

UNIVERSIDAD INTERNACIONAL DE LAS AMÉRICAS

ESCUELA DE INGENIERÍA INFORMÁTICA

**Sistema web para la gestión de pago de planilla en la Veterinaria
Invet Diagnóstico, ubicada en San Pedro Montes de Oca**

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

Fabián Esteban Chaves Delgado

Enero, 2026

DEDICATORIA

A mi papá, que nunca perdió la fe.

AGRADECIMIENTOS

Retomar y concluir este trabajo final de graduación me tomó años. Tuvieron que pasar cuatro años de haber terminado el programa de estudios para que pudiera encontrar la fuerza y la motivación necesarias para finalizar lo que inicié en 2018, cuando apenas tenía 19 años. Hoy, a mis 27, casi 28 años, reconozco en mí una mentalidad completamente distinta a la de aquel entonces. Ese crecimiento y esta meta alcanzada son gracias a mis personas; aquellas que puedo contar con los dedos de las manos.

Para empezar, es imposible no agradecer a mi papá y a mi mamá, quienes me han brindado toda la educación que tengo hasta el día de hoy. De ellos he aprendido mucho más que en cualquier aula. A mi pareja, porque sin su apoyo no sería la persona que soy hoy; gracias por mostrarme de lo que soy capaz y acompañarme a superar mis límites. Y, finalmente, a la razón principal y la fuente inagotable de mi motivación para culminar este arduo trabajo: mi hijo. Sin él no estaría aquí; él es mi mayor motivo para ser mejor cada día.

CONTENIDO

APROBACIÓN DEL TRIBUNAL EXAMINADOR.....	¡Error! Marcador no definido.
DECLARACIÓN JURADA DEL ESTUDIANTE.....	¡Error! Marcador no definido.
Descripción del Problema	12
Objetivos	13
Objetivo General.....	13
Objetivos Específicos.....	13
Justificación	13
Viabilidad Técnica	13
Viabilidad Operativa	15
Viabilidad Económica	16
Viabilidad Legal.....	17
Proyecciones	18
Alcance Funcional	19
Alcance Metodológico	21
Alcance Tecnológico.....	22
CAPÍTULO II: MARCO REFERENCIAL	25
Software	25
<i>Software de Planilla</i>	27
<i>Arquitectura del Software</i>	28
Base de Datos.....	29
Scrum	31
Lenguaje <i>Python</i>	32
Hardware.....	33

Gestión de Recursos Humanos	34
Seguridad de Datos	36
<i>Ley de Protección al Trabajador</i>	37
<i>Deducciones de Ley</i>	38
<i>Ley de Protección de la Persona Frente al Tratamiento de sus Datos Personales (Ley N° 8968)</i>	39
Enfoques de Investigación	41
<i>Enfoque Cuantitativo</i>	41
<i>Enfoque Cualitativo</i>	42
<i>Enfoque Mixto</i>	42
<i>Enfoque de Investigación Seleccionado</i>	44
Tipos de Investigación	44
<i>Investigación Exploratoria</i>	44
<i>Investigación Descriptiva</i>	45
<i>Investigación Correlacional</i>	46
<i>Investigación Explicativa</i>	46
<i>Tipo de Investigación Seleccionado</i>	47
Fuentes de Información.....	47
Fuentes de Información Primaria.....	48
Fuentes de Información Secundaria.....	49
Fuentes de Información Terciaria	49
Variables.....	50
Variable Conceptual	51
Variable Operacional.....	52
Variable Instrumental	52

Población.....	54
Muestra	55
Instrumentos de Recolección de Datos	55
Observación	56
Entrevista	56
Cuestionario	57
CAPÍTULO IV: ANÁLISIS DE RESULTADOS	59
CAPÍTULO V: PROPUESTA.....	70
Análisis Detallado del <i>Software</i> por Desarrollar.....	70
Análisis Detallado del <i>Hardware</i> Requerido	71
Especificaciones Técnicas y Costos de Adquisición	71
Análisis Detallado de los Elementos Relacionados con las Telecomunicaciones	72
Descripción Detallada de las Herramientas Técnicas Utilizadas.....	73
Descripción Detallada del Conocimiento Básico del Recurso Humano.....	74
Elaboración de los Casos de Uso	74
Arquitectura del Sistema.....	88
Arquitectura del <i>Software</i>	89
Diseño de Entradas	91
Diseño Físico de la Base de Datos.....	95
Diccionario de Datos.....	96
Diseño de Procesos	102
Diseño de Salidas.....	103
Diagramas UML	107
Generación de Planilla	107
Solicitud de Horas Extras.....	108

Registro de Asistencia.....	109
Programación	110
Salidas	112
Procesos	113
Validaciones	115
Módulos Señalados en el Alcance.....	117
CAPÍTULO VI: CONCLUSIONES Y RECOMENDACIONES.....	127
REFERENCIAS.....	129
ANEXOS	134
Anexo 1: Guía de Entrevista.	134
Anexo 2: Guía de Cuestionario.....	136
Anexo 3: Guía del Proceso de Observación	138

TABLAS

Tabla 1. Listado de hardware.	14
Tabla 2. Listado de software.	15
Tabla 3. Listado de recursos que se debe adquirir para la implementación del sistema.	16
Tabla 4. Listado por etapas de los costos de desarrollo del sistema.	17
Tabla 5. <i>Cuadro de variables.</i>	53
Tabla 6. Caso de uso: Iniciar Sesión en el Sistema.	76
Tabla 7. Consultar panel principal y notificaciones.	77
Tabla 8. Registrar marca de asistencia.	78
Tabla 9. Solicitar aprobación de horas extras.	79
Tabla 10. Gestionar solicitud de ausencias.	80
Tabla 11. Registro de incapacidades.	81
Tabla 12. Cálculo de liquidaciones.	82
Tabla 13. Reversión de pagos de planilla.	83
Tabla 14. Consulta y descarga de colilla.	84
Tabla 15. Auditoría de marcas y puntualidad.	85
Tabla 16. Registro de nuevos expedientes.	86
Tabla 17. Configuración de ley.	96
Tabla 18. Empleados.	96
Tabla 19. <i>Pagos planilla.</i>	97
Tabla 20. Detalle planilla.	97
Tabla 21. Evaluaciones.	98
Tabla 22. Liquidaciones.	98
Tabla 23. Logs auditoría.	99
Tabla 24. Módulos.	99
Tabla 25. Notificaciones.	100
Tabla 26. Perfiles.	100
Tabla 27. Permisos.	100
Tabla 28. Registro de asistencia.	101
Tabla 29. Solicitudes, ausencias.	101
Tabla 30. Solicitudes, horas extras.	102

