queda guardada en la tabla Liquidación.	
La acción queda registrada en Bitácora.	

El caso de uso Gestión de Liquidación Laboral permite calcular los montos correspondientes al finalizar la relación laboral de un colaborador. El sistema considera información histórica como salarios devengados, vacaciones pendientes y beneficios acumulados para generar el cálculo correspondiente. Durante el proceso se aplican validaciones que aseguran la correcta recopilación de datos antes de emitir el resultado final. Una vez completado el cálculo, el sistema registra la liquidación y permite su consulta o generación de documentación asociada. Este caso de uso garantiza un cierre administrativo ordenado y conforme a las disposiciones laborales vigentes.

Tabla 15

Caso de Uso CU-11 Gestionar Nómina.

Plantilla para la elaboración de Caso de Uso	
Número Caso de Uso: CU-11	Nombre del Caso de Uso: Gestionar Nómina
Fecha elaboración:	04/04/2026
Descripción Caso de Uso:	Permite al encargado de planilla administrar el proceso completo de nómina, incluyendo la apertura de periodos, revisión de planillas generadas y cierre administrativo del periodo salarial.
Autor caso de uso:	Emmanuel Cerdas Brenes
Actores relacionados:	Encargado de Planilla.
Precondiciones:	Usuario autenticado. Existencia de colaboradores activos.
Flujo Básico del caso de uso	
Detallar el paso a paso del Flujo Básico	
1.1.1. El actor accede a Nomina/CalcularNomina. 1.1.2. El sistema muestra combo de empleados activos y campos de mes y año. 1.1.3. El actor selecciona empleado, mes (1-12) y año y envía el formulario (FA-1). 1.1.4. El sistema valida los rangos de mes y año (FA-2). 1.1.5. El sistema calcula fechaInicio (primer día del mes) y fechaFin (último día del mes). 1.1.6. El sistema invoca NominaService.CalcularNominaEmpleado y muestra el DetalleNomina. 1.1.7. El actor confirma el guardado. 1.1.8. El sistema invoca GuardarNomina (FA-3). 1.1.9. El sistema registra en Bitácora GUARDAR NOMINA con datos del empleado, período y neto. 1.1.10. El sistema redirige a CalcularNomina con mensaje de éxito.	
Sub Flujos	
SF-1 Ver historial	El actor accede a Nomina/Index para ver todas las nóminas guardadas ordenadas por año y mes descendente.

SF-2 Detalle por período	El actor accede a DetalleNominaPeriodo para ver el desglose completo de un período específico.
Flujos Alternos	
FA-1 Mes inválido (fuera de 1-12)	El sistema muestra error.
FA-2 Año inválido (antes de 2000 o futuro lejano)	El sistema muestra error.
FA-3 Error en el cálculo (NominaService)	El sistema muestra el mensaje de excepción.
Requerimientos especiales	
El cálculo debe considerar deducciones de CCSS, impuesto de renta y otras deducciones configuradas. La acción de guardado debe quedar registrada en Bitácora.	
Post-Condiciones	
La nómina del empleado para el período queda guardada en las tablas Nomina y DetalleNomina. La acción queda registrada en Bitácora.	

El caso de uso Gestionar Nómina integra las operaciones administrativas relacionadas con el procesamiento salarial dentro del sistema. Este proceso permite controlar el ciclo completo de la nómina, desde la preparación del periodo hasta su cierre formal, garantizando trazabilidad y control organizacional sin añadir complejidad funcional innecesaria al desarrollo. así como un registro de este en la base de datos para su consulta posterior.

Tabla 16

Caso de Uso CU-12 Consultar Asistencia.

Plantilla para la elaboración de Caso de Uso	
Número Caso de Uso: CU-12	Nombre del Caso de Uso: Consultar Asistencia
Fecha elaboración:	04/04/2026
Descripción Caso de Uso:	Permite al colaborador visualizar su historial de registros de asistencia, incluyendo entradas, salidas y observaciones registradas durante el periodo laboral.
Autor caso de uso:	Emmanuel Cerdas Brenes
Actores relacionados:	Colaborador, Administrador, Encargado de Planilla.
Precondiciones:	Usuario autenticado. Colaborador activo con registros de asistencia.
Flujo Básico del caso de uso	
Detallar el paso a paso del Flujo Básico	
1.1.1 El colaborador accede al módulo de Consulta de Asistencia. 1.1.2 El sistema muestra filtros disponibles: rango de fechas y tipo de registro. 1.1.3 El colaborador selecciona el periodo que desea consultar. 1.1.4 El sistema recupera los registros de asistencia del colaborador en sesión. 1.1.5 El sistema presenta el listado con fecha, hora de entrada, hora de salida y observaciones. 1.1.6 El colaborador puede visualizar el detalle de un registro específico (SF-1). 1.1.7 El sistema muestra el resumen de días trabajados en el periodo seleccionado (FA-1).	
Sub Flujos	
SF-1: Ver detalle	2.1.1 El actor selecciona un registro específico. 2.1.2 El sistema muestra entrada, salida, horas totales y observaciones del día.
Flujos Alternos	
FA-1 Sin registros en el periodo	El sistema muestra mensaje indicando que no existen registros para el rango seleccionado.
Requerimientos especiales	
El colaborador únicamente puede visualizar sus propios registros. El administrador puede consultar registros de cualquier colaborador.	
Post-Condiciones	
Registros de asistencia mostrados correctamente según el periodo seleccionado. No se realizan modificaciones a los datos durante la consulta.	

El caso de uso Consultar Asistencia permite al colaborador revisar su historial de marcaciones registradas en el sistema durante un periodo determinado. A través de esta funcionalidad se ofrece transparencia en el registro de jornadas laborales, facilitando la verificación personal de entradas, salidas y observaciones almacenadas. El sistema aplica restricciones de acceso para que cada colaborador únicamente pueda visualizar su propia información, garantizando la confidencialidad de los datos. Por su parte, los usuarios con rol administrativo pueden acceder al historial de cualquier colaborador para efectos de control y auditoría. Este caso de uso contribuye a la trazabilidad operativa del sistema y al seguimiento del cumplimiento horario.

Tabla 17

Caso de Uso CU-13 Consultar Comprobante Salarial.

Plantilla para la elaboración de Caso de Uso	
Número Caso de Uso: CU-13	Nombre del Caso de Uso: Consultar Comprobante Salarial
Fecha elaboración:	04/04/2026
Descripción Caso de Uso:	Permite al colaborador consultar y descargar el comprobante salarial generado para un periodo específico, con el detalle de ingresos, deducciones y monto neto percibido.
Autor caso de uso:	Emmanuel Cerdas Brenes
Actores relacionados:	Colaborador.
Precondiciones:	Usuario autenticado con rol Colaborador. Existencia de nómina procesada para el periodo.
Flujo Básico del caso de uso	
Detallar el paso a paso del Flujo Básico	
1.1.1 El colaborador accede al módulo Consultar Comprobante Salarial. 1.1.2 El sistema muestra el listado de periodos con nómina disponible. 1.1.3 El colaborador selecciona el periodo que desea consultar (FA-1). 1.1.4 El sistema recupera el comprobante salarial correspondiente al colaborador en sesión. 1.1.5 El sistema presenta el detalle: salario bruto, deducciones aplicadas y salario neto. 1.1.6 El colaborador puede descargar el comprobante en formato digital (SF-1).	
Sub Flujos	
SF-1: Descargar comprobante	2.1.1 El actor selecciona la opción de descarga. 2.1.2 El sistema genera el documento en formato PDF. 2.1.3 El sistema habilita la descarga del archivo.
Flujos Alternos	
FA-1 No existe comprobante para el periodo	El sistema informa que no se ha generado nómina para el periodo seleccionado.
Requerimientos especiales	
El colaborador solo puede consultar sus propios comprobantes. No se permite modificación de datos durante la consulta.	
Post-Condiciones	
Comprobante salarial visualizado correctamente con todos sus componentes.	

El caso de uso Consultar Comprobante Salarial permite al colaborador acceder al detalle de su remuneración para un periodo determinado. Esta funcionalidad brinda transparencia en la gestión salarial al exponer de forma clara los componentes del salario bruto, las deducciones legales

aplicadas y el monto neto recibido. El sistema garantiza que cada colaborador únicamente pueda consultar sus propios comprobantes, protegiendo la confidencialidad de la información salarial. Además, se ofrece la opción de descarga en formato digital para facilitar la conservación del documento por parte del usuario. Este caso de uso complementa el proceso de generación de planilla al hacer accesible la información salarial directamente al colaborador.

Tabla 18

Caso de Uso CU-14 Consultar Historial Salarial.

Plantilla para la elaboración de Caso de Uso	
Número Caso de Uso: CU-14	Nombre del Caso de Uso: Consultar Historial Salarial
Fecha elaboración:	04/04/2026
Descripción Caso de Uso:	Permite al colaborador visualizar el historial completo de sus nóminas procesadas, mostrando la evolución de su remuneración a lo largo del tiempo.
Autor caso de uso:	Emmanuel Cerdas Brenes
Actores relacionados:	Colaborador.
Precondiciones:	Usuario autenticado. Existencia de nóminas procesadas asociadas al colaborador.
Flujo Básico del caso de uso	
Detallar el paso a paso del Flujo Básico	
1.1.1 El colaborador accede al módulo Historial Salarial. 1.1.2 El sistema recupera el historial de nóminas asociado al colaborador en sesión. 1.1.3 El sistema presenta el listado ordenado por fecha descendente con período, salario bruto y salario neto. 1.1.4 El colaborador puede seleccionar un período específico para ver el detalle (SF-1). 1.1.5 El sistema muestra el desglose completo del período seleccionado (FA-1).	
Sub Flujos	
SF-1: Ver detalle de período	2.1.1 El actor selecciona un período del historial. 2.1.2 El sistema muestra salario bruto, deducciones detalladas, horas extra y salario neto del período.
Flujos Alternos	
FA-1 Sin historial disponible	El sistema muestra mensaje indicando que no existen registros salariales para el colaborador.
Requerimientos especiales	
La consulta es de solo lectura. El colaborador no puede modificar información salarial.	
Post-Condiciones	
Historial salarial presentado correctamente al colaborador. No se realizan modificaciones a los datos durante la consulta.	

El caso de uso Consultar Historial Salarial permite al colaborador revisar la evolución de sus ingresos durante el tiempo que ha mantenido relación laboral con la organización. Al respecto, el sistema presenta el listado de nóminas procesadas ordenado cronológicamente, facilitando la comparación de periodos y el seguimiento del salario percibido. Cada registro del historial puede consultarse en detalle para obtener el desglose completo de ingresos y deducciones aplicadas. Esta

funcionalidad promueve la transparencia administrativa y otorga al colaborador acceso directo a su información salarial histórica. Por ello, resulta un componente de apoyo al control personal y a la verificación de datos desde la perspectiva del usuario.

Tabla 19

Caso de Uso CU-15 Gestionar Roles.

Plantilla para la elaboración de Caso de Uso	
Número Caso de Uso: CU-15	Nombre del Caso de Uso: Gestionar Roles
Fecha elaboración:	04/04/2026
Descripción Caso de Uso:	Permite al administrador crear, modificar y administrar los roles del sistema, definiendo los permisos y accesos asociados a cada perfil de usuario.
Autor caso de uso:	Emmanuel Cerdas Brenes
Actores relacionados:	Administrador.
Precondiciones:	Usuario autenticado con rol Administrador.
Flujo Básico del caso de uso	
Detallar el paso a paso del Flujo Básico	
1.1.1 El administrador accede al módulo Mantenimiento, Roles. 1.1.2 El sistema valida permisos administrativos (FA-1). 1.1.3 El sistema muestra el listado de roles registrados. 1.1.4 El administrador selecciona la acción a ejecutar. 1.1.5 Registrar Rol ir a SF-1. 1.1.6 Modificar Rol ir a SF-2. 1.1.7 Asignar Permisos ir a SF-3. 1.1.8 El sistema ejecuta el subflujo correspondiente.	
Sub Flujos	
SF-1: Registrar Rol	2.1.1 El administrador ingresa nombre y descripción del rol. 2.1.2 El sistema valida que el nombre no esté duplicado (FA-2). 2.1.3 El sistema registra el nuevo rol.
SF-2: Modificar Rol	2.2.1 El administrador selecciona el rol a modificar. 2.2.2 Actualiza nombre o descripción. 2.2.3 El sistema guarda los cambios.
SF-3: Asignar Permisos	2.3.1 El administrador selecciona el rol. 2.3.2 El sistema muestra los permisos disponibles. 2.3.3 El administrador activa o desactiva permisos. 2.3.4 El sistema guarda la configuración.
Flujos Alternos	
FA-1 Sin permisos administrativos	El sistema impide el acceso al módulo.
FA-2 Nombre de rol duplicado	El sistema informa que ya existe un rol con ese nombre.
Requerimientos especiales	
Los cambios en roles aplican de forma inmediata a todos los usuarios asociados. Los roles del sistema base no pueden eliminarse.	
Post-Condiciones	
Rol registrado, modificado o con permisos actualizados según la acción ejecutada. Los cambios quedan registrados en Bitácora.	

El caso de uso Gestionar Roles permite al administrador configurar los perfiles de acceso que controlan las funcionalidades disponibles para cada tipo de usuario dentro del sistema. En esa línea, este proceso posibilita definir, ajustar y asignar permisos de forma granular según las responsabilidades asociadas a cada rol organizacional. El sistema valida que no existan

duplicidades en los nombres de los roles y registra todos los cambios en el módulo de auditoría. De igual manera, la correcta definición de roles garantiza que el control de acceso basado en perfiles funcione de forma coherente en todos los módulos de la aplicación. Este caso de uso es fundamental para mantener la seguridad y la separación de funciones dentro del entorno del sistema.

Tabla 20

Caso de Uso CU-16 Configurar Parámetros del Sistema.

Plantilla para la elaboración de Caso de Uso	
Número Caso de Uso: CU-16	Nombre del Caso de Uso: Configurar Parámetros del Sistema
Fecha elaboración:	04/04/2026
Descripción Caso de Uso:	Permite al administrador definir y actualizar los parámetros generales del sistema, tales como deducciones, porcentajes salariales, beneficios y configuraciones de nómina.
Autor caso de uso:	Emmanuel Cerdas Brenes
Actores relacionados:	Administrador.
Precondiciones:	Usuario autenticado con rol Administrador.
Flujo Básico del caso de uso	
Detallar el paso a paso del Flujo Básico	
1.1.1 El administrador accede al módulo Configuración de Parámetros. 1.1.2 El sistema valida los permisos del usuario (FA-1). 1.1.3 El sistema muestra las secciones de configuración disponibles. 1.1.4 El administrador selecciona la categoría a configurar. 1.1.5 Configurar deducciones laborales ir a SF-1. 1.1.6 Configurar tipos de hora extra ir a SF-2. 1.1.7 Configurar parámetros de nómina ir a SF-3. 1.1.8 El sistema valida los valores ingresados (FA-2). 1.1.9 El sistema guarda la configuración y registra en Bitácora.	
Sub Flujos	
SF-1: Configurar deducciones	2.1.1 El administrador gestiona los tipos y porcentajes de deducción (CCSS, renta, otros). 2.1.2 El sistema valida que los porcentajes sean coherentes. 2.1.3 El sistema actualiza la configuración activa.
SF-2: Configurar horas extra	2.2.1 El administrador define categorías de horas extra y sus multiplicadores salariales. 2.2.2 El sistema guarda la configuración vigente.
SF-3: Parámetros de nómina	2.3.1 El administrador define períodos de pago, días hábiles y reglas de cálculo. 2.3.2 El sistema valida la coherencia de los datos ingresados. 2.3.3 El sistema actualiza la configuración de nómina.
Flujos Alternos	
FA-1 Sin permisos administrativos	El sistema impide el acceso al módulo.
FA-2 Valores fuera de rango	El sistema indica el error y solicita corrección antes de guardar.
Requerimientos especiales	
Los cambios en parámetros afectan los cálculos futuros de planilla. Los cambios históricos no se ven alterados.	
Post-Condiciones	
Parámetros del sistema actualizados y disponibles para los próximos cálculos de nómina. Acción registrada en Bitácora con el usuario y la fecha del cambio.	

El caso de uso Configurar Parámetros del Sistema permite al administrador ajustar las reglas de cálculo y los valores que rigen el procesamiento de la planilla. Por su parte, mediante esta funcionalidad se definen los porcentajes de deducción, las categorías de horas extra, los multiplicadores salariales y las reglas de nómina vigentes en la organización. El sistema valida que los valores ingresados sean coherentes antes de aplicar los cambios, previniendo configuraciones erróneas que puedan afectar los cálculos posteriores. Cada modificación realizada queda registrada en el módulo de auditoría con el identificador del usuario responsable. Este caso de uso garantiza que el sistema pueda adaptarse a cambios regulatorios o internos sin requerir modificaciones en el código de la aplicación.

Tabla 21

Caso de Uso CU-17 Administrar Seguridad.

Plantilla para la elaboración de Caso de Uso	
Número Caso de Uso: CU-17	Nombre del Caso de Uso: Administrar Seguridad
Fecha elaboración:	04/04/2026
Descripción Caso de Uso:	Permite al administrador gestionar las políticas de seguridad del sistema, incluyendo restablecimiento de contraseñas, bloqueo de cuentas y control de sesiones activas.
Autor caso de uso:	Emmanuel Cerdas Brenes
Actores relacionados:	Administrador.
Precondiciones:	Usuario autenticado con rol Administrador.
Flujo Básico del caso de uso	
Detallar el paso a paso del Flujo Básico	
1.1.1 El administrador accede al módulo Administración de Seguridad. 1.1.2 El sistema valida los permisos del usuario (FA-1). 1.1.3 El sistema muestra las opciones disponibles de seguridad. 1.1.4 El administrador selecciona la acción a ejecutar. 1.1.5 Restablecer contraseña ir a SF-1. 1.1.6 Bloquear o desbloquear cuenta ir a SF-2. 1.1.7 El sistema ejecuta el subflujo correspondiente y registra en Bitácora.	
Sub Flujos	
SF-1: Restablecer contraseña	2.1.1 El administrador selecciona el usuario. 2.1.2 El sistema genera una contraseña temporal cifrada. 2.1.3 El sistema marca la cuenta para cambio obligatorio al próximo acceso. 2.1.4 El sistema notifica al usuario.
SF-2: Bloquear/desbloquear cuenta	2.2.1 El administrador selecciona el usuario y la acción. 2.2.2 El sistema solicita confirmación (FA-2). 2.2.3 El sistema actualiza el estado de la cuenta. 2.2.4 El sistema registra la acción en Bitácora.
Flujos Alternos	
FA-1 Sin permisos administrativos	El sistema impide el acceso al módulo.
FA-2 Confirmación rechazada	El sistema cancela la operación sin realizar cambios.
Requerimientos especiales	
Las contraseñas se almacenan cifradas. El restablecimiento fuerza cambio obligatorio en el próximo acceso. Todas las acciones quedan registradas en auditoría.	
Post-Condiciones	
Cuenta de usuario actualizada según la acción ejecutada.	

Acción registrada en Bitácora con fecha, hora y usuario responsable.
--

El caso de uso Administrar Seguridad centraliza las operaciones que permiten mantener el control de acceso y la integridad de las cuentas del sistema. A través de este proceso, el administrador puede restablecer credenciales de usuarios que han perdido el acceso, así como bloquear cuentas comprometidas o inactivas que representen un riesgo de seguridad. El sistema garantiza que las contraseñas sean siempre almacenadas de forma cifrada y que cualquier restablecimiento obligue al usuario a establecer una nueva clave al primer acceso. Cada acción ejecutada dentro de este módulo queda registrada en la bitácora del sistema para garantizar trazabilidad y responsabilidad administrativa. Este caso de uso refuerza el esquema de seguridad institucional y protege la integridad de la información gestionada.

Tabla 22

Caso de Uso CU-18 Auditoría del Sistema.

Plantilla para la elaboración de Caso de Uso	
Número Caso de Uso: CU-18	Nombre del Caso de Uso: Auditoría del Sistema
Fecha elaboración:	04/04/2026
Descripción Caso de Uso:	Permite al administrador consultar el registro histórico de acciones realizadas en el sistema, identificando usuarios, fechas, módulos afectados y tipos de operación ejecutados.
Autor caso de uso:	Emmanuel Cerdas Brenes
Actores relacionados:	Administrador.
Precondiciones:	Usuario autenticado con rol Administrador. Existencia de registros en la Bitácora del sistema.
Flujo Básico del caso de uso	
Detallar el paso a paso del Flujo Básico	
1.1.1 El administrador accede al módulo Auditoría del Sistema. 1.1.2 El sistema valida los permisos del usuario (FA-1). 1.1.3 El sistema muestra el panel de consulta con filtros disponibles. 1.1.4 El administrador selecciona los criterios de búsqueda: rango de fechas, usuario, módulo o tipo de acción. 1.1.5 El sistema ejecuta la consulta y recupera los registros de Bitácora (FA-2). 1.1.6 El sistema presenta el listado con: fecha, usuario, módulo, acción ejecutada y resultado. 1.1.7 El administrador puede consultar el detalle de un registro específico (SF-1). 1.1.8 El administrador puede exportar los resultados de la consulta (SF-2).	
Sub Flujos	
SF-1: Ver detalle de registro	2.1.1 El administrador selecciona un registro del listado. 2.1.2 El sistema muestra el detalle completo: datos del usuario, módulo, parámetros de la operación y resultado obtenido.
SF-2: Exportar registros	2.2.1 El administrador selecciona la opción de exportar. 2.2.2 El sistema genera el reporte con los registros filtrados. 2.2.3 El sistema habilita la descarga del archivo.
Flujos Alternos	
FA-1 Sin permisos administrativos	El sistema impide el acceso al módulo de auditoría.

FA-2 Sin registros para los filtros aplicados	El sistema informa que no existen entradas en Bitácora para los criterios seleccionados.
Requerimientos especiales	
La consulta de auditoría es de solo lectura. No se permite modificar ni eliminar registros de Bitácora. El acceso está restringido exclusivamente al rol Administrador.	
Post-Condiciones	
Registros de Bitácora presentados correctamente según los filtros aplicados. No se realizan modificaciones a los datos durante la consulta.	

El caso de uso Auditoría del Sistema permite al administrador acceder al historial completo de acciones ejecutadas dentro de la plataforma, facilitando la supervisión y el control operativo del entorno digital.

Sobre el tema, cada evento registrado en la Bitácora incluye información sobre el usuario responsable, el módulo afectado, el tipo de operación realizada y el resultado obtenido, lo que permite reconstruir el historial de actividad con precisión.

El sistema ofrece filtros flexibles que facilitan la búsqueda de registros específicos en función de diferentes criterios, reduciendo el tiempo de análisis en situaciones de revisión o incidente.

La consulta es de solo lectura, lo que impide cualquier alteración de los registros históricos y garantiza la integridad del registro de auditoría. Este caso de uso fortalece el esquema de control interno del sistema y respalda los procesos de revisión administrativa y cumplimiento normativo.

Diseño

Arquitectura del sistema

La arquitectura del sistema propuesta se basa en un modelo cliente–servidor orientado a aplicaciones web empresariales, el cual permite centralizar el procesamiento de la información y facilitar el acceso de los usuarios desde distintos dispositivos dentro de la organización. En este enfoque, los colaboradores interactúan con el sistema mediante navegadores web instalados en equipos de escritorio, portátiles o dispositivos móviles conectados a la red institucional, ya sea de por medio de una conexión alámbrica o inalámbrica. La solución fue diseñada considerando un entorno organizacional real, donde la disponibilidad, seguridad y control de la información representan factores fundamentales para la gestión del sistema de planilla.

A diferencia de una arquitectura monolítica tradicional, la propuesta separa los componentes críticos del sistema en servidores independientes, estableciendo un Servidor de Aplicaciones y un Servidor de Base de Datos. Esta decisión responde a buenas prácticas de ingeniería de software y administración de infraestructura, ya que permite distribuir la carga de procesamiento, mejorar el rendimiento general y aumentar la seguridad de los datos. El sistema fue desarrollado utilizando ASP.NET MVC bajo .NET Framework en Visual Studio, aprovechando el patrón Modelo-Vista-Controlador para mantener una estructura organizada entre la interfaz, la lógica de negocio y el acceso a datos.

Acceso de Usuarios: El acceso al sistema se realiza a través de navegadores web, permitiendo que los usuarios autorizados utilicen la aplicación sin necesidad de instalar software adicional en sus equipos. Los dispositivos clientes se conectan mediante la red local empresarial, ya sea por conexión cableada o inalámbrica a través de un punto de acceso administrado por el router institucional. Este esquema facilita la accesibilidad, reduce costos de mantenimiento y permite que el sistema pueda utilizarse desde diferentes ubicaciones internas de la empresa manteniendo control de autenticación y permisos según los roles definidos.

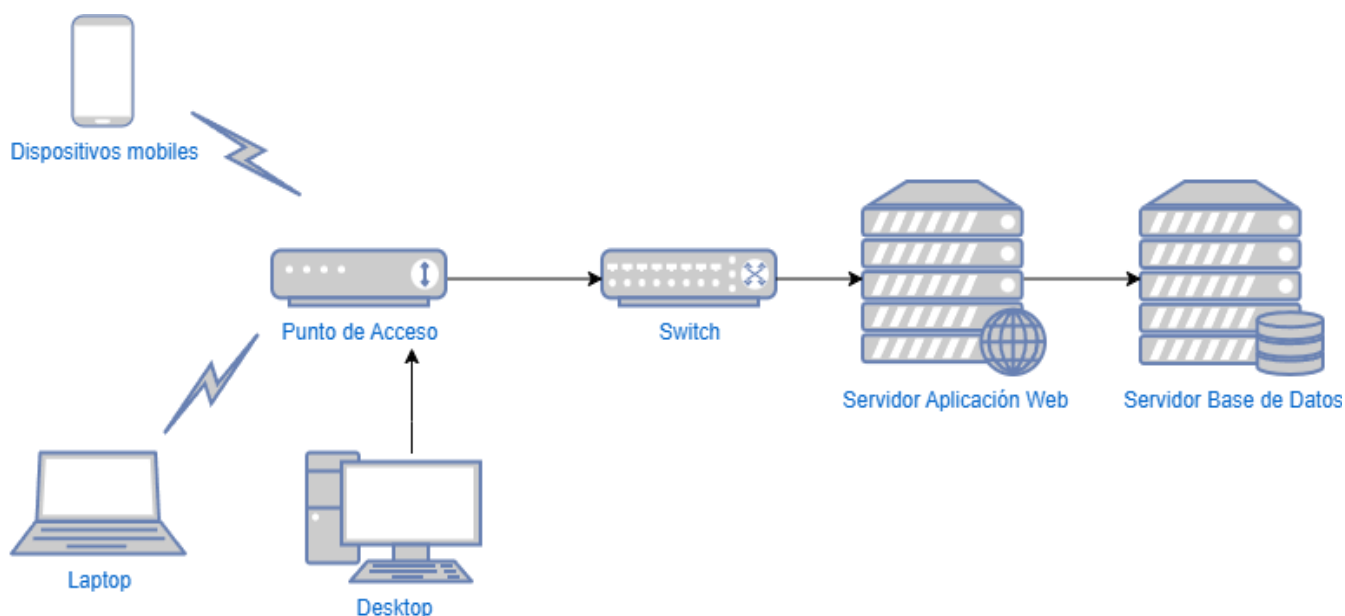
Servidor de Aplicaciones: El servidor de aplicaciones constituye el núcleo operativo del sistema, ya que aloja la aplicación web desarrollada en ASP.NET MVC. En este componente se ejecutan los procesos principales relacionados con la lógica de negocio, validaciones del sistema, gestión de usuarios, procesamiento de nómina y generación de reportes. La separación del servidor de aplicaciones permite aislar la ejecución del sistema respecto a los datos almacenados, lo que

mejora la escalabilidad y facilita futuras actualizaciones sin afectar directamente la información corporativa.

Servidor de Base de Datos: El servidor de base de datos se encarga exclusivamente del almacenamiento y administración de la información del sistema, incluyendo colaboradores, movimientos de planilla, permisos, cálculos salariales e historial operativo. Se implementa como un servidor independiente para fortalecer la seguridad, restringiendo el acceso directo a los datos únicamente al servidor de aplicaciones. Esta distribución reduce riesgos de pérdida o manipulación indebida de información y permite optimizar el rendimiento mediante estrategias propias de administración de bases de datos, respaldos y control de transacciones.

La Figura 17 presenta la vista general de la arquitectura del sistema dentro de la infraestructura tecnológica de la empresa. En el diagrama se observa la interacción entre los dispositivos de usuario, los componentes de red y los servidores principales que conforman la solución propuesta. Esta representación permite visualizar cómo fluye la información desde el acceso del usuario hasta el procesamiento en el servidor de aplicaciones y su posterior almacenamiento en el servidor de base de datos, evidenciando la separación de responsabilidades y la organización estructural definida para el prototipo desarrollado.

Figura 17.
Diagrama Arquitectura del sistema



Fuente: Elaboración propia.

Arquitectura del software

La arquitectura del software del sistema propuesto se fundamenta en un modelo por capas, alineado con las buenas prácticas de desarrollo utilizadas en aplicaciones empresariales basadas en tecnologías Microsoft. El prototipo fue desarrollado en Visual Studio utilizando ASP.NET MVC (.NET Framework), lo que permite separar claramente las responsabilidades del sistema y facilitar su mantenimiento, escalabilidad y evolución futura. Esta organización interna contribuye a reducir el acoplamiento entre componentes, permitiendo que cada parte del sistema cumpla una función específica dentro del procesamiento de la información.

El diseño adopta una arquitectura lógica dividida en tres capas principales: Presentación, Negocio y Datos, las cuales interactúan de forma controlada mediante el patrón Modelo-Vista-Controlador (MVC). Esta estructura permite que la interfaz de usuario, las reglas del negocio y el acceso a la base de datos funcionen de manera independiente, mejorando la claridad del desarrollo y simplificando las tareas de mantenimiento del sistema. Además, la separación por capas fortalece la seguridad, ya que los usuarios nunca acceden directamente a la base de datos, sino a través de procesos validados por la lógica del sistema.

Capa de Presentación: La capa de presentación corresponde a la interfaz gráfica con la que interactúan los usuarios del sistema mediante un navegador web. Está compuesta por las vistas, controladores y elementos de interacción desarrollados en ASP.NET MVC, encargados de mostrar formularios, consultas, reportes y módulos administrativos. Su función principal consiste en recibir las solicitudes del usuario, validar datos básicos y enviarlos hacia la capa de negocio para su procesamiento. Esta capa prioriza la usabilidad y accesibilidad, permitiendo operar el sistema sin requerir conocimientos técnicos avanzados.

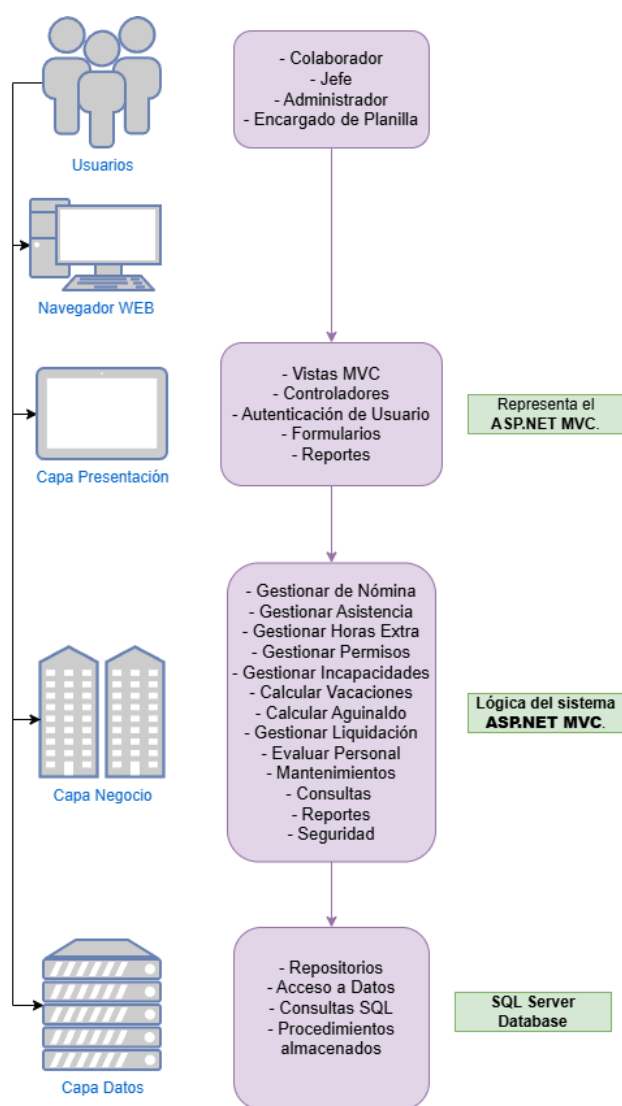
Capa de Negocio: La capa de negocio contiene la lógica funcional del sistema y representa el núcleo operativo de la aplicación. En este nivel se implementan las reglas relacionadas con asistencia, cálculo de nómina, gestión de permisos, control de horas extra, evaluaciones y generación de reportes administrativos. Los controladores y servicios procesan la información recibida desde la presentación, aplican validaciones y ejecutan los cálculos necesarios antes de interactuar con la base de datos. Esta separación garantiza que las reglas organizacionales permanezcan centralizadas y puedan modificarse sin afectar directamente la interfaz del sistema.

Capa de Datos: La capa de datos se encarga del acceso, almacenamiento y recuperación de la información del sistema utilizando SQL Server Management Studio 20 como herramienta de

administración del motor de base de datos. En esta capa se gestionan las consultas SQL, procedimientos almacenados y operaciones de persistencia necesarias para mantener la integridad de los registros. Su implementación permite controlar transacciones, asegurar consistencia en los datos y mantener un historial confiable de las operaciones realizadas dentro del sistema de planilla.

La Figura 18 Arquitectura del software del sistema presenta la vista interna del prototipo, mostrando la distribución de las capas y su interacción durante la ejecución de las operaciones principales. El diagrama permite visualizar el flujo de información desde la interacción del usuario hasta el almacenamiento de los datos, evidenciando la estructura técnica implementada durante el desarrollo del sistema de gestión y pago de planilla.

Figura 18
Diagrama Arquitectura del software



Fuente: Elaboración propia.

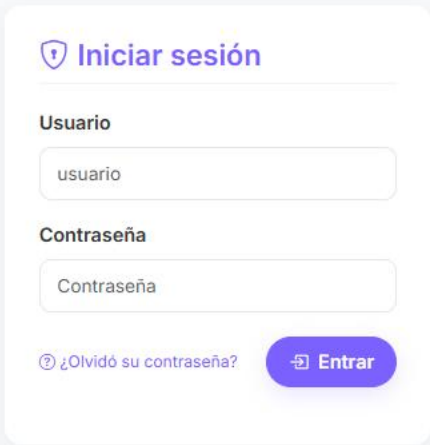
Diseño de entradas

El diseño de entradas del sistema constituye un componente esencial dentro del desarrollo del prototipo, ya que establece la forma en que los usuarios interactúan con la aplicación para registrar, consultar y actualizar la información requerida por los procesos de gestión de planilla. Esta etapa se orienta a definir interfaces funcionales que faciliten la captura correcta de datos, minimicen errores operativos y mejoren la eficiencia en la ejecución de las tareas administrativas. Las pantallas fueron diseñadas bajo criterios de usabilidad, consistencia visual y simplicidad operativa, manteniendo coherencia con los requerimientos funcionales identificados durante las fases de análisis y diseño del sistema.

Las interfaces implementadas responden directamente a los casos de uso definidos, permitiendo representar procesos clave como la administración de colaboradores, control de asistencia, gestión de permisos, cálculo de nómina y generación de información administrativa. Cada vista presenta únicamente los elementos necesarios para la actividad del usuario, evitando la sobrecarga visual y promoviendo una interacción clara dentro del entorno web desarrollado mediante ASP.NET MVC (.NET Framework). Por su parte, se incorporan mecanismos de validación de datos que contribuyen a garantizar la integridad de la información antes de su procesamiento en la capa de negocio.

En las subsecciones siguientes se presentan las pantallas representativas del prototipo, seleccionadas según su relevancia dentro del flujo operativo del sistema. Estas interfaces se documentan como figuras conforme a las normas APA 7, evidenciando la implementación práctica del diseño propuesto y mostrando la integración entre las entradas de información, la arquitectura del software y los procesos administrativos asociados a la gestión y pago de planilla.

La Figura 19 presenta la arquitectura general del sistema de planilla, mostrando la organización de sus componentes y la interacción entre ellos. Se observa una estructura por capas que separa la presentación, la lógica de negocio y el acceso a datos, facilitando el mantenimiento y la escalabilidad del sistema. Esta distribución permite gestionar adecuadamente los módulos funcionales y transversales definidos, garantizando control, seguridad y coherencia en el procesamiento de la información relacionada con la gestión de nómina.

Figura 19*Diseño de entrada - Pantalla Inicio de Sesión*The image shows a login form centered on a light blue background. The form is a white rounded rectangle with a subtle shadow. At the top, it has a purple shield icon followed by the text 'Iniciar sesión' in purple. Below this is a horizontal line. The form contains two input fields: the first is labeled 'Usuario' and contains the placeholder text 'usuario'; the second is labeled 'Contraseña' and contains the placeholder text 'Contraseña'. At the bottom left of the form, there is a link that says '¿Olvidó su contraseña?' with a question mark icon. To the right of this link is a purple button with a white arrow icon and the text 'Entrar'.

Fuente: Elaboración propia.

La Figura 20 muestra el diagrama general de casos de uso del sistema de planilla, donde se identifican los actores principales y las funcionalidades disponibles. El administrador interactúa con los módulos de mantenimiento, seguridad, consultas y generación de reportes, mientras que el sistema centraliza la gestión de procesos operativos relacionados con la nómina. Este modelo permite visualizar el alcance funcional del sistema y comprender cómo los distintos procesos apoyan la administración integral del personal y sus pagos.

Figura 20*Diseño de entrada - Pantalla Solicitar Vacaciones*

+ Nueva Solicitud de Vacaciones
Registre el período de vacaciones para el colaborador.

Métrica	Valor
Días acumulados	2,3
Días disfrutados	1,0
Días pendientes	0,0
Días disponibles	1,3

Colaborador: Rachel Perez Camacho

Fecha inicio: 07/07/2026

Fecha fin: 07/07/2026

Observaciones: Motivo u observaciones

1 día(s) hábil(es) del 7/7/2026 al 7/7/2026

[Volver](#) [Registrar solicitud](#)

Fuente: Elaboración propia.

La Figura 21 presenta el diagrama de despliegue del sistema de planilla, donde se identifican los nodos físicos y el entorno tecnológico necesarios para su funcionamiento. Se observa la interacción entre el servidor de aplicaciones, la base de datos y los dispositivos cliente utilizados por los usuarios del sistema. Este modelo permite comprender la distribución del software dentro de la infraestructura tecnológica, garantizando conectividad, disponibilidad y un adecuado flujo de información durante la ejecución de los procesos relacionados con la gestión de planillas.

Figura 21
Diseño de entrada - Nueva Evaluación de Desempeño

☆ **Nueva Evaluación de Desempeño**
Puntaje de 1 a 10 por criterio — la nota final es el promedio

Empleado Rachel Perez Camacho ▾ **Año** 2026 **Semestre** 2 — Julio a Diciem ▾ **Fecha evaluacion** 07/07/2026 📅

Evaludador Asistente

Criterios de evaluacion (1 = Minimo 10 = Maximo)

Puntualidad y asistencia	Calidad del trabajo	Trabajo en equipo
10 / 10	10 / 10	10 / 10
Iniciativa y proactividad	Cumplimiento de metas	
10 / 10	10 / 10	

Nota final promedio
10.00
Excelente

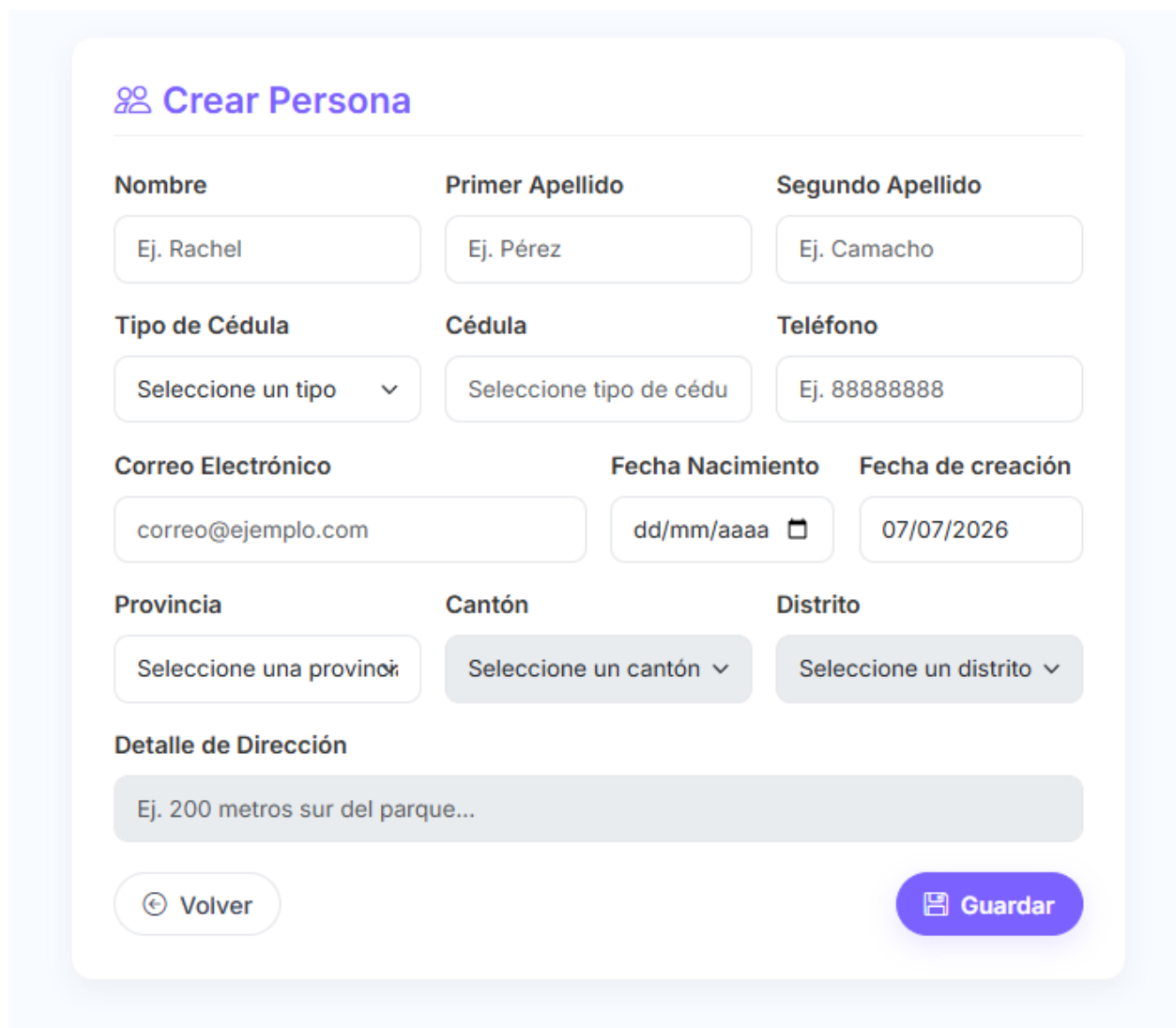
Comentarios
Observaciones generales de la evaluación

[Guardar evaluación](#) [Volver](#)

Fuente: Elaboración propia.

La Figura 22 muestra el formulario utilizado para registrar una nueva persona dentro del sistema de nómina. En esta interfaz se capturan los datos personales básicos necesarios para la identificación del empleado, incluyendo nombre completo, tipo y número de cédula, información de contacto y fecha de nacimiento. Además, el sistema permite registrar la ubicación geográfica mediante provincia, cantón y distrito, facilitando la estandarización de direcciones. La fecha de creación se asigna automáticamente para mantener trazabilidad del registro. Este formulario forma parte del módulo de mantenimiento, asegurando la correcta gestión y almacenamiento de la información del personal desde su incorporación al sistema.

Figura 22
Diseño de entrada - Crear Persona



El formulario 'Crear Persona' está diseñado con un encabezado que incluye un icono de personas y el título. Los campos de entrada están organizados en una cuadrícula de tres columnas. Las primeras tres columnas corresponden a los nombres y apellidos. Las siguientes tres columnas corresponden a la información de identificación y contacto. Las últimas tres columnas corresponden a la información geográfica. El formulario incluye botones para 'Volver' y 'Guardar'.

Nombre	Primer Apellido	Segundo Apellido	Tipo de Cédula	Cédula	Teléfono	Correo Electrónico	Fecha Nacimiento	Fecha de creación	Provincia	Cantón	Distrito	Detalle de Dirección
Ej. Rachel	Ej. Pérez	Ej. Camacho	Seleccione un tipo ▼	Seleccione tipo de cédu	Ej. 88888888	correo@ejemplo.com	dd/mm/aaaa 📅	07/07/2026	Seleccione una provincia	Seleccione un cantón ▼	Seleccione un distrito ▼	Ej. 200 metros sur del parque...

Botones: [Volver](#) [Guardar](#)

Fuente: Elaboración propia.

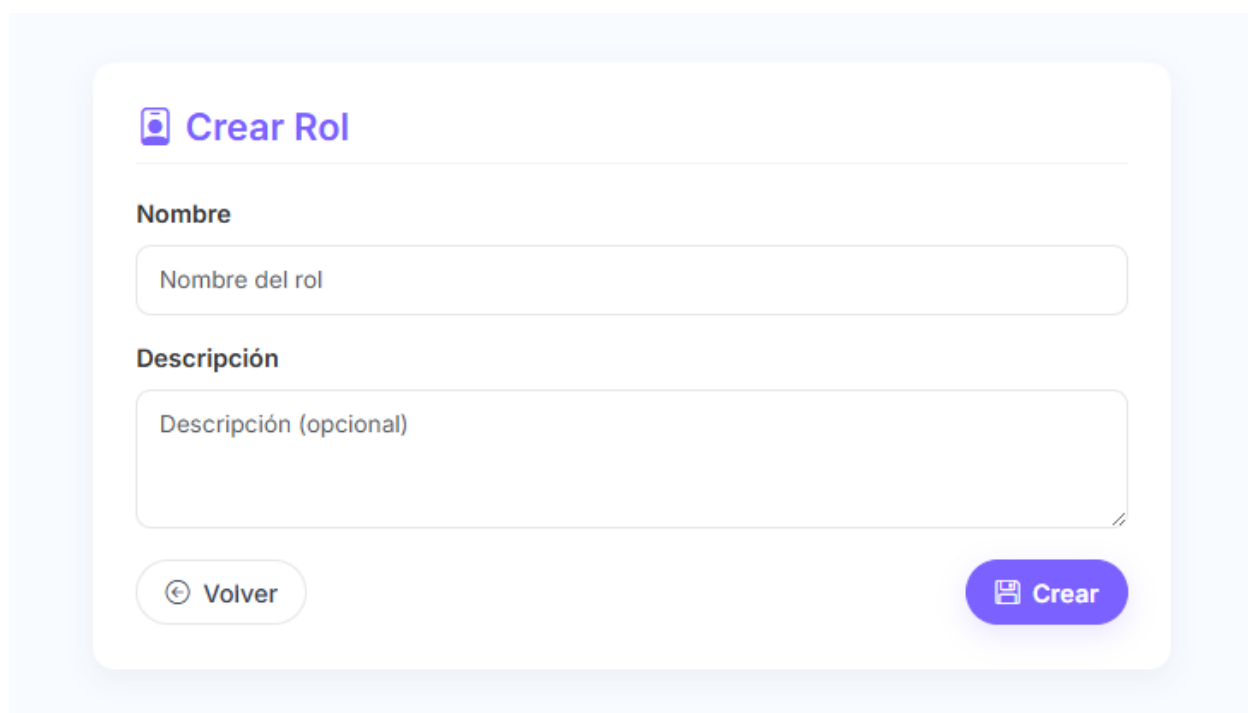
La Figura 23 presenta la interfaz correspondiente al proceso de creación de roles dentro del módulo de Seguridad del Sistema de Nómina, cuya finalidad es administrar los niveles de acceso según las funciones organizacionales definidas. En esta pantalla el usuario administrador puede registrar un nuevo rol mediante el ingreso del nombre identificador y una descripción opcional que permita documentar su alcance operativo. La correcta definición de roles constituye un elemento

fundamental para la implementación del control de accesos basado en permisos, fortaleciendo la segregación de responsabilidades dentro del sistema.

En esa misma línea, esta funcionalidad permite estructurar la asignación posterior de privilegios a los usuarios, garantizando coherencia entre los procesos del sistema y las políticas internas de seguridad. En consecuencia, el sistema asegura una gestión ordenada de autorizaciones, contribuyendo a la protección de la información y al adecuado funcionamiento de los módulos transversales de seguridad y auditoría.

Figura 23

Diseño de entrada – Crear Rol

El formulario 'Crear Rol' tiene un encabezado con un icono de documento y el título 'Crear Rol'. A continuación, hay un campo de texto etiquetado 'Nombre' con el placeholder 'Nombre del rol'. Debajo de eso, hay un campo de texto etiquetado 'Descripción' con el placeholder 'Descripción (opcional)'. En la parte inferior izquierda, hay un botón 'Volver' con un icono de flecha hacia atrás. En la parte inferior derecha, hay un botón 'Crear' con un icono de guardar y el texto 'Crear'.

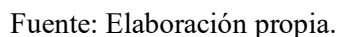
Fuente: Elaboración propia.

Diseño físico de la base de datos

El diseño físico de la base de datos representa la etapa en la cual el modelo conceptual y lógico previamente definido se transforma en una estructura implementable dentro del gestor de bases de datos seleccionado. En esta fase se establecen las tablas, tipos de datos, relaciones, claves primarias y foráneas necesarias para garantizar la integridad y consistencia de la información del sistema de planilla. Para la representación gráfica de esta estructura se utilizó MySQL Workbench,

empleada únicamente como herramienta de diagramación del modelo entidad-relación. La implementación se realizó en Microsoft SQL Server, motor sobre el cual se programó y se mantiene la base de datos del sistema, razón por la cual los tipos de dato documentados en el diccionario corresponden a la nomenclatura propia de SQL Server.

Este diseño busca asegurar un almacenamiento eficiente de la información laboral, administrativa y salarial, facilitando posteriormente los procesos de consulta, cálculo y generación de reportes del sistema. Asimismo, se aplicaron principios básicos de normalización con el objetivo de evitar redundancia de datos y mejorar el rendimiento de las operaciones realizadas sobre la base de datos. Como complemento al modelo físico, se presenta el diccionario de datos, donde se documentan detalladamente los atributos, tipos de datos y descripciones funcionales de cada tabla, permitiendo una comprensión clara de la estructura implementada y sirviendo como referencia técnica para futuras fases de desarrollo y mantenimiento.



El diccionario de datos constituye un elemento fundamental dentro del diseño físico de la base de datos, ya que documenta de manera estructurada todos los componentes que conforman el almacenamiento de la información del sistema.

En esta sección se describe formalmente la estructura de la base de datos BD_Nomina, utilizada en el Sistema de Gestión y Pago de Planilla desarrollado para la empresa AraWeb CR, permitiendo comprender cómo se organizan y relacionan los datos dentro del entorno implementado.

A continuación, se presenta el diccionario de datos generado a partir de la implementación física de la base de datos BD_Nomina en Microsoft SQL Server, donde se detallan las tablas que conforman el sistema junto con la definición de sus atributos.

Para cada entidad se especifican los campos correspondientes, tipo de dato, longitud, restricciones de nulidad, así como la identificación de llaves primarias (PK) y llaves foráneas (FK), lo que garantiza la integridad referencial entre los distintos módulos funcionales del sistema.

Junto a las llaves primarias y foráneas se documentan las restricciones de unicidad (UQ), aplicadas sobre aquellos campos o combinaciones de campos que no admiten duplicidad lógica dentro del modelo. Estas restricciones complementan la integridad referencial al impedir el registro de valores repetidos en atributos que identifican de forma no ambigua a una entidad o a un evento del dominio laboral. Su declaración en el motor de base de datos traslada la validación desde la capa de aplicación hacia el propio esquema, lo que evita inconsistencias originadas por accesos concurrentes o por operaciones ejecutadas fuera de la interfaz del sistema. Estas restricciones no constituyen por sí mismas una forma normal, pero resultan necesarias para preservar la integridad del modelo normalizado.

Este documento sirve como referencia técnica tanto para el desarrollo como para el mantenimiento futuro del sistema, ya que permite estandarizar la comprensión de la estructura de datos y facilitar la integración entre las capas del software.

Además, el diccionario de datos respalda la trazabilidad del diseño implementado, asegurando coherencia entre el modelo conceptual, lógico y físico definidos durante el proceso de construcción del prototipo.

Tabla 23*Estructura de la tabla Aguinaldo*

Campo	Tipo de Dato	Long.	Nulo	PK	FK	UQ	Descripción
idAguinaldo	INT	-	NO	SI	NO	NO	Identificador único del aguinaldo
Empleado_idEmpleado	INT	-	NO	NO	SI	SI	Referencia al empleado, Empleado
año	INT	-	NO	NO	NO	SI	Año de cálculo
monto	DECIMAL	12,2	NO	NO	NO	NO	Monto monetario del concepto
fecha_calculo	DATE	-	NO	NO	NO	NO	Fecha en que se realizó el cálculo
fecha_pago	DATE	-	NO	NO	NO	NO	Fecha programada de pago
estado	NVARCHAR	50	NO	NO	NO	NO	Estado actual del registro

Nota. Campos de la tabla Aguinaldo. PK = Llave Primaria. FK = Llave Foránea. Long. = Longitud máxima del campo. UQ = Restricción de unicidad; los campos Empleado_idEmpleado y año conforman una restricción de unicidad compuesta.

Tabla 24*Estructura de la tabla AportesSeguridadSocial*

Campo	Tipo de Dato	Long.	Nulo	PK	FK	UQ	Descripción
idAporte	INT	-	NO	SI	NO	NO	Identificador único del aporte
nombre	NVARCHAR	50	NO	NO	NO	NO	Nombre descriptivo del registro
porcentaje	DECIMAL	6,2	NO	NO	NO	NO	Porcentaje aplicable
descripcion	NVARCHAR	255	SI	NO	NO	NO	Descripción detallada
fecha_actualizacion	DATETIME2	7	NO	NO	NO	NO	Fecha de la última actualización

Nota. Campos de la tabla AportesSeguridadSocial. PK = Llave Primaria. FK = Llave Foránea. Long. = Longitud máxima del campo. UQ = Restricción de unicidad.

Tabla 25*Estructura de la tabla Asistencia*

Campo	Tipo de Dato	Long.	Nulo	PK	FK	UQ	Descripción
idAsistencia	INT	-	NO	SI	NO	NO	Identificador único de asistencia
Empleado_idEmpleado	INT	-	NO	NO	SI	SI	Referencia al empleado, Empleado
fecha	DATE	-	NO	NO	NO	SI	Fecha del evento
estado	VARCHAR	20	NO	NO	NO	NO	Estado actual del registro
es_justificada	BIT	-	NO	NO	NO	NO	Indica si la ausencia está justificada
hora_entrada	TIME	7	SI	NO	NO	NO	Hora de entrada registrada
hora_salida	TIME	7	SI	NO	NO	NO	Hora de salida registrada
horas_trabajadas	DECIMAL	5,2	NO	NO	NO	NO	Total de horas trabajadas en el día
horas_netas	DECIMAL	5,2	NO	NO	NO	NO	Horas netas efectivas trabajadas
horas_extra	DECIMAL	5,2	NO	NO	NO	NO	Horas extra generadas en el día
tiene_tardia	BIT	-	NO	NO	NO	NO	Indica si el empleado tuvo tardía
minutos_tardia	INT	-	NO	NO	NO	NO	Minutos de tardía registrados
observacion	VARCHAR	255	SI	NO	NO	NO	Observaciones o notas adicionales
fecha_registro	DATETIME	-	NO	NO	NO	NO	Fecha y hora de creación del registro
fecha_actualizacion	DATETIME	-	SI	NO	NO	NO	Fecha de la última actualización

Nota. Campos de la tabla Asistencia. PK = Llave Primaria. FK = Llave Foránea. Long. = Longitud máxima del campo. UQ = Restricción de unicidad; los campos Empleado_idEmpleado y fecha conforman una restricción de unicidad compuesta.

Tabla 26*Estructura de la tabla AuditoriaGeneral*

Campo	Tipo de Dato	Long.	Nulo	PK	FK	UQ	Descripción
idAuditoria	INT	-	NO	SI	NO	NO	Identificador único de la auditoría
tabla	NVARCHAR	100	NO	NO	NO	NO	Nombre de la tabla auditada
idRegistro	NVARCHAR	50	NO	NO	NO	NO	Identificador del registro afectado
accion	NVARCHAR	50	NO	NO	NO	NO	Tipo de acción ejecutada (INSERT/UPDATE/DELETE)
usuario	NVARCHAR	100	NO	NO	NO	NO	Nombre de usuario para autenticación
fecha	DATETIME2	7	NO	NO	NO	NO	Fecha del evento
detalle	NVARCHAR	max	SI	NO	NO	NO	Descripción detallada o señas de la dirección

Nota. Campos de la tabla AuditoriaGeneral. PK = Llave Primaria. FK = Llave Foránea. Long. = Longitud máxima del campo. UQ = Restricción de unicidad.

Tabla 27*Estructura de la tabla Ausencia*

Campo	Tipo de Dato	Long.	Nulo	PK	FK	UQ	Descripción
idAusencia	INT	-	NO	SI	NO	NO	Identificador único de la ausencia
fecha_inicio	DATE	-	NO	NO	NO	NO	Fecha de inicio del período
fecha_fin	DATE	-	NO	NO	NO	NO	Fecha de finalización del período
motivo	NVARCHAR	255	NO	NO	NO	NO	Motivo o justificación
Empleado_idEmpleado	INT	-	NO	NO	SI	NO	Referencia al empleado, Empleado

Nota. Campos de la tabla Ausencia. PK = Llave Primaria. FK = Llave Foránea. Long. = Longitud máxima del campo. UQ = Restricción de unicidad.

Tabla 28*Estructura de la tabla Bitácora*

Campo	Tipo de Dato	Long.	Nulo	PK	FK	UQ	Descripción
idBitacora	INT	-	NO	SI	NO	NO	Identificador único de la bitácora
Credencial_idCredencial	INT	-	NO	NO	SI	NO	Referencia a las credenciales del usuario, Credencial
accion	NVARCHAR	255	NO	NO	NO	NO	Tipo de acción ejecutada (INSERT/UPDATE/DELETE)
fecha	DATETIME2	7	NO	NO	NO	NO	Fecha del evento
ip_origen	NVARCHAR	50	SI	NO	NO	NO	Dirección IP desde donde se realizó la acción
descripcion	NVARCHAR	255	SI	NO	NO	NO	Descripción detallada

Nota. Campos de la tabla Bitácora. PK = Llave Primaria. FK = Llave Foránea. Long. = Longitud máxima del campo. UQ = Restricción de unicidad.

Tabla 29*Estructura de la tabla Bonificacion*

Campo	Tipo de Dato	Long.	Nulo	PK	FK	UQ	Descripción
idBonificacion	INT	-	NO	SI	NO	NO	Identificador único de la bonificación
Empleado_idEmpleado	INT	-	NO	NO	SI	NO	Referencia al empleado, Empleado
tipo	NVARCHAR	100	NO	NO	NO	NO	Tipo o categoría del concepto
monto	DECIMAL	12,2	NO	NO	NO	NO	Monto monetario del concepto
descripcion	NVARCHAR	255	SI	NO	NO	NO	Descripción detallada
fecha_asignacion	DATE	-	NO	NO	NO	NO	Fecha en que fue asignada la bonificación
estado	NVARCHAR	50	NO	NO	NO	NO	Estado actual del registro

Nota. Campos de la tabla Bonificacion. PK = Llave Primaria. FK = Llave Foránea. Long. = Longitud máxima del campo. UQ = Restricción de unicidad.

Tabla 30*Estructura de la tabla Canton*

Campo	Tipo de Dato	Long.	Nulo	PK	FK	UQ	Descripción
idCanton	INT	-	NO	SI	NO	NO	Identificador único del cantón
nombre	NVARCHAR	100	NO	NO	NO	SI	Nombre descriptivo del registro
Provincia_ idProvincia	INT	-	NO	NO	SI	SI	Referencia a la provincia, Provincia

Nota. Campos de la tabla Canton. PK = Llave Primaria. FK = Llave Foránea. Long. = Longitud máxima del campo. UQ = Restricción de unicidad; los campos Provincia_idProvincia y nombre conforman una restricción de unicidad compuesta.

Tabla 31*Estructura de la tabla ClaseHoraExtra*

Campo	Tipo de Dato	Long.	Nulo	PK	FK	UQ	Descripción
idClaseHoraExtra	INT	-	NO	SI	NO	NO	Identificador único de la clase de hora extra
nombre	NVARCHAR	100	NO	NO	NO	NO	Nombre descriptivo del registro
porcentaje	DECIMAL	5,2	NO	NO	NO	NO	Porcentaje aplicable
descripcion	NVARCHAR	255	SI	NO	NO	NO	Descripción detallada

Nota. Campos de la tabla ClaseHoraExtra. PK = Llave Primaria. FK = Llave Foránea. Long. = Longitud máxima del campo. UQ = Restricción de unicidad.

Tabla 32*Estructura de la tabla Credencial*

Campo	Tipo de Dato	Long.	Nulo	PK	FK	UQ	Descripción
idCredencial	INT	-	NO	SI	NO	NO	Identificador único de credencial
usuario	NVARCHAR	50	NO	NO	NO	SI	Nombre de usuario para autenticación
contrasena	NVARCHAR	255	NO	NO	NO	NO	Contraseña cifrada del usuario
activo	BIT	-	NO	NO	NO	NO	Indica si el registro está activo
fecha_creacion	DATETIME2	7	NO	NO	NO	NO	Fecha de creación del registro
Rol_idRol	INT	-	NO	NO	SI	NO	Referencia al rol asignado, Rol
requiere_cambio	BIT	-	NO	NO	NO	NO	Indica si se requiere cambio de contraseña
fecha_ultimo_cambio	DATETIME2	7	SI	NO	NO	NO	Fecha del último cambio de contraseña

Nota. Campos de la tabla Credencial. PK = Llave Primaria. FK = Llave Foránea. Long. = Longitud máxima del campo. UQ = Restricción de unicidad; el campo usuario no admite valores duplicados.

Tabla 33*Estructura de la tabla Departamento*

Campo	Tipo de Dato	Long.	Nulo	PK	FK	UQ	Descripción
idDepartamento	INT	-	NO	SI	NO	NO	Identificador único del departamento
nombre	NVARCHAR	100	NO	NO	NO	NO	Nombre descriptivo del registro
descripcion	NVARCHAR	255	NO	NO	NO	NO	Descripción detallada

Nota. Campos de la tabla Departamento. PK = Llave Primaria. FK = Llave Foránea. Long. = Longitud máxima del campo. UQ = Restricción de unicidad.

Tabla 34*Estructura de la tabla DetalleAporte*

Campo	Tipo de Dato	Long.	Nulo	PK	FK	UQ	Descripción
idDetalleAporte	INT	-	NO	SI	NO	NO	Identificador único del detalle de aporte
Aporte_idAporte	INT	-	NO	NO	SI	NO	Referencia al aporte de seguridad social. AportesSeguridadSocial
DetalleNomina_ idDetalleNomina	INT	-	NO	NO	SI	NO	Referencia al detalle de nómina -> DetalleNomina
monto	DECIMAL	12,2	NO	NO	NO	NO	Monto monetario del concepto

Nota. Campos de la tabla DetalleAporte. PK = Llave Primaria. FK = Llave Foránea. Long. = Longitud máxima del campo. UQ = Restricción de unicidad.

Tabla 35*Estructura de la tabla DetalleNomina*

Campo	Tipo de Dato	Long.	Nulo	PK	FK	UQ	Descripción
idDetalleNomina	INT	-	NO	SI	NO	NO	Identificador único del detalle de nómina
Nomina_idNomina	INT	-	NO	NO	SI	NO	Referencia a la nómina, Nomina
Empleado_idEmpleado	INT	-	NO	NO	SI	NO	Referencia al empleado, Empleado
SalarioBrutoMensual_ idSalarioBrutoMensual	INT	-	NO	NO	SI	NO	Referencia al salario bruto mensual, SalarioBrutoMensual
DetalleRenta_idDetalleRenta	INT	-	SI	NO	SI	NO	Referencia al detalle de renta, DetalleRenta
total_bonificaciones	DECIMAL	12,2	NO	NO	NO	NO	Sumatoria de bonificaciones del período
total_horas_extra	DECIMAL	12,2	NO	NO	NO	NO	Sumatoria de horas extra del período
total_deducciones	DECIMAL	12,2	NO	NO	NO	NO	Sumatoria de deducciones del período
total_percepciones	DECIMAL	12,2	NO	NO	NO	NO	Sumatoria de percepciones del período
salario_netto	DECIMAL	12,2	NO	NO	NO	NO	Salario neto a pagar al empleado
fecha_calculo	DATETIME2	7	NO	NO	NO	NO	Fecha en que se realizó el cálculo

Nota. Campos de la tabla *DetalleNomina*. PK = Llave Primaria. FK = Llave Foránea.
Long. = Longitud máxima del campo. UQ = Restricción de unicidad.

Tabla 36*Estructura de la tabla DetalleRenta*

Campo	Tipo de Dato	Long.	Nulo	PK	FK	UQ	Descripción
idDetalleRenta	INT	-	NO	SI	NO	NO	Identificador único del detalle de renta
TramosParametrosRenta_ idTramosParametrosRenta	INT	-	NO	NO	SI	NO	Referencia al tramo de renta aplicado, TramosParametrosRenta
base_calculo	DECIMAL	12,2	NO	NO	NO	NO	Base utilizada para el cálculo
monto_impuesto	DECIMAL	12,2	NO	NO	NO	NO	Monto del impuesto calculado
observacion	NVARCHAR	255	SI	NO	NO	NO	Observaciones o notas adicionales

Nota. Campos de la tabla DetalleRenta. PK = Llave Primaria. FK = Llave Foránea. Long. = Longitud máxima del campo. UQ = Restricción de unicidad.

Tabla 37*Estructura de la tabla Direccion*

Campo	Tipo de Dato	Long.	Nulo	PK	FK	UQ	Descripción
idDireccion	INT	-	NO	SI	NO	NO	Identificador único de dirección
detalle	NVARCHAR	255	NO	NO	NO	NO	Descripción detallada o señas de la dirección
Distrito_idDistrito	INT	-	NO	NO	SI	NO	Referencia al distrito, Distrito

Nota. Campos de la tabla Direccion. PK = Llave Primaria. FK = Llave Foránea. Long. = Longitud máxima del campo. UQ = Restricción de unicidad.

Tabla 38*Estructura de la tabla Distrito*

Campo	Tipo de Dato	Long.	Nulo	PK	FK	UQ	Descripción
idDistrito	INT	-	NO	SI	NO	NO	Identificador único del distrito
nombre	NVARCHAR	100	NO	NO	NO	SI	Nombre descriptivo del registro
Canton_idCanton	INT	-	NO	NO	SI	SI	Referencia al cantón, Canton

Nota. Campos de la tabla Distrito. PK = Llave Primaria. FK = Llave Foránea. Long. = Longitud máxima del campo. UQ = Restricción de unicidad; los campos Canton_idCanton y nombre conforman una restricción de unicidad compuesta.

Tabla 39*Estructura de la tabla Empleado*

Campo	Tipo de Dato	Long.	Nulo	PK	FK	UQ	Descripción
idEmpleado	INT	-	NO	SI	NO	NO	Identificador único del empleado
Persona_idPersona	INT	-	NO	NO	SI	NO	Referencia a los datos personales. Persona
Credencial_idCredencial	INT	-	NO	NO	SI	NO	Referencia a las credenciales del usuario, Credencial
Puesto_idPuesto	INT	-	NO	NO	SI	NO	Referencia al puesto del empleado, Puesto
Jornada_idJornada	INT	-	NO	NO	SI	NO	Referencia a la jornada laboral, Jornada
Horario_idHorario	INT	-	SI	NO	SI	NO	Referencia al horario asignado, Horario
fecha_ingreso	DATE	-	NO	NO	NO	NO	Fecha de ingreso a la empresa
fecha_salida	DATE	-	SI	NO	NO	NO	Fecha de salida o retiro del empleado
estado	NVARCHAR	50	NO	NO	NO	NO	Estado actual del registro
Direccion_idDireccion	INT	-	SI	NO	SI	NO	Referencia a la dirección, Dirección
HorarioSemanal_idHorarioSemanal	INT	-	SI	NO	SI	NO	Referencia al horario semanal, HorarioSemanal

Nota. Campos de la tabla Empleado. PK = Llave Primaria. FK = Llave Foránea. Long. = Longitud máxima del campo. UQ = Restricción de unicidad.

Tabla 40*Estructura de la tabla EmpleadoHorario*

Campo	Tipo de Dato	Long.	Nulo	PK	FK	UQ	Descripción
idEmpleadoHorario	INT	-	NO	SI	NO	NO	Identificador único de asignación de horario
Empleado_idEmpleado	INT	-	NO	NO	SI	SI	Referencia al empleado, Empleado
HorarioSemanal_idHorarioSemanal	INT	-	NO	NO	SI	SI	Referencia al horario semanal, HorarioSemanal
fecha_inicio	DATE	-	NO	NO	NO	SI	Fecha de inicio del período
fecha_fin	DATE	-	SI	NO	NO	NO	Fecha de finalización del período
fecha_asignacion	DATETIME2	7	NO	NO	NO	NO	Fecha en que fue asignada la bonificación

Nota. Campos de la tabla EmpleadoHorario. PK = Llave Primaria. FK = Llave Foránea. Long. = Longitud máxima del campo. UQ = Restricción de unicidad; los campos Empleado_idEmpleado, HorarioSemanal_idHorarioSemanal y fecha_inicio conforman una restricción de unicidad compuesta.

Tabla 41*Estructura de la tabla Feriado*

Campo	Tipo de Dato	Long.	Nulo	PK	FK	UQ	Descripción
idFeriado	INT	-	NO	SI	NO	NO	Identificador único del feriado
nombre	NVARCHAR	100	NO	NO	NO	NO	Nombre descriptivo del registro
fecha	DATE	-	NO	NO	NO	NO	Fecha del evento
pago_obligatorio	BIT	-	NO	NO	NO	NO	Indica si el feriado tiene pago obligatorio
Empleado_idEmpleado	INT	-	NO	NO	SI	NO	Referencia al empleado, Empleado

Nota. Campos de la tabla Feriado. PK = Llave Primaria. FK = Llave Foránea. Long. = Longitud máxima del campo. UQ = Restricción de unicidad.

Tabla 42*Estructura de la tabla FeriadoLaborado*

Campo	Tipo de Dato	Long.	Nulo	PK	FK	UQ	Descripción
idFeriadoLaborado	INT	-	NO	SI	NO	NO	Identificador único del feriado laborado
feriado_id	INT	-	NO	NO	SI	NO	Referencia al feriado, Feriado
horas	DECIMAL	5,2	NO	NO	NO	NO	Cantidad de horas del período
pago_extra	DECIMAL	12,2	SI	NO	NO	NO	Monto de pago adicional generado

Nota. Campos de la tabla FeriadoLaborado. PK = Llave Primaria. FK = Llave Foránea. Long. = Longitud máxima del campo. UQ = Restricción de unicidad.

Tabla 43*Estructura de la tabla HistorialBonificacion*

Campo	Tipo de Dato	Long.	Nulo	PK	FK	UQ	Descripción
idHistorialBonificacion	INT	-	NO	SI	NO	NO	Identificador del historial de bonificación
bonificacion_id	INT	-	NO	NO	SI	NO	Referencia a la bonificación, Bonificación
fecha	DATE	-	NO	NO	NO	NO	Fecha del evento

Nota. Campos de la tabla HistorialBonificacion. PK = Llave Primaria. FK = Llave Foránea. Long. = Longitud máxima del campo. UQ = Restricción de unicidad.