Tabla 31. Usuarios.....	102
--------------------------------	-----

FIGURAS

Figura 1. Diagrama de metodología Scrum.	22
Figura 2. Arquitectura Modelo Vista Controlador.	23
Figura 3. Diferencia entre software y hardware.	27
Figura 4. Python: una visión integral para el desarrollo de software.	33
Figura 5. Proforma de Extreme Tech.....	72
Figura 6. Diagrama de casos de uso.	75
Figura 7. Arquitectura del sistema.	89
Figura 8. Arquitectura del software.	90
Figura 9. Formulario de registro de nuevos expedientes.	92
Figura 10. Interfaz del reloj marcador.	93
Figura 11. Formulario de solicitud de ausencias.....	93
Figura 12. Matriz de evaluación de desempeño.	94
Figura 13. Configuración de parámetros de ley.....	95
Figura 14. Diseño físico de la base de datos.....	95
Figura 15. Diseño de procesos.	103
Figura 16. Dashboard de indicadores gerenciales.	104
Figura 17. Colilla de pago en formato PDF.	105
Figura 18. Reporte del auditor de asistencia.....	105
Figura 19. Comprobante de liquidación laboral.	106
Figura 20. Vista integral del expediente del colaborador.....	107
Figura 21. Diagrama de generación de planilla.	108
Figura 22. Diagrama de solicitud de horas extras.....	109
Figura 23. Diagrama de registro de asistencia.	110
Figura 24. Captura múltiple y asignación de variables de sesión en Mantenimiento de Empleados.	111
Figura 25. <i>Captura dinámica de múltiples variables para Parámetros de Ley.</i>	112
Figura 26. Serialización de datos JSON para Indicadores Gerenciales.	112
Figura 27. Generación de salidas inmutables (descarga de colillas en PDF).	113
Figura 28. Algoritmo de cálculo de liquidaciones.	114
Figura 29. Delegación de procesos matemáticos a la base de datos (auditoría).	115

Figura 30. <i>Validación de reglas de negocio mediante iteración (módulo de ausencias).</i>	116
Figura 31. Validación secuencial de estado (reloj marcador).	117
Figura 32. Módulo: Pagar planilla.	118
Figura 33. Módulo: Gestionar horas extras.	119
Figura 34. Módulo: Gestionar permisos (ausencias).	119
Figura 35. Módulo: Gestionar vacaciones.	120
Figura 36. Módulo: Gestionar liquidaciones.	121
Figura 37. Módulo: Gestionar incapacidades.	122
Figura 38. Módulo: Gestionar aguinaldo.	122
Figura 39. Módulo: Evaluar empleado.	123
Figura 40. Módulo: Mantenimiento.	124
Figura 41. <i>Módulo: Consultas (expediente integral).</i>	124
Figura 42. Módulo: Reportes (auditoría).	125
Figura 43. Módulo: Seguridad (roles y permisos).	126

CAPÍTULO I: INTRODUCCIÓN

Descripción del Problema

El proyecto se lleva a cabo en la empresa Invet Diagnóstico, fundada en 2020 por el veterinario Dr. Leonel Altamirano. En un inicio, la empresa tenía sede únicamente en Los Yoses, para posteriormente abrir dos nuevas sedes en Escazú y Cartago. Actualmente, la empresa cuenta con mayor cantidad de personal, por lo que se ha visto en la necesidad de implementar nuevas herramientas para el pago de las planillas. Los empleados han manifestado una molestia debido a que dichos pagos pueden atrasarse o venir incorrectos, esto debido a que, en la actualidad, los cálculos se realizan de manera manual, y los pocos sistemas que utilizan no tienen en cuenta todas las variables de pago, como deducciones, bonificaciones, horas extras, entre otros factores.

Partiendo desde esta misma problemática, no existe un registro centralizado y preciso de las horas extras trabajadas por los empleados. Esto ha generado inconsistencias debido a pagos insuficientes o incompletos o, por el contrario, pagos excesivos de estas horas. De la mano con esto viene el cálculo del monto del aguinaldo, el cual no es consistente y genera desconfianza, ya que suelen existir errores en la recopilación de datos de salarios o en la aplicación de las fórmulas correspondientes para su cálculo.

En el tema de permisos y ausencias también se detectaron puntos de mejora, ya que, al no existir un registro adecuado de estos, se dificulta la gestión de indemnizaciones, los cuales se podrían ver afectados, incluso, de manera legal.

En esta misma línea se añade una problemática con la gestión de incapacidades, ya que no se tiene un control sistemático de estas, lo que puede generar errores en la nómina y complicaciones en la gestión de los recursos humanos disponibles.

Por otro lado, no existe una herramienta para medir el desempeño de los empleados, lo cual dificulta la identificación de áreas de mejora, la planificación de ascensos, la asignación de bonificaciones y la toma de decisiones estratégicas.

Por último, existe una problemática en el tema de control de marcas de entrada y salida ante la falta de un sistema automatizado, lo cual genera errores en el control de asistencia y retrasos en la validación de horas laboradas. Además, el registro manual dificulta la supervisión en tiempo real y la generación de reportes confiables.

Objetivos

Objetivo General

Desarrollar un sistema web para la gestión de pago de planilla, utilizando las herramientas HTML5, CSS3, *JavaScript*, *Python*, con el *framework MySQL* como base de datos, y empleando *Visual Studio Code* como entorno de desarrollo.

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*.
- Programar los diferentes módulos del sistema mediante la herramienta *Visual Studio Code*.
- Probar los diferentes módulos de forma independiente y también de forma integral, garantizando los resultados correctos del sistema.

Justificación

El presente proyecto busca crear una solución al problema actual que presenta la empresa Invet Diagnóstico en el pago de planillas, cálculo de vacaciones y registro de horas. Es de suma importancia que este sistema sea ágil y confiable, ya que anteriormente dependían de registros físicos y del sistema de Excel, lo cual generaba inconsistencias e, incluso, pérdida de datos.

La investigación busca beneficiar a la empresa Invet Diagnóstico y a sus empleados, creando una herramienta que permita a los encargados, brindar mejores y más rápidas soluciones. Los beneficiados con esta implementación no serán únicamente los encargados del Departamento de Recursos Humanos, sino también los empleados, ya que se podrá obtener datos fidedignos para mejorar el pago de horas extras, ascensos, vacaciones e incapacidades; además de que los pagos podrán ser realizados sin atrasos y por los montos correctos, entre otras mejoras que se podrán determinar dentro de la empresa.

Viabilidad Técnica

El proyecto es técnicamente viable, ya que la empresa dispone del equipo necesario para el uso adecuado del sistema y demuestra disposición para, en caso de ser necesario, adquirir el *hardware* y el *software* necesarios.

Seguidamente, se adjunta el listado detallado de todo el equipo que se requiere para su implementación, tanto en *software* como en *hardware*, y se detalla con cuál ya cuenta la empresa y cuál debe adquirir.

Tabla 1.

Listado de hardware.

Componente	Descripción	¿La empresa cuenta con el equipo o debe adquirirlo?
Monitor.	Monitor 22" FHD con panel IPS 75Hz y bordes ultradelgados, marca Samsung.	Sí cuenta con el equipo.
Gabinete de pared.	Gabinete 6U Pared Armado 1 Abanico 540W 450D 313H 50 kgs, LANTEK LTK-GAB6	Sí cuenta con el equipo.
Teclado y <i>mouse</i> .	Teclado y <i>Mouse Logitech Slim</i> MK470 – Español.	Sí cuenta con el equipo.
Silla de oficina.	Silla secretarial de <i>mesh</i> color negro, ergonómica 4tune.	Sí cuenta con el equipo.
Almacenamiento.	Disco duro: DD 6TB HPE SAS 7200RPM 600MB #765259-B21#326078589085	Se debe adquirir el equipo.
Procesador.	AMD Ryzen 7 5700X	Se debe adquirir el equipo.
Memoria RAM.	Memoria RAM DDR4 32GB Kingston 3200MT ECC#KTD-PE432/32G	Se debe adquirir el equipo.
Fuente de Poder.	XPG Kyber 850W – 80 <i>Gold Plus</i> – ATX 3.0	Se debe adquirir el equipo.