Tabla 44*Estructura de la tabla HistorialHoraExtra*

Campo	Tipo de Dato	Long.	Nulo	PK	FK	UQ	Descripción
idHistorialHoraExtra	INT	-	NO	SI	NO	NO	Identificador del historial de horas extra
HoraExtraid HoraExtra	INT	-	NO	NO	SI	NO	Referencia a la hora extra, HoraExtra
idUsuario	INT	-	SI	NO	NO	NO	Referencia al usuario que registró la acción
accion	NVARCHAR	100	NO	NO	NO	NO	Tipo de acción ejecutada (INSERT/UPDATE/DELETE)
comentario	NVARCHAR	255	SI	NO	NO	NO	Comentario adicional sobre la acción
fecha	DATETIME2	7	NO	NO	NO	NO	Fecha del evento

Nota. Campos de la tabla HistorialHoraExtra. PK = Llave Primaria. FK = Llave Foránea. Long. = Longitud máxima del campo. UQ = Restricción de unicidad.

Tabla 45*Estructura de la tabla HistorialPermiso*

Campo	Tipo de Dato	Long.	Nulo	PK	FK	UQ	Descripción
idHistorialPermiso	INT	-	NO	SI	NO	NO	Identificador del historial de permiso
Permiso_idPermiso	INT	-	NO	NO	SI	NO	Referencia al permiso, Permiso
idUsuario	INT	-	NO	NO	NO	NO	Referencia al usuario que registró la acción
accion	NVARCHAR	100	NO	NO	NO	NO	Tipo de acción ejecutada (INSERT/UPDATE/DELETE)
comentario	NVARCHAR	255	SI	NO	NO	NO	Comentario adicional sobre la acción
fecha	DATETIME2	7	NO	NO	NO	NO	Fecha del evento

Nota. Campos de la tabla HistorialPermiso. PK = Llave Primaria. FK = Llave Foránea. Long. = Longitud máxima del campo. UQ = Restricción de unicidad.

Tabla 46*Estructura de la tabla HistorialVacaciones*

Campo	Tipo de Dato	Long.	Nulo	PK	FK	UQ	Descripción
idHistorialVacaciones	INT	-	NO	SI	NO	NO	Identificador del historial de vacaciones
Vacaciones_id Vacaciones	INT	-	NO	NO	SI	NO	Referencia a las vacaciones, Vacaciones
idUsuario	INT	-	SI	NO	NO	NO	Referencia al usuario que registró la acción
accion	NVARCHAR	100	NO	NO	NO	NO	Tipo de acción ejecutada (INSERT/UPDATE/DELETE)
comentario	NVARCHAR	255	SI	NO	NO	NO	Comentario adicional sobre la acción
fecha	DATETIME2	7	NO	NO	NO	NO	Fecha del evento

Nota. Campos de la tabla HistorialVacaciones. PK = Llave Primaria. FK = Llave Foránea. Long. = Longitud máxima del campo. UQ = Restricción de unicidad.

Tabla 47*Estructura de la tabla HoraExtra*

Campo	Tipo de Dato	Long.	Nulo	PK	FK	UQ	Descripción
idHoraExtra	INT	-	NO	SI	NO	NO	Identificador único de hora extra
Empleado_id Empleado	INT	-	NO	NO	SI	NO	Referencia al empleado, Empleado
ClaseHoraExtra_id ClaseHoraExtra	INT	-	NO	NO	SI	NO	Referencia a la clase de hora extra, ClaseHoraExtra
fecha	DATE	-	NO	NO	NO	NO	Fecha del evento
cantidad_horas	DECIMAL	5,2	NO	NO	NO	NO	Cantidad de horas del concepto
aprobado	BIT	-	NO	NO	NO	NO	Indica si fue aprobado

Nota. Campos de la tabla HoraExtra. PK = Llave Primaria. FK = Llave Foránea. Long. = Longitud máxima del campo. UQ = Restricción de unicidad.

Tabla 48

Estructura de la tabla Horario

Campo	Tipo de Dato	Long.	Nulo	PK	FK	UQ	Descripción
idHorario	INT	-	NO	SI	NO	NO	Identificador único del horario
dia_semana	NVARCHAR	50	NO	NO	NO	NO	Día de la semana (Lunes, Martes, etc.)
hora_inicio	TIME	7	NO	NO	NO	NO	Hora de inicio de la jornada
hora_fin	TIME	7	NO	NO	NO	NO	Hora de finalización de la jornada
HorarioSemanal_ idHorarioSemanal	INT	-	SI	NO	SI	NO	Referencia al horario semanal, HorarioSemanal

Nota. Campos de la tabla Horario. PK = Llave Primaria. FK = Llave Foránea. Long. = Longitud máxima del campo. UQ = Restricción de unicidad.

Tabla 49

Estructura de la tabla HorarioSemanal

Campo	Tipo de Dato	Long.	Nulo	PK	FK	UQ	Descripción
idHorarioSemanal	INT	-	NO	SI	NO	NO	Identificador único del horario semanal
nombre	NVARCHAR	150	NO	NO	NO	NO	Nombre descriptivo del registro
descripcion	NVARCHAR	255	SI	NO	NO	NO	Descripción detallada
Jornada_idJornada	INT	-	SI	NO	SI	NO	Referencia a la jornada laboral, Jornada
fecha_creacion	DATETIME2	7	NO	NO	NO	NO	Fecha de creación del registro

Nota. Campos de la tabla HorarioSemanal. PK = Llave Primaria. FK = Llave Foránea. Long. = Longitud máxima del campo. UQ = Restricción de unicidad.

Tabla 50*Estructura de la tabla Incapacidad*

Campo	Tipo de Dato	Long.	Nulo	PK	FK	UQ	Descripción
idIncapacidad	INT	-	NO	SI	NO	NO	Identificador único de incapacidad
Empleado_ idEmpleado	INT	-	NO	NO	SI	NO	Referencia al empleado, Empleado
TipoIncapacidad_ idTipoIncapacidad	INT	-	NO	NO	SI	NO	Referencia al tipo de incapacidad, TipoIncapacidad
entidad	NVARCHAR	100	NO	NO	NO	NO	Entidad emisora de la incapacidad (CCSS, INS, etc.)
fecha_inicio	DATE	-	NO	NO	NO	NO	Fecha de inicio del período
fecha_fin	DATE	-	NO	NO	NO	NO	Fecha de finalización del período
porcentaje_pago	DECIMAL	5,2	NO	NO	NO	NO	Porcentaje del salario pagado durante la incapacidad
descripcion	NVARCHAR	255	SI	NO	NO	NO	Descripción detallada
estado	NVARCHAR	50	NO	NO	NO	NO	Estado actual del registro
numero_boleta	NVARCHAR	50	SI	NO	NO	NO	Número de boleta de la incapacidad
fecha_emision	DATE	-	SI	NO	NO	NO	Fecha de emisión de la incapacidad
dias_incapacidad	INT	-	SI	NO	NO	NO	Días de duración de la incapacidad
origen	NVARCHAR	50	NO	NO	NO	NO	Origen o fuente del registro
es_prorroga	BIT	-	NO	NO	NO	NO	Indica si es una prórroga de incapacidad anterior
incapacidad_origen_id	INT	-	SI	NO	SI	NO	Referencia a la incapacidad original (si es prórroga), Incapacidad
afecta_asistencia	BIT	-	NO	NO	NO	NO	Indica si afecta el registro de asistencia

Campo	Tipo de Dato	Long.	Nulo	PK	FK	UQ	Descripción
observaciones_rrhh	NVARCHAR	255	SI	NO	NO	NO	Observaciones del departamento de RRHH
fecha_registro	DATETIME2	7	NO	NO	O	O	Fecha y hora de creación del registro
fecha_actualizacion	DATETIME2	7	SI	NO	NO	NO	Fecha de la última actualización

Nota. Campos de la tabla Incapacidad. PK = Llave Primaria. FK = Llave Foránea. Long. = Longitud máxima del campo. UQ = Restricción de unicidad.

Tabla 51

Estructura de la tabla Jornada

Campo	Tipo de Dato	Long.	Nulo	PK	FK	UQ	Descripción
idJornada	INT	-	NO	SI	NO	NO	Identificador único de la jornada
nombre	NVARCHAR	100	NO	NO	NO	NO	Nombre descriptivo del registro
horas_semanales	INT	-	NO	NO	NO	NO	Total de horas de trabajo por semana
descripcion	NVARCHAR	255	NO	NO	NO	NO	Descripción detallada

Nota. Campos de la tabla Jornada. PK = Llave Primaria. FK = Llave Foránea. Long. = Longitud máxima del campo. UQ = Restricción de unicidad.

Tabla 52

Estructura de la tabla Liquidacion

Campo	Tipo de Dato	Long.	Nulo	PK	FK	UQ	Descripción
idLiquidacion	INT	-	NO	SI	NO	NO	Identificador único de la liquidación
Empleado_idEmpleado	INT	-	NO	NO	SI	NO	Referencia al empleado, Empleado

Campo	Tipo de Dato	Long.	Nulo	PK	FK	UQ	Descripción
TipoLiquidacion_ idTipoLiquidacion	INT	-	NO	NO	SI	NO	Referencia al tipo de liquidación, TipoLiquidacion
fecha	DATE	-	NO	NO	NO	NO	Fecha del evento
total_percepciones	DECIMAL	12,2	NO	NO	NO	NO	Sumatoria de percepciones del período
total_deducciones	DECIMAL	12,2	NO	NO	NO	NO	Sumatoria de deducciones del período
neto_pagar	DECIMAL	12,2	NO	NO	NO	NO	Monto neto a pagar en la liquidación
observaciones	NVARCHAR	255	SI	NO	NO	NO	Observaciones o notas del proceso

Nota. Campos de la tabla Liquidacion. PK = Llave Primaria. FK = Llave Foránea. Long. = Longitud máxima del campo. UQ = Restricción de unicidad.

Tabla 53

Estructura de la tabla Nomina

Campo	Tipo de Dato	Long.	Nulo	PK	FK	UQ	Descripción
idNomina	INT	-	NO	SI	NO	NO	Identificador único de la nómina
mes	TINYINT	-	NO	NO	NO	NO	Mes del período de nómina
anio	INT	-	NO	NO	NO	NO	Año de cálculo
estado	NVARCHAR	50	NO	NO	NO	NO	Estado actual del registro
fecha_creacion	DATETIME2	7	NO	NO	NO	NO	Fecha de creación del registro
fecha_cierre	DATETIME2	7	SI	NO	NO	NO	Fecha de cierre y aprobación de la nómina

Nota. Campos de la tabla Nomina. PK = Llave Primaria. FK = Llave Foránea. Long. = Longitud máxima del campo. UQ = Restricción de unicidad.

Tabla 54

Estructura de la tabla Permiso

Campo	Tipo de Dato	Long.	Nulo	PK	FK	UQ	Descripción
idPermiso	INT	-	NO	SI	NO	NO	Identificador único del permiso
Empleado_idEmpleado	INT	-	NO	NO	SI	NO	Referencia al empleado, Empleado
TipoPermiso_id	INT	-	NO	NO	SI	NO	Referencia al tipo de permiso, TipoPermiso

Campo	Tipo de Dato	Long.	Nulo	PK	FK	UQ	Descripción
TipoPermiso							
fecha_inicio	DATE	-	NO	NO	NO	NO	Fecha de inicio del período
fecha_fin	DATE	-	NO	NO	NO	NO	Fecha de finalización del período
motivo	NVARCHAR	255	NO	NO	NO	NO	Motivo o justificación
estado	NVARCHAR	50	NO	NO	NO	NO	Estado actual del registro
fecha_registro	DATETIME2	7	NO	NO	NO	NO	Fecha y hora de creación del registro
fecha_actualizacion	DATETIME2	7	SI	NO	NO	NO	Fecha de la última actualización

Nota. Campos de la tabla Permiso. PK = Llave Primaria. FK = Llave Foránea. Long. = Longitud máxima del campo. UQ = Restricción de unicidad.

Tabla 55

Estructura de la tabla Persona

Campo	Tipo de Dato	Long.	Nulo	PK	FK	UQ	Descripción
idPersona	INT	-	NO	SI	NO	NO	Identificador único de la persona
nombre	NVARCHAR	100	NO	NO	NO	NO	Nombre descriptivo del registro
apellido1	NVARCHAR	100	NO	NO	NO	NO	Primer apellido
apellido2	NVARCHAR	100	NO	NO	NO	NO	Segundo apellido
cedula	NVARCHAR	20	NO	NO	NO	NO	Número de cédula de identidad
telefono	NVARCHAR	20	NO	NO	NO	NO	Número de teléfono de contacto
correo	NVARCHAR	150	NO	NO	NO	NO	Dirección de correo electrónico
fecha_nacimiento	DATE	-	NO	NO	NO	NO	Fecha de nacimiento
fecha_creacion	DATE	-	NO	NO	NO	NO	Fecha de creación del registro
Cedula_id Cedula	INT	-	NO	NO	SI	NO	Referencia al tipo de cédula, TipoCedula
Direccion_id Direccion	INT	-	SI	NO	SI	NO	Referencia a la dirección, Direccion

Nota. Campos de la tabla Persona. PK = Llave Primaria. FK = Llave Foránea. Long. = Longitud máxima del campo. UQ = Restricción de unicidad.

Tabla 56*Estructura de la tabla Provincia*

Campo	Tipo de Dato	Long.	Nulo	PK	FK	UQ	Descripción
idProvincia	INT	-	NO	SI	NO	NO	Identificador único de la provincia
nombre	NVARCHAR	100	NO	NO	NO	SI	Nombre descriptivo del registro

Nota. Campos de la tabla Provincia. PK = Llave Primaria. FK = Llave Foránea. Long. = Longitud máxima del campo. UQ = Restricción de unicidad; el campo nombre no admite valores duplicados.

Tabla 57*Estructura de la tabla Puesto*

Campo	Tipo de Dato	Long.	Nulo	PK	FK	UQ	Descripción
idPuesto	INT	-	NO	SI	NO	NO	Identificador único del puesto
nombre	NVARCHAR	150	NO	NO	NO	NO	Nombre descriptivo del registro
planilla	BIT	-	NO	NO	NO	NO	Indica si el puesto aplica a planilla
salario_base	DECIMAL	12,2	NO	NO	NO	NO	Salario base mensual del puesto
salario_por_hora	DECIMAL	12,4	NO	NO	NO	NO	Valor del salario por hora
estado	NVARCHAR	50	NO	NO	NO	NO	Estado actual del registro
fecha_inicio	DATE	-	NO	NO	NO	NO	Fecha de inicio del período
fecha_fin	DATE	-	NO	NO	NO	NO	Fecha de finalización del período
Departamento_id Departamento	INT	-	NO	NO	SI	NO	Referencia al departamento, Departamento

Nota. Campos de la tabla Puesto. PK = Llave Primaria. FK = Llave Foránea. Long. = Longitud máxima del campo. UQ = Restricción de unicidad.

Tabla 58*Estructura de la tabla RegistroMarcacion*

Campo	Tipo de Dato	Long.	Nulo	PK	FK	UQ	Descripción
idRegistroMarcacion	INT	-	NO	SI	NO	NO	Identificador único del registro de marcación
Empleado_idEmpleado	INT	-	NO	NO	SI	NO	Referencia al empleado, Empleado
fecha	DATE	-	NO	NO	NO	NO	Fecha del evento
hora	TIME	7	NO	NO	NO	NO	Hora exacta de la marcación
tipo	NVARCHAR	10	NO	NO	NO	NO	Tipo o categoría del concepto
origen	NVARCHAR	50	SI	NO	NO	NO	Origen o fuente del registro
observacion	NVARCHAR	255	SI	NO	NO	NO	Observaciones o notas adicionales
fecha_registro	DATETIME2	7	NO	NO	NO	NO	Fecha y hora de creación del registro

Nota. Campos de la tabla RegistroMarcacion. PK = Llave Primaria. FK = Llave Foránea. Long. = Longitud máxima del campo. UQ = Restricción de unicidad.

Tabla 59*Estructura de la tabla Rol*

Campo	Tipo de Dato	Long.	Nulo	PK	FK	UQ	Descripción
idRol	INT	-	NO	SI	NO	NO	Identificador único del rol
nombre	NVARCHAR	100	NO	NO	NO	SI	Nombre descriptivo del registro
descripcion	NVARCHAR	255	SI	NO	NO	NO	Descripción detallada

Nota. Campos de la tabla Rol. PK = Llave Primaria. FK = Llave Foránea. Long. = Longitud máxima del campo. UQ = Restricción de unicidad; el campo nombre no admite valores duplicados.

Tabla 60*Estructura de la tabla SalarioBrutoMensual*

Campo	Tipo de Dato	Long.	Nulo	PK	FK	UQ	Descripción
idSalarioBrutoMensual	INT	-	NO	SI	NO	NO	Identificador único del salario bruto mensual
Empleado_idEmpleado	INT	-	NO	NO	SI	NO	Referencia al empleado, Empleado
mes	TINYINT	-	NO	NO	NO	NO	Mes del período de nómina
año	INT	-	NO	NO	NO	NO	Año de cálculo
salario_bruto	DECIMAL	12,2	NO	NO	NO	NO	Monto del salario bruto del período
fecha_calculo	DATETIME2	7	NO	NO	NO	NO	Fecha en que se realizó el cálculo

Nota. Campos de la tabla SalarioBrutoMensual. PK = Llave Primaria. FK = Llave Foránea. Long. = Longitud máxima del campo. UQ = Restricción de unicidad.

Tabla 61*Estructura de la tabla SesionUsuario*

Campo	Tipo de Dato	Long.	Nulo	PK	FK	UQ	Descripción
idSesion	INT	-	NO	SI	NO	NO	Identificador único de la sesión
Credencial_ idCredencial	INT	-	NO	NO	SI	NO	Referencia a las credenciales del usuario, Credencial
fecha_inicio	DATETIME2	7	NO	NO	NO	NO	Fecha de inicio del período
fecha_fin	DATETIME2	7	SI	NO	NO	NO	Fecha de finalización del período
ip	NVARCHAR	45	SI	NO	NO	NO	Dirección IP del dispositivo
dispositivo	NVARCHAR	100	SI	NO	NO	NO	Identificación del dispositivo utilizado
activo	BIT	-	NO	NO	NO	NO	Indica si el registro está activo

Nota. Campos de la tabla SesionUsuario. PK = Llave Primaria. FK = Llave Foránea. Long. = Longitud máxima del campo. UQ = Restricción de unicidad.

Tabla 62*Estructura de la tabla SolicitudHorasExtra*

Campo	Tipo de Dato	Long.	Nulo	PK	FK	UQ	Descripción
idSolicitudHorasExtra	INT	-	NO	SI	NO	NO	Identificador único de solicitud de horas extra
fecha_solicitud	DATE	-	NO	NO	NO	NO	Fecha en que se realizó la solicitud
estado	NVARCHAR	50	NO	NO	NO	NO	Estado actual del registro
aprobado_por	INT	-	SI	NO	NO	NO	Identificador del usuario que aprobó
fecha_aprobacion	DATE	-	SI	NO	NO	NO	Fecha de la aprobación
HoraExtra_idHoraExtra	INT	-	NO	NO	SI	NO	Referencia a la hora extra, HoraExtra

Nota. Campos de la tabla SolicitudHorasExtra. PK = Llave Primaria. FK = Llave Foránea. Long. = Longitud máxima del campo. UQ = Restricción de unicidad.

Tabla 63*Estructura de la tabla SolicitudVacaciones*

Campo	Tipo de Dato	Long.	Nulo	PK	FK	UQ	Descripción
idSolicitudVacaciones	INT	-	NO	SI	NO	NO	Identificador único de solicitud de vacaciones
fecha_inicio	DATE	-	NO	NO	NO	NO	Fecha de inicio del período
fecha_fin	DATE	-	NO	NO	NO	NO	Fecha de finalización del período
fecha_solicitud	DATE	-	NO	NO	NO	NO	Fecha en que se realizó la solicitud
estado	NVARCHAR	50	NO	NO	NO	NO	Estado actual del registro
aprobado_por	INT	-	SI	NO	NO	NO	Identificador del usuario que aprobó
fecha_aprobacion	DATE	-	SI	NO	NO	NO	Fecha de la aprobación
Vacaciones_idVacaciones	INT	-	NO	NO	SI	NO	Referencia a las vacaciones, Vacaciones

Nota. Campos de la tabla SolicitudVacaciones. PK = Llave Primaria. FK = Llave Foránea. Long. = Longitud máxima del campo. UQ = Restricción de unicidad.

Tabla 64*Estructura de la tabla Tardia*

Campo	Tipo de Dato	Long.	Nulo	PK	FK	UQ	Descripción
idTardia	INT	-	NO	SI	NO	NO	Identificador único de la tardía
fecha	DATE	-	NO	NO	NO	NO	Fecha del evento
minutos	INT	-	NO	NO	NO	NO	Minutos de tardía registrados
justificada	BIT	-	NO	NO	NO	NO	Indica si la tardía está justificada
observacion	NVARCHAR	200	NO	NO	NO	NO	Observaciones o notas adicionales
Empleado_id Empleado	INT	-	NO	NO	SI	NO	Referencia al empleado, Empleado

Nota. Campos de la tabla Tardia. PK = Llave Primaria. FK = Llave Foránea. Long. = Longitud máxima del campo. UQ = Restricción de unicidad.

Tabla 65*Estructura de la tabla TipoCedula*

Campo	Tipo de Dato	Long.	Nulo	PK	FK	UQ	Descripción
idTipoCedula	INT	-	NO	SI	NO	NO	Identificador único del tipo de cédula
tipo_cedula	NVARCHAR	15	NO	NO	NO	NO	Descripción del tipo de cédula

Nota. Campos de la tabla TipoCedula. PK = Llave Primaria. FK = Llave Foránea. Long. = Longitud máxima del campo. UQ = Restricción de unicidad.

Tabla 66*Estructura de la tabla TipoIncapacidad*

Campo	Tipo de Dato	Long.	Nulo	PK	FK	UQ	Descripción
idTipoIncapacidad	INT	-	NO	SI	NO	NO	Identificador único del tipo de incapacidad
nombre	NVARCHAR	100	NO	NO	NO	NO	Nombre descriptivo del registro
descripcion	NVARCHAR	255	NO	NO	NO	NO	Descripción detallada

Nota. Campos de la tabla TipoIncapacidad. PK = Llave Primaria. FK = Llave Foránea. Long. = Longitud máxima del campo. UQ = Restricción de unicidad.

Tabla 67*Estructura de la tabla TipoLiquidacion*

Campo	Tipo de Dato	Long.	Nulo	PK	FK	UQ	Descripción
idTipoLiquidacion	INT	-	NO	SI	NO	NO	Identificador único del tipo de liquidación
codigo	NVARCHAR	50	NO	NO	NO	NO	Código único del concepto
nombre	NVARCHAR	100	NO	NO	NO	NO	Nombre descriptivo del registro
descripcion	NVARCHAR	255	NO	NO	NO	NO	Descripción detallada
aplica_cesantia	BIT	-	NO	NO	NO	NO	Indica si aplica pago de cesantía
aplica_preaviso	BIT	-	NO	NO	NO	NO	Indica si aplica pago de preaviso

Nota. Campos de la tabla TipoLiquidacion. PK = Llave Primaria. FK = Llave Foránea. Long. = Longitud máxima del campo. UQ = Restricción de unicidad.

Tabla 68*Estructura de la tabla TipoPermiso*

Campo	Tipo de Dato	Long.	Nulo	PK	FK	UQ	Descripción
idTipoPermiso	INT	-	NO	SI	NO	NO	Identificador único del tipo de permiso
nombre	NVARCHAR	100	NO	NO	NO	NO	Nombre descriptivo del registro
descripcion	NVARCHAR	255	SI	NO	NO	NO	Descripción detallada
tiene_goce	BIT	-	NO	NO	NO	NO	Indica si el permiso es con goce de salario
es_administrativo	BIT	-	NO	NO	NO	NO	Indica si es permiso de tipo administrativo
afecta_asistencia	BIT	-	NO	NO	NO	NO	Indica si afecta el registro de asistencia

Nota. Campos de la tabla TipoPermiso. PK = Llave Primaria. FK = Llave Foránea. Long. = Longitud máxima del campo. UQ = Restricción de unicidad.

Tabla 69*Estructura de la tabla TramosParametrosRenta*

Campo	Tipo de Dato	Long.	Nulo	PK	FK	UQ	Descripción
idTramosParametrosRenta	INT	-	NO	SI	NO	NO	Identificador único del tramo de renta
anio	INT	-	NO	NO	NO	NO	Año de cálculo
tramo	NVARCHAR	50	NO	NO	NO	NO	Nombre o etiqueta del tramo
limite_inferior	DECIMAL	12,2	NO	NO	NO	NO	Límite inferior del tramo de renta
limite_superior	DECIMAL	12,2	NO	NO	NO	NO	Límite superior del tramo de renta
porcentaje	DECIMAL	5,2	NO	NO	NO	NO	Porcentaje aplicable
exceso	DECIMAL	12,2	NO	NO	NO	NO	Monto fijo de exceso del tramo anterior
detalle_tramo	NVARCHAR	150	NO	NO	NO	NO	Descripción detallada del tramo
fecha_actualizacion	DATETIME2	7	NO	NO	NO	NO	Fecha de la última actualización

Nota. Campos de la tabla TramosParametrosRenta. PK = Llave Primaria. FK = Llave Foránea. Long. = Longitud máxima del campo. UQ = Restricción de unicidad.

Tabla 70*Estructura de la tabla Vacaciones*

Campo	Tipo de Dato	Long.	Nulo	PK	FK	UQ	Descripción
idVacaciones	INT	-	NO	SI	NO	NO	Identificador único de vacaciones
Empleado_idEmpleado	INT	-	NO	NO	SI	NO	Referencia al empleado, Empleado
fecha_inicio	DATE	-	NO	NO	NO	NO	Fecha de inicio del período
fecha_fin	DATE	-	NO	NO	NO	NO	Fecha de finalización del período
dias_habiles	INT	-	NO	NO	NO	NO	Días hábiles del período
estado	NVARCHAR	50	NO	NO	NO	NO	Estado actual del registro
observaciones	NVARCHAR	255	SI	NO	NO	NO	Observaciones o notas del proceso
fecha_solicitud	DATE	-	NO	NO	NO	NO	Fecha en que se realizó la solicitud
fecha_aprobacion	DATE	-	SI	NO	NO	NO	Fecha de la aprobación

Nota. Campos de la tabla Vacaciones. PK = Llave Primaria. FK = Llave Foránea. Long. = Longitud máxima del campo. UQ = Restricción de unicidad.

Diseño de procesos

El diseño de procesos constituye una etapa fundamental dentro del desarrollo del prototipo, dado que permite representar de forma gráfica y estructurada los flujos operativos que rigen el comportamiento del sistema. A través de los diagramas de flujo se documentan las secuencias lógicas de cada proceso principal, identificando las actividades del sistema, los puntos de decisión relevantes y los actores que participan en cada etapa de ejecución. Esta representación facilita la comprensión técnica del funcionamiento interno de la aplicación y sirve como referencia para la implementación, validación y mantenimiento posterior del sistema. Igualmente, permite verificar que los flujos definidos guardan coherencia con los casos de uso identificados durante el análisis de requerimientos del proyecto.

Para el sistema web de gestión y pago de planilla desarrollado para AraWeb CR, se identificaron tres procesos principales cuya documentación resulta esencial para comprender el flujo operativo de la solución propuesta. El primer proceso corresponde a la autenticación e inicio de sesión, el cual controla el acceso al sistema mediante la validación de credenciales en la base de datos y la asignación de permisos según el rol del usuario registrado. El segundo proceso abarca la gestión integral de solicitudes, que centraliza el registro, validación y aprobación de vacaciones,

horas extra, permisos e incapacidades bajo un flujo unificado de resolución por parte del jefe inmediato correspondiente. El tercer proceso representa la generación de planilla, el cual constituye el núcleo funcional del sistema y agrupa las etapas de validación del período seleccionado, el cálculo salarial automatizado y la emisión digital de comprobantes de pago.

Los diagramas fueron elaborados empleando la simbología estándar de diagramas de flujo, en la que los terminadores representan el inicio y el fin de cada proceso, los rectángulos identifican actividades o procedimientos ejecutados por el sistema, los rombos señalan puntos de decisión con ramificaciones condicionales, y los paralelogramos indican entradas o salidas de información.

Cada diagrama documenta tanto el flujo básico del proceso como los flujos alternos derivados de condiciones de error o validaciones fallidas, reflejando el comportamiento real del sistema ante distintos escenarios de operación. De esta manera, la representación gráfica complementa la documentación técnica del proyecto, establece una base sólida para las pruebas funcionales del prototipo y facilita la comunicación entre los distintos componentes del diseño.

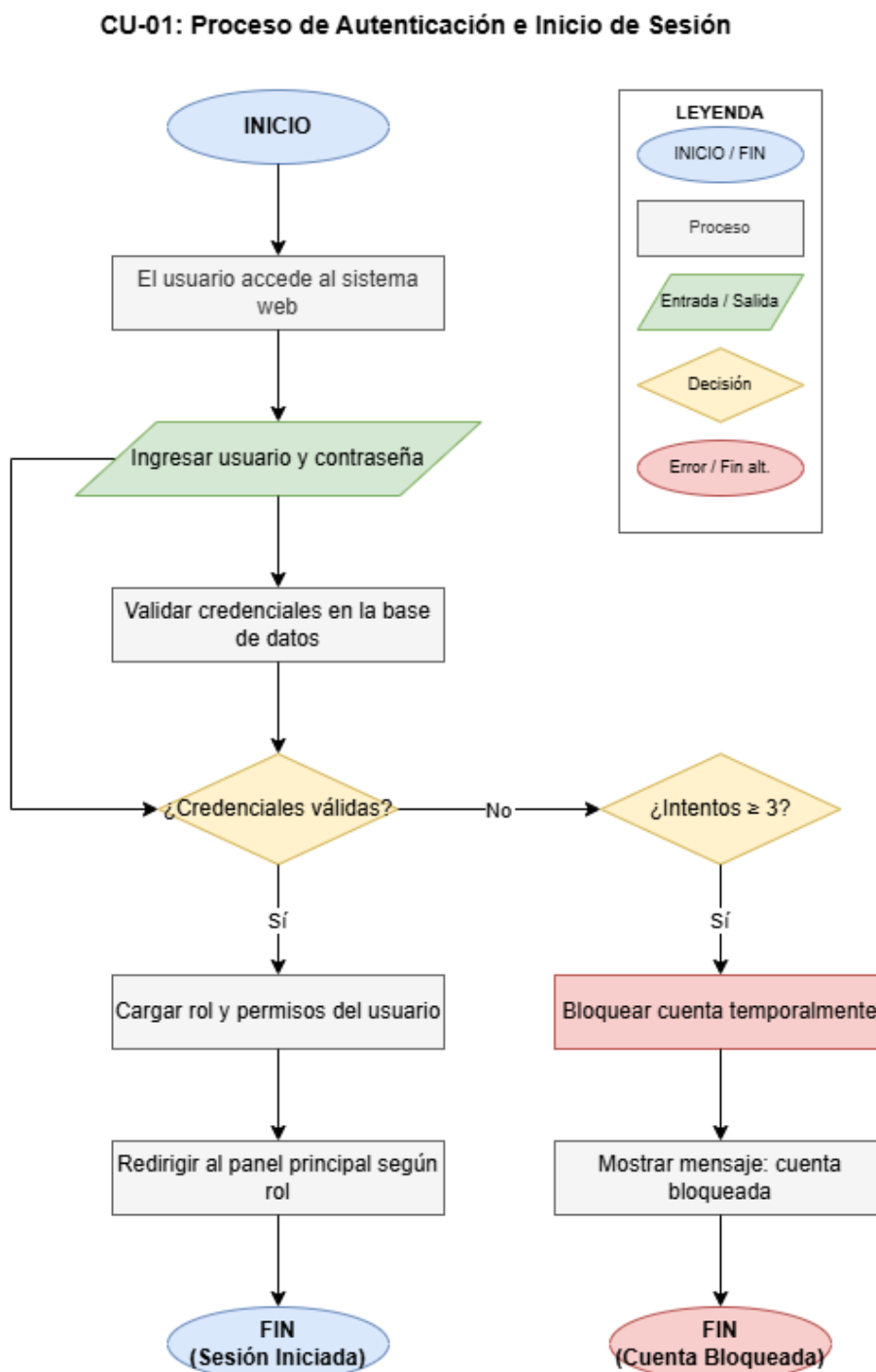
La Figura 25 presenta el diagrama de flujo correspondiente al proceso de autenticación e inicio de sesión del sistema. El flujo inicia cuando el usuario accede a la página principal y completa el formulario con sus credenciales, las cuales son validadas contra la tabla Credencial de la base de datos para verificar su existencia y estado activo.

En caso de que las credenciales sean incorrectas, el sistema registra el intento fallido y permite reintentar hasta alcanzar el límite definido, momento en el que la cuenta es bloqueada temporalmente como medida de seguridad. Si la autenticación resulta exitosa, el sistema identifica el rol asignado al usuario, carga los permisos correspondientes y registra el evento de acceso en el módulo de auditoría de la bitácora.

Finalmente, el usuario es redirigido al panel principal según el rol autenticado, garantizando control de acceso diferenciado para los perfiles de colaborador, jefe inmediato, encargado de planilla y administrador.

Figura 25

Diagrama de flujo – Proceso de Autenticación e Inicio de Sesión.



Fuente: Elaboración propia.

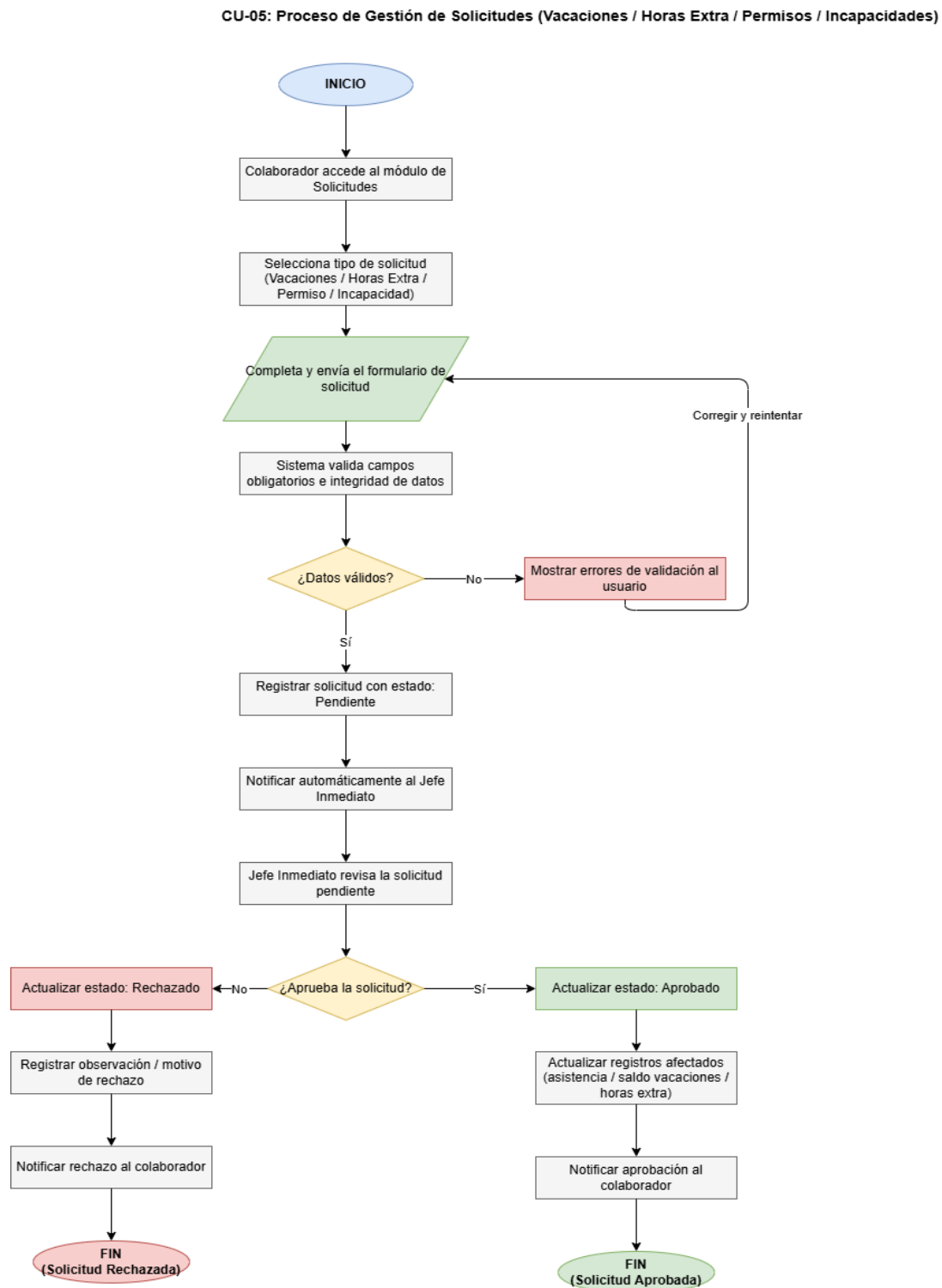
La Figura 26 presenta el diagrama de flujo del proceso de gestión de solicitudes laborales del sistema.

El flujo inicia cuando el colaborador autenticado accede al módulo de Solicitudes, selecciona el tipo de solicitud requerida vacaciones, horas extra, permiso o incapacidad y completa el formulario correspondiente, el cual es validado por el sistema antes de ser registrado con estado pendiente.

Una vez registrada la solicitud, el sistema notifica automáticamente al jefe inmediato, quien accede al módulo para revisar la información y emitir su resolución mediante la acción de aprobar o rechazar. Ante una aprobación, el sistema actualiza el estado de la solicitud, aplica el impacto correspondiente sobre los registros afectados como asistencia o saldo vacacional y notifica al colaborador sobre el resultado.

En caso de rechazo, el jefe registra el motivo de la decisión, el sistema actualiza el estado ha rechazado y notifica igualmente al colaborador, asegurando trazabilidad completa del proceso independientemente del resultado.

Figura 26
Diagrama de flujo – Gestión de solicitudes.



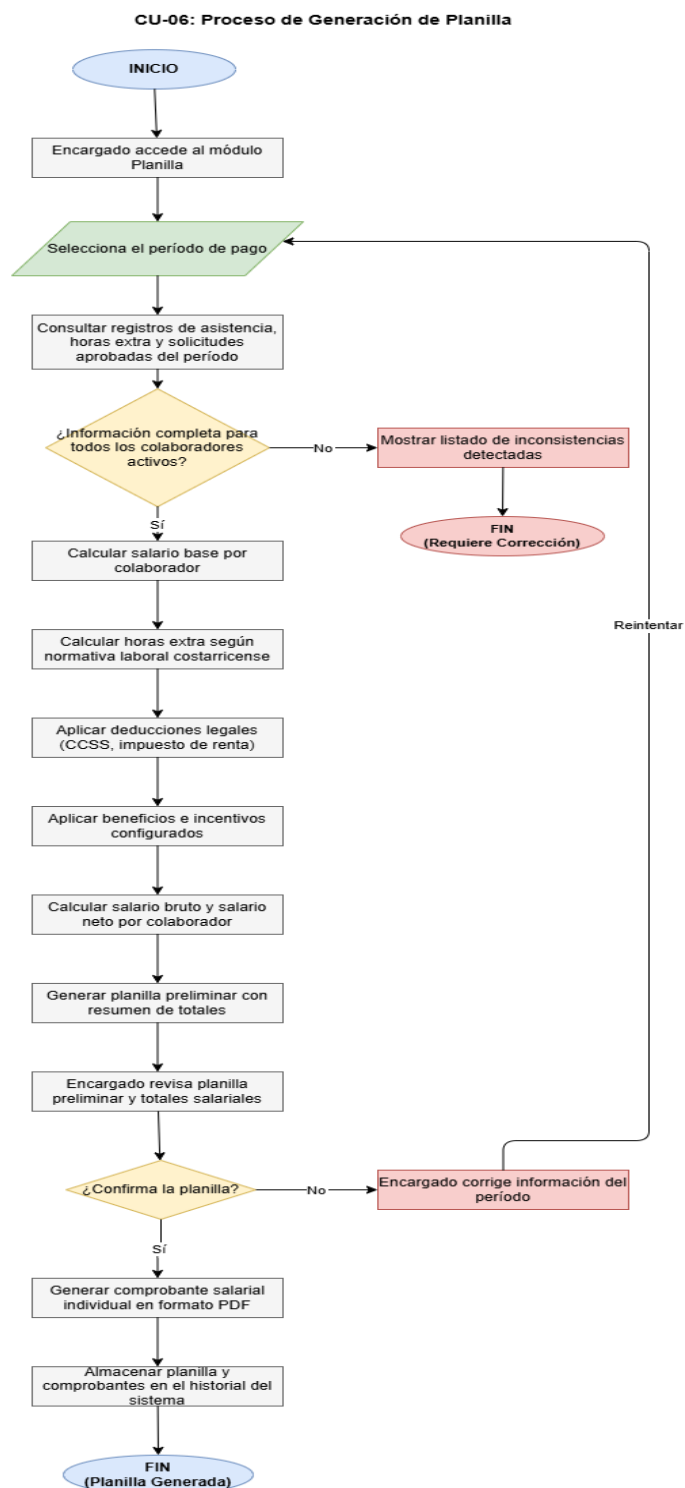
La Figura 27 presenta el diagrama de flujo del proceso de generación de planilla, el cual constituye el proceso central del sistema desarrollado.

El flujo inicia cuando el encargado de planilla selecciona el período de pago, momento en el que el sistema ejecuta una validación integral que consulta los registros de asistencia, horas extra aprobadas y solicitudes resueltas del período, verificando que todos los colaboradores activos cuenten con información completa antes de continuar.

De no existir inconsistencias, el sistema ejecuta el cálculo automático, determinando el salario base, las horas extra conforme a la normativa laboral costarricense, las deducciones legales obligatorias CCSS e impuesto de renta y los beneficios configurados, obteniendo el salario bruto y neto de cada colaborador.

El encargado revisa la planilla preliminar con el resumen de totales y confirma su aprobación; en caso de detectar errores, puede corregir la información y reiniciar el proceso desde la selección del período. Tras la confirmación, el sistema genera los comprobantes salariales individuales en formato PDF, los almacena en el historial del colaborador correspondiente y habilita su descarga dentro de la plataforma.

Figura 27
Diagrama de flujo – Generación de Planilla.



Fuente: Elaboración propia.

Diseño de salidas

El diseño de salidas corresponde a la etapa del desarrollo del sistema en la que se define cómo la información procesada será presentada al usuario final mediante reportes, consultas y visualizaciones estructuradas. En el sistema web de gestión de planilla, este proceso se orienta a transformar los datos almacenados en información clara, verificable y útil para las operaciones administrativas.

Se establecieron criterios de uniformidad visual, organización lógica de datos y facilidad de lectura, con el objetivo de apoyar la interpretación rápida de resultados relacionados con cálculos salariales y controles de nómina. Así, las salidas del sistema no solo cumplen una función informativa, sino que también contribuyen al control operativo y a la validación de los procesos ejecutados.

La figura 28 presenta la vista de detalle de la nómina correspondiente a un período específico, donde se visualiza la información consolidada del cálculo realizado para cada colaborador incluido en la planilla.

La interfaz muestra datos esenciales como empleado, puesto, percepciones, horas extra, deducciones, monto neto a pagar y fecha de cálculo, permitiendo verificar de forma inmediata los resultados generados por el sistema.

Además, se incorpora un resumen de totales que facilita el control general del proceso antes de su cierre o aprobación. Este diseño permite realizar revisiones operativas rápidas y garantiza trazabilidad sobre la información procesada, fortaleciendo la transparencia y confiabilidad del sistema de gestión de planillas.

Figura 28
Diseño de salida – Detalle Nomina Mayo

Detalle Nomina Mayo 2026

Nómina — Mayo 2026							Cerrada
Creada: 13/05/2026 20:09 Cerrada: 24/06/2026 19:20							
Empleado	Puesto	Percepciones	Horas Extra	Deducciones	Neto a Pagar	Calculado	
Sofía Cano Molina 325006520	Gerente General	€ 0,00	€ 0,00	- € 0,00	€ 0,00	24/06/26	
Andrés Chaves Araya 112340006	Gerente General	€ 1 260 000,00	€ 0,00	- € 134 442,00	€ 1 125 558,00	24/06/26	
Sofía González Ulate 112340003	Gerente General	€ 1 260 000,00	€ 0,00	- € 134 442,00	€ 1 125 558,00	24/06/26	
Diego Hernández Brenes 112340004	Administrador General	€ 840 000,00	€ 0,00	- € 89 628,00	€ 750 372,00	24/06/26	
Priscilla Méndez Castro 112340009	Gerente General	€ 1 260 000,00	€ 0,00	- € 134 442,00	€ 1 125 558,00	24/06/26	
Carlos Eduardo Mora Jiménez 112340002	Ejecutivo de Ventas	€ 420 000,00	€ 0,00	- € 44 814,00	€ 375 186,00	24/06/26	
Jefe Perez Fallas 000000003	Administrador General	€ 0,00	€ 0,00	- € 0,00	€ 0,00	24/06/26	
Rachel Perez Camacho 000000001	Disenador UX UI	€ 0,00	€ 0,00	- € 0,00	€ 0,00	24/06/26	
Valentina Quirós López 112340007	Administrador General	€ 840 000,00	€ 0,00	- € 89 628,00	€ 750 372,00	24/06/26	
Fabián Rodríguez Monge 112340008	Ejecutivo de Ventas	€ 420 000,00	€ 0,00	- € 44 814,00	€ 375 186,00	24/06/26	
Administrador Sistema AraWeb 000000000	Administrador General	€ 457 307,69	€ 17 307,69	- € 48 794,73	€ 408 512,96	24/06/26	
Laura Solís Cascante 112340005	Ejecutivo de Ventas	€ 420 000,00	€ 0,00	- € 44 814,00	€ 375 186,00	24/06/26	
Ana María Vargas Rojas 112340001	Administrador General	€ 840 000,00	€ 0,00	- € 89 628,00	€ 750 372,00	24/06/26	
Marco Vargas Segura 1-0456-0789	Asistente Administrativo	€ 0,00	€ 0,00	- € 0,00	€ 0,00	24/06/26	
Totales del periodo		€ 8 017 307,69	€ 17 307,69	- € 855 446,73	€ 7 161 860,96		

Volver

Nómina cerrada — no se permiten modificaciones

Reabrir nómina

Fuente: Elaboración propia.

La Figura 29 presenta el diseño de salida correspondiente al detalle de solicitud de vacaciones dentro del sistema de nómina, cuyo propósito es centralizar la información necesaria para la gestión y validación administrativa del permiso solicitado. La interfaz muestra de forma estructurada los datos del empleado, período de vacaciones, cantidad de días hábiles y estado actual de la solicitud, permitiendo una lectura rápida orientada a la toma de decisiones. Además, incorpora controles operativos para aprobar o rechazar la solicitud, incluyendo campos de observación que fortalecen la trazabilidad del proceso. El historial de acciones registra los eventos asociados a la solicitud, garantizando auditoría y seguimiento conforme a los principios de control interno del sistema. Este diseño prioriza la claridad visual, la reducción de errores operativos y la estandarización del flujo de aprobación dentro del módulo de gestión de vacaciones.

Figura 29*Diseño de salida – Detalle Vacaciones*

Detalle Vacaciones

Detalle de Solicitud de Vacaciones

Solicitud #62 — 07/07/2026 00:00

Empleado
 Nombre
Jefe Perez Fallas

Cedula
000000003

Puesto
Administrador General

Estado
Pendiente

Periodo de vacaciones
 Fecha inicio
12/07/2026

Fecha fin
17/07/2026

Dias habiles
5 dias

Observaciones: Vacaciones Ordinarias

Aprobar solicitud

Comentario (opcional)

Aprobar

Rechazar solicitud

Motivo del rechazo

Rechazar

Historial

Accion	Comentario	Fecha
Solicitud creada	5 días hábiles del 12/07/2026 al 17/07/2026	07/07/2026 10:53

Volver

Fuente: Elaboración propia.

La Figura 30 muestra el reporte de aguinaldo generado por el sistema, donde se presenta el resumen del cálculo correspondiente al periodo anual definido. En esta salida se visualizan los datos del empleado, puesto desempeñado, monto calculado y estado del proceso, además del total general a pagar, lo que permite validar rápidamente la información antes de su emisión oficial. El diseño incorpora opciones de generación de nuevos reportes e impresión, facilitando la trazabilidad y formalización del proceso. Esta salida cumple una función clave dentro del sistema al servir como documento de respaldo financiero y administrativo para la organización.

Figura 30
Diseño de salida – Reporte Aguinaldo

Reporte Aguinaldo

 **Aguinaldo 2026**

14 empleado(s) | Fecha máxima de pago: 20 de diciembre 2026

Empleado	Cedula	Puesto	Monto aguinaldo	Estado	Fecha calculo
Sofía Cano Molina	325006520	Gerente General	€0,00	Calculado	27/06/2026
Andrés Chaves Araya	112340006	Gerente General	€105 000,00	Calculado	27/06/2026
Sofía González Ulate	112340003	Gerente General	€105 000,00	Calculado	27/06/2026
Diego Hernández Brenes	112340004	Administrador General	€70 000,00	Calculado	27/06/2026
Priscilla Méndez Castro	112340009	Gerente General	€105 000,00	Calculado	27/06/2026
Carlos Eduardo Mora Jiménez	112340002	Ejecutivo de Ventas	€35 000,00	Calculado	27/06/2026
Jefe Perez Fallas	000000003	Administrador General	€0,00	Calculado	27/06/2026
Rachel Perez Camacho	000000001	Diseñador UX UI	€2 083,33	Calculado	27/06/2026
Valentina Quirós López	112340007	Administrador General	€70 000,00	Calculado	27/06/2026
Fabían Rodríguez Monge	112340008	Ejecutivo de Ventas	€35 000,00	Calculado	27/06/2026
Administrador Sistema AraWeb	000000000	Administrador General	€63 308,49	Calculado	27/06/2026
Laura Solís Cascante	112340005	Ejecutivo de Ventas	€35 000,00	Calculado	27/06/2026
Ana María Vargas Rojas	112340001	Administrador General	€70 000,00	Calculado	27/06/2026
Marco Vargas Segura	1-0456-0789	Asistente Administrativo	€0,00	Calculado	27/06/2026
TOTAL A PAGAR			€695 391,82		

 Nuevo reporte
  Imprimir

Fuente: Elaboración propia.

La Figura 31 presenta el reporte de asistencia generado por el sistema de nómina, el cual consolida la información diaria de marcaciones realizadas por el colaborador dentro de un período específico. La salida muestra de forma estructurada la fecha, hora de entrada, hora de salida y el cálculo automático de horas netas laboradas, permitiendo evaluar el cumplimiento de la jornada laboral.

Del mismo modo, el sistema identifica condiciones relevantes como tardías y ausencias mediante indicadores visuales que facilitan la interpretación operativa del reporte. Este diseño permite apoyar procesos administrativos, auditorías internas y validaciones previas al cálculo de planilla. La organización tabular garantiza claridad en la lectura y trazabilidad histórica de la información registrada. Además, la funcionalidad contribuye a mejorar el control del tiempo laboral y la toma de decisiones basada en datos verificables.

Figura 31
Diseño de salida – Reporte Asistencia

Reporte de Asistencia

Reporte de asistencia
 Período: 22/05/2026 al 02/06/2026 | Colaborador: Administrador Sistema AraWeb | Cédula:000000000

Fecha	Entrada	Salida	Horas netas	Tardía	Ausencia
22/05/2026	19:31:23.8496744	20:14:24.6732873	0,716895448027778	No	No
23/05/2026			0	No	Si
24/05/2026			0	No	Si
25/05/2026			0	No	Si
26/05/2026			0	No	Si
27/05/2026			0	No	Si
28/05/2026	21:22:34.0916596	21:23:01.2336587	0,00753944419444444	No	No
29/05/2026	09:39:57.4363651	09:40:32.4747127	0,00973287433333333	No	No
30/05/2026	10:08:45.0785472	10:08:48.1104369	0,000842191583333333	No	No
31/05/2026	10:11:40.0230667	10:11:41.8710264	0,000513322138888889	No	No
01/06/2026	08:00:00	17:00:00	9	No	No
02/06/2026	08:20:07.3731977	19:52:25.4869224	11,5383649235278	20 min	No

[⏮ Volver](#) [🖨 Imprimir](#)

Fuente: Elaboración propia.

La Figura 32 presenta el resumen mensual de asistencia del colaborador, el cual consolida los indicadores principales generados a partir del registro diario de marcaciones.

En esta salida se muestran métricas clave como días laborados, cantidad de ausencias, tardías registradas y el total de horas netas trabajadas durante el período seleccionado.

Este diseño permite visualizar de forma inmediata el comportamiento general de asistencia del empleado, facilitando el análisis administrativo y el control del cumplimiento laboral. Además, la información resumida sirve como insumo directo para procesos posteriores del sistema de nómina, tales como cálculos salariales, aplicación de deducciones y generación de reportes gerenciales.

Figura 32*Diseño de salida – Resumen Mensual de Asistencia*

Fuente: Elaboración propia.

Diagramas UML

Los diagramas UML (Unified Modeling Language) son representaciones gráficas estandarizadas que permiten documentar tanto la estructura como el comportamiento de un sistema de software de manera clara y precisa. En el presente proyecto se desarrollaron dos tipos de diagramas: el diagrama de clases y el diagrama de secuencia. El diagrama de clases describe la estructura estática del sistema, mostrando las entidades principales del modelo, sus atributos y las relaciones existentes entre ellas, tomando como base el diseño físico de la base de datos y los módulos funcionales implementados.

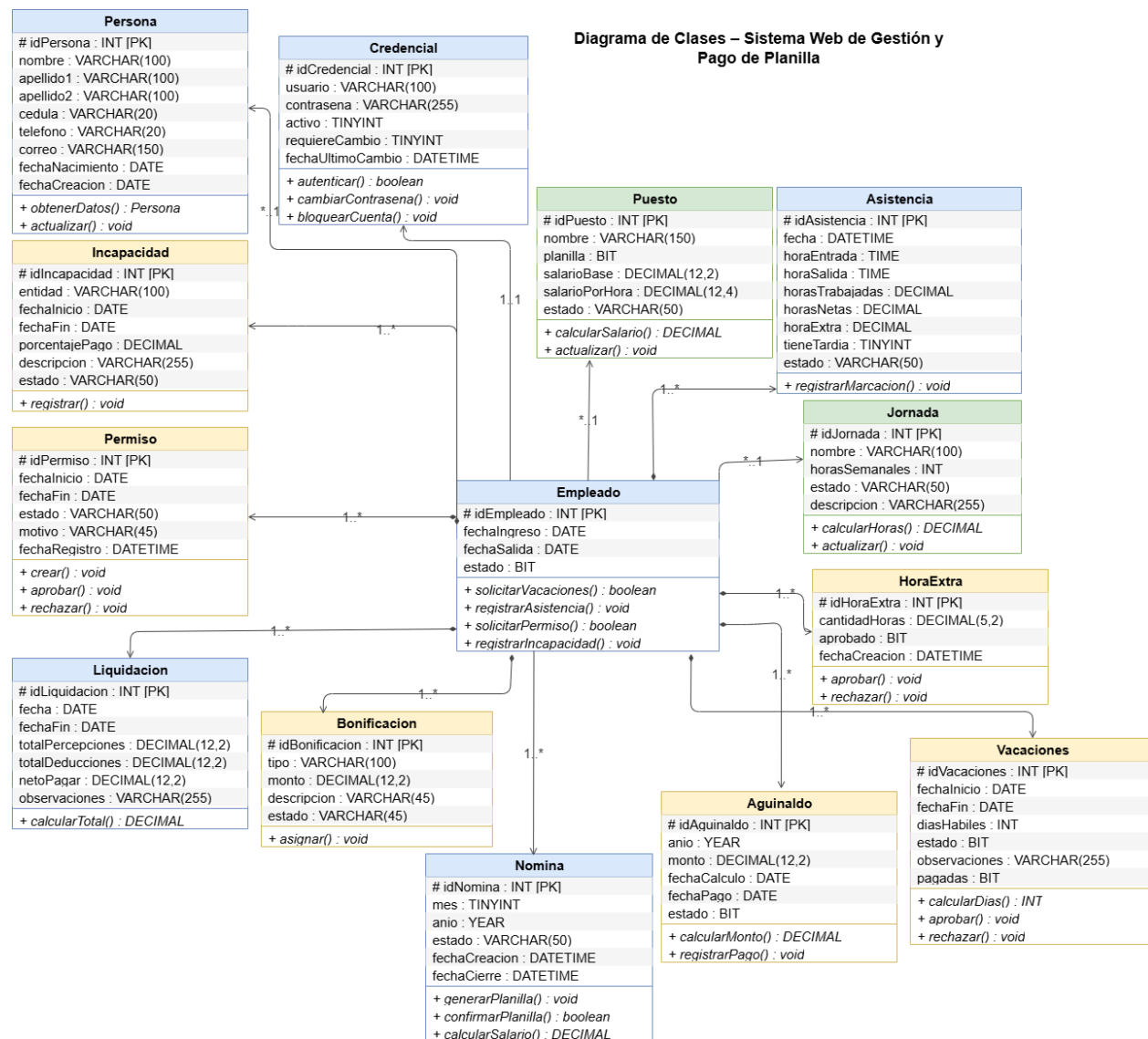
Por su parte, el diagrama de secuencia representa la interacción dinámica entre los actores y los componentes del sistema durante la ejecución de un proceso específico, en este caso el proceso de generación de planilla, ilustrando el flujo de mensajes y operaciones en orden cronológico. Ambos diagramas complementan los artefactos de diseño previamente documentados y fortalecen la trazabilidad técnica del prototipo desarrollado.

Diagrama de Clases

El diagrama de clases presenta la estructura estática del sistema web de gestión y pago de planilla, identificando las entidades principales que conforman el modelo de dominio y las relaciones existentes entre ellas. Este modelo se construyó a partir del diseño físico de la base de datos y de los módulos funcionales definidos durante las etapas de análisis y diseño, representando las clases del modelo orientado a objetos implementado en C# bajo el framework ASP.NET MVC.

Cada clase incluye sus atributos principales y las asociaciones que reflejan las dependencias establecidas entre los distintos componentes del sistema, como la relación entre el empleado y sus registros de asistencia, nómina, solicitudes y beneficios laborales. La representación formal de estas estructuras permite comprender la organización lógica del sistema y la forma en que los datos interactúan entre sus entidades durante la ejecución de los procesos operativos.

Figura 33
Diagrama de Clases



Fuente: Elaboración propia.

Diagrama de Secuencia

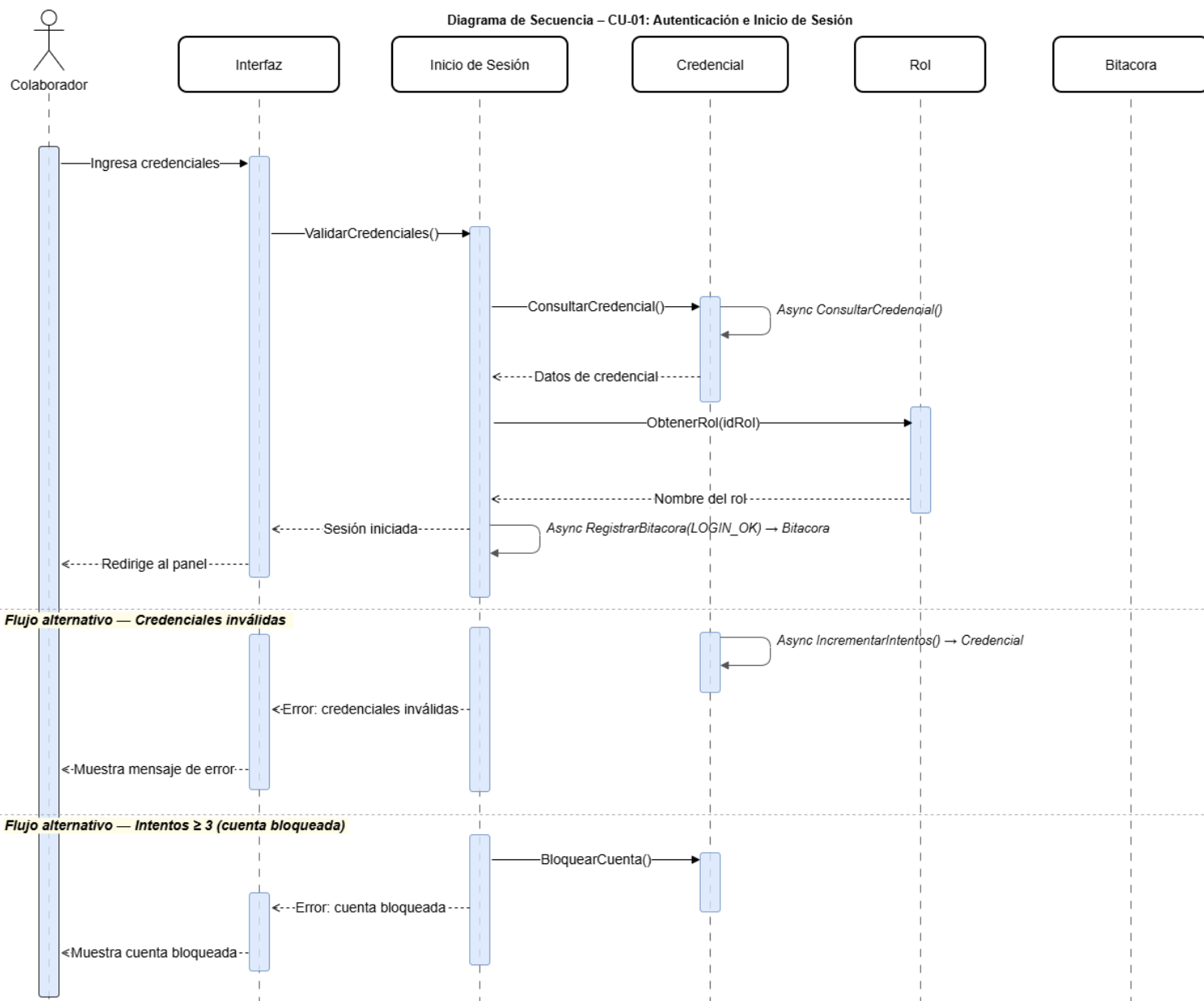
El diagrama de secuencia modela el comportamiento dinámico del sistema, mostrando la interacción temporal entre los actores y los componentes internos durante la ejecución de un proceso específico. Para el presente prototipo, se seleccionó el proceso de generación de planilla como caso representativo, dado que constituye el flujo operativo de mayor complejidad e importancia dentro del sistema.

El diagrama refleja la secuencia de mensajes intercambiados entre el encargado de planilla, la interfaz de usuario, la capa de lógica de negocio y la base de datos, permitiendo visualizar el orden de las operaciones, las validaciones aplicadas y los registros generados durante el cálculo salarial. Este tipo de modelado complementa el diagrama de clases al agregar la dimensión temporal al diseño, evidenciando cómo los objetos colaboran entre sí para cumplir con los requerimientos funcionales definidos.

En la figura 34 se puede visualizar el diagrama de secuencia del proceso de autenticación, el cual modela la interacción entre el usuario, la interfaz del sistema, el servicio de autenticación y la base de datos durante el inicio de sesión. El flujo principal describe cómo el usuario ingresa sus credenciales, las cuales son enviadas al Inicio de Sesión para su validación contra la base de datos; de ser correctas, se registra la sesión activa y el sistema redirige al usuario al panel correspondiente según su rol asignado dentro del sistema.

Adicionalmente, el diagrama contempla dos flujos alternativos que cubren los escenarios de error durante la autenticación. El primero modela el caso en que las credenciales ingresadas son inválidas, donde el sistema retorna un mensaje de error directamente al usuario sin generar ningún registro de sesión. El segundo flujo alternativo representa el bloqueo automático de la cuenta cuando el usuario supera tres intentos fallidos consecutivos, acción que es procesada por el Inicio de Sesión mediante una llamada asíncrona que actualiza el estado de la credencial en la base de datos y notifica al usuario que su acceso ha sido restringido temporalmente.

Figura 34
Diagrama de Secuencia – Inicio de Sesión



Fuente: Elaboración propia.

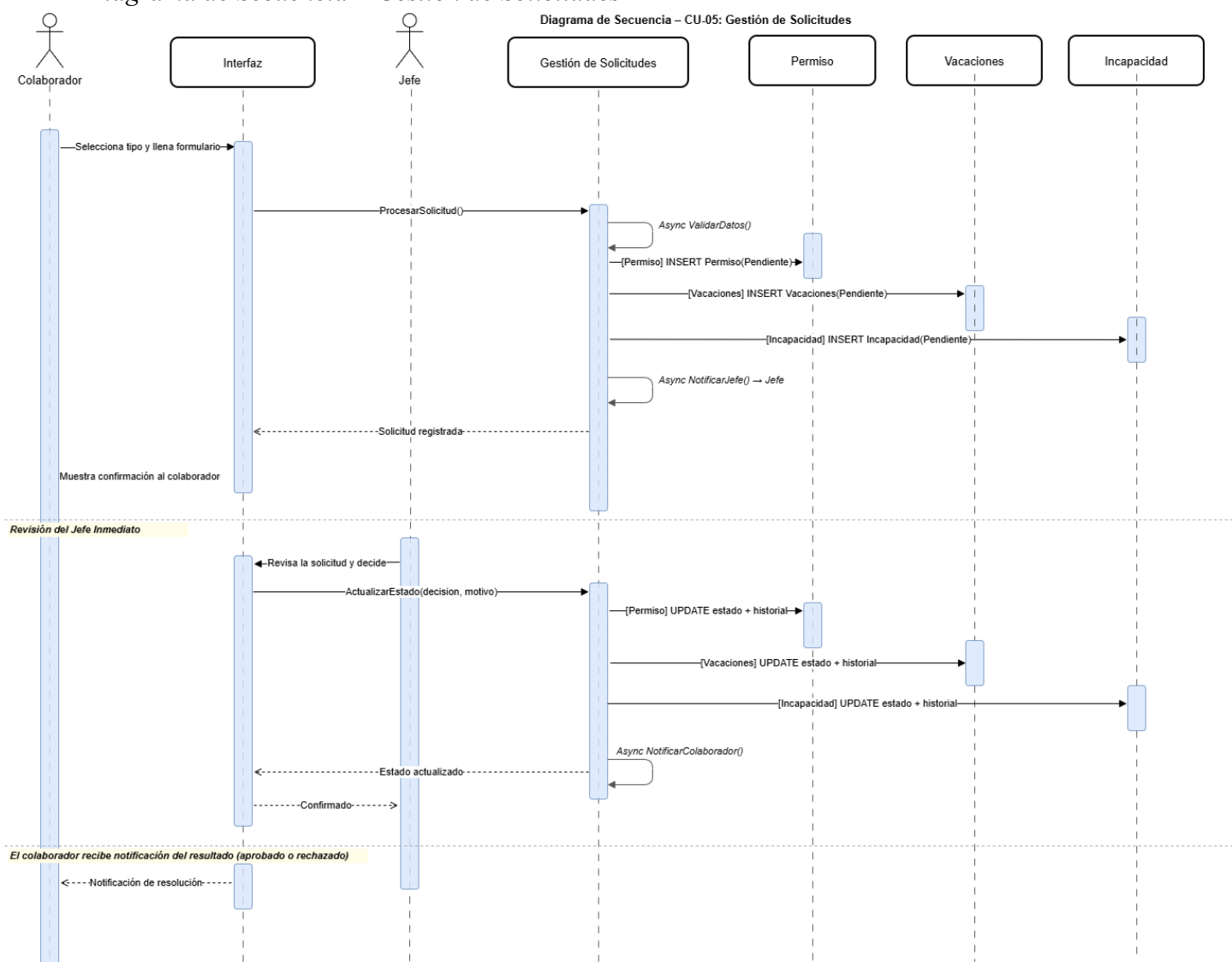
En la figura 35 se puede visualizar el diagrama de secuencia del proceso de gestión de solicitudes, el cual representa el flujo de comunicación entre el colaborador, la interfaz, el servicio de solicitudes, la base de datos y el jefe inmediato a lo largo del ciclo completo de una solicitud. En la primera etapa, el colaborador completa el formulario correspondiente al tipo de solicitud

requerido, el cual es procesado y validado por el sistema antes de ser registrado en la base de datos con estado pendiente; una vez almacenado, se genera una notificación asíncrona dirigida al jefe inmediato para su revisión.

En la segunda etapa del flujo, el jefe inmediato accede a la solicitud desde la interfaz del sistema y emite su decisión de aprobación o rechazo, proporcionando en caso de rechazo una observación o motivo que queda registrado junto al cambio de estado en la base de datos. Esta actualización desencadena de forma automática el envío de una notificación asíncrona al colaborador, informándole el resultado de su gestión sin necesidad de intervención adicional por parte del encargado, lo que garantiza la trazabilidad y la comunicación oportuna dentro del proceso.

Figura 35

Diagrama de Secuencia – Gestión de Solicitudes



Fuente: Elaboración propia.

En la figura 36 se puede visualizar el diagrama de secuencia del proceso de generación de planilla, el cual ilustra la interacción entre el encargado de planilla, la interfaz del sistema, el servicio de nómina y la base de datos durante el cálculo y registro del pago mensual de un colaborador. En esta primera etapa, el encargado selecciona el empleado, el mes y el año correspondiente; el sistema consulta de forma asíncrona la asistencia registrada, las horas extra aprobadas y las solicitudes del período, con cuyos datos el servicio ejecuta el cálculo del salario bruto, las deducciones legales y el salario neto, retornando un detalle preliminar para revisión del encargado.

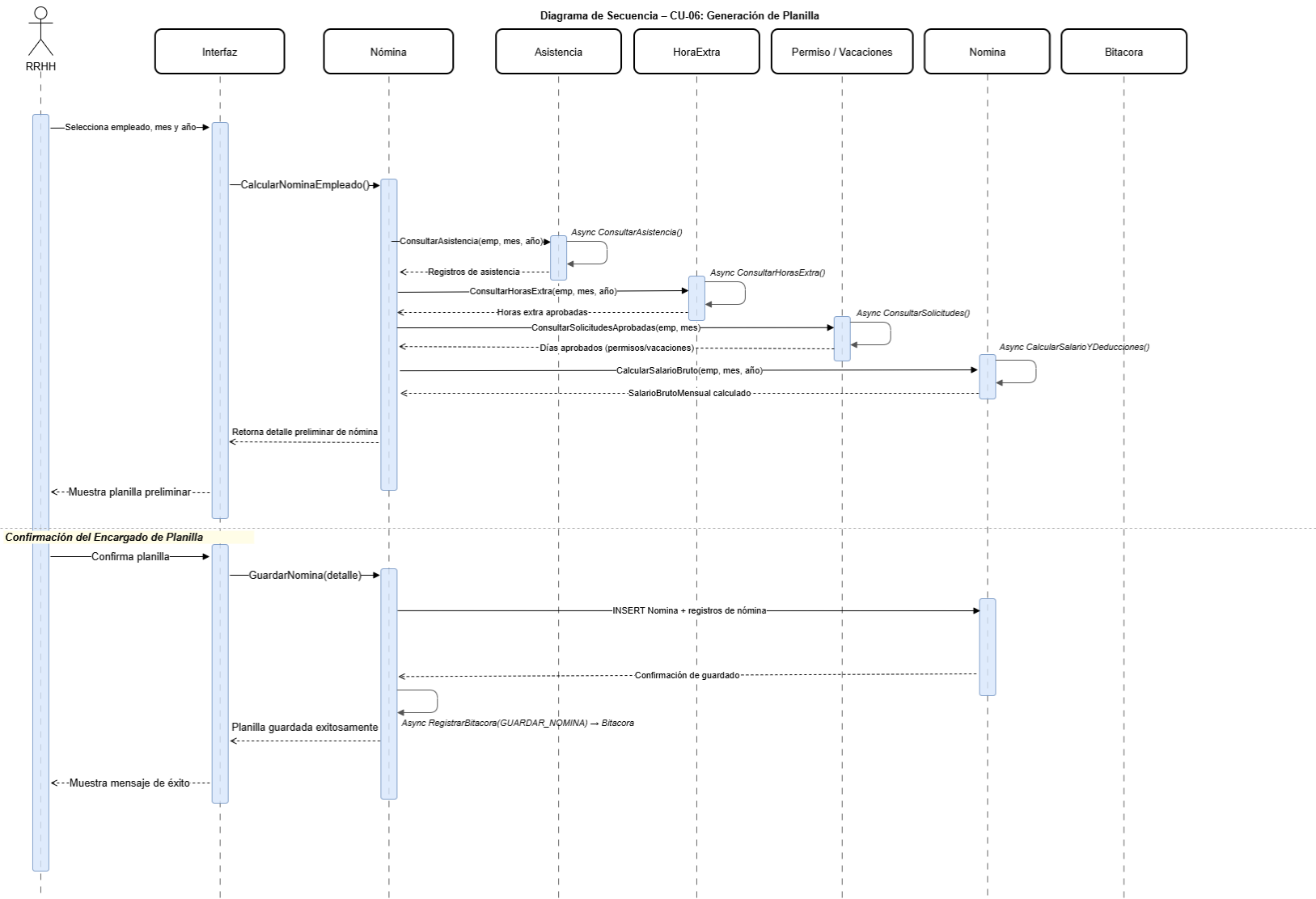
Una vez que el encargado revisa y confirma la planilla generada, el sistema procede a persistir los registros de nómina y detalle en la base de datos mediante las operaciones de inserción correspondientes. Posterior a la confirmación del guardado, el servicio de nómina ejecuta de forma asíncrona el registro del evento en la bitácora del sistema, dejando constancia de la operación realizada con los datos del usuario y la marca de tiempo. Finalmente, el sistema retorna una respuesta de éxito a la interfaz, la cual despliega al encargado un mensaje de confirmación que indica que el proceso de generación de planilla ha sido completado satisfactoriamente.