Fuente: Elaboración propia (2026).

Tabla 2.*Listado de software.*

Componente	Descripción	¿La empresa cuenta con el sistema o debe adquirirlo?
Entorno de desarrollo.	<i>Visual Studio Code</i> , versión: 1.105.1	Ya cuenta con el sistema, ya que es de uso libre, por lo cual no requiere una inversión.
Base de datos.	<i>My SQL Workbench Community Edition</i> , versión: 8.0.44	Ya cuenta con el sistema, ya que es de uso libre, por lo cual no requiere una inversión.
Lenguaje.	<i>Python 3.14</i>	No aplica.
Estilos.	CSS, Versión 3 y HTML 5	No aplica.

Fuente: Elaboración propia (2026).***Viabilidad Operativa***

El sistema propuesto posee una alta viabilidad operativa, ya que se fundamenta en un diseño enfocado en la experiencia del usuario y la optimización de los procesos de análisis y cálculo de la planilla, vacaciones, incapacidades y entre otros.

Inicialmente, la interfaz gráfica ha sido diseñada implementando estándares reconocidos de diseño, e incorporando elementos que cualquier usuario de sistemas modernos, como redes sociales u otros, sabría utilizar de manera intuitiva, tales como una pantalla de autenticación, también llamada *Log In*, y un sistema de navegación por menús jerárquicos.

Debido a esta familiaridad del sistema, es posible reducir la curva de aprendizaje, al permitir que el proceso de adaptación y capacitación sea más corto. Además, el sistema automatiza los cálculos complejos del pago de la planilla de una empresa mediana, lo cual elimina la posibilidad del error humano en un tema tan delicado como es éste y reduce la carga operativa manual ya ejecutada en la empresa. Al simplificar estas tareas, se garantiza que la implementación del sistema no será percibida como un proceso o una carga adicional, si no como una herramienta facilitadora.

En la parte operativa se cuenta con la disponibilidad del recurso. Tal como se detalló párrafos atrás, la empresa cuenta con el capital para invertir en equipo necesario, además del que ya tiene a disposición. Esto hace que la viabilidad operativa sea aún mayor, porque se entiende

que Invet Diagnóstico cuenta con los recursos físicos y económicos para la correcta implementación del sistema.

Por otro lado, el sistema como tal no requiere de una gran inversión por concepto de mantenimiento o de soporte técnico constante, lo que hace que la empresa pueda hacer uso de la herramienta sin tener que incurrir en gastos adicionales, o inclusive contratación de ingenieros en sistemas para su mantenimiento, tanto de equipos como del sistema. Por lo anterior, se entiende que con las mejoras al sistema no se busca una reducción del personal; por el contrario, se pretende brindar mejores recursos y herramientas para la implementación de nuevos puestos, ascensos y remuneraciones óptimas a los colaboradores.

Viabilidad Económica

El proyecto cuenta con la viabilidad económica, ya que la empresa dispone de un fondo exclusivo para el desarrollo de este sistema. Además, en el tema de *software* no se requiere inversión, ya que el entorno de desarrollo, la base de datos y demás recursos son de uso libre. En cuanto al *hardware*, ya Invet Diagnóstico cuenta con ciertos componentes; por su parte, aquellos que es necesario adquirir, se detallan en un listado de la tabla 3.

Tabla 3.

Listado de recursos que se debe adquirir para la implementación del sistema.

Componente	Descripción	Precio
Almacenamiento.	Disco duro: DD 6TB HPE SAS 7200RPM 600MB #765259-B21#326078589085	€299.727,86
Procesador.	AMD Ryzen 7 5700X	€105.000,00
Memoria RAM.	Memoria RAM DDR4 32GB Kingston 3200MT ECC#KTD-PE432/32G	€82.156,03
Fuente de Poder.	XPG Kyber 850W – 80 <i>Gold Plus</i> – ATX 3.0	€37.900,00
Total		€524.783,89

Fuente: Elaboración propia (2026).

Posterior a la identificación de los costos de *hardware*, es necesario proyectar la inversión en términos de horas de desarrollo técnico. Aunque estos costos no serán facturados a la empresa por tratarse de un proyecto de graduación, representan el valor real de mercado del esfuerzo de

ingeniería aplicado. La tabla 4 desglosa la valoración económica por etapas, fundamentada en los salarios mínimos vigentes para técnicos en sistemas en Costa Rica.

Tabla 4.

Listado por etapas de los costos de desarrollo del sistema.

Etapas del desarrollo	Costo
Análisis	₡ 81.222,50
Diseño	₡487.335
Construcción	₡812.225
Pruebas	₡113.711,50
Total	₡1.494.494

Fuente: Elaboración propia (2026).

Es importante mencionar que la empresa no incurre en los costos que se detallan en la tabla 4, ya que el desarrollo se hará como parte del proyecto de graduación. Además, dichos montos se contemplan con base en el salario mínimo bruto en colones de un Técnico en Sistemas, según la página web del Ministerio de Trabajo y Seguridad Social (MTSS, 2026), el cual es de ₡16.244,50 por hora.

Viabilidad Legal

Para el buen desarrollo del sistema que empleará Invet Diagnóstico, es de suma importancia conocer la viabilidad legal que este tiene, ya que se debe saber que el investigador se compromete a trabajar este proyecto bajo las leyes de seguridad de la información y delitos informáticos. A continuación, se mencionan las leyes aplicables a la propuesta.

La viabilidad legal asegura que el sistema opere bajo el marco jurídico costarricense, protegiendo tanto los intereses de Invet Diagnóstico como los derechos de sus colaboradores. El proyecto se adhiere a la Ley N° 8148 de delitos informáticos (Asamblea Legislativa, 2001), garantizando que el investigador no incurra en conductas que entorpezcan o inutilicen el funcionamiento de sistemas informáticos ajenos. Se respeta la Ley de Derechos de Autor N° 6683 (Asamblea Legislativa, 1982), mediante el uso de herramientas de código abierto y la generación

de código original para el negocio. Finalmente, se aplica rigurosamente la Ley N° 8968 (Asamblea Legislativa, 2011), sobre la protección de datos personales, asegurando la confidencialidad, veracidad y el tratamiento lícito de la información privada de cada empleado dentro de la base de datos.

Para efectos de este proyecto es importante considerar lo mencionado en la Ley N° 8148, *Adición de los artículos 196 BIS, 217 BIS y 229 BIS al Código Penal, Ley N° 4573 para reprimir y sancionar los delitos informáticos* (Asamblea Legislativa, 2001, párr. 4): “Si como resultado de las conductas indicadas se entorpece o inutiliza el funcionamiento de un programa de cómputo, una base de datos o un sistema informático, la pena será de tres a seis años de prisión”.

Por otro lado, en la Ley N° 8968 (Asamblea Legislativa, 2011) se menciona un punto importante a considerar en el tratamiento de los datos utilizados en la base de datos, este es mencionado en el artículo 6, punto 2, Veracidad: “La persona responsable de la base de datos está obligado a modificar o suprimir los datos que falten a la verdad. De la misma manera, velará por que los datos sean tratados de manera leal y lícita. (Párr. 26)

A partir de lo anterior queda en evidencia que los datos serán utilizados de manera discreta, correcta y, además, serán datos veraces, siempre cumpliendo los principios de cada ley citada y teniendo en cuenta el secreto a la información de Invet Diagnóstico.