Cabe señalar que el diseño del diagrama refleja una separación clara de responsabilidades entre las distintas clases del modelo. La clase Asistencia provee los registros de entrada y salida del período seleccionado, mientras que HoraExtra suministra únicamente las horas adicionales que han sido previamente aprobadas por el jefe inmediato, con lo que se asegura que no se incluyan horas pendientes de validación en el cálculo. La clase Permiso y Vacaciones aporta los días aprobados que afectan el monto a pagar, y la clase SalarioBrutoMensual centraliza la lógica de cálculo de deducciones legales como la CCSS y el INS, produciendo como resultado el salario neto definitivo. Esta distribución permite que cada componente del sistema tenga una única responsabilidad dentro del flujo, facilitando el mantenimiento y la trazabilidad de los cálculos realizados.

Desde la perspectiva del control de errores e integridad del proceso, la operación de guardado se ejecuta de forma transaccional, de modo que los registros en la clase Nomina únicamente se persisten una vez que todos los datos del detalle han sido validados y calculados correctamente. El uso de operaciones asíncronas para las consultas a Asistencia, HoraExtra y Permiso y Vacaciones, así como para el registro en Bitacora, garantiza que el hilo principal de ejecución no se bloquee durante la recuperación de datos, mejorando la eficiencia del proceso ante

períodos con un alto volumen de registros. El registro en la bitácora al final del flujo asegura la auditoría completa de cada planilla generada, almacenando el identificador del usuario responsable, la fecha y hora de la operación, y los parámetros del empleado y período procesado.

Figura 36
Diagrama de Secuencia – Generación de Planilla



Fuente: Elaboración propia.

Programación

La presente sección tiene como propósito exponer fragmentos representativos del código fuente desarrollado durante la construcción del Sistema de Gestión de Nómina. De acuerdo con las indicaciones del trabajo final de graduación, no se incluye el código completo, sino ejemplos específicos que ilustran las categorías de entradas, salidas, procesos y validaciones, así como un fragmento por cada módulo contemplado dentro del alcance del proyecto.

Entradas

Recepción de credenciales de inicio de sesión (Módulo de Seguridad)

El primer ejemplo de entrada corresponde al proceso de autenticación del usuario en el sistema. El método Login del controlador AccountController recibe como parámetros de entrada el nombre de usuario y la contraseña ingresados a través del formulario. Antes de proceder con la búsqueda en la base de datos, se valida que ambos campos contengan información; si alguno está vacío, el sistema devuelve un mensaje de error al usuario. Este fragmento pertenece al módulo de Seguridad y representa el punto de ingreso de datos más crítico de la aplicación, dado que controla el acceso a todas las funcionalidades del sistema.

Figura 37

Recepción de credenciales de inicio de sesión.

```
[HttpPost]
[AllowAnonymous]
[ValidateAntiForgeryToken]
0 referencias
public async Task<ActionResult> Login(string usuario, string contraseña)
{
    if (string.IsNullOrEmpty(usuario) || string.IsNullOrEmpty(contraseña))
    {
        TempData["Error"] = "Debe ingresar usuario y contraseña.";
        return RedirectToAction("Login");
    }

    var cred = await db.Credencial
        .Include(c => c.Rol)
        .FirstOrDefaultAsync(c => c.usuario == usuario);

    if (cred == null || !PasswordHelper.VerifyPassword(contraseña, cred.contrasena))
    {
        TempData["Error"] = "Usuario o contraseña incorrectos.";
        return RedirectToAction("Login");
    }

    // Migración automática de hash SHA256 legado + PBKDF2
    if (PasswordHelper.IsLegacyHash(cred.contrasena))
    {
        cred.contrasena = PasswordHelper.HashPassword(contraseña);
        await db.SaveChangesAsync();
    }

    if (!cred.activo)
    {
        TempData["Error"] = "La cuenta está inactiva.";
        return RedirectToAction("Login");
    }
}
```

Fuente: elaboración propia.

El fragmento de código mostrado se ubica en el archivo `Controllers\AccountController.cs`, desde la línea 33 hasta la línea 65. El método está decorado con los atributos `[HttpPost]` y `[ValidateAntiForgeryToken]`, lo que garantiza que únicamente acepta envíos de formulario y protege contra ataques CSRF. Los parámetros `string` usuario y `string` contraseña constituyen las entradas recibidas desde la vista. La verificación se realiza con `PasswordHelper.VerifyPassword`, comparando la contraseña ingresada contra el hash PBKDF2 almacenado en la base de datos, asegurando que la entrada sensible nunca se compare en texto plano.

Registro de solicitud de permiso laboral (Módulo Gestionar Permisos)

El segundo ejemplo de entrada corresponde al formulario de creación de solicitudes de permiso en el módulo Gestionar Permisos. El método `CrearSolicitudPermiso` del `PermisoController` recibe un objeto de tipo `Permiso` que encapsula los datos del formulario, incluyendo el tipo de permiso, las fechas de inicio y fin, y una justificación. La verificación de `ModelState.IsValid` garantiza que todos los campos obligatorios fueron completados según las anotaciones de validación definidas en el modelo. Este tipo de entrada representa la captura de una solicitud formal del colaborador que deberá seguir un flujo de aprobación hasta ser procesada por la jefatura.

Figura 38

Registro de solicitud de permiso laboral.

```
[HttpPost]
[ValidateAntiForgeryToken]
0 referencias
public ActionResult CrearSolicitudPermiso(Permiso permiso)
{
    if (ModelState.IsValid)
    {
        permiso.Empleado_idEmpleado = ObtenerEmpleadoIdSesion();
        permiso.estado = "Pendiente";
        permiso.fecha_registro = DateTime.Now;

        _db.Permiso.Add(permiso);
        _db.SaveChanges();

        return RedirectToAction("MisSolicitudes");
    }

    CargarTiposPermiso();
    return View(permiso);
}
```

Fuente: elaboración propia.

El fragmento se localiza en `Controllers\PermisoController.cs`, líneas 186 a 203. Una vez validado el modelo, el sistema asigna automáticamente el identificador del empleado desde la sesión activa, establece el estado inicial como "Pendiente" y registra la fecha y hora del sistema como `fecha_registro`, con lo que se asegura que el colaborador no pueda manipular estos campos desde el formulario. Finalmente, el objeto se persiste en la base de datos mediante `SaveChanges` y el usuario es redirigido a su lista de solicitudes. Si la validación falla, se vuelve a cargar el formulario con los mensajes de error correspondientes.

Salidas

Listado de sesiones activas del sistema (Módulo de seguridad)

La primera salida ejemplificada es el listado de sesiones activas de usuarios, visible exclusivamente para el Administrador del sistema desde el módulo de Seguridad. El método `SesionesActivas` del `SeguridadController` consulta la tabla `SesionUsuario` incluyendo las relaciones con `Credencial` y `Rol`, filtrando únicamente los registros donde el campo `activo` es verdadero y ordenando los resultados de forma descendente por fecha de inicio. Esta salida permite al administrador visualizar en tiempo real quién tiene sesión abierta en el sistema, con la posibilidad de cerrar sesiones de forma remota en caso de ser necesario.

Figura 39

Listado de sesiones activas del sistema.

```
public ActionResult SesionesActivas()
{
    var sesiones = _db.SesionUsuario
        .Include(s => s.Credencial)
        .Include(s => s.Credencial.Rol)
        .Where(s => s.activo)
        .OrderByDescending(s => s.fecha_inicio)
        .ToList();

    return View(sesiones);
}
```

Fuente: elaboración propia.

El código se localiza en `Controllers\SeguridadController.cs`, líneas 25 a 35. El controlador está restringido con el atributo `[Authorize (Roles = "Administrador")]`, lo que impide que cualquier otro rol acceda a esta información. La lista resultante se pasa directamente a la vista como modelo fuertemente tipado, donde se presentan columnas como el usuario, el rol asignado, la dirección IP de origen y la fecha y hora de inicio de sesión.

Adicionalmente, el método `CerrarSesion` permite terminar una sesión específica, registrando la acción en la bitácora del sistema para fines de auditoría.

Comprobante de nómina individual del colaborador (Módulo Gestionar Nóminas)

La segunda salida presentada corresponde al comprobante de nómina individual que el sistema pone a disposición de cada colaborador.

El método `MiDetalleNomina` en `NominaController` consulta el detalle de nómina vinculado al empleado en sesión, cargando mediante `Include` los datos relacionados como `Nomina`, `Empleado`, `Persona`, `Puesto` y `SalarioBrutoMensual`.

Adicionalmente, el método calcula de forma dinámica el desglose de tramos de renta según la tabla `TramosParametrosRenta` del año correspondiente, lo que garantiza que el comprobante refleje con exactitud los cálculos aplicados durante el período de pago.

Figura 40*Salida de comprobante de nómina individual.*

```

public ActionResult MDetalleNomina(int id)
{
    int empId = Session["EmpleadoId"] != null ? (int)Session["EmpleadoId"] : 0;
    if (empId == 0) return RedirectToAction("Index", "Home");

    var detalle = _db.DetalleNomina
        .Include(d => d.Nomina)
        .Include(d => d.Empleado.Persona)
        .Include(d => d.Empleado.Puesto)
        .Include(d => d.SalarioBrutoMensual)
        .Include(d => d.DetalleRenta)
        .FirstOrDefault(d => d.idDetalleNomina == id && d.Empleado_idEmpleado == empId);

    if (detalle == null)
    {
        TempData["Error"] = "No se encontró el detalle de nómina.";
        return RedirectToAction("MHistorialNomina");
    }

    // Cargar aportes con catálogo en consulta directa para evitar el Include anidado
    var aportes = _db.DetalleAporte
        .Include(a => a.CatalogoDeduction)
        .Where(a => a.DetalleNomina_idDetalleNomina == id)
        .ToList();

    ViewBag.DetalleAportes = aportes
        .Where(a => a.CatalogoDeduction != null && !a.CatalogoDeduction.es_carga_patronal)
        .OrderBy(a => a.CatalogoDeduction.orden)
        .ToList();

    // Recalcular renta escalonada para mostrar desglose por tramo en el comprobante.
    // Se usa total_percepciones (salario bruto guardado) con los tramos del año de la nómina.
    int anioNomina = detalle.Nomina.anio;
    decimal salarioBruto = detalle.total_percepciones;

    var tramos = _db.TramosParametrosRenta
        .Where(t => t.anio == anioNomina)
        .OrderBy(t => t.limite_inferior)
        .ToList();

    var tramosAplicados = new List<Servicios.TramoRentaAplicado>();
    foreach (var t in tramos)
    {
        if (t.porcentaje == 0m || salarioBruto <= t.limite_inferior) continue;
        decimal techo = Math.Min(salarioBruto, t.limite_superior);
        decimal porcion = techo - t.limite_inferior;
        if (porcion > 0m)
            tramosAplicados.Add(new Servicios.TramoRentaAplicado
            {
                NombreTramo = t.tramo,
                LimiteInferior = t.limite_inferior,
                LimiteSuperior = t.limite_superior,
                Porcentaje = t.porcentaje,
                BaseGravable = porcion,
                Impuesto = Math.Round(porcion * (t.porcentaje / 100m), 2)
            });
    }

    ViewBag.TramosRenta = tramosAplicados;
    return View(detalle);
}

```

Fuente: elaboración propia.

El fragmento se encuentra en `Controllers\NominaController.cs`, líneas 393 a 454. La salida se construye cargando los aportes del detalle mediante `DetalleAporte` incluyendo el catálogo de deducciones, y separando los aportes del empleado de las cargas patronales con la propiedad `es_carga_patronal`. Los tramos de renta aplicados se calculan iterando sobre los registros de `TramosParametrosRenta` para el año de la nómina, determinando la porción gravable y el impuesto de cada tramo. Toda esta información se envía a la vista a través de `ViewBag` para ser renderizada en el comprobante en formato estructurado y comprensible para el empleado.

Procesos

Cálculo de aguinaldo según legislación costarricense (Módulo Calcular Aguinaldo)

El primer proceso presentado corresponde al cálculo del aguinaldo, implementado en la capa de servicios dentro de la clase `AguinaldoService`. El método `CalcularAguinaldo` recibe el

identificador del empleado y el año de cálculo, determinando automáticamente el período comprendido entre el 1 de diciembre del año anterior y el 30 de noviembre del año en curso, conforme a lo establecido en el artículo 166 del Código de Trabajo costarricense. Si el empleado ingresó después del inicio del período, la fecha de inicio se ajusta a la fecha de ingreso para garantizar la proporcionalidad del cálculo.

Figura 41
Proceso de cálculo del aguinaldo.

```

5 referencias
public ResultadoAguinaldo CalcularAguinaldo(int empleadoId, int anio)
{
    var empleado = _db.Empleado
        .Include(e => e.Persona)
        .Include(e => e.Puesto)
        .FirstOrDefault(e => e.idEmpleado == empleadoId);

    if (empleado == null)
        throw new Exception("Empleado no encontrado.");

    if (empleado.Persona == null)
        throw new Exception("El empleado no tiene datos personales registrados.");

    DateTime fechaInicioPeriodo = new DateTime(anio - 1, 12, 1);
    DateTime fechaFinPeriodo = new DateTime(anio, 11, 30);

    // Si ingresó después del inicio del periodo, ajustar
    if (empleado.fecha_ingreso > fechaInicioPeriodo)
        fechaInicioPeriodo = empleado.fecha_ingreso;

    // Claves comparables: año*100+mes
    int claveInicio = fechaInicioPeriodo.Year * 100 + fechaInicioPeriodo.Month;
    int claveFin = fechaFinPeriodo.Year * 100 + fechaFinPeriodo.Month;

    var salarios = _db.SalarioBrutoMensual
        .Where(s => s.Empleado_idEmpleado == empleadoId)
        .ToList()
        .Where(s =>
        {
            int clave = s.anio * 100 + s.mes;
            return clave >= claveInicio && clave <= claveFin;
        })
        .OrderBy(s => s.anio).ThenBy(s => s.mes)
        .ToList();

    decimal sumatoria = salarios.Sum(s => s.salario_bruto);
    decimal monto = Math.Round(sumatoria / 12m, 2);

    bool yaGuardado = _db.Aguinaldo.Any(a =>
        a.Empleado_idEmpleado == empleadoId && a.anio == anio);

    var resultado = new ResultadoAguinaldo
    {
        EmpleadoId = empleadoId,
        NombreEmpleado = $"{empleado.Persona.nombre} {empleado.Persona.apellido1} {empleado.Persona.apellido2}".Trim(),
        CedulaEmpleado = empleado.Persona.cedula,
        Anio = anio,
        FechaInicioPeriodo = fechaInicioPeriodo,
        FechaFinPeriodo = fechaFinPeriodo,
        SumatoriaSalarios = sumatoria,
        MesesConsiderados = salarios.Count,
        MontoAguinaldo = monto,
        YaGuardado = yaGuardado
    };
}

```

Fuente: elaboración propia.

El fragmento se ubica en Servicios\AguinaldoService.cs, líneas 21 a 75. El proceso consiste en obtener todos los registros de SalarioBrutoMensual del empleado que correspondan al período definido, sumar los salarios brutos de los meses incluidos y dividir entre doce para obtener el monto del aguinaldo, redondeando a dos decimales. La clave de comparación año*100+mes permite filtrar con precisión los registros dentro del rango sin depender de consultas de rango de fechas complejas. El resultado se encapsula en un objeto ResultadoAguinaldo que incluye el detalle mes a mes para su presentación en pantalla y eventual almacenamiento.

Sincronización automática de marcaciones a registros de asistencia (Módulo Gestionar Asistencia)

El segundo proceso ilustra la sincronización automática que ocurre cada vez que un empleado registra una marcación de entrada o salida. El método privado SincronizarAsistencia en MarcacionController obtiene todas las marcaciones del día para el empleado y determina, con base en el horario configurado, si hubo tardía, cuántas horas trabajó en su jornada regular y si los minutos de almuerzo deben descontarse del tiempo neto. Este proceso opera de forma transparente al usuario y garantiza que la tabla Asistencia siempre refleje el estado actualizado tras cada marcación registrada.

Figura 42

Proceso de sincronización de marcaciones a asistencia.

```
private void SincronizarAsistencia(int empleadoId, DateTime fecha)
{
    var marcaciones = db.RegistroMarcacion
        .Where(m => m.Emppleado_idEmpleado == empleadoId && m.fecha == fecha)
        .OrderBy(m => m.hora)
        .ToList();

    var asistencia = db.Asistencia
        .FirstOrDefault(a => a.Emppleado_idEmpleado == empleadoId && a.fecha == fecha);

    // Crear registro de asistencia si no existe
    if (asistencia == null)
    {
        asistencia = new Asistencia
        {
            Emppleado_idEmpleado = empleadoId,
            fecha = fecha,
            estado = "Pendiente",
            es_justificada = false,
            horas_trabajadas = 0,
            horas_netas = 0,
            horas_extra = 0,
            tiene_tardia = false,
            minutos_tardia = 0,
            observacion = null,
            fecha_registro = DateTime.Now,
            fecha_actualizacion = DateTime.Now
        };
        db.Asistencia.Add(asistencia);
    }

    // Solo la PRIMERA entrada y la PRIMERA salida corresponden a la jornada normal
    var primeraEntrada = marcaciones.FirstOrDefault(m => m.tipo == "Entrada");
    var primeraSalida = marcaciones.FirstOrDefault(m => m.tipo == "Salida");

    asistencia.hora_entrada = primeraEntrada?.hora;
    asistencia.hora_salida = primeraSalida?.hora;

    // --- Restablecer tardía ---
    asistencia.tiene_tardia = false;
    asistencia.minutos_tardia = 0;

    if (primeraEntrada != null)
    {
        // Cargar el horario del empleado para este día
        var empleado = db.Emppleado.FirstOrDefault(e => e.idEmpleado == empleadoId);

        // Minutos de almuerzo configurados en la jornada (default 60 si no tiene jornada asignada)
        double minutosAlmuerzo = 60.0;
        if (empleado?.Jornada_idJornada != null)
        {
            var jornada = db.Jornada.Find(empleado.Jornada_idJornada);
            if (jornada != null)
            {
                minutosAlmuerzo = jornada.minutos_almuerzo;
            }
            double horasAlmuerzo = minutosAlmuerzo / 60.0;
        }

        if (empleado?.HorarioSemanal_idHorarioSemanal != null)
        {
            string nombreDia = ObtenerNombreDia(fecha.DayOfWeek);

            var horario = db.Horario.FirstOrDefault(h =>
                h.HorarioSemanal_idHorarioSemanal == empleado.HorarioSemanal_idHorarioSemanal
                && h.dia_semana == nombreDia);

            if (horario == null)
            {
                // El empleado tiene HorarioSemanal pero hoy no está en él = día fuera de horario
                asistencia.estado = "Fuera de Horario";
                asistencia.es_justificada = false;
                asistencia.horas_trabajadas = 0;
                asistencia.horas_netas = 0;
                asistencia.horas_extra = 0;
                asistencia.tiene_tardia = false;
                asistencia.minutos_tardia = 0;
                asistencia.hora_entrada = primeraEntrada?.hora;
                asistencia.hora_salida = db.RegistroMarcacion
                    .Where(m => m.Emppleado_idEmpleado == empleadoId && m.fecha == fecha && m.tipo == "Salida")
                    .OrderBy(m => m.hora)
                    .FirstOrDefault()?.hora;
                asistencia.observacion = "[FUERA DE HORARIO] Marcación en día no laborable según horario asignado. No se contabiliza en nómina.";
                asistencia.fecha_actualizacion = DateTime.Now;
                return;
            }
        }
    }
}
```

Fuente: elaboración propia.

El código se encuentra en `Controllers\MarcacionController.cs`, líneas 453 a 535. Si el empleado tiene un `HorarioSemanal` asignado, el proceso lo compara contra la primera entrada y la primera salida del día para detectar tardías superando la tolerancia de diez minutos configurada como constante. Las horas efectivas se calculan limitando la entrada al inicio oficial del horario y la salida al fin del horario, de modo que el tiempo fuera de jornada no se contabiliza en horas normales, sino que se registra como horas extra. Las horas netas se obtienen restando los minutos de almuerzo definidos en la jornada del empleado.

Validaciones

Reglas de negocio para el registro de marcaciones (Módulo Gestionar Asistencia)

La primera validación corresponde al conjunto de reglas de negocio que el sistema aplica antes de aceptar cualquier marcación de asistencia. En el método `MarcarEntradaSalida` del `MarcacionController`, se evalúan condiciones como: impedir que el colaborador marque entrada más de treinta minutos antes de su horario configurado, bloquear una segunda entrada si la primera salida aún no se ha registrado, y requerir confirmación explícita del colaborador antes de iniciar un segundo lote de marcaciones correspondiente a horas extra. Cada validación devuelve un mensaje descriptivo a través de `TempData["Error"]` para orientar al usuario sobre la acción correcta.

Figura 43

Validaciones de reglas de negocio para marcaciones de asistencia.

```
if (tipo == "Entrada")
{
    // --- Llegada demasiado temprana (más de 30 min antes del horario) ---
    if (horario != null
        && ahora.TimeOfDay < horario.hora_inicio.Subtract(TimeSpan.FromMinutes(MINUTOS_MAXIMO_ANTICIPADO)))
    {
        int min = (int)(horario.hora_inicio - ahora.TimeOfDay).TotalMinutes;
        TempData["Error"] = $"No puede marcar entrada aún. "
            + $"Está (min) minuto(s) antes de su horario ({horario.hora_inicio:hh\\:mm}). "
            + $"El registro se habilita {MINUTOS_MAXIMO_ANTICIPADO} minutos antes del inicio.";
        return RedirectToAction("Marcar");
    }

    // --- Ya tiene entrada pero no salida + debe marcar salida primero ---
    if (ne == 1 && ns == 0)
    {
        TempData["Error"] = "Ya registró su entrada hoy. "
            + "Debe marcar salida antes de iniciar una nueva jornada.";
        return RedirectToAction("Marcar");
    }

    // --- Segunda entrada (Horas Extra) + requiere confirmación ---
    if (esSegundoLote && !confirmarHorasExtra)
    {
        TempData["Error"] = "Debe confirmar que desea registrar horas extra antes de continuar.";
        return RedirectToAction("Marcar");
    }

    // --- Tercera entrada o más + jornada ya completa ---
    if (ne >= 2)
    {
        TempData["Error"] = "Ya completó su jornada regular y sus horas extra del día.";
        return RedirectToAction("Marcar");
    }

    // Si es horas extra, anotar en la observación
    if (esSegundoLote)
        observacion = "[HORA EXTRA] " + (string.IsNullOrWhiteSpace(observacion) ? "" : observacion.Trim());

    // Si es día fuera de horario, anotar en la observación
    if (esDiaFueraDeHorario)
        observacion = "[FUERA DE HORARIO] " + (string.IsNullOrWhiteSpace(observacion) ? "" : observacion.Trim());
}
```

Fuente: elaboración propia.

El fragmento se localiza en `Controllers\MarcacionController.cs`, líneas 200 a 244. Las validaciones están organizadas según el tipo de marcación recibida, diferenciando el bloque `if` (tipo `== "Entrada"`) del bloque `else if` (tipo `== "Salida"`). Para la entrada, se verifica que la hora actual no supere el límite anticipado definido en la constante `MINUTOS_MAXIMO_ANTICIPADO`, que no exista ya una entrada sin salida correspondiente, y que las horas extra sean confirmadas explícitamente mediante el parámetro `confirmarHorasExtra`. Para la salida, se valida que exista al menos una entrada previa y que la jornada no se encuentre ya completa, evitando marcaciones duplicadas o fuera de secuencia.

Validación de datos de incapacidad laboral (Módulo Gestionar Incapacidad)

La segunda validación presentada corresponde a las comprobaciones que se ejecutan al registrar una incapacidad laboral en el sistema. El método `CrearIncapacidad` del `IncapacidadController` verifica que la fecha de fin no sea anterior a la fecha de inicio, garantizando la coherencia del período de incapacidad. Adicionalmente, valida que el porcentaje de pago esté dentro del rango permitido entre cero y uno, equivalente a 0% y 100% respectivamente, conforme a las reglas de pago del Seguro Social costarricense. El cálculo automático de días de incapacidad se realiza como resultado derivado de estas fechas validadas.

Figura 44

Validaciones de datos en el registro de incapacidades.

```
// =====
// Validaciones legales
// =====
if (incapacidad.fecha_fin < incapacidad.fecha_inicio)
    throw new Exception("La fecha fin no puede ser menor a la fecha inicio.");

if (incapacidad.porcentaje_pago < 0 || incapacidad.porcentaje_pago > 1)
    throw new Exception("El porcentaje de pago debe estar entre 0 y 1 (ejemplo: 0.60 = 60%).");

// =====
// Origen (OBLIGATORIO)
// =====
if (User.IsInRole("Colaborador"))
    incapacidad.origen = "Colaborador";
else if (User.IsInRole("RRHH"))
    incapacidad.origen = "RRHH";
else if (User.IsInRole("Jefe") || User.IsInRole("Jefa"))
    incapacidad.origen = "Jefatura";
else
    incapacidad.origen = "Administrador";

// =====
// Campos del sistema
// =====
incapacidad.estado = "Pendiente";
incapacidad.fecha_registro = DateTime.Now;
incapacidad.fecha_actualizacion = null;
incapacidad.es_prorroga = false;
incapacidad.afecta_asistencia = true;

incapacidad.dias_incapacidad =
    (incapacidad.fecha_fin - incapacidad.fecha_inicio).Days + 1;

_db.Incapacidad.Add(incapacidad);
_db.SaveChanges();

TempData["Success"] = "Incapacidad registrada correctamente.";
return RedirectToAction("Index");
}
```

Fuente: elaboración propia.

El fragmento se ubica en `Controllers\IncapacidadController.cs`, líneas 115 a 143. Las validaciones se implementan mediante instrucciones `if` que lanzan excepciones descriptivas capturadas en el bloque `catch` externo, lo que centraliza el manejo de errores en un único punto y evita código duplicado.

Si las fechas son válidas y el porcentaje está dentro del rango permitido, el sistema asigna automáticamente los campos del sistema como el estado inicial "Pendiente", la fecha de registro, la bandera `afecta_asistencia` y el campo `es_prorroga`. La cantidad de días de incapacidad se calcula restando las fechas y sumando uno para incluir el día de inicio en el conteo total.

Módulos señalados en el alcance

Gestionar Nóminas – Proceso masivo de cálculo y guardado de nómina

El módulo de Gestionar Nóminas es uno de los más complejos del sistema, ya que centraliza el cálculo de los salarios de todos los empleados para un período determinado. El método `CalcularMasiva` del `NominaController` recibe un arreglo de identificadores de empleados junto con el mes y el año del período a procesar.

Antes de ejecutar cualquier cálculo, se valida que se haya seleccionado al menos un empleado y que el período ingresado sea válido dentro de los rangos permitidos.

Para cada empleado, se invoca el servicio `NominaService` que centraliza toda la lógica de cálculo, y los resultados se acumulan para su presentación.

Figura 45*Proceso masivo de cálculo de nómina por período*

```

public ActionResult CalcularMasiva(int[] empleadoIds, int mes, int anno)
{
    if (empleadoIds == null || empleadoIds.Length == 0)
    {
        TempData["Error"] = "Debe seleccionar al menos un empleado.";
        return RedirectToAction("CalcularNomina");
    }

    if (mes < 1 || mes > 12 || anno < 2000 || anno > DateTime.Now.Year + 1)
    {
        TempData["Error"] = "Periodo inválido.";
        return RedirectToAction("CalcularNomina");
    }

    DateTime fechaInicio = new DateTime(anno, mes, 1);
    DateTime fechaFin = fechaInicio.AddMonths(1).AddDays(-1);
    var service = new NominaService(_db);

    var resultados = new List<ResultadoNomina>();
    var errores = new List<string>();
    var nominaCabecera = _db.Nomina.FirstOrDefault(n => n.mes == (byte)mes && n.anno == anno);

    foreach (var empId in empleadoIds)
    {
        try
        {
            var res = service.CalcularNominaEmpleado(empId, fechaInicio, fechaFin);

            // Marcar si ya tiene detalle guardado en este período
            res.TieneNominaGuardada = nominaCabecera != null &&
                _db.DetalleNomina.Any(d =>
                    d.Nomina_idNomina == nominaCabecera.idNomina &&
                    d.Empleado_idEmpleado == empId);

            // Datos de puesto/departamento para la tabla resumen
            var emp = _db.Empleado
                .Include(e => e.Puesto)
                .Include(e => e.Puesto.Departamento)
                .Include(e => e.Jornada)
                .FirstOrDefault(e => e.idEmpleado == empId);

            res.Departamento = emp?.Puesto?.Departamento?.nombre ?? "-";
            res.NombrePuesto = emp?.Puesto?.nombre ?? "-";
            res.NombreJornada = emp?.Jornada?.nombre ?? "-";
            res.EnPlanilla = emp?.estado == "Activo";

            resultados.Add(res);
        }
        catch (Exception ex)
        {
            var empFail = _db.Empleado.Include(e => e.Persona).FirstOrDefault(e => e.idEmpleado == empId);
            string nombre = empFail?.Persona != null
                ? $"{empFail.Persona.nombre} {empFail.Persona.apellido}"
                : $"{ID {empId}}";
            errores.Add($"{nombre}: {ex.Message}");
        }
    }

    ViewBag.Mes = mes;
    ViewBag.Anno = anno;
    ViewBag.Errores = errores;

    return View("ResultadosMasivo", resultados);
}

```

Fuente: elaboración propia.

El fragmento se localiza en `Controllers\NominaController.cs`, líneas 63 a 131. El proceso itera sobre cada empleado seleccionado e invoca `CalcularNominaEmpleado`, marcando si ya tiene nómina guardada en el período mediante una consulta a `DetalleNomina`. Los datos de departamento, puesto y jornada se enriquecen con `Include` para la tabla de resultados. Los errores por empleado se capturan individualmente con `try-catch` para que un fallo en un registro no interrumpa el procesamiento de los demás. El resultado final es una lista de objetos `ResultadoNomina` que se presenta al usuario antes de confirmar el guardado.

Gestionar Asistencia – Registro de marcación de entrada y salida

El módulo de Gestionar Asistencia implementa el control de la jornada laboral a través del registro de marcaciones digitales. El método POST MarcarEntradaSalida del MarcacionController recibe el tipo de marcación Entrada o Salida y una observación opcional. Tras verificar la identidad del empleado desde la sesión, el sistema determina cuántas entradas y salidas existen en el día actual para establecer si corresponde una jornada normal o el inicio de horas extra. El horario programado del día se obtiene de la tabla Horario vinculada al HorarioSemanal asignado al empleado, y se usa como referencia para calcular tardías y tiempo fuera de jornada.

Figura 46

Registro de marcación de asistencia de entrada y salida (Parte 1)

```
[HttpPost]
[ValidateAntiForgeryToken]
0 referencias
public async Task<ActionResult> MarcarEntradaSalida(
    string tipo,
    string observacion,
    bool confirmarHorasExtra = false)
{
    int? empId = Session["EmpleadoId"] as int?;

    if (empId == null)
    {
        TempData["Error"] = "No se pudo identificar al empleado.";
        return RedirectToAction("Marcar");
    }

    if (string.IsNullOrWhiteSpace(tipo))
    {
        TempData["Error"] = "Debe seleccionar Entrada o Salida.";
        return RedirectToAction("Marcar");
    }

    DateTime ahora = DateTime.Now;
    DateTime hoy = ahora.Date;

    // — Marcaciones de hoy —
    var marcacionesHoy = await db.RegistroMarcacion
        .Where(r => r.Empleado_idEmpleado == empId.Value && r.fecha == hoy)
        .OrderBy(r => r.hora)
        .ToListAsync();

    var entradasHoy = marcacionesHoy.Where(m => m.tipo == "Entrada").ToList();
    var salidasHoy = marcacionesHoy.Where(m => m.tipo == "Salida").ToList();

    int ne = entradasHoy.Count;
    int ns = salidasHoy.Count;

    // — Horario del empleado —
    var empleado = await db.Empleado
        .FirstOrDefaultAsync(e => e.idEmpleado == empId.Value);

    var horario = await ObtenerHorarioDelDiaAsync(
        empleado?.HorarioSemanal_idHorarioSemanal, ahora.DayOfWeek);

    // Día fuera de horario: el empleado TIENE HorarioSemanal pero hoy no está configurado
    bool esDiaFueraDeHorario = empleado?.HorarioSemanal_idHorarioSemanal.HasValue == true && horario == null;

    bool esSegundoLote = (ne >= 1 && ns >= 1); // jornada normal completa
```

Fuente: elaboración propia.

Figura 47*Registro de marcación de asistencia de entrada y salida (Parte 2)*

```

if (tipo == "Entrada")
{
    // — Llegada demasiado temprana (más de 30 min antes del horario) —
    if (horario != null
        && ahora.TimeOfDay < horario.hora_inicio.Subtract(TimeSpan.FromMinutes(MINUTOS_MAXIMO_ANTICIPADO)))
    {
        int min = (int)(horario.hora_inicio - ahora.TimeOfDay).TotalMinutes;
        TempData["Error"] = $"No puede marcar entrada aún. "
            + $"Está {min} minuto(s) antes de su horario ({horario.hora_inicio:hh\\:mm}). "
            + $"El registro se habilita {MINUTOS_MAXIMO_ANTICIPADO} minutos antes del inicio.";
        return RedirectToAction("Marcar");
    }

    // — Ya tiene entrada pero no salida → debe marcar salida primero —
    if (ne == 1 && ns == 0)
    {
        TempData["Error"] = "Ya registró su entrada hoy. "
            + "Debe marcar salida antes de iniciar una nueva jornada.";
        return RedirectToAction("Marcar");
    }

    // — Segunda entrada (Horas Extra) → requiere confirmación —
    if (esSegundoLote && !confirmarHorasExtra)
    {
        TempData["Error"] = "Debe confirmar que desea registrar horas extra antes de continuar.";
        return RedirectToAction("Marcar");
    }

    // — Tercera entrada o más → jornada ya completa —
    if (ne >= 2)
    {
        TempData["Error"] = "Ya completó su jornada regular y sus horas extra del día.";
        return RedirectToAction("Marcar");
    }

    // Si es horas extra, anotar en la observación
    if (esSegundoLote)
        observacion = "[HORA EXTRA] " + (string.IsNullOrWhiteSpace(observacion) ? "" : observacion.Trim());

    // Si es día fuera de horario, anotar en la observación
    if (esDiaFueraDeHorario)
        observacion = "[FUERA DE HORARIO] " + (string.IsNullOrWhiteSpace(observacion) ? "" : observacion.Trim());
}
else if (tipo == "Salida")
{
    // — Sin entrada previa —
    if (ne == 0)
    {
        TempData["Error"] = "No puede registrar salida sin haber marcado entrada primero.";
        return RedirectToAction("Marcar");
    }
}

```

Fuente: elaboración propia.

Figura 48*Registro de marcación de asistencia de entrada y salida (Parte 3)*

```

// — Jornada normal completa, pero sin entrada HE —————
if (ne == 1 && ns == 1)
{
    TempData["Error"] = "Su jornada normal ya fue completada. "
        + "Para registrar salida de horas extra, primero debe marcar la entrada de horas extra.";
    return RedirectToAction("Marcar");
}

// — Todo ya completo —————
if (ne >= 2 && ns >= 2)
{
    TempData["Error"] = "Ya completó su jornada regular y sus horas extra del día.";
    return RedirectToAction("Marcar");
}

// Si es salida de horas extra, anotar en la observación
if (ne >= 2 && ns >= 1)
    observacion = "[HORA EXTRA] " + (string.IsNullOrWhiteSpace(observacion) ? "" : observacion.Trim());

// Si es día fuera de horario, anotar en la observación
if (esDiaFueraDeHorario)
    observacion = "[FUERA DE HORARIO] " + (string.IsNullOrWhiteSpace(observacion) ? "" : observacion.Trim());
}

// =====
// GUARDAR EL REGISTRO DE MARCACIÓN
// =====
var registro = new RegistroMarcacion
{
    Empleado_idEmpleado = empId.Value,
    fecha                = hoy,
    hora                 = ahora.TimeOfDay,
    tipo                 = tipo,
    origen               = "Colaborador",
    observacion           = string.IsNullOrWhiteSpace(observacion) ? null : observacion.Trim(),
    fecha_registro       = ahora
};

db.RegistroMarcacion.Add(registro);
await db.SaveChangesAsync();

// — Sincronizar con la tabla Asistencia —————
SincronizarAsistencia(empId.Value, hoy);
await db.SaveChangesAsync();

```

Fuente: elaboración propia.

El fragmento se ubica en `Controllers\MarcacionController.cs`, líneas 150 a 296. Luego de pasar las validaciones de negocio, el sistema crea un objeto `RegistroMarcacion` con la fecha, hora actual, tipo, origen y observación, y lo persiste en la base de datos. Inmediatamente después se invoca `SincronizarAsistencia` para actualizar o crear el registro correspondiente en la tabla `Asistencia`. Si se detecta una salida que supera el horario fin, el sistema calcula automáticamente las horas extra y crea una `SolicitudHorasExtra` en estado `Pendiente` mediante el método auxiliar `RegistrarHoraExtraAsync`, sin requerir acción adicional del colaborador.

Gestionar Horas Extra – Aprobación de solicitudes por jefaturas

El módulo de Gestionar Horas Extras contempla el flujo de aprobación de solicitudes generadas tanto por el sistema al detectar tiempo extra como por solicitudes manuales del colaborador. El método Aprobar del HorasExtraController verifica que el aprobador no sea el mismo empleado que realizó la solicitud, aplicando la regla de no autoAprobación establecida en los estándares del sistema. Una vez confirmada la aprobación mediante el servicio AprobarSolicitud, el sistema envía una notificación automática al colaborador a través de NotificacionHelper y registra la acción en la bitácora mediante BitacoraHelper.

Figura 49

Aprobación de solicitudes de horas extra por jefatura

```
[HttpPost]
[ValidateAntiForgeryToken]
[Authorize(Roles = "Administrador,Jefe,Jefa,RRHH")]
0 referencias
public ActionResult Aprobar(int id, string comentario)
{
    if (Session["CredencialId"] == null)
        return RedirectToAction("Login", "Account");

    int empSesion = Session["EmpleadoId"] != null ? Convert.ToInt32(Session["EmpleadoId"]) : 0;
    var sol = _db.SolicitudHorasExtra
        .Include(s => s.HoraExtra)
        .FirstOrDefault(s => s.idSolicitudHorasExtra == id);

    if (sol != null && sol.HoraExtra.Empleado_idEmpleado == empSesion)
    {
        TempData["Error"] = "No puede aprobar su propia solicitud de horas extra.";
        return RedirectToAction("Detalle", new { id });
    }

    try
    {
        int credencialId = (int)Session["CredencialId"];
        _service.AprobarSolicitud(id, credencialId);

        if (sol != null)
            NotificacionHelper.Crear(_db, sol.HoraExtra.Empleado_idEmpleado, "Horas Extra", "Aprobado",
                $"Tu solicitud de horas extra del {sol.HoraExtra.fecha:dd/MM/yyyy} fue aprobada.");

        BitacoraHelper.Registrar(_db, credencialId,
            "APROBAR HORAS EXTRA",
            $"id={id} aprobadas. Comentario: {comentario}",
            this.HttpContext);

        TempData["Success"] = "Solicitud de horas extra aprobada correctamente.";
    }
    catch (Exception ex)
    {
        TempData["Error"] = ex.Message;
    }

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

Fuente: elaboración propia.

El fragmento se encuentra en `Controllers\HorasExtraController.cs`, líneas 176 a 217. El método está protegido con `[Authorize(Roles = "Administrador,Jefe,Jefa,RRHH")]` para restringir su acceso a los roles autorizados. La verificación de autoAprobación compara el `Empleado_idEmpleado` de la solicitud contra el `EmpleadoId` almacenado en la sesión activa, asegurando que ningún colaborador pueda aprobar sus propias horas extra. El servicio `AprobarSolicitud` encapsula la lógica de actualización del estado, separando la responsabilidad del controlador de los detalles de la capa de datos. La notificación incluye la fecha de la hora extra para que el colaborador pueda identificar a cuál solicitud corresponde.

Gestionar Permisos – Aprobación de permiso con registro de historial

El módulo de Gestionar Permisos administra las solicitudes de ausencias autorizadas de los colaboradores, clasificadas por tipo de permiso según si tienen o no goce de salario. El método `Aprobar` del `PermisoController` actualiza el estado del permiso a "Aprobado", registra la fecha de actualización y crea un registro en la tabla `HistorialPermiso` con el comentario del aprobador, el identificador del usuario que ejecutó la acción y la fecha exacta. Esta trazabilidad permite auditar en el futuro quién aprobó cada permiso y bajo qué comentario. La regla de no autoAprobación también se aplica en este módulo, verificando que el aprobador sea distinto al solicitante.

Figura 50

Aprobación de solicitud de permiso con registro en historial

```
[HttpPost]
[Authorize(Roles = "Administrador,Jefe,Jefa,RRHH")]
[ValidateAntiForgeryToken]
0 referencias
public ActionResult Aprobar(int id, string comentario)
{
    var permiso = _db.Permiso.Find(id);
    if (permiso == null) return HttpNotFound();

    int empSession = Session["EmpleadoId"] != null ? Convert.ToInt32(Session["EmpleadoId"]) : 0;
    if (permiso.Empleado_idEmpleado == empSession)
    {
        TempData["Error"] = "No puede aprobar su propio permiso.";
        return RedirectToAction("Detalle", new { id });
    }

    if (permiso.estado != "Pendiente")
        return RedirectToAction("SolicitudesPendientes");

    permiso.estado = "Aprobado";
    permiso.fecha_actualizacion = DateTime.Now;

    int usuarioId = Session["CredencialId"] != null ? (int)Session["CredencialId"] : 0;
    _db.HistorialPermiso.Add(new HistorialPermiso
    {
        Permiso_idPermiso = id,
        idUsuario = usuarioId,
        accion = "Aprobado",
        comentario = comentario ?? "",
        fecha = DateTime.Now
    });

    NotificacionHelper.Crear(_db, permiso.Empleado_idEmpleado, "Permisos", "Aprobado",
        $"Tu solicitud de permiso del {permiso.fecha_inicio:dd/MM/yyyy} al {permiso.fecha_fin:dd/MM/yyyy} fue aprobada.");

    _db.SaveChanges();

    TempData["Success"] = "Permiso aprobado correctamente.";
    return RedirectToAction("SolicitudesPendientes");
}
```

Fuente: elaboración propia.

El fragmento se localiza en `Controllers\PermisoController.cs`, líneas 90 a 128. Tras verificar que el permiso existe, que el aprobador no es el solicitante y que el estado actual es "Pendiente", el sistema modifica el objeto en memoria y agrega el registro al `DbSet` de `HistorialPermiso`. La notificación al empleado incluye las fechas de inicio y fin del permiso para contextualizarla. El método `SaveChanges` persiste simultáneamente tanto el cambio de estado del permiso como el nuevo registro de historial en una sola transacción, garantizando la consistencia entre ambas tablas sin posibilidad de que uno se guarde sin el otro.

Gestionar Incapacidades – Impacto en registros de asistencia al aprobar

El módulo de Gestionar Incapacidades incluye una funcionalidad diferenciadora que consiste en impactar automáticamente los registros de asistencia del período cubierto por la incapacidad al momento de su aprobación. El método privado `AplicarImpactoAsistencia` del `IncapacidadController` itera día a día desde la fecha de inicio hasta la fecha de fin de la incapacidad. Para cada fecha, busca si existe un registro en la tabla `Asistencia`; si no existe lo crea con estado "Incapacidad" y `es_justificada` en verdadero, y si ya existe lo actualiza con los mismos valores, con lo que se asegura que la asistencia refleje correctamente el período de baja.

Figura 51

Proceso de impacto de incapacidad en registros de asistencia

```
private void AplicarImpactoAsistencia(Incapacidad incapacidad)
{
    DateTime fecha = incapacidad.fecha_inicio;

    while (fecha <= incapacidad.fecha_fin)
    {
        var asistencia = _db.Asistencia.FirstOrDefault(a =>
            a.Empleado_idEmpleado == incapacidad.Empleado_idEmpleado &&
            DbFunctions.TruncateTime(a.fecha) == fecha.Date);

        if (asistencia == null)
        {
            asistencia = new Asistencia
            {
                Empleado_idEmpleado = incapacidad.Empleado_idEmpleado,
                fecha = fecha,
                estado = "Incapacidad",
                es_justificada = true,
                horas_trabajadas = 0,
                observacion = "Incapacidad aprobada",
                fecha_registro = DateTime.Now
            };

            _db.Asistencia.Add(asistencia);
        }
        else
        {
            asistencia.estado = "Incapacidad";
            asistencia.es_justificada = true;
            asistencia.horas_trabajadas = 0;
            asistencia.observacion = "Incapacidad aprobada";
            asistencia.fecha_actualizacion = DateTime.Now;
        }

        fecha = fecha.AddDays(1);
    }
}
```

Fuente: elaboración propia.

El fragmento se encuentra en `Controllers\IncapacidadController.cs`, líneas 280 a 316. La iteración utiliza un ciclo `while` que avanza la fecha en un día con `AddDays(1)` hasta sobrepasar la fecha de fin. La búsqueda en la tabla `Asistencia` usa `DbFunctions.TruncateTime` para comparar únicamente la parte de fecha del campo, independientemente de la hora almacenada. Las horas trabajadas se establecen en cero para todos los días del período, las observaciones se documentan con el texto "Incapacidad aprobada" y la bandera es `_justificada` en verdadero evita que esos días sean contabilizados como ausencias injustificadas en los reportes de nómina.

Calcular Vacaciones – Cálculo y presentación del saldo disponible

El módulo de Calcular Vacaciones permite a los colaboradores conocer su saldo disponible de días de vacaciones antes de realizar una solicitud. El método `Index` del `VacacionesController` invoca el método auxiliar `ObtenerOCalcularSaldo` cuando detecta un `EmpleadoId` válido en sesión, y almacena el resultado en `ViewBag.Saldo` para que la vista lo presente de forma prominente en la pantalla principal del módulo. Adicionalmente, si el usuario tiene un rol de jefatura o administración, el sistema contabiliza el total de solicitudes pendientes de aprobación para alertar sobre la carga de gestión disponible en el módulo.

Figura 52

Cálculo y presentación del saldo de vacaciones disponible

```
public ActionResult Index()
{
    bool esGestor = User.IsInRole("Administrador") || User.IsInRole("RRHH") ||
                   User.IsInRole("Jefe") || User.IsInRole("Jefa");
    ViewBag.EsGestor = esGestor;

    if (Session["EmpleadoId"] != null)
    {
        int empId = Convert.ToInt32(Session["EmpleadoId"]);
        if (empId > 0)
            ViewBag.Saldo = ObtenerOCalcularSaldo(empId);
    }

    if (esGestor)
        ViewBag.TotalPendientes = _db.Vacaciones.Count(v => v.estado == "Pendiente");

    return View();
}
```

Fuente: elaboración propia.

El fragmento se localiza en `Controllers\VacacionesController.cs`, líneas 18 a 35. El cálculo del saldo se delega al método `ObtenerOCalcularSaldo`, que considera la fecha de ingreso del empleado para determinar cuántos días le han acumulado conforme al artículo 153 del Código de Trabajo costarricense, restando los días ya utilizados según los registros de vacaciones aprobadas existentes.

El atributo del controlador `[Authorize (Roles ="Administrador,RRHH,Jefe,Jefa,Colaborador")]` garantiza que únicamente los roles definidos en el alcance del módulo puedan acceder a esta funcionalidad, excluyendo roles como Contador que no tienen relevancia en la gestión de vacaciones.

Calcular Aguinaldo – Servicio de cálculo proporcional por período

El módulo de Calcular Aguinaldo implementa la fórmula establecida en el artículo 166 del Código de Trabajo de Costa Rica. El servicio `AguinaldoService` contiene el método `CalcularAguinaldo` que determina el período de diciembre a noviembre, obtiene todos los salarios brutos mensuales registrados para el empleado dentro de ese período, los suma y los divide entre doce para obtener el monto proporcional.

El ajuste de la fecha de inicio al momento de ingreso del empleado garantiza la proporcionalidad para los casos en que el colaborador no haya laborado el período completo de doce meses.

Figura 53
Servicio de cálculo de aguinaldo proporcional

```

public ResultadoAguinaldo CalcularAguinaldo(int empleadoId, int anio)
{
    var empleado = _db.Empleado
        .Include(e => e.Persona)
        .Include(e => e.Puesto)
        .FirstOrDefault(e => e.idEmpleado == empleadoId);

    if (empleado == null)
        throw new Exception("Empleado no encontrado.");

    if (empleado.Persona == null)
        throw new Exception("El empleado no tiene datos personales registrados.");

    DateTime fechaInicioPeriodo = new DateTime(anio - 1, 12, 1);
    DateTime fechaFinPeriodo = new DateTime(anio, 11, 30);

    // Si ingresó después del inicio del periodo, ajustar
    if (empleado.fecha_ingreso > fechaInicioPeriodo)
        fechaInicioPeriodo = empleado.fecha_ingreso;

    // Claves comparables: año*100+mes
    int claveInicio = fechaInicioPeriodo.Year * 100 + fechaInicioPeriodo.Month;
    int claveFin = fechaFinPeriodo.Year * 100 + fechaFinPeriodo.Month;

    var salarios = _db.SalarioBrutoMensual
        .Where(s => s.Empleado_idEmpleado == empleadoId)
        .ToList()
        .Where(s =>
        {
            int clave = s.anio * 100 + s.mes;
            return clave >= claveInicio && clave <= claveFin;
        })
        .OrderBy(s => s.anio).ThenBy(s => s.mes)
        .ToList();

    decimal sumatoria = salarios.Sum(s => s.salario_bruto);
    decimal monto = Math.Round(sumatoria / 12m, 2);

    bool yaGuardado = _db.Aguinaldo.Any(a =>
        a.Empleado_idEmpleado == empleadoId && a.anio == anio);

    var resultado = new ResultadoAguinaldo
    {
        EmpleadoId = empleadoId,
        NombreEmpleado = $"{empleado.Persona.nombre} {empleado.Persona.apellido1} {empleado.Persona.apellido2}".Trim(),
        CedulaEmpleado = empleado.Persona.cedula,
        Anio = anio,
        FechaInicioPeriodo = fechaInicioPeriodo,
        FechaFinPeriodo = fechaFinPeriodo,
        SumatoriaSalarios = sumatoria,
        MesesConsiderados = salarios.Count,
        MontoAguinaldo = monto,
        YaGuardado = yaGuardado
    };

    foreach (var s in salarios)
    {
        resultado.Meses.Add(new DetalleAguinaldoMes
        {
            Mes = s.mes,
            Anio = s.anio,
            SalarioBruto = s.salario_bruto
        });
    }
}

```

Fuente: elaboración propia.

El fragmento se ubica en Servicios\AguinaldoService.cs, líneas 21 a 82. El filtro de salarios utiliza la clave compuesta año*100+mes para comparar de forma eficiente los rangos de tiempo sin necesidad de consultas de tipo BETWEEN en SQL, dado que EF6 tiene limitaciones con comparaciones de tuplas compuestas en LINQ. La sumatoria se calcula con el método LINQ Sum sobre la propiedad salario_bruto, y el monto final se redondea a dos decimales con Math.Round.

El campo YaGuardado indica si el aguinaldo del año ya fue persistido, lo que permite a la vista mostrar una advertencia antes de sobrescribir un cálculo previo.

Gestionar Liquidación – Validaciones previas al cálculo de liquidación laboral

El módulo de Gestionar Liquidación administra los cálculos de liquidación laboral por cesación de contrato. El método POST Calcular del LiquidacionController implementa validaciones exhaustivas antes de invocar el servicio de cálculo, verificando que se haya seleccionado un empleado, que se haya indicado el tipo de liquidación, que la fecha ingresada sea válida mediante TryParse y que la fecha de liquidación no sea futura. Este conjunto de validaciones previas al cálculo protege la integridad del proceso y asegura que los cálculos se realicen únicamente con datos coherentes y completos antes de persistirlos en la base de datos.

Figura 54

Validaciones en el cálculo de liquidación laboral

```
public ActionResult Calcular(int empleadoId, int tipoLiquidacionId,
    string fechaLiquidacion, string observaciones)
{
    CargarEmpleados();
    CargarTipos();
    ViewBag.FechaHoy = DateTime.Today.ToString("yyyy-MM-dd");

    if (empleadoId <= 0)
    {
        TempData["Error"] = "Debe seleccionar un empleado.";
        return View();
    }

    if (tipoLiquidacionId <= 0)
    {
        TempData["Error"] = "Debe seleccionar el tipo de liquidacion.";
        return View();
    }

    if (!DateTime.TryParse(fechaLiquidacion, out DateTime fecha))
    {
        TempData["Error"] = "Fecha de liquidacion invalida.";
        return View();
    }

    if (fecha > DateTime.Today)
    {
        TempData["Error"] = "La fecha de liquidacion no puede ser una fecha futura.";
        return View();
    }

    try
    {
        var resultado = _service.CalcularLiquidacion(
            empleadoId, tipoLiquidacionId, fecha, observaciones ?? "");
        return View("DetalleLiquidacion", resultado);
    }
    catch (Exception ex)
    {
        TempData["Error"] = ex.Message;
        return View();
    }
}
```

Fuente: elaboración propia.

El fragmento se encuentra en `Controllers\LiquidacionController.cs`, líneas 49 a 80. El controlador está restringido con `[Authorize(Roles = "Administrador,RRHH,Contador")]`, dado que la liquidación es un proceso administrativo-financiero que requiere permisos especializados. Cada validación fallida devuelve inmediatamente un mensaje en `TempData["Error"]` y regresa la vista con los listados de empleados y tipos de liquidación recargados, para que el usuario corrija el dato incorrecto sin perder lo que ya había seleccionado. La restricción de fecha futura impide calcular liquidaciones con fechas incorrectas que afectarían los montos proporcionales de los beneficios laborales.

Evaluar Personal – Creación de evaluación con criterios y ponderación

El módulo de Evaluar Personal permite a la jefatura registrar evaluaciones de desempeño semestrales para los colaboradores a su cargo. El método POST Crear del `EvaluacionController` recibe las cinco notas de criterios de evaluación puntualidad, calidad, trabajo en equipo, iniciativa y cumplimiento y verifica que cada una esté dentro del rango válido de uno a diez. También valida que el empleado haya sido seleccionado, que el evaluador haya ingresado su nombre y que la fecha de evaluación sea parseable como `DateTime`. Esta estructura garantiza que ninguna evaluación se guarde con datos incompletos o fuera de rango.

Figura 55

Creación de evaluación de personal con validación de criterios (Parte 1)

```
public ActionResult Crear(int empleadoId, int anio, byte semestre,
    decimal notaPuntualidad, decimal notaCalidad, decimal notaTrabajoEquipo,
    decimal notaIniciativa, decimal notaCumplimiento,
    string evaluador, string comentarios, string fechaEvaluacion)
{
    CargarEmpleados();
    ViewBag.AnioActual = DateTime.Today.Year;
    ViewBag.SemestreActual = DateTime.Today.Month <= 6 ? 1 : 2;

    if (empleadoId <= 0)
    {
        TempData["Error"] = "Debe seleccionar un empleado.";
        return View();
    }

    if (string.IsNullOrEmpty(evaluador))
    {
        TempData["Error"] = "Debe ingresar el nombre del evaluador.";
        return View();
    }

    decimal[] notas = { notaPuntualidad, notaCalidad, notaTrabajoEquipo, notaIniciativa, notaCumplimiento };
    if (notas.Any(n => n < 1m || n > 10m))
    {
        TempData["Error"] = "Todas las notas deben estar entre 1 y 10.";
        return View();
    }

    if (!DateTime.TryParse(fechaEvaluacion, out DateTime fechaEval))
    {
        TempData["Error"] = "Fecha de evaluacion invalida.";
        return View();
    }

    bool existe = _db.EvaluacionPersonal.Any(e =>
        e.Empleado_idEmpleado == empleadoId &&
        e.anio == anio && e.semestre == semestre);

    if (existe)
    {
        TempData["Error"] = $"Ya existe una evaluacion para este empleado en ese periodo.";
        return View();
    }
}
```

Fuente: elaboración propia.

Figura 56

Creación de evaluación de personal con validación de criterios (Parte 2)

```

decimal notaFinal = Math.Round(
    (notaPuntualidad + notaCalidad + notaTrabajoEquipo + notaIniciativa + notaCumplimiento) / 5m, 2);

string calificacion;
if (notaFinal >= 9m) calificacion = "Excelente";
else if (notaFinal >= 7m) calificacion = "Bueno";
else if (notaFinal >= 5m) calificacion = "Regular";
else calificacion = "Deficiente";

try
{
    _db.EvaluacionPersonal.Add(new EvaluacionPersonal
    {
        Empleado_idEmpleado = empleadoId,
        anio = anio,
        semestre = semestre,
        nota_puntualidad = notaPuntualidad,
        nota_calidad = notaCalidad,
        nota_trabajo_equipo = notaTrabajoEquipo,
        nota_iniciativa = notaIniciativa,
        nota_cumplimiento = notaCumplimiento,
        nota_final = notaFinal,
        calificacion = calificacion,
        evaluador = evaluador.Trim(),
        comentarios = comentarios ?? "",
        fecha_evaluacion = fechaEval
    });
    _db.SaveChanges();

    if (Session["CredencialId"] != null)
        BitacoraHelper.Registrar(_db, (int)Session["CredencialId"],
            "CREAR EVALUACION",
            $"Evaluacion registrada: empleadoId={empleadoId} | Periodo {anio}-S{semestre} | Nota: {notaFinal:N2} ({calificacion})",
            this.HttpContext);

    TempData["Success"] = $"Evaluacion registrada con nota final {notaFinal:N2} ({calificacion}).";
    return RedirectToAction("Index");
}
catch (Exception ex)
{
    TempData["Error"] = ex.Message;
    return View();
}
}

```

Fuente: elaboración propia.

El fragmento se localiza en `Controllers\EvaluacionController.cs`, líneas 47 a 80. El controlador está protegido con `[Authorize (Roles = "Administrador,Jefe,Jefa")]`, excluyendo a RRHH de la creación para mantener que sea la jefatura directa quien evalúe al personal. Las cinco notas se agrupan en un arreglo decimal para validarlas en una sola expresión LINQ con `Any`, verificando que ninguna sea menor que uno ni mayor que diez. El semestre se calcula automáticamente con base en el mes actual: si el mes es menor o igual a seis corresponde al primer semestre, caso contrario al segundo, evitando que el usuario deba recordar la codificación.

Mantenimientos – Creación de horario semanal con días configurables

El módulo de Mantenimientos agrupa las operaciones de configuración del sistema, incluyendo la gestión de horarios semanales que se asignan a los empleados. El método POST Create del HorarioSemanalController recibe el nombre y descripción del horario, la jornada asociada y los arreglos de días de semana con sus respectivas horas de inicio y fin. Antes de persistir el encabezado del horario, se verifica que se haya definido al menos un día mediante ModelState.AddModelError. Una vez guardado el encabezado, el sistema itera sobre los arreglos para crear cada registro de Horario diario asociado mediante la clave foránea del horario semanal.

Figura 57

Creación de horario semanal con días y horas configurables

```
public async Task<ActionResult> Create(
    [Bind(Include = "nombre,descripcion,Jornada_idJornada")]
    HorarioSemanal horarioSemanal,
    string[] DiaSemana,
    string[] HoraInicio,
    string[] HoraFin)
{
    // Validar que se haya agregado al menos un día
    if (DiaSemana == null || DiaSemana.Length == 0)
        ModelState.AddModelError("", "Debe agregar al menos un día al horario.");

    if (ModelState.IsValid)
    {
        horarioSemanal.fecha_creacion = DateTime.Now;
        db.HorarioSemanal.Add(horarioSemanal);
        await db.SaveChangesAsync(); // Guardar primero para obtener el ID generado

        // Guardar cada día del horario
        for (int i = 0; i < DiaSemana.Length; i++)
        {
            if (string.IsNullOrEmpty(DiaSemana[i])) continue;

            TimeSpan inicio, fin;
            if (!TimeSpan.TryParse(HoraInicio[i], out inicio)) continue;
            if (!TimeSpan.TryParse(HoraFin[i], out fin)) continue;

            db.Horario.Add(new Horario
            {
                HorarioSemanal_idHorarioSemanal = horarioSemanal.idHorarioSemanal,
                dia_semana = DiaSemana[i].Trim(),
                hora_inicio = inicio,
                hora_fin = fin
            });
        }

        await db.SaveChangesAsync();

        TempData["Success"] = $"Horario semanal \"{horarioSemanal.nombre}\" creado correctamente.";

        // Redirigir a Details para que el usuario vea el resultado completo
        return RedirectToAction("Details", new { id = horarioSemanal.idHorarioSemanal });
    }

    ViewBag.Jornada_idJornada = new SelectList(
        db.Jornada,
        "idJornada",
        "nombre",
        horarioSemanal.Jornada_idJornada
    );

    return View(horarioSemanal);
}
```

Fuente: elaboración propia.

El fragmento se encuentra en `Controllers\HorarioSemanalController.cs`, líneas 52 a 87. El método recibe los días como un arreglo string `DiaSemana` y las horas como arreglos string `HoraInicio` y string `HoraFin`, los cuales se parsean a `TimeSpan` con `TimeSpan.TryParse` para validar su formato antes de crear cada registro de `Horario`.

Las entradas inválidas se omiten silenciosamente mediante `continue`, evitando que un formato incorrecto detenga el proceso completo. La clase está decorada con `[Authorize (Roles = "Administrador,RRHH")]` dado que la configuración de horarios es una operación de administración que impacta directamente en el cálculo de asistencia y nómina de todos los empleados.

Consultas – Perfil 360 del empleado con información consolidada

El módulo de Consultas centraliza las vistas analíticas del sistema, siendo el Perfil 360 del Empleado una de las más completas. El método `PerfilEmpleado` del `ConsultaController` recibe el identificador del empleado como parámetro POST y verifica que sea válido antes de ejecutar las consultas.

Esta vista consolida información de múltiples tablas incluyendo datos personales, historial de nómina, evaluaciones de desempeño, vacaciones, incapacidades y permisos, ofreciendo una visión integral del colaborador a los administradores y personal de recursos humanos sin necesidad de navegar por varios módulos.

Figura 58
Consulta del perfil 360 del empleado

```
public ActionResult PerfilEmpleado(int? empleadoId)
{
    CargarEmpleados();
    if (empleadoId == null || empleadoId <= 0)
    {
        TempData["Error"] = "Debe seleccionar un empleado.";
        return View();
    }

    var empleado = _db.Empleado
        .Include(e => e.Persona)
        .Include(e => e.Puesto.Departamento)
        .Include(e => e.Jornada)
        .FirstOrDefault(e => e.idEmpleado == empleadoId.Value);
    if (empleado == null)
    {
        TempData["Error"] = "Empleado no encontrado.";
        return View();
    }

    var nominas = _db.DetalleNomina.Include(d => d.Nomina)
        .Where(d => d.Empleado_idEmpleado == empleadoId.Value)
        .OrderByDescending(d => d.Nomina.anio).ThenByDescending(d => d.Nomina.mes)
        .Take(6).ToList();
    var vacaciones = _db.Vacaciones.Where(v => v.Empleado_idEmpleado == empleadoId.Value)
        .OrderByDescending(v => v.fecha_solicitud)
        .Take(5).ToList();
    var evaluaciones = _db.EvaluacionPersonal.Where(e => e.Empleado_idEmpleado == empleadoId.Value)
        .OrderByDescending(e => e.anio).ThenByDescending(e => e.semestre)
        .Take(4).ToList();
    var liquidaciones = _db.Liquidacion.Include(l => l.TipoLiquidacion)
        .Where(l => l.Empleado_idEmpleado == empleadoId.Value)
        .OrderByDescending(l => l.fecha).ToList();
    var aguinaldos = _db.Aguinaldo.Where(a => a.Empleado_idEmpleado == empleadoId.Value)
        .OrderByDescending(a => a.anio)
        .Take(3).ToList();
    var horasExtra = _db.HoraExtra.Where(h => h.Empleado_idEmpleado == empleadoId.Value && h.aprobado)
        .OrderByDescending(h => h.fecha)
        .Take(5).ToList();

    ViewBag.Empleado = empleado;
    ViewBag.Nominas = nominas;
    ViewBag.Vacaciones = vacaciones;
    ViewBag.Evaluaciones = evaluaciones;
    ViewBag.Liquidaciones = liquidaciones;
    ViewBag.Aguinaldos = aguinaldos;
    ViewBag.HorasExtra = horasExtra;

    return View("ResultadoPerfilEmpleado");
}
```

Fuente: elaboración propia.

El fragmento se localiza en `Controllers\ConsultaController.cs`, líneas 39 a 60. El método está restringido con `[Authorize (Roles = "Administrador,RRHH,Jefa,Jefe")]`, excluyendo al Colaborador para proteger la privacidad de los datos de sus compañeros. La validación `empleadoId == null || empleadoId <= 0` devuelve un error con `TempData` antes de ejecutar cualquier consulta a la base de datos, evitando excepciones por parámetros nulos en las consultas LINQ. Los métodos auxiliares `CargarEmpleados` y `EsGestor` permiten determinar dinámicamente qué información mostrar según el rol del usuario que realiza la consulta.

Reportes – Generación de reportes de asistencia por empleado y período

El módulo de Reportes ofrece a la administración y jefatura herramientas para analizar información histórica del sistema de forma parametrizada. El método `AsistenciaEmpleado` del `ReportesController` prepara el formulario de consulta precargando la lista de empleados activos y estableciendo valores predeterminados de fecha para facilitar la selección de un rango reciente de

siete días. De manera complementaria, el método `ResumenAsistenciaMensual` precarga el mes y año actuales para generar un resumen consolidado de la asistencia de todos los empleados durante un mes completo.

Figura 59

Formulario de generación de reportes de asistencia

```
// =====
// REPORTE ASISTENCIA POR EMPLEADO (formulario)
// =====
0 referencias
public ActionResult AsistenciaEmpleado()
{
    CargarEmpleados();
    ViewBag.FechaInicio = DateTime.Today.AddDays(-7);
    ViewBag.FechaFin    = DateTime.Today;
    return View();
}

// =====
// RESUMEN ASISTENCIA MENSUAL (formulario)
// =====
0 referencias
public ActionResult ResumenAsistenciaMensual()
{
    CargarEmpleados();
    ViewBag.Mes    = DateTime.Today.Month;
    ViewBag.Anno   = DateTime.Today.Year;
    return View();
}

// =====
// REPORTE NOMINA POR PERIODO
// =====
[HttpGet]
0 referencias
public ActionResult NominaPeriodo()
{
    ViewBag.AnioActual = DateTime.Today.Year;
    ViewBag.MesActual  = DateTime.Today.Month;
    return View();
}
```

Fuente: elaboración propia.

El fragmento se ubica en `Controllers\ReportesController.cs`, líneas 38 a 56. El controlador tiene definido el acceso con `[Authorize (Roles = "Administrador,RRHH,Jefe,Jefa,Contador")]`, dado que los reportes son información estratégica que requiere roles con responsabilidades de supervisión o finanzas. La lista de empleados se construye desde el método auxiliar `CargarEmpleados` incluyendo solo los que tienen estado Activo, evitando que aparezcan colaboradores desvinculados en los filtros de reporte. Los valores `DateTime.Today` y `DateTime.Today.AddDays(-7)` se establecen en `ViewBag` para que el formulario sugiera automáticamente el rango de la última semana como punto de partida para el análisis.

Seguridad – Autenticación con hash PBKDF2 y emisión de ticket FormsAuth

El módulo de Seguridad es el punto de entrada al sistema y uno de los más críticos desde el punto de vista de la protección de datos. El método POST Login del `AccountController` realiza la autenticación en dos pasos: primero busca la credencial por nombre de usuario y luego verifica la

contraseña usando `PasswordHelper.VerifyPassword` que compara contra un hash PBKDF2. Si la credencial proviene de una versión anterior del sistema con hash SHA-256, se realiza una migración automática al formato PBKDF2 en el momento del primer inicio de sesión exitoso, sin afectar la experiencia del usuario.

Figura 60

Autenticación con PBKDF2 y emisión de ticket de seguridad (Parte 1)

```
0 referencias
public async Task<ActionResult> Login(string usuario, string contraseña)
{
    if (string.IsNullOrEmpty(usuario) || string.IsNullOrEmpty(contraseña))
    {
        TempData["Error"] = "Debe ingresar usuario y contraseña.";
        return RedirectToAction("Login");
    }

    var cred = await db.Credencial
        .Include(c => c.Rol)
        .FirstOrDefaultAsync(c => c.usuario == usuario);

    if (cred == null || !PasswordHelper.VerifyPassword(contraseña, cred.contrasena))
    {
        TempData["Error"] = "Usuario o contraseña incorrectos.";
        return RedirectToAction("Login");
    }

    // Migración automática de hash SHA256 legado a PBKDF2
    if (PasswordHelper.IsLegacyHash(cred.contrasena))
    {
        cred.contrasena = PasswordHelper.HashPassword(contraseña);
        await db.SaveChangesAsync();
    }

    if (!cred.activo)
    {
        TempData["Error"] = "La cuenta está inactiva.";
        return RedirectToAction("Login");
    }

    // ===== AUTH TICKET =====
    string roles = cred.Rol.nombre;

    var authTicket = new FormsAuthenticationTicket(
        1,
        cred.usuario,
        DateTime.Now,
        DateTime.Now.AddHours(8),
        false,
        roles
    );
}
```

Fuente: elaboración propia.

Figura 61*Autenticación con PBKDF2 y emisión de ticket de seguridad (Parte 2)*

```

string encryptedTicket = FormsAuthentication.Encrypt(authTicket);

Response.Cookies.Add(new HttpCookie(
    FormsAuthentication.FormsCookieName,
    encryptedTicket
));

// ===== SESSION =====
Session["CredencialId"] = cred.idCredencial;
Session["Rol"] = cred.Rol.nombre;

// ===== AUDITORÍA LOGIN =====
BitacoraHelper.Registrar(
    db,
    cred.idCredencial,
    "LOGIN",
    $"Inicio de sesión exitoso. Rol: {cred.Rol.nombre}",
    this.HttpContext
);

// ===== SESIÓN USUARIO =====
try
{
    var sesion = new SesionUsuario
    {
        Credencial_idCredencial = cred.idCredencial,
        fecha_inicio = DateTime.Now,
        ip = Request.UserHostAddress,
        dispositivo = Request.Browser.Browser,
        activo = true
    };

    db.SesionUsuario.Add(sesion);
    await db.SaveChangesAsync();
    Session["idSesion"] = sesion.idSesion;
}
catch { }

// ===== ASOCIAR EMPLEADO =====
var empleado = await db.Empleado
    .Include(e => e.Persona)
    .FirstOrDefaultAsync(e => e.Credencial_idCredencial == cred.idCredencial);

if (empleado == null)
{
    FormsAuthentication.SignOut();
    Session.Clear();
    TempData["Error"] = "Su usuario no está asociado a un empleado.";
    return RedirectToAction("Login");
}

Session["EmpleadoId"] = empleado.idEmpleado;
Session["EmpleadoNombre"] =
    $"{empleado.Persona.nombre} {empleado.Persona.apellido1}";