Proyecciones

Con la creación de este sistema en mención, se pretende crear una herramienta funcional para el día a día de los diferentes departamentos de la empresa Invet Diagnóstico, por lo cual no necesariamente se verá beneficiado únicamente el personal de Recursos Humanos, sino también personal administrativo y de atención al público.

El sistema busca crear una transformación digital dentro de esta empresa en específico, al mismo tiempo que ofrece una solución a los pagos de planillas, registros de marcas, cálculos de horas extras y bonificaciones, además de los cálculos de cargas sociales, como el seguro y la renta. Asimismo, la finalidad es llevar un control real de la efectividad de los colaboradores, para mejorar el control de vacaciones, incapacidades e, inclusive, de posibles ascensos.

Para Invet Diagnóstico, el contar con este sistema le representa un ahorro no solo de espacio físico, sino también del tiempo que se desperdiciaba en llevar controles manuales o buscar documentos que muchas veces se terminaban dando por perdidos. También se benefician

sus empleados, ya que con esta herramienta, los jefes o gerentes pueden llevar un control certero de marcas de entradas y salidas, lo cual generaba descontento entre el personal, ya que el método empleado consistía en enviar un mensaje a un chat de *WhatsApp*, por lo que tenían que hacer uso de sus dispositivos móviles personales.

Por otro lado, al mejorar el control de vacaciones, incapacidades y marcas, el Departamento de Recursos Humanos podrá optimizar el control del personal necesario para cubrir horarios, vacaciones y, en caso de ser necesario, hacer los rebajos correspondientes de cargas sociales por incapacidades. Estos puntos le permitirán a la empresa hacer los pagos correctos y no tener pérdidas en el pago de planilla; además, sus colaboradores dispondrán de sus pagos oportunamente, incluso si se encuentran fuera por alguna de las razones expuestas.

Alcance Funcional

En el alcance funcional se describirán todas las capacidades que tiene el sistema, además del detalle de cada uno de los módulos con los que contará el sistema. Cada módulo será de suma importancia para que la herramienta cuente con todo lo necesario para que sea útil y funcional.

Módulo 1: Pagar Planilla.

Este módulo se encargará de automatizar el cálculo de salarios, deducciones, bonificaciones y pagos adicionales. Incluirá funcionalidades para generar nóminas electrónicas, programar pagos automáticos y enviar notificaciones a los empleados sobre sus recibos de pago. También integrará verificaciones de errores para garantizar la precisión de los montos antes de proceder con los pagos.

Módulo 2: Gestionar Horas Extras.

Este módulo permitirá el registro automático de las horas de entrada y salida de los empleados, así como el cálculo de horas extras laboradas. Incluirá opciones para la validación y aprobación de las horas extras por parte de los supervisores y proporcionará reportes detallados de las horas trabajadas por cada empleado.

El colaborador realizará la solicitud vía sistema a su jefatura inmediata, si esta la aprueba, entonces se notifica vía sistema a Recursos Humanos y este departamento se encargará de notificar al usuario vía sistema. Si la jefatura inmediata no la aprueba, entonces se notifica vía sistema al colaborador, que fue rechazada su solicitud.

Módulo 3: Gestionar Permisos.

Este módulo permitirá a los empleados solicitar permisos de ausencia laboral a través del sistema, que luego podrán ser aprobados o rechazados por los supervisores. Se mantendrá un registro de todas las solicitudes y aprobaciones, proporcionando una vista clara y actualizada de las ausencias.

El colaborador realizará la solicitud vía sistema a su jefatura inmediata; si esta la aprueba, entonces se notifica vía sistema al Departamento de Recursos Humanos y esta dependencia se encargará de notificar al usuario igualmente vía sistema. Si la jefatura inmediata no la aprueba, entonces se notifica al colaborador, vía sistema, que fue rechazada su solicitud.

Módulo 4: Gestionar Vacaciones.

Este módulo rastreará automáticamente la acumulación de días de vacaciones de acuerdo con las políticas de la empresa y permitirá a los empleados solicitar vacaciones a través del sistema. También facilitará la aprobación por parte de los supervisores y mantendrá un registro actualizado de los días de vacaciones disponibles y los ya disfrutados.

Módulo 5: Gestionar Liquidaciones.

Este módulo automatizará el cálculo de las liquidaciones con base en las políticas de la empresa y la legislación laboral vigente. Incluirá todos los elementos necesarios, como indemnizaciones, saldo de vacaciones, aguinaldo proporcional, entre otros, para garantizar pagos correctos y oportunos.

Módulo 6: Gestionar Incapacidades.

Este módulo permitirá registrar y hacer seguimiento a las incapacidades de los empleados. Incluirá opciones para cargar certificados médicos y calcular automáticamente el impacto de las incapacidades en la nómina, asegurando que se realicen los ajustes salariales adecuados.

Módulo 7: Gestionar Aguinaldo.

Este módulo calculará automáticamente los aguinaldos de los empleados con base en los salarios devengados durante el periodo correspondiente. Permitirá efectuar ajustes manuales en caso de ser necesario, y generará reportes que muestren claramente cómo se llegó a cada monto, asegurando transparencia y precisión.

Módulo 8: Evaluar Empleado.

Este módulo permitirá la creación y gestión de evaluaciones de desempeño para empleados. Incluirá herramientas para establecer metas, criterios de evaluación, recopilación de *feedback* de supervisores y compañeros, y generación de informes de desempeño que ayuden a tomar decisiones informadas sobre el desarrollo y la gestión del talento en la organización.

Módulo 9: Mantenimiento.

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

Módulo 10: Consultas.

Este módulo se encargará de generar información proporcionada a partir de las diferentes tablas.

Módulo 11: Reportes.

Este módulo se encargará de generar información proporcionada a partir de las diferentes tablas y procesos, pero con un formato específico, según lo solicite el usuario. Permite ser visualizado en pantalla y podrá ser impreso.

Módulo 12: Seguridad.

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

Alcance Metodológico

El método utilizado será Scrum, ya que es una herramienta que ayuda a agilizar procesos dentro del desarrollo de un sistema de *software*. Como menciona Lomelí (2023),

Scrum se enfoca en la planificación del trabajo dividiendo el proyecto en partes llamadas *sprints*. Cada *sprint* tiene una duración de dos a cuatro semanas y su objetivo es construir las características más importantes y generar un producto potencialmente entregable. (párr. 5).

Por medio de esta metodología se pretende crear un producto final potencialmente funcional, ya que Scrum permite dividir procesos en pequeñas partes, también conocidas como *sprints*. Es un método que ayuda a tener un sistema en menor tiempo, por lo cual, debido a la segmentación de tareas, permite un avance más constante debido a que los periodos de desarrollo se ejecutan en plazos de entre una y cuatro semanas.

También, el método permite la retroalimentación por parte del cliente; en este caso, la empresa. Esto favorece efectuar ajustes en tiempo prácticamente real, ya que se realizan al finalizar cada tarea o *sprint*.