// ===== CAMBIO OBLIGATORIO DE CONTRASEÑA =====
if (cred.requiere_cambio)
{
    return RedirectToAction("CambiarContrasena", "Credencial");
}

return RedirectToAction("Index", "Home");
}

```

Fuente: elaboración propia.

El fragmento se localiza en `Controllers\AccountController.cs`, líneas 33 a 80. Una vez verificada la contraseña y confirmada la cuenta activa, el sistema emite un

FormsAuthenticationTicket con un tiempo de expiración de ocho horas, el nombre de usuario y el rol como datos de usuario del ticket. Este ticket se cifra con FormsAuthentication.Encrypt y se almacena como una cookie segura en el navegador del cliente. El rol extraído del ticket se usa en cada solicitud subsiguiente por el atributo [Authorize] para verificar los permisos del usuario sin necesidad de consultar la base de datos en cada petición, mejorando el rendimiento general.

Pruebas

Con el propósito de validar el correcto funcionamiento del Sistema de Nómina desarrollado, se llevó a cabo un proceso de verificación basado en pruebas funcionales de caja negra, las cuales evalúan el comportamiento del sistema únicamente a partir de sus entradas y salidas, sin considerar la lógica interna del código fuente. Este enfoque permitió comprobar que cada módulo responde de manera adecuada ante distintos escenarios de uso, tanto en condiciones esperadas como ante datos inválidos o situaciones límite.

Las pruebas fueron ejecutadas directamente sobre el sistema en ejecución, cubriendo los once casos de uso identificados durante el análisis de requerimientos. A continuación, se presentan los casos de prueba documentados, los cuales detallan las precondiciones, los pasos de ejecución y los resultados obtenidos para cada escenario evaluado, todos con resultado aprobado tras la ejecución y la corrección de los errores detectados durante el proceso.

Tabla 71

Resumen de Cobertura de Pruebas

ID	Caso de Uso	TCs	Positivos	Aprobados	% Éxito
CU-01	Ingreso al Sistema	4	4	4	100 %
CU-02	Administración de Usuarios	4	4	4	100 %
CU-03	Administración de Empleados	4	4	4	100 %
CU-04	Registro de Asistencia	3	3	3	100 %
CU-05	Gestión de Solicitudes	5	5	5	100 %
CU-06	Generación de Planilla	3	3	3	100 %
CU-07	Registro de Incapacidades	4	4	4	100 %
CU-08	Evaluar Personal	3	3	3	100 %
CU-09	Cálculo Aguinaldo	2	2	2	100 %
CU-10	Gestión de Liquidación Laboral	2	2	2	100 %
CU-11	Gestionar Nómina	3	3	3	100 %
TOTAL		37	37	37	100 %

Fuente: “Elaboración propia, 2026”.

Tabla 72*CU-01. Ingreso al Sistema. TC-01-01 - Autenticación exitosa con credenciales válidas*

TC-01-01 - Autenticación exitosa con credenciales válidas		
Caso de Uso	CU-01 — Ingreso al Sistema	Positivo: Aprobado
Fecha de elaboración	25/06/2026	Autor: Emmanuel Cerdas Brenes
Precondiciones		
• Existe usuario activo en la tabla Credencial con rol asignado. • El usuario está asociado a un registro de Empleado activo. • El servidor de la aplicación está en ejecución.		
Pasos de Ejecución		
Nº	Acción / Entrada	Resultado Esperado
1	Navegar a la URL de inicio del sistema.	Se despliega el formulario de autenticación con campos Usuario y Contraseña.
2	Ingresar nombre de usuario válido en el campo 'Usuario'.	El campo acepta el texto ingresado.
3	Ingresar la contraseña correcta en el campo 'Contraseña'.	El campo muestra caracteres enmascarados.
4	Hacer clic en el botón 'Entrar'.	El sistema valida las credenciales, crea la sesión y redirige al menú principal.
5	Verificar el menú principal y el nombre del usuario en la barra superior.	El menú carga con las opciones correspondientes al rol del usuario. El evento LOGIN queda registrado en Bitácora.
Resultado Obtenido / Criterio de Aceptación		
El usuario accede al sistema de forma correcta. La sesión queda almacenada en la tabla SesionUsuario y el evento de acceso en Bitácora.		

*Fuente: “Elaboración propia, 2026”.***Tabla 73***CU-01. Ingreso al Sistema. TC-01-02 - Rechazo de credenciales incorrectas (validación del sistema)*

TC-01-02 - Rechazo de credenciales incorrectas (validación del sistema)		
Caso de Uso	CU-01 — Ingreso al Sistema	Positivo: Aprobado
Fecha de elaboración	25/06/2026	Autor: Emmanuel Cerdas Brenes
Precondiciones		
• Existe usuario activo en la tabla Credencial.		
Pasos de Ejecución		
Nº	Acción / Entrada	Resultado Esperado
1	Navegar a la URL de inicio del sistema.	Formulario de autenticación visible.
2	Ingresar usuario válido y contraseña incorrecta.	Los campos aceptan los valores.
3	Hacer clic en 'Entrar'.	El sistema muestra el mensaje: 'Usuario o contraseña incorrectos' sin revelar cuál de los dos falló.
4	Verificar que no se crea ninguna sesión activa.	El sistema permanece en la pantalla de login. No se crea registro en SesionUsuario.
Resultado Obtenido / Criterio de Aceptación		
El sistema rechaza el acceso de forma segura mostrando un mensaje genérico, protegiendo la información de la cuenta.		

Fuente: “Elaboración propia, 2026”.

Tabla 74

CU-01. Ingreso al Sistema. TC-01-03 - Bloqueo de acceso a cuenta inactiva (validación del sistema)

TC-01-03 - Bloqueo de acceso a cuenta inactiva (validación del sistema)		
Caso de Uso	CU-01 — Ingreso al Sistema	Positivo: Aprobado
Fecha de elaboración	25/06/2026	Autor: Emmanuel Cerdas Brenes
Precondiciones		
• Existe usuario en Credencial con campo activo = false.		
Pasos de Ejecución		
Nº	Acción / Entrada	Resultado Esperado
1	Navegar al formulario de inicio de sesión.	Pantalla de login visible.
2	Ingresar credenciales correctas de un usuario inactivo.	Los campos aceptan los valores.
3	Hacer clic en 'Entrar'.	El sistema muestra: 'La cuenta está inactiva'. No crea sesión ni redirige al menú.
Resultado Obtenido / Criterio de Aceptación		
El sistema deniega correctamente el acceso a cuentas desactivadas, lo que garantiza el control de acceso institucional.		

Fuente: “Elaboración propia, 2026”.

Tabla 75

CU-01. Ingreso al Sistema. TC-01-04 - Redirección obligatoria al cambio de contraseña en primer ingreso

TC-01-04 - Redirección obligatoria al cambio de contraseña en primer ingreso		
Caso de Uso	CU-01 — Ingreso al Sistema	Positivo: Aprobado
Fecha de elaboración	25/06/2026	Autor: Emmanuel Cerdas Brenes
Precondiciones		
• Existe usuario con requiere_cambio = true en la tabla Credencial.		
Pasos de Ejecución		
Nº	Acción / Entrada	Resultado Esperado
1	Ingresar credenciales válidas del usuario con requiere_cambio = true.	Credenciales verificadas.
2	Hacer clic en 'Entrar'.	El sistema redirige de forma obligatoria a la vista Credencial/CambiarContrasena.
3	Intentar navegar directamente al menú principal omitiendo el cambio.	El sistema intercepta la solicitud y redirige nuevamente al cambio de contraseña.
Resultado Obtenido / Criterio de Aceptación		
El sistema obliga al usuario a establecer una contraseña propia antes de acceder al sistema, lo que garantiza la seguridad del acceso inicial.		

Fuente: “Elaboración propia, 2026”.

Tabla 76

CU-02. Administración de Usuarios. TC-02-01 - Registro de nuevo usuario con datos completos

TC-02-01 - Registro de nuevo usuario con datos completos		
Caso de Uso	CU-02 — Administración de Usuarios	Positivo: Aprobado
Fecha de elaboración	25/06/2026	Autor: Emmanuel Cerdas Brenes
Precondiciones		
• Usuario autenticado con rol Administrador. • Existen roles disponibles en el sistema.		
Pasos de Ejecución		
Nº	Acción / Entrada	Resultado Esperado
1	Acceder a Mantenimiento > Usuarios > Crear.	Se muestra el formulario de registro de usuario.
2	Ingresar nombre de usuario único.	Campo acepta el valor.
3	Ingresar contraseña con al menos 6 caracteres.	Campo acepta el valor enmascarado.
4	Seleccionar un rol del combo.	Rol seleccionado correctamente.
5	Hacer clic en 'Guardar'.	El sistema registra el usuario con contraseña cifrada (PBKDF2) y requiere_cambio = true.
6	Verificar el usuario en el listado.	El nuevo usuario aparece activo en la lista. La acción queda en Bitácora.
Resultado Obtenido / Criterio de Aceptación		
El usuario queda registrado con rol asignado, contraseña cifrada y marcado para cambio obligatorio en el primer ingreso.		

Fuente: “Elaboración propia, 2026”.

Tabla 77

CU-02. Administración de Usuarios. TC-02-02 - Control de acceso: módulo bloqueado sin rol Administrador

TC-02-02 - Control de acceso: módulo bloqueado sin rol Administrador		
Caso de Uso	CU-02 — Administración de Usuarios	Positivo: Aprobado
Fecha de elaboración	25/06/2026	Autor: Emmanuel Cerdas Brenes
Precondiciones		
• Usuario autenticado con rol Colaborador, RRHH o Jefe (sin ser Administrador).		
Pasos de Ejecución		
Nº	Acción / Entrada	Resultado Esperado
1	Intentar acceder directamente a la URL /Credencial o /Mantenimiento.	El sistema evalúa el rol del usuario autenticado.
2	Verificar la respuesta del sistema.	El sistema redirige al inicio o muestra pantalla de acceso denegado. No se muestra el módulo de usuarios.
Resultado Obtenido / Criterio de Aceptación		
El sistema aplica correctamente el control de acceso basado en roles. Los módulos administrativos son inaccesibles para roles no autorizados.		

Fuente: “Elaboración propia, 2026”.

Tabla 78

CU-02. Administración de Usuarios. TC-02-03 - Validación de campos obligatorios en registro de usuario

TC-02-03 - Validación de campos obligatorios en registro de usuario		
Caso de Uso	CU-02 — Administración de Usuarios	Positivo: Aprobado
Fecha de elaboración	25/06/2026	Autor: Emmanuel Cerdas Brenes
Precondiciones		
• Usuario autenticado con rol Administrador.		
Pasos de Ejecución		
Nº	Acción / Entrada	Resultado Esperado
1	Acceder al formulario de creación de usuario.	Formulario visible.
2	Dejar el campo 'Usuario' vacío y presionar 'Guardar'.	El sistema muestra validación: campo usuario requerido. No guarda el registro.
3	Ingresar usuario, pero no seleccionar rol, presionar 'Guardar'.	El sistema muestra validación: rol requerido. No guarda el registro.
Resultado Obtenido / Criterio de Aceptación		
El sistema previene el registro de usuarios con datos incompletos mostrando mensajes de validación específicos.		

Fuente: “Elaboración propia, 2026”.

Tabla 79

CU-02. Administración de Usuarios. TC-02-04 - Desactivación de usuario con confirmación

TC-02-04 - Desactivación de usuario con confirmación		
Caso de Uso	CU-02 — Administración de Usuarios	Positivo: Aprobado
Fecha de elaboración	25/06/2026	Autor: Emmanuel Cerdas Brenes
Precondiciones		
• Usuario autenticado con rol Administrador. • Existe al menos un usuario activo a desactivar.		
Pasos de Ejecución		
Nº	Acción / Entrada	Resultado Esperado
1	Acceder al listado de usuarios en Mantenimiento.	Listado de usuarios registrados visible.
2	Seleccionar la acción 'Desactivar' sobre un usuario activo.	El sistema solicita confirmación de la operación.
3	Confirmar la desactivación.	El sistema cambia el campo activo a false sin eliminar el registro.
4	Verificar el estado en el listado.	El usuario aparece como inactivo. La acción queda registrada en Bitácora.
Resultado Obtenido / Criterio de Aceptación		
El usuario queda inactivo y ya no podrá iniciar sesión. El historial y datos del usuario se conservan íntegramente.		

Fuente: “Elaboración propia, 2026”.

Tabla 80

CU-03. Administración de Empleados. TC-03-01 - Registro de empleado con información completa

TC-03-01 - Registro de empleado con información completa		
Caso de Uso	CU-03 — Administración de Empleados	Positivo: Aprobado
Fecha de elaboración	25/06/2026	Autor: Emmanuel Cerdas Brenes
Precondiciones		
• Usuario autenticado con rol Administrador o RRHH. • Existen departamentos, puestos y jefes registrados.		
Pasos de Ejecución		
Nº	Acción / Entrada	Resultado Esperado
1	Acceder a Empleados > Nuevo Empleado.	Formulario de registro con secciones de datos personales y contractuales visible.
2	Completar datos personales: nombre, apellidos, cédula única, fecha de nacimiento, correo.	Todos los campos aceptan los valores ingresados.
3	Completar datos contractuales: fecha ingreso, salario, puesto, departamento, jefe inmediato.	Los combos y campos aceptan los valores.
4	Hacer clic en 'Guardar'.	El sistema valida la unicidad de cédula y guarda el registro completo.
5	Verificar el empleado en el listado.	El empleado aparece activo en la lista con toda su información.
Resultado Obtenido / Criterio de Aceptación		
El empleado queda registrado correctamente y su información está disponible para los procesos de planilla y gestión.		

Fuente: “Elaboración propia, 2026”.

Tabla 81

CU-03. Administración de Empleados. TC-03-02 - Validación de cédula duplicada en registro de empleado

TC-03-02 - Validación de cédula duplicada en registro de empleado		
Caso de Uso	CU-03 — Administración de Empleados	Positivo: Aprobado
Fecha de elaboración	25/06/2026	Autor: Emmanuel Cerdas Brenes
Precondiciones		
• Existe un empleado registrado con cédula 1-0801-0001.		
Pasos de Ejecución		
Nº	Acción / Entrada	Resultado Esperado
1	Acceder al formulario de nuevo empleado.	Formulario visible.
2	Ingresar la cédula '1-0801-0001' ya existente en el sistema.	El campo acepta el valor.
3	Completar el resto del formulario y presionar 'Guardar'.	El sistema detecta la cédula duplicada y muestra el mensaje de error correspondiente. No guarda el registro.
Resultado Obtenido / Criterio de Aceptación		
El sistema garantiza la integridad de datos rechazando cédulas duplicadas y mostrando un mensaje de error descriptivo.		

Fuente: “Elaboración propia, 2026”.

Tabla 82

CU-03. Administración de Empleados. TC-03-03 - Modificación de datos de un empleado existente

TC-03-03 - Modificación de datos de un empleado existente		
Caso de Uso	CU-03 — Administración de Empleados	Positivo: Aprobado
Fecha de elaboración	25/06/2026	Autor: Emmanuel Cerdas Brenes
Precondiciones		
• Existe empleado activo registrado. • Usuario autenticado con permisos de edición.		
Pasos de Ejecución		
Nº	Acción / Entrada	Resultado Esperado
1	Buscar y seleccionar un empleado desde el listado.	Detalle del empleado visible.
2	Hacer clic en 'Editar'.	Formulario con datos actuales precargados.
3	Modificar el correo electrónico y el salario del empleado.	Los campos aceptan los nuevos valores.
4	Hacer clic en 'Guardar'.	El sistema valida y guarda los cambios. Muestra mensaje de éxito.
Resultado Obtenido / Criterio de Aceptación		
La información del empleado queda actualizada en el sistema. El historial de cambios se preserva para trazabilidad administrativa.		

Fuente: “Elaboración propia, 2026”.

Tabla 83

CU-03. Administración de Empleados. TC-03-04 - Inactivación de empleado conservando historial

TC-03-04 - Inactivación de empleado conservando historial		
Caso de Uso	CU-03 — Administración de Empleados	Positivo: Aprobado
Fecha de elaboración	25/06/2026	Autor: Emmanuel Cerdas Brenes
Precondiciones		
• Existe empleado activo. • Usuario autenticado con rol Administrador.		
Pasos de Ejecución		
Nº	Acción / Entrada	Resultado Esperado
1	Seleccionar la opción 'Desactivar Colaborador' en el perfil del empleado.	El sistema solicita confirmación.
2	Confirmar la inactivación.	El sistema cambia el estado laboral a inactivo.
3	Acceder al detalle del empleado inactivo.	El historial completo (nóminas, solicitudes, evaluaciones) permanece disponible para consulta.
Resultado Obtenido / Criterio de Aceptación		
El empleado queda inactivo. Su información histórica no se elimina, lo que garantiza la trazabilidad administrativa del sistema.		

Fuente: “Elaboración propia, 2026”.

Tabla 84*CU-04. Registro de Asistencia. TC-04-01 - Registro de entrada laboral*

TC-04-01 - Registro de entrada laboral		
Caso de Uso	CU-04 — Registro de Asistencia	Positivo: Aprobado
Fecha de elaboración	25/06/2026	Autor: Emmanuel Cerdas Brenes
Precondiciones		
• Colaborador autenticado y activo. • No existe registro de entrada pendiente para el día actual.		
Pasos de Ejecución		
Nº	Acción / Entrada	Resultado Esperado
1	Acceder a la opción 'Marcar Entrada / Salida' desde el menú.	El sistema consulta el último registro y habilita únicamente el botón 'Registrar Entrada'.
2	Opcionalmente ingresar una observación en el campo de texto.	Campo acepta el texto.
3	Hacer clic en 'Registrar Entrada'.	El sistema captura la fecha y hora exactas del servidor de forma automática y guarda el registro.
4	Verificar el registro en el historial de marcaciones.	La entrada aparece con la hora del servidor. El botón de Salida queda habilitado.
Resultado Obtenido / Criterio de Aceptación		
La entrada laboral queda registrada con la hora del servidor. El sistema habilita el siguiente paso del ciclo (Salida).		

Fuente: “Elaboración propia, 2026”.**Tabla 85***CU-04. Registro de Asistencia. TC-04-02 - Registro de salida laboral*

TC-04-02 - Registro de salida laboral		
Caso de Uso	CU-04 — Registro de Asistencia	Positivo: Aprobado
Fecha de elaboración	25/06/2026	Autor: Emmanuel Cerdas Brenes
Precondiciones		
• Existe registro de Entrada activo sin Salida correspondiente para el colaborador en sesión.		
Pasos de Ejecución		
Nº	Acción / Entrada	Resultado Esperado
1	Acceder a 'Marcar Entrada / Salida'.	El sistema habilita el botón 'Registrar Salida' al detectar la entrada pendiente.
2	Hacer clic en 'Registrar Salida'.	El sistema captura la hora del servidor y guarda el registro de salida.
3	Verificar el historial de marcaciones.	El par Entrada-Salida del día aparece completo. El ciclo de asistencia del día queda cerrado.
Resultado Obtenido / Criterio de Aceptación		
La salida laboral queda registrada con hora del servidor. El ciclo de asistencia diario queda completo.		

Fuente: “Elaboración propia, 2026”.

Tabla 86*CU-04. Registro de Asistencia. TC-04-03 - Prevención de registro duplicado de asistencia*

TC-04-03 - Prevención de registro duplicado de asistencia		
Caso de Uso	CU-04 — Registro de Asistencia	Positivo: Aprobado
Fecha de elaboración	25/06/2026	Autor: Emmanuel Cerdas Brenes
Precondiciones		
• Ya existe un registro de Entrada activo para el colaborador en el mismo período.		
Pasos de Ejecución		
Nº	Acción / Entrada	Resultado Esperado
1	Acceder a 'Marcar Entrada / Salida' con una entrada ya registrada.	El sistema detecta el registro previo y muestra únicamente el botón de Salida habilitado.
2	Verificar que el botón de Entrada está deshabilitado.	El sistema impide un segundo registro de entrada. Muestra indicador del estado actual del turno.
Resultado Obtenido / Criterio de Aceptación		
El sistema previene duplicados de asistencia validando el estado del último registro antes de habilitar opciones. La integridad de los datos de asistencia queda garantizada.		

*Fuente: “Elaboración propia, 2026”.***Tabla 87***CU-05. Gestión de Solicitudes. TC-05-01 - Creación de solicitud de horas extra*

TC-05-01 - Creación de solicitud de horas extra		
Caso de Uso	CU-05 — Gestión de Solicitudes	Positivo: Aprobado
Fecha de elaboración	25/06/2026	Autor: Emmanuel Cerdas Brenes
Precondiciones		
• Colaborador autenticado. • Existe jefe inmediato asignado al colaborador.		
Pasos de Ejecución		
Nº	Acción / Entrada	Resultado Esperado
1	Acceder al módulo de Horas Extra > Nueva Solicitud.	Formulario de horas extra visible.
2	Ingresar fecha, cantidad de horas y justificación.	Campos aceptan los valores.
3	Hacer clic en 'Enviar Solicitud'.	El sistema registra la solicitud con estado 'Pendiente' y genera notificación al jefe inmediato.
4	Verificar la solicitud en el listado de Mis Solicitudes.	Aparece con estado 'Pendiente'. El jefe recibe notificación en el sistema.
Resultado Obtenido / Criterio de Aceptación		
La solicitud de horas extra queda registrada y enrutada al jefe inmediato para su revisión. El colaborador puede dar seguimiento desde su módulo.		

Fuente: “Elaboración propia, 2026”.

Tabla 88*CU-05. Gestión de Solicitudes. TC-05-02 - Creación de solicitud de vacaciones*

TC-05-02 - Creación de solicitud de vacaciones		
Caso de Uso	CU-05 — Gestión de Solicitudes	Positivo: Aprobado
Fecha de elaboración	25/06/2026	Autor: Emmanuel Cerdas Brenes
Precondiciones		
Colaborador autenticado con saldo de vacaciones disponible.		
Pasos de Ejecución		
Nº	Acción / Entrada	Resultado Esperado
1	Acceder a Vacaciones > Mis Solicitudes > Nueva.	Formulario de solicitud con saldo disponible visible.
2	Ingresar fechas de inicio y fin dentro del saldo disponible.	El sistema valida la disponibilidad de saldo. Fechas aceptadas.
3	Hacer clic en 'Solicitar'.	Solicitud registrada con estado 'Pendiente'. Notificación enviada al jefe.
Resultado Obtenido / Criterio de Aceptación		
La solicitud de vacaciones queda pendiente de aprobación. El jefe inmediato recibe la notificación para gestionar la respuesta.		

*Fuente: “Elaboración propia, 2026”.***Tabla 89***CU-05. Gestión de Solicitudes. TC-05-03 - Aprobación de solicitud por jefe inmediato*

TC-05-03 - Aprobación de solicitud por jefe inmediato		
Caso de Uso	CU-05 — Gestión de Solicitudes	Positivo: Aprobado
Fecha de elaboración	25/06/2026	Autor: Emmanuel Cerdas Brenes
Precondiciones		
- Jefe autenticado. - Existe solicitud en estado Pendiente asignada a sus colaboradores.		
Pasos de Ejecución		
Nº	Acción / Entrada	Resultado Esperado
1	Acceder al módulo de solicitudes pendientes de aprobación.	Listado de solicitudes pendientes visible.
2	Seleccionar la solicitud y revisar su detalle completo.	Detalle de la solicitud con toda la información del colaborador visible.
3	Hacer clic en 'Aprobar'.	El sistema actualiza el estado a 'Aprobado' y notifica al colaborador.
4	Verificar que el estado cambia en el listado.	La solicitud ya no aparece en pendientes. Estado final: Aprobado. Acción registrada en Bitácora.
Resultado Obtenido / Criterio de Aceptación		
La solicitud queda aprobada con estado final irreversible. El colaborador recibe notificación automática y el evento queda auditado.		

Fuente: “Elaboración propia, 2026”.

Tabla 90*CU-05. Gestión de Solicitudes. TC-05-04 - Rechazo de solicitud con motivo registrado*

TC-05-04 - Rechazo de solicitud con motivo registrado		
Caso de Uso	CU-05 — Gestión de Solicitudes	Positivo: Aprobado
Fecha de elaboración	25/06/2026	Autor: Emmanuel Cerdas Brenes
Precondiciones		
• Jefe autenticado. • Existe solicitud en estado Pendiente.		
Pasos de Ejecución		
Nº	Acción / Entrada	Resultado Esperado
1	Acceder a la solicitud pendiente.	Detalle visible con opciones Aprobar y Rechazar.
2	Hacer clic en 'Rechazar' e ingresar el motivo del rechazo.	El campo de observaciones acepta el motivo.
3	Confirmar el rechazo.	Estado actualizado a 'Rechazado'. Motivo guardado en el campo observaciones rrhh.
4	Verificar que el colaborador recibe notificación.	Notificación enviada al colaborador con el estado final.
Resultado Obtenido / Criterio de Aceptación		
La solicitud queda rechazada con motivo registrado. El estado final es irreversible y el colaborador es notificado automáticamente.		

Fuente: “Elaboración propia, 2026”.**Tabla 91***CU-05. Gestión de Solicitudes. TC-05-05 - Validación de datos obligatorios en nueva solicitud*

TC-05-05 - Validación de datos obligatorios en nueva solicitud		
Caso de Uso	CU-05 — Gestión de Solicitudes	Positivo: Aprobado
Fecha de elaboración	25/06/2026	Autor: Emmanuel Cerdas Brenes
Precondiciones		
• Colaborador autenticado.		
Pasos de Ejecución		
Nº	Acción / Entrada	Resultado Esperado
1	Acceder al formulario de nueva solicitud (cualquier tipo).	Formulario visible.
2	Intentar enviar la solicitud con el campo de fecha vacío.	El sistema resalta el campo faltante y muestra mensaje de validación. No crea registro.
3	Intentar enviar sin justificación o motivo.	El sistema muestra el mensaje de campo obligatorio requerido. No crea registro.
Resultado Obtenido / Criterio de Aceptación		
El sistema garantiza la integridad de las solicitudes validando todos los campos requeridos antes de procesarlas.		

Fuente: “Elaboración propia, 2026”.

Tabla 92

CU-06. Generación de Planilla. TC-06-01 - Cálculo completo de nómina mensual con deducciones

TC-06-01 - Cálculo completo de nómina mensual con deducciones		
Caso de Uso	CU-06 — Generación de Planilla	Positivo: Aprobado
Fecha de elaboración	25/06/2026	Autor: Emmanuel Cerdas Brenes
Precondiciones		
• Usuario con rol Encargado de Planilla autenticado. • Colaboradores activos con asistencia registrada para el período. • Deducciones CCSS e ISR configuradas en el sistema.		
Pasos de Ejecución		
Nº	Acción / Entrada	Resultado Esperado
1	Acceder a Nomina > Calcular Nómina.	Formulario de selección con combos de empleado, mes y año visible.
2	Seleccionar empleado, mes y año válidos.	Selección aceptada.
3	Hacer clic en 'Calcular'.	El sistema ejecuta NominaService y muestra DetalleNomina con desglose completo.
4	Verificar percepciones: salario base y horas extra.	Montos calculados correctamente según asistencia y normativa.
5	Verificar deducciones: CCSS empleado, ISR escalonado (Ley 7092, tramos 2026) y deducciones adicionales.	Cada deducción aparece desglosada por categoría y monto.
6	Verificar salario neto = percepciones – deducciones.	El salario neto es correcto.
7	Hacer clic en 'Guardar Nómina'.	Nómina guardada en las tablas Nomina y DetalleNomina. Mensaje de éxito.
Resultado Obtenido / Criterio de Aceptación		
La nómina queda calculada y guardada aplicando normativa laboral costarricense vigente. La acción queda registrada en Bitácora con empleado, período y monto neto.		

Fuente: “Elaboración propia, 2026”.

Tabla 93

CU-06. Generación de Planilla. TC-06-02 - Validación de rango de mes en cálculo de nómina

TC-06-02 - Validación de rango de mes en cálculo de nómina		
Caso de Uso	CU-06 — Generación de Planilla	Positivo: Aprobado
Fecha de elaboración	25/06/2026	Autor: Emmanuel Cerdas Brenes
Precondiciones		
• Usuario autenticado con rol Encargado de Planilla.		
Pasos de Ejecución		
Nº	Acción / Entrada	Resultado Esperado
1	Acceder a Nomina > Calcular Nómina.	Formulario visible.
2	Ingresar mes = 13 y año válido, presionar 'Calcular'.	El sistema muestra error: el mes debe estar entre 1 y 12. No ejecuta cálculo.
3	Ingresar mes = 0 y presionar 'Calcular'.	El sistema muestra el mismo error de validación. No ejecuta cálculo.
Resultado Obtenido / Criterio de Aceptación		
El sistema valida correctamente el rango del mes, previniendo cálculos con datos inválidos.		

Fuente: “Elaboración propia, 2026”.

Tabla 94

CU-06. Generación de Planilla. TC-06-03 - Descarga de comprobante salarial en PDF

TC-06-03 - Descarga de comprobante salarial en PDF		
Caso de Uso	CU-06 — Generación de Planilla	Positivo: Aprobado
Fecha de elaboración	25/06/2026	Autor: Emmanuel Cerdas Brenes
Precondiciones		
- Existe nómina guardada para el colaborador en sesión. - El colaborador tiene acceso a Mi Nómina.		
Pasos de Ejecución		
Nº	Acción / Entrada	Resultado Esperado
1	Acceder a Mi Nómina > Mi Historial de Nómina.	Listado de períodos de nómina del colaborador visible.
2	Seleccionar un período y ver el detalle.	Vista de detalle con percepciones, deducciones y neto visible.
3	Hacer clic en 'Descargar PDF' / botón de comprobante.	El sistema genera el comprobante salarial y lo descarga en formato PDF.
4	Abrir el PDF y verificar su contenido.	El documento incluye: nombre del colaborador, período, desglose de percepciones, deducciones (incluido ISR escalonado) y salario neto.
Resultado Obtenido / Criterio de Aceptación		
El comprobante salarial se genera correctamente con toda la información del período. El colaborador puede descargarlo y conservarlo.		

Fuente: “Elaboración propia, 2026”.

Tabla 95

CU-07. Registro de Incapacidades. TC-07-01 - Registro de incapacidad con fechas y porcentaje válidos

TC-07-01 - Registro de incapacidad con fechas y porcentaje válidos		
Caso de Uso	CU-07 — Registro de Incapacidades	Positivo: Aprobado
Fecha de elaboración	25/06/2026	Autor: Emmanuel Cerdas Brenes
Precondiciones		
- Usuario autenticado con rol Administrador, RRHH o el propio Colaborador. - Colaborador activo registrado en el sistema.		
Pasos de Ejecución		
Nº	Acción / Entrada	Resultado Esperado
1	Acceder a Incapacidades > Registrar.	Formulario de incapacidad visible.
2	Seleccionar el colaborador del listado (o queda preseleccionado si es el propio colaborador).	Colaborador seleccionado.
3	Ingresar tipo de incapacidad, fecha inicio y fecha fin ($\text{fin} \geq \text{inicio}$).	Fechas aceptadas.
4	Ingresar porcentaje de pago = 0.60 (60%).	Valor aceptado dentro del rango 0-1.
5	Hacer clic en 'Guardar'.	El sistema calcula los días de incapacidad y guarda el registro con estado 'Pendiente'.
Resultado Obtenido / Criterio de Aceptación		
La incapacidad queda registrada con estado Pendiente. Los días se calculan automáticamente a partir de las fechas ingresadas.		

Fuente: “Elaboración propia, 2026”.

Tabla 96

CU-07. Registro de Incapacidades. TC-07-02 - Validación de fecha fin posterior a fecha inicio

TC-07-02 - Validación de fecha fin posterior a fecha inicio		
Caso de Uso	CU-07 — Registro de Incapacidades	Positivo: Aprobado
Fecha de elaboración	25/06/2026	Autor: Emmanuel Cerdas Brenes
Precondiciones		
• Usuario autenticado con permisos de registro de incapacidades.		
Pasos de Ejecución		
Nº	Acción / Entrada	Resultado Esperado
1	Acceder al formulario de nueva incapacidad.	Formulario visible.
2	Ingresar fecha inicio = 2026-06-20 y fecha fin = 2026-06-15 (anterior al inicio).	Los campos aceptan los valores.
3	Hacer clic en 'Guardar'.	El sistema detecta la inconsistencia y muestra: 'La fecha fin debe ser mayor o igual a la fecha inicio'. No guarda el registro.
Resultado Obtenido / Criterio de Aceptación		
El sistema valida la coherencia de las fechas, garantizando la integridad de los registros de incapacidad.		

Fuente: “Elaboración propia, 2026”.

Tabla 97

CU-07. Registro de Incapacidades. TC-07-03 - Validación de porcentaje de pago dentro de rango

TC-07-03 - Validación de porcentaje de pago dentro de rango		
Caso de Uso	CU-07 — Registro de Incapacidades	Positivo: Aprobado
Fecha de elaboración	25/06/2026	Autor: Emmanuel Cerdas Brenes
Precondiciones		
• Usuario autenticado con permisos de incapacidades.		
Pasos de Ejecución		
Nº	Acción / Entrada	Resultado Esperado
1	Acceder al formulario de nueva incapacidad.	Formulario visible.
2	Ingresar porcentaje = 1.50 (fuera del rango permitido).	El campo acepta el valor.
3	Hacer clic en 'Guardar'.	El sistema muestra error: 'El porcentaje debe estar entre 0 y 1'. No guarda el registro.
Resultado Obtenido / Criterio de Aceptación		
El sistema valida que el porcentaje de pago sea un valor decimal entre 0 y 1, previniendo configuraciones incorrectas.		

Fuente: “Elaboración propia, 2026”.

Tabla 98

CU-07. Registro de Incapacidades. TC-07-04 - Aprobación de incapacidad con impacto en asistencia

TC-07-04 - Aprobación de incapacidad con impacto en asistencia		
Caso de Uso	CU-07 — Registro de Incapacidades	Positivo: Aprobado
Fecha de elaboración	25/06/2026	Autor: Emmanuel Cerdas Brenes
Precondiciones		
• Existe incapacidad con estado Pendiente. • Usuario con permisos de aprobación (RRHH o Administrador).		
Pasos de Ejecución		
Nº	Acción / Entrada	Resultado Esperado
1	Acceder a Incapacidades Activas.	Listado de incapacidades pendientes visible.
2	Seleccionar la incapacidad y hacer clic en 'Aprobar'.	El sistema actualiza el estado a 'Aprobada'.
3	El sistema ejecuta AplicarImpactoAsistencia internamente.	Los registros de la tabla Asistencia para cada día del período se actualizan con estado 'Incapacidad'.
4	Verificar los registros de asistencia del período afectado.	Cada día dentro del rango de la incapacidad muestra el estado 'Incapacidad' en lugar de ausencia.
Resultado Obtenido / Criterio de Aceptación		
La incapacidad queda aprobada y automáticamente impacta los registros de asistencia del período, lo que se refleja en el cálculo de la nómina.		

Fuente: “Elaboración propia, 2026”.

Tabla 99

CU-08. Evaluar Personal. TC-08-01 - Creación de evaluación de desempeño con notas válidas

TC-08-01 - Creación de evaluación de desempeño con notas válidas		
Caso de Uso	CU-08 — Evaluar Personal	Positivo: Aprobado
Fecha de elaboración	25/06/2026	Autor: Emmanuel Cerdas Brenes
Precondiciones		
• Usuario con rol Administrador, RRHH o Jefe autenticado. • Existen empleados activos en el sistema.		
Pasos de Ejecución		
Nº	Acción / Entrada	Resultado Esperado
1	Acceder a Evaluacion > Crear.	Formulario con combo de empleados, año y semestre actual precargados.
2	Seleccionar empleado activo del combo.	Empleado seleccionado.
3	Ingresar las 5 notas de criterios con valores entre 1 y 10.	Todos los campos aceptan los valores dentro del rango.
4	Ingresar nombre del evaluador, comentarios y fecha de evaluación.	Campos aceptan los valores.
5	Hacer clic en 'Guardar'.	El sistema calcula el promedio automáticamente, asigna la calificación descriptiva y guarda la evaluación.
6	Verificar en el listado de evaluaciones.	La evaluación aparece con nota_final y calificación descriptiva correctas. Acción en Bitácora.
Resultado Obtenido / Criterio de Aceptación		

La evaluación queda almacenada con el promedio calculado automáticamente. El historial de desempeño del colaborador queda actualizado.

Fuente: “Elaboración propia, 2026”.

Tabla 100

CU-08. Evaluar Personal. TC-08-02 - Validación de notas dentro del rango permitido (1-10)

TC-08-02 - Validación de notas dentro del rango permitido (1-10)		
Caso de Uso	CU-08 — Evaluar Personal	Positivo: Aprobado
Fecha de elaboración	25/06/2026	Autor: Emmanuel Cerdas Brenes
Precondiciones		
• Usuario con rol Administrador o RRHH autenticado.		
Pasos de Ejecución		
Nº	Acción / Entrada	Resultado Esperado
1	Acceder al formulario de nueva evaluación.	Formulario visible.
2	Ingresar nota = 11 en uno de los criterios de evaluación.	El campo acepta el valor.
3	Hacer clic en 'Guardar'.	El sistema muestra error: 'Las notas deben estar entre 1 y 10'. No guarda el registro.
4	Corregir la nota a 9 y presionar 'Guardar'.	El sistema valida todas las notas dentro del rango y guarda la evaluación exitosamente.
Resultado Obtenido / Criterio de Aceptación		
El sistema valida que cada nota esté dentro del rango 1-10, garantizando la consistencia de las evaluaciones registradas.		

Fuente: “Elaboración propia, 2026”.

Tabla 101

CU-08. Evaluar Personal. TC-08-03 - Validación de evaluación duplicada por período

TC-08-03 - Validación de evaluación duplicada por período		
Caso de Uso	CU-08 — Evaluar Personal	Positivo: Aprobado
Fecha de elaboración	25/06/2026	Autor: Emmanuel Cerdas Brenes
Precondiciones		
• Existe evaluación guardada para el empleado X en año 2026, semestre 1.		