Por la naturaleza propia de este método, la cual se constituye a partir de equipos de trabajo, no se puede implementar en este proyecto de graduación debido a que se requiere de más personal. Por ende, el investigador debe asumir los roles que se consideren más importantes para el desarrollo del sistema, siempre manteniendo la estructura Scrum.

Se adjunta una imagen que permite la comprensión visual del proceso de trabajo del método Scrum.

Figura 1.

Diagrama de metodología Scrum.



Fuente: Lomelí (2023).

Alcance Tecnológico

Para la implementación del sistema, se utilizará el aplicativo web. Por la naturaleza del proyecto se implementan diversas tecnologías y programas que se entrelazarán para el funcionamiento correcto del sistema.

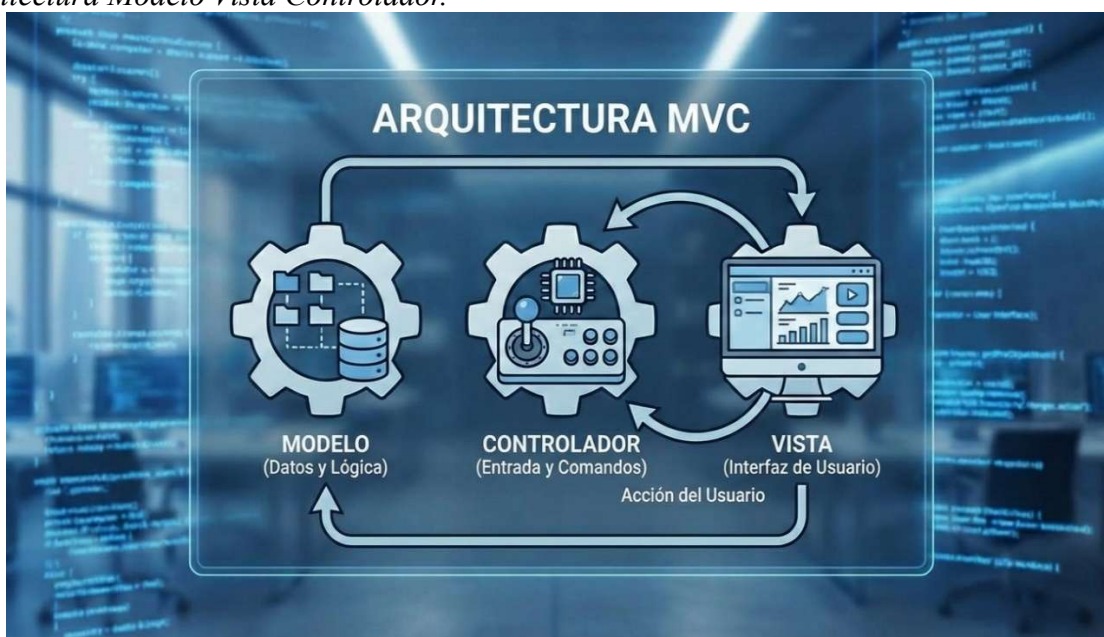
Inicialmente, se utilizará el ambiente de desarrollo *Visual Studio Code*, versión: 1.105.1, el cual actualmente pertenece a la compañía *Microsoft*. Esta es una herramienta que nace para la comunidad de desarrolladores, gratuita y de libre uso, lo cual la hace perfecta para la implementación del presente proyecto. Además, por su amplia documentación y soporte, la convierte en un sistema desarrollo de alta fidelidad.

En el ámbito de bases de datos, será usado el sistema llamado *MySQL Workbench*, cuyo actual dueño es *Oracle Corporation*. Dentro de las principales ventajas que brinda el uso de este motor de base de datos está la herramienta *ER Diagramming*, la cual permite la creación de las tablas y las relaciones entre estas de una manera visual. También la flexibilidad para realizar la migración de datos a diferentes motores de bases de datos.

Por otra parte, como lenguaje de programación se aplicará *Python*, el cual permite una amplia agilidad en el sistema. Para el *back end*, o para la lógica de este, se realizará con el *framework* llamado *Flask*, el cual manejará la conexión entre el sistema propio y la base de datos, el inicio de sesión y procesamiento de datos. Otro *framework* empleado será *Bootstrap*, que es una biblioteca de CSS y JS, que provee los elementos visuales de la página web.

Finalmente, como modelo de arquitectura será implementado el Modelo-Vista-Controlador (MVC). Este modelo es reconocido por ser ampliamente utilizado en desarrollos web, ya que segmenta el patrón de diseño en tres componentes principales. Esa segmentación se da, como sus siglas lo mencionan, primero por medio del modelo, que es el encargado de gestionar la lógica de datos e interacción de la base; segundo, la vista, que es la interfaz gráfica con la que interactúa el usuario, y por último el controlador, que es la conexión entre el modelo y la vista.

Figura 2.
Arquitectura Modelo Vista Controlador.



Fuente: IA Flow (2026).

El modelo MVC, para el presente proyecto, se puede definir como Modelo igual a conexión de *Python* con *MySQL*, Vista igual a *Bootstrap* y el HTML, y Controlador igual a *Flask*. De esta manera, se entrelazarían todos los sistemas implementados en el proyecto.

CAPÍTULO II: MARCO REFERENCIAL

En el presente apartado se pretende definir los conceptos y leyes esenciales para una mejor comprensión del presente proyecto. Es importante que el lector pueda tener una base sólida escrita, ya que a partir de ahí se podrá obtener una mejor visión del trabajo programado. Este marco es de suma importancia, ya que se utilizarán los conceptos obtenidos a lo largo de la carrera de Ingeniería en Sistemas, pero también se nutrirán estos conceptos con la información obtenida a partir de diversos autores investigados.

El marco referencial permite, además, tener bases sólidas para la conceptualización de los temas, esto debido a que, como su nombre lo dice, es una referencia a la literatura ya existente con base en un tema específico.

En investigaciones para mejorar procesos, desarrollar algoritmos y generar nuevas tecnologías y productos, parte de la revisión de la literatura consiste en considerar y evaluar los anteriores procesos, algoritmos, tecnologías o productos, recordando que el planteamiento implica analizar en qué variables puede optimizarse. (Hernández Sampieri *et al.*, 2014, p. 74)

Este proceso de referencias, desarrollo de conceptos y temas por citar, es circunstancial para que un trabajo escrito sea bien entendido, y con ello su proyecto tecnológico. Si solo existiese el proyecto, sin las bases teóricas, este no podría ser analizado ni entendido por cualquier persona con o sin conocimiento en sistemas de desarrollo web.

A continuación, se especifican por tema, los principales conceptos empleados dentro de la presente propuesta.

Software

El termino *software* suele ser muy utilizado dentro del mundo de la Ingeniería en Sistemas, ya que es un elemento esencial. Este término se refiere a toda la parte intangible de un sistema informático, es decir, la parte lógica; se comete el error de creer que es solamente de una computadora, pero abarca más que solo esto: son todos los datos y procesos que se llevan a cabo dentro de los equipos para que estos funcionen. Como menciona Casals (2025) “en términos sencillos, el *software* engloba los programas, las reglas y las instrucciones que permiten que un

ordenador ejecute diversas funciones” (párr. 1); en otras palabras, es la base sobre la que existe la tecnología actual.

Es importante entender que el *software* existe para que se puedan ejecutar programas, sistemas operativos como en el caso de la presente investigación: sistemas de planillas. Éste en conjunto con el *hardware* (el cual se definirá más adelante), son elementos determinantes en el mundo de la tecnología. El *software* tiene como función esencial controlar todo lo que hace una computadora, un celular o un dispositivo tecnológico. “La función básica del *software* es permitir que el usuario interactúe con el *hardware* de manera eficiente”. (Casals, 2025, párr. 6)

El IEEE (*Institute of Electrical and Electronics Engineers*) es una de las organizaciones más importantes dentro del mundo tecnológico y científico. Fue fundada en 1963 y actualmente es líder en el desarrollo de normas y estándares que son aplicados a nivel mundial. Esta organización define *software* como un conjunto de programas informáticos, procedimientos y documentos asociados al funcionamiento de un sistema informático, como ejemplo archivos de comandos y lenguaje de control de trabajos ejecutables o no. (ISO/IEC/IEEE, 2017)

Dentro del *software* existen diferentes tipos, este puede ser de sistemas, de aplicación, de programación, malicioso, antivirus, *software* CRM (para gestión de clientes) y *software* especializado. Todos son útiles dentro del mundo de la programación; sin embargo, para el presente proyecto, el tipo de *software* seleccionado es de aplicación, ya que, como lo define Casals (2025, párr. 14), este puede usarse para que el usuario ejecute diversas tareas específicas dentro de un sistema.

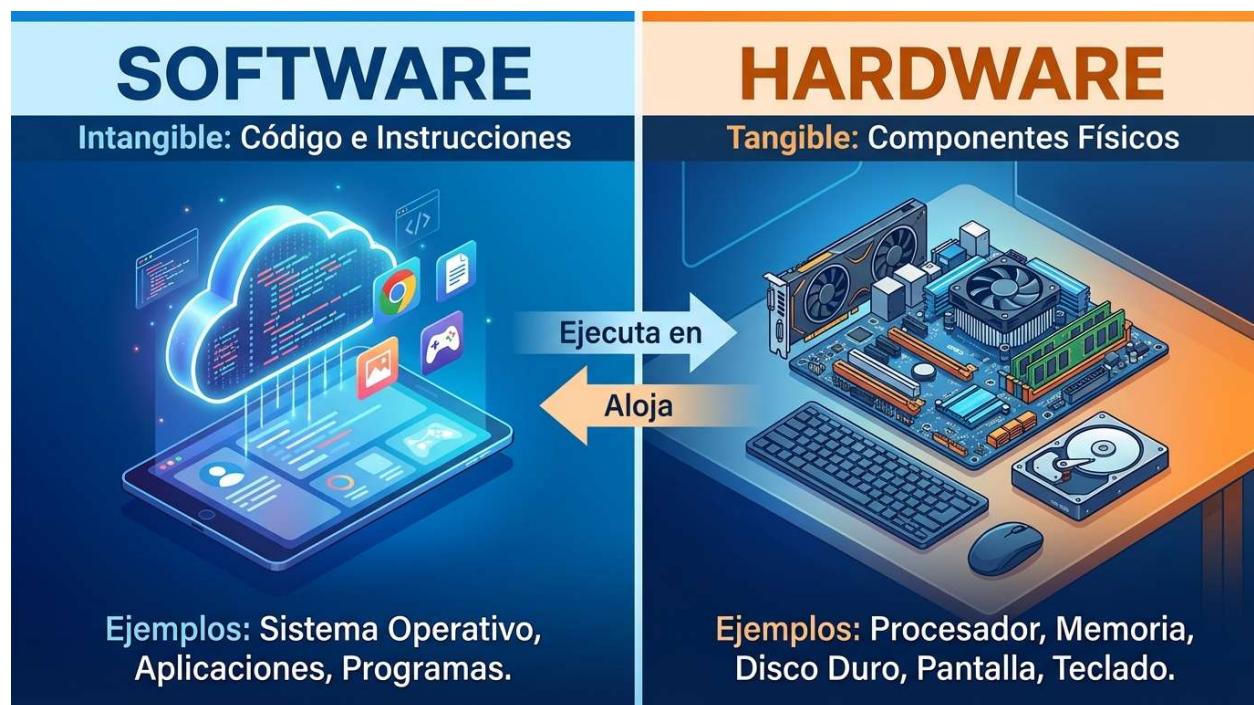
Este *software* no debe ser confundido con el de sistemas, el cual se encarga de gestionar, como tal, los procesos de la computadora. El *software* de aplicación se enfoca más en procesos técnicos dentro de la empresa, como en el caso del presente proyecto: la ejecución de las tareas dentro del sistema de nómina. Según Pressman (2020, p. 6), el *software* de aplicación se utiliza para controlar funciones de negocios o facilitar la toma de decisiones técnicas o de gestión; es decir, este funciona como controlador de funciones en tiempo real.

En resumen, el *software* comprende los componentes no visibles que existen dentro de un programa, sistema, computadora o dispositivo tecnológico. Es usado en todos los campos de la Ingeniería de Sistemas y para el presente proyecto no será la excepción, ya que se empleará el *software* de aplicación, el cual, por sus características, es el adecuado para la ejecución del proyecto.

Como ya se indicó, el *software* trabaja de la mano con el *hardware*, el cual se definirá ampliamente más adelante en este proyecto; sin embargo, se muestra una imagen (figura 3) para entender mejor estas diferencias.

Figura 3.

Diferencia entre software y hardware.



Fuente: IA Flow (2026).

Software de Planilla

Para determinar correctamente lo que es un *software* de planilla, se debía definir primero el principal concepto de “*software*”. Una vez definido, se puede entender que este tipo de *software*, también conocido como *software* de nómina, está ligado directamente con la creación de sistema de gestión de recursos humanos. Para Softland Inversiones S. L. (2024), “es una herramienta digital diseñada para gestionar el pago de los empleados, incluyendo el cálculo de salarios, deducciones y otros beneficios” (párr. 6). Este tipo de *software* es una herramienta base, ya que aporta beneficios a la empresa.

Dentro de las características principales que debe tener un buen sistema de nómina, destacan la “automatización de cálculos y deducciones, cumplimiento de leyes laborales y fiscales costarricenses, accesibilidad y seguridad de datos, integración con otros sistemas empresariales, generación de reportes y análisis, soporte y actualizaciones continuas” (Softland

Inversiones S. L., 2024). Es importante que este *software* pueda llevar a cabo todos estos procesos, ya que le permitirá al patrono, o en este caso al área correspondiente del pago de nómina, tener un mayor control sobre los pagos y la gestión de Recursos Humanos en general.

Un *software* de planilla evitará llevar el control en hojas de Excel o documentos físicos, los cuales se pueden perder y, con ellos, la información de la empresa. Este tipo de sistema, por su parte, permitirá no solo hacer pagos de nómina, sino también disponer de una base de datos sólida con la información de los colaboradores. De esta manera, la empresa podrá tomar decisiones basadas en estos datos, los cuales no se eliminarán con el pasar del tiempo y serán de importancia para procesos internos de la empresa.