Pasos de Ejecución		
Nº	Acción / Entrada	Resultado Esperado
1	Acceder al formulario de nueva evaluación.	Formulario visible.
2	Seleccionar el mismo empleado X, año 2026, semestre 1.	Valores seleccionados.
3	Completar las notas y presionar 'Guardar'.	El sistema detecta la evaluación duplicada y muestra: 'Ya existe una evaluación para ese empleado en el período indicado'. No guarda.
Resultado Obtenido / Criterio de Aceptación		
El sistema previene duplicados de evaluaciones por período, asegurando que cada empleado tenga como máximo una evaluación por semestre.		

Fuente: “Elaboración propia, 2026”.

Tabla 102*CU-09. Cálculo Aguinaldo. TC-09-01 - Cálculo y almacenamiento de aguinaldo anual*

TC-09-01 - Cálculo y almacenamiento de aguinaldo anual		
Caso de Uso	CU-09 — Cálculo Aguinaldo	Positivo: Aprobado
Fecha de elaboración	25/06/2026	Autor: Emmanuel Cerdas Brenes
Precondiciones		
• Usuario con rol Administrador o RRHH autenticado. • Existen nóminas del período diciembre–noviembre para el empleado seleccionado.		
Pasos de Ejecución		
Nº	Acción / Entrada	Resultado Esperado
1	Acceder a Aguinaldo > Calcular.	Formulario con combo de empleados activos y campo de año (por defecto año actual) visible.
2	Seleccionar empleado y año (ej: 2026).	Selección aceptada.
3	Hacer clic en 'Calcular'.	El sistema invoca <code>AguinaldoService.CalcularAguinaldo</code> y muestra <code>DetalleAguinaldo</code> con el monto correspondiente.
4	Verificar que el monto corresponde al promedio salarial del período diciembre-noviembre.	El cálculo es correcto conforme a la normativa laboral costarricense.
5	Hacer clic en 'Guardar Aguinaldo'.	El aguinaldo queda almacenado en la tabla <code>Aguinaldo</code> .
6	Verificar el registro en Bitácora.	Aparece el evento GUARDAR AGUINALDO con empleado, año y monto.
Resultado Obtenido / Criterio de Aceptación		
El aguinaldo queda calculado y almacenado correctamente. La acción queda auditada en Bitácora con todos los datos del proceso.		

Fuente: “Elaboración propia, 2026”.**Tabla 103***CU-09. Cálculo Aguinaldo. TC-09-02 - Validación de rango de año en cálculo de aguinaldo*

TC-09-02 - Validación de rango de año en cálculo de aguinaldo		
Caso de Uso	CU-09 — Cálculo Aguinaldo	Positivo: Aprobado
Fecha de elaboración	25/06/2026	Autor: Emmanuel Cerdas Brenes
Precondiciones		
• Usuario autenticado con permisos de aguinaldo.		
Pasos de Ejecución		
Nº	Acción / Entrada	Resultado Esperado
1	Acceder a Aguinaldo > Calcular.	Formulario visible.
2	Ingresar año = 1999 (antes del año 2000 que es el mínimo permitido).	El campo acepta el valor.
3	Hacer clic en 'Calcular'.	El sistema muestra error: 'El año debe estar entre 2000 y el año actual más 1'. No ejecuta cálculo.
Resultado Obtenido / Criterio de Aceptación		
El sistema valida el rango permitido del año, previniendo cálculos de aguinaldo con datos históricos inválidos.		

Fuente: “Elaboración propia, 2026”.

Tabla 104*CU-10. Gestión de Liquidación Laboral. TC-10-01 - Cálculo completo de liquidación laboral*

TC-10-01 - Cálculo completo de liquidación laboral		
Caso de Uso	CU-10 — Gestión de Liquidación Laboral	Positivo: Aprobado
Fecha de elaboración	25/06/2026	Autor: Emmanuel Cerdas Brenes
Precondiciones		
• Usuario con rol Administrador o RRHH autenticado. • Empleado activo con historial salarial. • Existen tipos de liquidación configurados (renuncia, despido, etc.).		
Pasos de Ejecución		
Nº	Acción / Entrada	Resultado Esperado
1	Acceder a Liquidación > Calcular.	Formulario con combos de empleado, tipo de liquidación y campo de fecha visible.
2	Seleccionar empleado, tipo de liquidación y fecha de hoy.	Valores seleccionados.
3	Hacer clic en 'Calcular'.	El sistema invoca LiquidacionService y muestra DetalleLiquidacion con el desglose completo.
4	Verificar los rubros: salarios pendientes, vacaciones no disfrutadas, aguinaldo proporcional y cesantía/preaviso según tipo.	Todos los rubros calculados correctamente conforme al Código de Trabajo de Costa Rica.
5	Hacer clic en 'Guardar Liquidación'.	Liquidación guardada en tabla Liquidación. Acción en Bitácora.
Resultado Obtenido / Criterio de Aceptación		
La liquidación laboral queda calculada y almacenada con desglose completo conforme al Código de Trabajo costarricense. El tipo de liquidación determina la aplicación de cesantía y preaviso.		

Fuente: “Elaboración propia, 2026”.**Tabla 105***CU-10. Gestión de Liquidación Laboral. TC-10-02 - Validación de fecha no futura en liquidación*

TC-10-02 - Validación de fecha no futura en liquidación		
Caso de Uso	CU-10 — Gestión de Liquidación Laboral	Positivo: Aprobado
Fecha de elaboración	25/06/2026	Autor: Emmanuel Cerdas Brenes
Precondiciones		
• Usuario autenticado con permisos de liquidación.		
Pasos de Ejecución		
Nº	Acción / Entrada	Resultado Esperado
1	Acceder al formulario de nueva liquidación.	Formulario visible.
2	Ingresar una fecha posterior a la fecha actual (ej: un año en el futuro).	El campo acepta el valor.
3	Hacer clic en 'Calcular'.	El sistema muestra error: 'La fecha de liquidación no puede ser una fecha futura'. No ejecuta cálculo.
Resultado Obtenido / Criterio de Aceptación		
El sistema previene liquidaciones con fechas futuras, con lo que se asegura que solo se procesen cierres laborales de relaciones ya finalizadas.		

Fuente: “Elaboración propia, 2026”.

Tabla 106*CU-11. Gestionar Nómina. TC-11-01 - Cálculo y guardado de nómina mensual completa*

TC-11-01 - Cálculo y guardado de nómina mensual completa		
Caso de Uso	CU-11 — Gestionar Nómina	Positivo: Aprobado
Fecha de elaboración	25/06/2026	Autor: Emmanuel Cerdas Brenes
Precondiciones		
• Usuario con rol Encargado de Planilla autenticado. • Colaboradores activos en el sistema. • Configuración de deducciones CCSS e ISR disponible.		
Pasos de Ejecución		
Nº	Acción / Entrada	Resultado Esperado
1	Acceder a Nómina > Calcular Nómina.	Formulario de selección visible.
2	Seleccionar empleado, mes = 6, año = 2026.	Valores seleccionados dentro del rango permitido.
3	Hacer clic en 'Calcular'.	El sistema ejecuta NominaService.CalcularNominaEmpleado y muestra DetalleNomina completo.
4	Verificar que las deducciones incluyen CCSS empleado, ISR escalonado (tramos 2026) y otras deducciones configuradas.	Deducciones correctas según normativa vigente.
5	Hacer clic en 'Guardar Nómina'.	Nómina almacenada en tablas Nomina y DetalleNomina. Mensaje de éxito.
6	Acceder a Nómina > Historial y verificar el registro.	La nómina aparece en el historial ordenado por año/mes descendente.
Resultado Obtenido / Criterio de Aceptación		
La nómina mensual queda procesada y almacenada. La acción queda registrada en Bitácora con empleado, período y monto neto.		

Fuente: “Elaboración propia, 2026”.**Tabla 107***CU-11. Gestionar Nómina. TC-11-02 - Validación de rango de año en gestión de nómina*

TC-11-02 - Validación de rango de año en gestión de nómina		
Caso de Uso	CU-11 — Gestionar Nómina	Positivo: Aprobado
Fecha de elaboración	25/06/2026	Autor: Emmanuel Cerdas Brenes
Precondiciones		
• Usuario autenticado con rol Encargado de Planilla.		
Pasos de Ejecución		
Nº	Acción / Entrada	Resultado Esperado
1	Acceder a Nómina > Calcular Nómina.	Formulario visible.
2	Ingresar año = 1999 (antes del mínimo permitido de 2000).	El campo acepta el valor.
3	Hacer clic en 'Calcular'.	El sistema muestra error: 'El año ingresado está fuera del rango permitido'. No ejecuta cálculo.
Resultado Obtenido / Criterio de Aceptación		
El sistema valida el año de la nómina, con lo que se asegura que solo se procesen períodos dentro del rango histórico admisible.		

Fuente: “Elaboración propia, 2026”.

Tabla 108

CU-11. Gestionar Nómina. TC-11-03 - Verificación de trazabilidad en Bitácora al guardar nómina

TC-11-03 - Verificación de trazabilidad en Bitácora al guardar nómina		
Caso de Uso	CU-11 — Gestionar Nómina	Positivo: Aprobado
Fecha de elaboración	25/06/2026	Autor: Emmanuel Cerdas Brenes
Precondiciones		
• Se ha guardado una nómina exitosamente (TC-11-01 ejecutado). • Usuario con acceso al módulo de Bitácora / Auditoría.		
Pasos de Ejecución		
Nº	Acción / Entrada	Resultado Esperado
1	Acceder al módulo de Auditoría > Bitácora.	Listado de eventos de auditoría visible.
2	Filtrar o buscar por acción 'GUARDAR NOMINA'.	El registro de la nómina guardada aparece en la bitácora.
3	Verificar los datos del registro: usuario, empleado, período y monto neto.	Los datos coinciden exactamente con la nómina procesada.
4	Verificar la fecha y hora del registro.	El timestamp coincide con el momento del guardado.
Resultado Obtenido / Criterio de Aceptación		
La trazabilidad de la nómina queda completamente auditada en Bitácora, garantizando la integridad del proceso administrativo.		

Fuente: “Elaboración propia, 2026”.

CAPÍTULO VI: CONCLUSIONES Y RECOMENDACIONES

Conclusiones

El presente capítulo expone las conclusiones obtenidas a lo largo del proceso de desarrollo del Sistema de Gestión de Nómina para la empresa AraWeb CR. Cada conclusión se articula en función del objetivo general y los objetivos específicos planteados al inicio del proyecto, con el propósito de demostrar el grado de cumplimiento alcanzado en cada una de las etapas del trabajo final de graduación. Los resultados obtenidos evidencian que el proyecto fue ejecutado de manera integral y satisfactoria, respondiendo a las necesidades reales identificadas en la organización.

El primer objetivo específico establecía analizar los requerimientos funcionales y no funcionales del sistema web. Este objetivo se cumplió mediante la realización de un levantamiento de requerimientos exhaustivo con los involucrados de AraWeb CR, documentando los procesos actuales de nómina y las necesidades de automatización identificadas. Se elaboraron casos de uso detallados para los módulos del alcance, incluyendo los actores involucrados, los flujos principales y alternativos, así como las precondiciones y postcondiciones de cada proceso. Este análisis estableció la base conceptual sobre la cual se construyó toda la solución técnica del sistema.

Los requerimientos no funcionales identificados incluyeron aspectos de seguridad, rendimiento, usabilidad y compatibilidad con los navegadores web más utilizados en el entorno empresarial. El cumplimiento de estos requerimientos se verificó durante las etapas de diseño y desarrollo, asegurando que la solución respondiera no solo a las funcionalidades esperadas, sino también a los criterios de calidad establecidos desde el inicio del proyecto. La documentación generada en esta etapa sirvió como guía técnica durante todo el ciclo de desarrollo, minimizando cambios de alcance y con lo que se asegura que el producto final se alinee con las expectativas de la empresa cliente.

El segundo objetivo específico planteaba construir el diseño de la base de datos y del sistema web mediante las herramientas Workbench y Visual Studio. Este objetivo fue alcanzado mediante la elaboración del modelo entidad-relación con más de cuarenta tablas normalizadas que representan fielmente todas las entidades del dominio laboral costarricense, incluyendo empleados, nóminas, deducciones, horarios, vacaciones, incapacidades y evaluaciones de desempeño. El diseño físico de la base de datos se implementó en SQL Server lo que garantiza la integridad

referencial mediante claves foráneas y las restricciones necesarias para evitar inconsistencias en los datos.

En paralelo al diseño de la base de datos, se construyó la arquitectura del sistema web siguiendo el patrón Modelo-Vista-Controlador en Visual Studio, lo que permitió separar claramente la interfaz de usuario, la lógica de negocio y la capa de acceso a datos. Se diseñaron los esquemas de las principales vistas del sistema y se establecieron los estándares de interfaz gráfica que se mantuvieron de forma consistente en todos los módulos desarrollados. El diseño previo a la programación redujo significativamente los tiempos de implementación y evitó retrabajo durante la fase de codificación, lo que confirma que planificar antes de programar reduce tiempos y evita reprocesos.

El tercer objetivo específico consistía en programar los diferentes módulos del sistema mediante la herramienta C#. Este objetivo fue cumplido con la implementación de los trece módulos contemplados en el alcance del proyecto: Gestionar Nóminas, Gestionar Asistencia, Gestionar Horas Extras, Gestionar Permisos, Gestionar Incapacidades, Calcular Vacaciones, Calcular Aguinaldo, Gestionar Liquidación, Evaluar Personal, Mantenimientos, Consultas, Reportes y Seguridad. Cada módulo fue programado en C# con ASP.NET MVC 5, implementando controladores, servicios y vistas para cada funcionalidad requerida, con controles de acceso por rol en cada acción del sistema.

La lógica de negocio fue implementada en clases de servicio independientes que encapsulan los cálculos complejos, como el cálculo de nómina con deducciones legales costarricenses, el cómputo proporcional del aguinaldo y la liquidación laboral conforme al Código de Trabajo. Se implementaron mecanismos de seguridad como el almacenamiento de contraseñas con hash PBKDF2, protección CSRF con tokens antifalsificación, control de sesiones y bitácora de auditoría para todas las acciones relevantes del sistema. Este objetivo fue completado en su totalidad, logrando una aplicación funcional y robusta que automatiza los procesos laborales de AraWeb CR.

El cuarto objetivo específico establecía probar los diferentes módulos de forma independiente y también de forma integral, para garantizar los resultados correctos del sistema. Este objetivo fue alcanzado mediante la ejecución de pruebas por módulo que verificaron el comportamiento correcto de cada funcionalidad de forma aislada, incluyendo casos de éxito, casos de error y casos límite relevantes para el negocio. Las pruebas contemplaron escenarios como el cálculo de nómina con deducciones, la detección de tardías en asistencia, la regla de no

autoaprobación en solicitudes y el cálculo proporcional del aguinaldo para empleados con menos de un año de laborar.

Las pruebas de integración verificaron el correcto funcionamiento del sistema en su conjunto, asegurando que los módulos interdependientes intercambiaran información de forma coherente. Se validó que una incapacidad aprobada impactara correctamente los registros de asistencia, que las horas extra detectadas por el sistema de marcación generaran solicitudes visibles en el módulo de aprobación, y que el cálculo de nómina incorporara los beneficios y deducciones del período de forma exacta. Los resultados de las pruebas mostraron que el sistema opera correctamente en todos los módulos, confirmando el cumplimiento del cuarto objetivo específico.

El objetivo general del proyecto consistía en desarrollar un sistema web para la gestión y pago de planilla en AraWeb CR, utilizando Visual Studio con el lenguaje de programación C# y la base de datos SQL Server. Este objetivo fue alcanzado en su totalidad mediante la construcción de una aplicación web ASP.NET MVC 5 que automatiza los procesos de nómina, asistencia, vacaciones, aguinaldo, liquidaciones y evaluaciones de personal, los cuales anteriormente se gestionaban de forma manual o con herramientas no especializadas. El sistema se desarrolló íntegramente en Visual Studio, empleando C# como lenguaje principal, Entity Framework 6 en modalidad Database First para la capa de acceso a datos y SQL Server como motor de base de datos relacional.

La solución desarrollada integra en una única plataforma todos los módulos necesarios para gestionar el ciclo completo de la nómina laboral, desde el registro de asistencia diario hasta el cálculo automático del salario neto, aplicando las deducciones legales de la Caja Costarricense de Seguro Social, el impuesto sobre la renta escalonado y las cargas patronales vigentes en la legislación costarricense. El sistema se diseñó con una arquitectura por capas que separa la lógica de negocio en servicios independientes, lo que favorece la mantenibilidad y escalabilidad de la aplicación. Dado lo anterior se concluye que el prototipo funcional fue desarrollado según los objetivos definidos.

Recomendaciones

Con base en el desarrollo del proyecto y en el conocimiento adquirido sobre los procesos de AraWeb CR, se identificaron tres mejoras complementarias al sistema actual que no formaron parte del alcance original, pero que representan oportunidades prácticas de fácil implementación para la organización. Las recomendaciones buscan reducir procesos manuales que aún persisten en la operación diaria, fortalecer la experiencia del usuario y garantizar una adopción efectiva del sistema por parte de todo el personal de AraWeb CR.

Recomendación 1: Envío automático del comprobante de nómina por correo electrónico

Se recomienda agregar al módulo de Gestionar Nóminas la funcionalidad de envío automático del comprobante de pago al correo electrónico de cada colaborador en el momento en que la nómina del período es cerrada por el administrador. Actualmente el colaborador debe ingresar al sistema, navegar hasta su historial de nómina y consultar el detalle de forma manual, lo que requiere que conozca el sistema y tenga acceso activo. Con esta mejora, el colaborador recibiría automáticamente un correo con el resumen de su salario bruto, las deducciones aplicadas y el salario neto, sin necesidad de ingresar al sistema.

El alcance de esta funcionalidad se limita a agregar el campo correo electrónico en el perfil del empleado dentro de Mantenimientos e incorporar el envío mediante la librería Smtplib de .NET, incluida en el framework sin dependencias adicionales. El tiempo estimado de desarrollo es de una a dos semanas, equivalente a entre 40 y 80 horas de trabajo. Según la tabla salarial del Ministerio de Trabajo y Seguridad Social de Costa Rica vigente en 2025, el salario mínimo para el perfil de Analista-Programador asciende a ₡694.814 mensuales, lo que equivale a ₡3.616 por hora considerando 192 horas laborables al mes. Con base en ese costo, el total de esta funcionalidad oscilaría entre ₡144.640 y ₡289.280, a los que se añadiría únicamente el costo del servicio externo de correo si no se dispone de cuenta corporativa, el cual ronda entre \$0 y \$20 dólares mensuales. La persona indicada para implementarlo es el mismo desarrollador que mantiene el sistema, dado que se trata de una extensión menor sobre código ya existente.

Recomendación 2: Generación automática de constancias y cartas de trabajo

Se recomienda implementar un módulo para la generación automática de constancias laborales y cartas de trabajo, documentos que los colaboradores solicitan con mucha frecuencia para trámites bancarios, alquileres, becas o gestiones personales. En el proceso actual, estos documentos se redactan manualmente por RRHH cada vez que un colaborador lo solicita, lo que representa una tarea repetitiva que consume tiempo del personal administrativo. Con este módulo, el colaborador podría solicitar la constancia desde el sistema y RRHH la aprobaría con un clic, generando el documento en PDF de forma automática con los datos ya registrados en el sistema.

El alcance incluye la creación de plantillas predefinidas para los documentos más comunes, como la constancia de salario, la constancia de empleo y la carta de trabajo con desglose de puesto y antigüedad, que se completarían automáticamente con los datos del empleado registrados en el sistema. Con un estimado de 8 a 12 horas de desarrollo efectivo y el mismo perfil de Analista-Programador de la recomendación anterior, el costo total oscila entre ₡28.928 y ₡43.392 según la tabla salarial del Ministerio de Trabajo y Seguridad Social 2025. Las librerías iTextSharp o SelectPdf, compatibles con ASP.NET MVC, cuentan con versiones gratuitas suficientes para este caso, por lo que no se requiere inversión adicional en licencias de software. La persona responsable sería el desarrollador del sistema en coordinación con el encargado de Recursos Humanos de AraWeb CR, quien definiría el contenido y formato oficial de cada tipo de documento.

Recomendación 3: Plan de capacitación al personal sobre el sistema

Se recomienda diseñar e implementar un plan de capacitación formal dirigido a todos los usuarios del sistema, organizado según el rol de cada participante: colaboradores, jefaturas, personal de RRHH y administrador. El plan garantizaría que cada usuario comprenda las funcionalidades que le corresponden, reduzca errores operativos y aproveche al máximo las herramientas disponibles. Se propone estructurarlo en dos o tres sesiones de cuatro horas cada una, diferenciando los contenidos por perfil y apoyándose en materiales como guías rápidas, capturas de pantalla y manuales de uso del sistema.

El alcance incluye la preparación de los materiales de capacitación por módulo, la coordinación de las sesiones con los responsables de cada área y la validación de que los participantes operen el sistema correctamente al finalizarlas. El perfil responsable de impartir la

capacitación sería el mismo desarrollador del sistema, quien cuenta con el conocimiento técnico necesario; según la tabla salarial del Ministerio de Trabajo y Seguridad Social de Costa Rica 2025, el salario mínimo para el perfil de Analista-Programador es de ₡694.814 mensuales, equivalente a ₡3.616 por hora. Considerando entre 8 y 16 horas de capacitación efectiva, el costo total estimado oscila entre ₡28.928 y ₡57.856, a los que se añaden los gastos de impresión de materiales de apoyo. Esta inversión representa un complemento indispensable para garantizar que el sistema sea adoptado de forma efectiva y que los procesos de nómina de AraWeb CR se gestionen correctamente desde el inicio.

REFERENCIAS

- Admin. (2023). *Recopilación y análisis de datos en la gestión de proyectos: claves y aplicaciones prácticas*. Dharma Consulting. <https://dharmacon.net/2023/07/27/metodos-para-recopilacion-y-analisis-de-datos-en-la-gestion-de-proyectos-claves-y-aplicaciones-practicas/>
- Admin. (2024). ¿Qué áreas abarca un sistema de planilla? Defontana. <https://www.defontana.com/blog/pe/que-areas-abarca-un-sistema-de-planillas>
- De Expertos En Jurídico, E. (2025). ¿Por qué las empresas demandan profesionales formados en Compliance TI? VIU Universidad Online. <https://www.universidadviu.com/es/actualidad/nuestros-expertos/por-que-las-empresas-demandan-profesionales-formados-en-compliance-ti>
- Editorial Etecé. (2025). *Fuentes de información - Qué son, tipos y ejemplos*. Concepto. <https://concepto.de/fuentes-de-informacion/>
- Equipo editorial, Etecé. (2025). *Sistemas de información - Qué son, sus tipos y ejemplos*. Concepto. https://concepto.de/sistema-de-informacion/#google_vignette
- Finn, T., & Downie, A. (2025). Automatización de recursos humanos. IBM. <https://www.ibm.com/es-es/think/topics/hr-automation>
- Gómez, I. (2024). ¿Por qué son tan importantes los sistemas informáticos en las empresas? Universidad San Marcos. <https://www.usanmarcos.ac.cr/blogs/sistemas-informaticos-en-las-empresas>
- IBM Documentation. (2026). *Db2 for Linux, UNIX and Windows*. IBM. <https://www.ibm.com/docs/es/db2/12.1.x?topic=databases-designing>
- IBM. (2025). Gestión del capital humano. IBM. <https://www.ibm.com/es-es/think/topics/human-capital-management>
- Iwashyna, T. J., & Kennedy, E. H. (2013). Instrumental Variable Analyses. Exploiting Natural Randomness to Understand Causal Mechanisms. *Annals Of The American Thoracic Society*, 10(3), 255-260. <https://doi.org/10.1513/annalsats.201303-054fr>
- KBIBenefits. (2022). ¿Qué son los sistemas de información en la gestión de recursos Humanos? Kbibenefits. https://www-kbibenefits-com.translate.goog/what-are-information-systems-in-human-resource-management?_x_tr_sl=en&_x_tr_tl=es&_x_tr_hl=es&_x_tr_pto=tc

- Lifeder. (2021). *7 Técnicas e Instrumentos para la Recolección de Datos*. <https://www.lifeder.com/tecnicas-instrumentos-recoleccion-datos/>
- Maranto, M., & González, M. (2015). *Fuentes de Información* [Universidad Autónoma del Estado de Hidalgo]. <https://dspace.uaeh.edu.mx/server/api/core/bitstreams/624c644f-fe81-42be-9a5d-2ccde73a78e6/content>
- MC Library Montgomery College. (s. f.). *Identificar diferentes tipos de fuentes de información: gubernamentales, primarias y secundarias*. https://libguides-montgomerycollege-edu.translate.goog/c.php?g=857161&p=8836564&_x_tr_sl=en&_x_tr_tl=es&_x_tr_hl=es&_x_tr_pto=tc
- Medina Romero, M. Á., Rojas León, C., Bustamante Hoces, W., Loaiza Carrasco, R., Martel Carranza, C., & Castillo Acobo, R. (2023). *Metodología de la investigación: Técnicas e instrumentos de investigación* (Primera edición digital) [Instituto Universitario de Innovación Ciencia y Tecnología Inudi Perú S.A.C]. Instituto Universitario de Innovación Ciencia y Tecnología Inudi Perú S.A.C. <https://doi.org/10.35622/inudi.b.080>
- Morganelli, M. (2024). *Una guía sobre los tipos de investigación y cómo se utilizan*. Southern New Hampshire University. <https://www.snhu.edu/about-us/newsroom/education/types-of-research>
- Narváez, O., & Villegas, L. (2014). *Introducción a la investigación: Guía interactiva*. Universidad Veracruzana. <https://www.uv.mx/apps/bdh/investigacion/unidad1/investigacion-tipos.html>
- Open University. (2018). *Enfoques para el desarrollo de software*. Open Learning. https://www-open-edu.translate.goog/openlearn/science-maths-technology/approaches-software-development/content-section-1.5.2?_x_tr_sl=en&_x_tr_tl=es&_x_tr_hl=es&_x_tr_pto=tc
- Oracle América Latina. (2025). *¿Qué son los sistemas de información de recursos humanos (HRIS)?* <https://www.oracle.com/latam/human-capital-management/what-is-hris/>
- Powell, P., & Smalley, I. (2025). Pruebas del sistema. IBM. <https://www.ibm.com/es-es/think/topics/system-testing>
- QuestionPro. (2026). *Tamaño de la muestra. ¿Qué es y cómo calcularla?* <https://www.questionpro.com/es/tama%C3%B1o-de-la-muestra.html>
- Quiroz Vázquez, C. (2026). Desarrollo de software. IBM. <https://www.ibm.com/mx-es/think/topics/software-development>

- Research, A. (2024). ¿Qué es la investigación aplicada? definición, tipos, ejemplos | Appinio. Appinio. <https://www.appinio.com/es/blog/investigacion-de-mercados/investigacion-aplicada>
- Rholetter, E., Purdy, Elena, Popan, P., & MA. (2023). *Investigación descriptiva*. EBSCO. <https://www-ebSCO-com.translate.goog/research-starters/social-sciences-and-humanities/descriptive-research? x tr sl=en& x tr tl=es& x tr hl=es& x tr pto=tc& x tr hist=true>
- Salarios mínimos en Costa Rica*. (2026). Ministerio de Trabajo y Seguridad Social. https://www.mtss.go.cr/temas-laborales/salarios/salario_minimo.html
- Salazar, C. (2021). MARCO TEÓRICO O REFERENCIAL. <https://repositorio.usam.ac.cr/xmlui/bitstream/handle/11506/2623/LEC%20MET%200017%202021.pdf?sequence=1&isAllowed=y>
- Salazar, C. (2021). *POBLACIÓN, MUESTRA y MUESTREO* [Repositorio U San Marcos]. <https://repositorio.usam.ac.cr/xmlui/bitstream/handle/11506/2634/LEC%20MET%200020%202021.pdf?sequence=1&isAllowed=y>
- SalusPlay. (s. f.). *Las variables de investigación*. <https://www.salusplay.com/apuntes/apuntes-metodologia-de-la-investigacion/tema-2-las-variables-de-investigacion>
- Sampieri, R. H., & Torres, C. P. M. (2018). *Metodología de la investigación* (7.^a ed.) [Universidad Pedagógica Nacional - Facultad de Artes]. MCGRAW-HILL / INTERAMERICANA EDITORES, S.A. DE C.V. https://apiperiodico.jalisco.gob.mx/api/sites/periodicooficial.jalisco.gob.mx/files/metodologia_de_la_investigacion_-_roberto_hernandez_sampieri.pdf
- Softland Costa Rica. (2025). Guía de Software de Planillas para Empresarios en Costa Rica - Softland Costa Rica. Softland Costa Rica. <https://softland.com/cr/guia-software-planillas-empresarios/>
- Sommerville, I. (2011). *Ingeniería de software* (9.^a ed.) [Scalared]. Pearson Education, Inc. https://gc..com/recursos/files/r161r/w25469w/ingdelsoftwarelibro9_compressed.pdf
- Stewart, L. (2025). *Variables en la Investigación | Tipos, Definición y Ejemplos*. ATLAS.ti. <https://atlasti.com/es/research-hub/variables-investigacion>
- T. Pham, L.-H., Desai-Naik, T., Hammond, L., & Abdeljabbar, W. (2024). 1.2: Identificación de los componentes de los sistemas de información. LibreTexts Español.

https://espanol.libretexts.org/Bookshelves/Vocacional/Sistemas_de_Informacion_para_Negocios/01%3A_Que_es_un_Sistema_de_Informacion/01%3A_%C2%BFQu%C3%A9_es_un_Sistema_de_Informaci%C3%B3n%3F/1.02%3A_Identificaci%C3%B3n_de_los_componentes_de_los_sistemas_de_informaci%C3%B3n

Tegan, G. (2023). *Investigación exploratoria | Definición, guía y ejemplos*. Scribbr. <https://www.scribbr.com/methodology/exploratory-research/>

U San Marcos. (2020). *Técnicas de investigación* [Repositorio U San Marcos]. Ilumno. <https://repositorio.usam.ac.cr/xmlui/bitstream/handle/11506/1268/LEC%20MET%200008%202020.pdf?sequence=1&isAllowed=y>

Villuendas, B. (2025). *¿Cuáles son las características de un documento de archivo?* 480. <https://cuatroochenta.com/caracteristicas-documento-de-archivo/#:~:text=Los%20documentos%20de%20archivo%20son,contabilidad%20y%20un%20lugar%20etc%C3%A9tera.>

Apéndice A: Cuestionario.

CUESTIONARIO

En el marco de una investigación sobre Sistema web para la gestión y pago de planilla en AraWeb CR, ubicada en San José, le invitamos a completar este cuestionario. Su participación es de gran importancia para comprender cómo el tema en estudio influye en la actividad de la organización.

Este cuestionario es confidencial. Sus respuestas solo se utilizarán con fines académicos y no serán compartidas con otras personas. Completar el cuestionario tomará aproximadamente 10 minutos.

1. ¿El proceso actual de gestión de planilla en la empresa se realiza de manera eficiente?
2. ¿Considera que el proceso actual de pago de planilla presenta retrasos frecuentes?
3. ¿Con qué frecuencia detecta errores en su cálculo de salario?
4. ¿El sistema actual (manual) facilita el acceso rápido a la información de planilla?
5. ¿La información relacionada con su salario es clara y comprensible?
6. ¿Considera que el proceso actual garantiza la confidencialidad de la información salarial?
7. ¿Cree que la implementación de un sistema web mejoraría la organización de la información de planilla?
8. ¿Un sistema web permitiría reducir errores en el cálculo de pagos y deducciones?
9. ¿Considera que un sistema automatizado agilizaría el proceso de pago de planilla?
10. ¿Le gustaría poder consultar su información de planilla en línea de forma segura?
11. ¿Considera que la empresa necesita modernizar el proceso actual de gestión de planilla?
12. ¿La implementación de un sistema web contribuiría a una mayor transparencia en los pagos?
13. ¿Estaría dispuesto(a) a recibir capacitación para utilizar un nuevo sistema web de planilla?
14. ¿Considera que la digitalización del proceso de planilla mejoraría la eficiencia administrativa de la empresa?
15. ¿Recomendaría la implementación de un sistema web para la gestión y pago de planilla en AraWeb CR?

Apéndice B: Guía de Observación.

GUÍA DE OBSERVACIÓN

Entidad:	AraWeb CR.
Dirección física de la entidad:	Pérez Zeledón San José, Costa Rica.
Fecha de la actividad de observación:	14/04/2026
Nombre del estudiante:	Emmanuel Cerdas Brenes.

Tabla de control de aspectos observados:

No	Aspectos por observar	Cumple	No Cumple	Oportunidad de mejora	Detalle de Observación
1	El cálculo de la planilla salarial se realiza mediante un sistema digital automatizado.		X	X	El cálculo de planilla se realiza de forma manual mediante hojas de cálculo, generando riesgo de errores y retrasos.
2	Existe un procedimiento estandarizado para el registro de asistencia de los colaboradores.		X	X	No se evidencia un sistema digital formal; el control depende de registros manuales.
3	Las horas extraordinarias se registran formalmente y cuentan con aprobación documentada.		X	X	El registro no sigue un flujo automatizado ni control centralizado de aprobación.

No	Aspectos por observar	Cumple	No Cumple	Oportunidad de mejora	Detalle de Observación
4	El proceso de solicitud y aprobación de permisos laborales se encuentra documentado y organizado.		X	X	Las solicitudes se gestionan informalmente sin trazabilidad digital.
5	Las incapacidades médicas se registran en un sistema que permita control de fechas y respaldo documental.		X	X	No existe repositorio digital que centralice incapacidades y documentación médica.
6	Existe control actualizado sobre el saldo de vacaciones disponibles por colaborador.		X	X	El control se realiza manualmente, dificultando la consulta inmediata.
7	El cálculo del aguinaldo se realiza mediante un procedimiento automatizado conforme a la normativa laboral vigente.		X	X	El cálculo depende de procesos manuales y revisiones posteriores.
8	La liquidación laboral se calcula utilizando herramientas que minimicen errores manuales.		X	X	No se evidencian herramientas automatizadas especializadas.
9	Se aplican evaluaciones periódicas de desempeño con criterios previamente definidos.	X		X	Se realizan evaluaciones ocasionales, pero sin integración tecnológica formal.

No	Aspectos por observar	Cumple	No Cumple	Oportunidad de mejora	Detalle de Observación
10	La empresa cuenta con un sistema que permita la inserción, modificación y actualización segura de datos administrativos.		X	X	La gestión de datos administrativos no posee control centralizado ni auditoría digital.
11	Se generan reportes formales de planilla, asistencia y pagos con formato estructurado.		X	X	Los reportes se elaboran manualmente, aumentando tiempos operativos.
12	Existe un sistema de autenticación con control de usuarios y perfiles de acceso.		X	X	No se observan mecanismos de login o control de roles para acceso a información.
13	La información relacionada con planilla y recursos humanos se encuentra organizada y disponible para consulta inmediata.		X	X	El acceso a información requiere intermediación administrativa.
14	Se evidencia confidencialidad en el manejo de información salarial y datos personales.		X	X	El manejo manual limita controles de seguridad y trazabilidad de accesos.
15	Los procesos administrativos actuales presentan atrasos atribuibles al manejo manual de información.	X		X	Se observan demoras operativas asociadas a registros manuales y validaciones repetitivas.