Un *software* de nómina debe cumplir con funciones que ayuden y agilicen procesos dentro de la empresa, es por ello que *NCQ Technologies* (2021), empresa costarricense dedicada específicamente al desarrollo de *software* y soluciones comerciales, describe las funciones que debe cumplir un sistema de nómina. La primera de ellas son los procesos automáticos: “es importante que ofrezca funciones de generación automática de informes que simplifiquen la gestión de RRHH y la elaboración de presupuestos y procesos estratégicos similares”. (párr. 7)

Por otro lado, están la facilidad de uso y las funcionalidades del sistema. Aquí se menciona la importancia de que este *software* tenga una interfaz amigable con el usuario, y que las funciones que ofrezca permitan llevar registros de control de marcas, permisos y vacaciones. También se añade dentro de las funciones con las que debe contar el *software*, las políticas de seguridad, donde siempre se asegure que la información que se ingrese en estos sistemas sea totalmente confidencial y que no haya riesgo de fuga de datos. (*NCQ Technologies*, 2021, párr. 7-10)

Es importante conocer estos conceptos dentro de la creación de un marco referencial, ya que ayudan a la comprensión del por qué se crea este sistema de pago de planilla, más allá de solo el proyecto tecnológico. Como afirma *NCQ Technologies* (2021), se debe tener un soporte técnico como respaldo, el cual se encargará de ir automatizando procesos que pueden variar con el tiempo.

Arquitectura del Software

La arquitectura del *software* es un elemento de suma importancia dentro del desarrollo del *software*; según Pressman (2020), “permite analizar la efectividad del diseño para cumplir los requerimientos establecidos, considerar alternativas arquitectónicas en una etapa en la que hacer

cambios al diseño todavía es relativamente fácil y reducir los riesgos asociados con la construcción del *software*”. (p. 207)

Es importante aclarar que en la arquitectura del *software* no se especifican propiedades internas, como los detalles del algoritmo, por ejemplo; esta arquitectura involucra más las características que se requieren para que los procesos del *software* interactúen entre sí (Pressman, 2020, p. 207). No se deben confundir los términos de arquitectura y diseño, ya que estos son conceptos que sí cuentan con un distintito entre sí dentro del *software*.

Los autores Bass, Clements y Kazman (2003, citados por Pressman, 2020), definen la arquitectura del *software* como “la estructura o estructuras del sistema, lo que comprende a los componentes del *software*, sus propiedades externas visibles y las relaciones entre ellos” (p. 207). Esto confirma lo descrito, donde se explica la diferencia que existe entre la arquitectura del *software* y el diseño de este. Es importante que se tomen en cuenta estos conceptos, ya que el sistema de planilla se basará en lo aquí mencionado para su correcto funcionamiento.

Para efectos del presente trabajo, la arquitectura específica que se implementará será una arquitectura centrada en los datos, complementada con el patrón de diseño arquitectónico Modelo-Vista-Controlador (MVC). En este enfoque, el repositorio de datos (la base de datos relacional) se ubica en el centro de la arquitectura, y los demás componentes del *software* acceden a él para agregar, actualizar o manipular la información de forma directa.

Se aplicará este estilo de arquitectura debido a que el núcleo de un sistema de planillas depende de la integridad, precisión matemática y el manejo transaccional ágil de la información. Al utilizar un enfoque centrado en los datos (apoyado por programación imperativa y consultas SQL directas), se garantiza un control algorítmico absoluto sobre el cálculo de las deducciones, horas extras y salarios netos. Esta decisión técnica evita la sobrecarga de procesamiento (*overhead*) que típicamente generan los mapeadores objeto-relacionales (ORM) en las arquitecturas puramente orientadas a objetos, permitiendo que el motor de la base de datos asuma el trabajo pesado, mientras el marco de trabajo (*Flask*) enruta las vistas e interfaces de manera eficiente.

Base de Datos

Una base de datos es un sistema computarizado para almacenar información. Estos datos que se almacenan pueden ser modificados, eliminados, actualizados e, incluso, recuperados. Para

Bermúdez León (2020), “una base de datos se entenderá como una colección de datos relacionados entre sí y que tienen un significado implícito” (p. 4). Estos datos deben ser reales y pueden contener información con diversos significados; por ejemplo, una base de datos que contenga la información del personal que trabaja para una empresa.

Las bases de datos surgen de la necesidad de tener un mejor control y manejo de datos, ya que, en el pasado, estos solían estar almacenados en archivos tradicionales, los cuales solían presentar múltiples errores, como pérdida total de información, duplicación de información e, incluso, que la información estuviese desorganizada y fuese más difícil su análisis.

Como afirma Bermúdez León (2020), en una base de datos, los datos se almacenan de una manera más formal, lo que permite tener un control sobre estos, inclusive por parte de diversos usuarios.

Las bases de datos permiten, en resumen, tener un control de la información, en este caso de la empresa Invet Diagnóstico, lo que asegura un mejor manejo de los recursos, y para este proyecto en específico, una mayor eficiencia en el pago de nóminas.

En los sistemas de *software*, al gestionar y almacenar los datos, estas bases facilitan el acceso a la información y, además, su manipulación directa. Son un elemento esencial dentro de la creación de un sistema de *software*.

Desde una perspectiva técnica, la implementación de una base de datos en este proyecto responde a la necesidad de superar las limitaciones de los sistemas de archivos tradicionales. No se trata solo de guardar datos, sino de garantizar su calidad y disponibilidad. El sistema requiere que la información cumpla con criterios de integridad, asegurando que los datos sean exactos y consistentes a lo largo de su ciclo de vida; y de seguridad, permitiendo que solo los usuarios autorizados accedan a módulos sensibles del sistema.

Finalmente, la interacción con estos datos se realizará a través de un Sistema Gestor de Bases de Datos (SGBD), el cual actuará como la interfaz entre el *software* de aplicación (*backend* en *Python*) y los datos físicos almacenados. El uso de un SGBD robusto permite centralizar el control de la información, garantizando propiedades esenciales para las transacciones financieras, como la atomicidad y la durabilidad. De esta manera, la veterinaria podrá contar con un respaldo histórico confiable de todos los movimientos de planilla, facilitando no solo el cumplimiento de las obligaciones laborales actuales, sino también la auditoría futura de los procesos administrativos.

Scrum

Scrum se define como un marco de trabajo liviano que ayuda a las personas, equipos y organizaciones a generar valor a través de soluciones adaptativas para problemas complejos. Según Schwaber y Sutherland (2020), Scrum no es una metodología rígida, sino una estructura que contiene un conjunto de valores, principios y prácticas que permiten la transparencia, la inspección y la adaptación continua. En la actualidad, el desarrollo de sistemas de información requiere metodologías que permitan responder rápidamente a los cambios en los requerimientos. Por esta razón, para este proyecto se ha seleccionado Scrum como marco de trabajo.

La base teórica de Scrum es el empirismo, esto significa que el conocimiento se adquiere a través de la experiencia y la toma de decisiones basada en lo que realmente se observa durante el desarrollo. Schwaber y Sutherland (2020) explican que el control empírico de procesos se sostiene en tres pilares: transparencia, inspección y adaptación. A partir de estos pilares, el flujo de trabajo que debe implementarse obligatoriamente para que el marco funcione es el siguiente:

1. Un propietario del producto (*Product Owner*) ordena el trabajo de un problema complejo en un *Product Backlog*.
2. El equipo de Scrum convierte una selección del trabajo en un incremento de valor durante un *sprint*.
3. El equipo de Scrum y sus partes interesadas (*stakeholders*) inspeccionan los resultados y realizan los ajustes necesarios para el próximo *sprint*.
4. Repetir. (Schwaber y Sutherland, 2020, p. 3)

La implementación de Scrum permite organizar el proyecto en bloques fijos de tiempo llamados *sprints*, esto facilita la gestión de los entregables y asegura que el sistema final cumpla con las necesidades reales de la organización, permitiendo ajustes técnicos sobre la marcha sin comprometer la totalidad del proyecto.

Los artefactos de Scrum representan trabajo o valor y están diseñados para maximizar la transparencia de la información clave, estos son de suma importancia dentro del desarrollo del proyecto, ya que permiten a los creadores llevar un orden sistemático de lo que se está realizando dentro del proyecto. Por un lado, se tiene el *Product Backlog*, también conocida como “la pila del producto”, esta es una lista ordenada de todo lo que se sabe que el producto final necesita tener. Por otro lado, está lo que son los *sprints* y, por último, está el “incremento”, que es la

suma de todos los elementos del *Product Backlog* que se han completado durante un *sprint*. (Schwaber y Sutherland, 2020)

Lenguaje *Python*

Python es un lenguaje de programación orientado a objetos de alto nivel y fácil de interpretar, actualmente es uno de los lenguajes de programación más usados por desarrolladores a nivel mundial. Debido a su posibilidad de ajuste, este lenguaje permite al usuario una mayor adaptabilidad a los proyectos que se desee crear; es decir, se puede emplear en la creación de sitios web y *software*, como en el caso del presente proyecto, se puede utilizar en la automatización de tareas y también en el análisis de datos. “*Python* es un lenguaje de propósito general, lo que significa que se puede utilizar para crear variedad de programas diferentes y no está especializado en ningún problema específico”. (Coursera Staff, 2023, párr. 3)

La *Python Software Foundation* (2023) lo define de manera técnica, como un lenguaje interpretado, interactivo y orientado a objetos. Incorpora módulos, excepciones, tipado dinámico y clases de muy alto nivel. Su naturaleza interpretada permite que el código se ejecute línea por línea, lo que facilita la depuración y el desarrollo rápido de prototipos, una característica esencial para el ciclo de vida de este proyecto. Esta característica es fundamental en el desarrollo de *software* moderno, ya que reduce la complejidad del mantenimiento y facilita la colaboración entre programadores. Esta cualidad es primordial, ya que asegura que el código fuente sea comprensible para futuros desarrolladores o auditores del sistema.

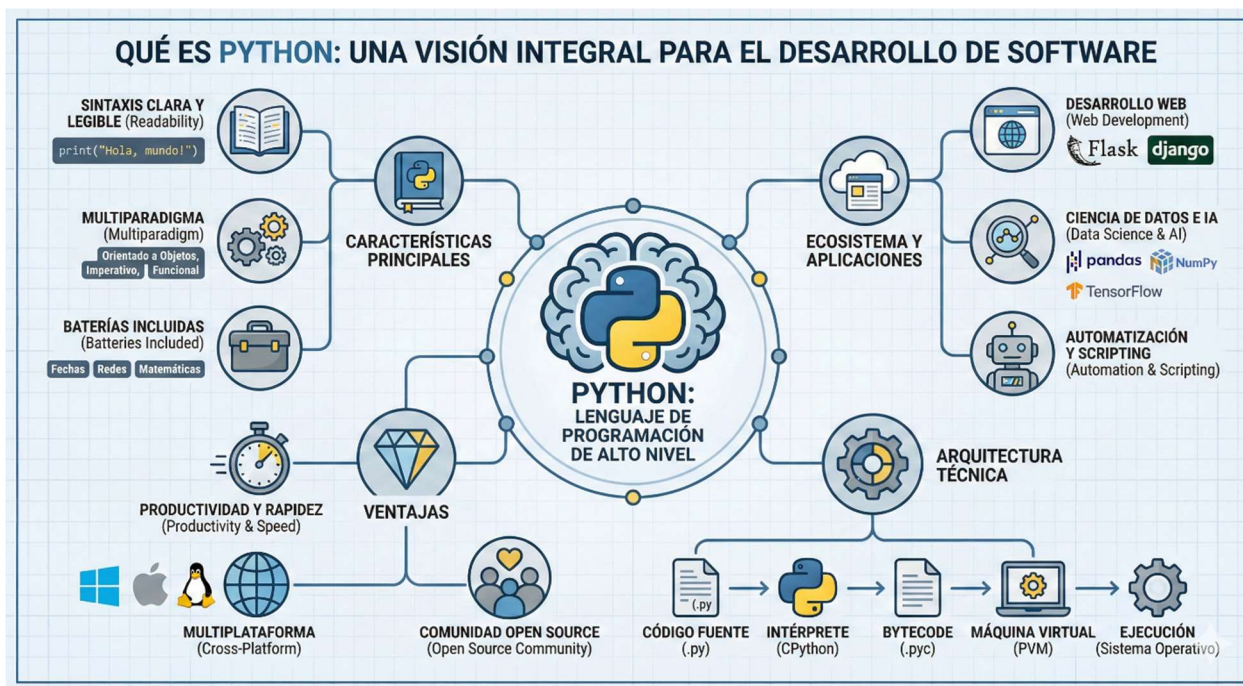
Una de las mayores ventajas de este lenguaje es su gran biblioteca estándar. La *Python Software Foundation* (2023) destaca que el lenguaje proporciona herramientas preconstruidas para realizar multitud de tareas complejas sin necesidad de código externo, desde la manipulación de expresiones regulares hasta la implementación de protocolos de red (párr. 2). Para un sistema de gestión de planillas, esto implica que funcionalidades críticas, como el manejo de fechas para los periodos de pago, ya están optimizadas y probadas, reduciendo significativamente el riesgo de errores en la implementación.

Finalmente, la elección de *Python* se justifica por su precisión en el manejo de datos y su integración con bases de datos. *Python* es fuertemente tipado, lo que evita operaciones ilógicas entre datos incompatibles. Lutz (2013) argumenta que esta característica, sumada a su facilidad para conectarse con motores como *MySQL*, asegura la integridad de la información procesada.

Para este proyecto, dicha robustez es indispensable para garantizar la exactitud de los cálculos salariales, las deducciones de la Caja Costarricense del Seguro Social (CCSS) y los reportes financieros que exige la ley.

Figura 4.

Python: una visión integral para el desarrollo de software.



Fuente: IA Gemini (2026).

Hardware

Como se mencionó, el *hardware* es otra de las partes esenciales dentro del desarrollo de cualquier sistema, esta es la parte física que se requiere para el procesamiento de los datos y de las aplicaciones que se han descrito. Es decir, sin el *hardware* no se puede ejecutar ningún sistema, son los componentes que permiten que haya interacción entre los programadores y todo lo demás. Cómo describe Graglia (2025), “se refiere a los componentes físicos y tangibles de una computadora o sistema electrónico. A diferencia del *software*, que existe como código o datos, el *hardware* es el equipo que se puede ver, tocar e incluso oír”. (párr. 4)

Es importante entender que el *hardware* no es únicamente la computadora, abarca muchos más elementos que son necesarios para el funcionamiento de los sistemas, dentro de estos se pueden definir los *routers*, que son los que permiten el uso de Internet en los

dispositivos, los conmutadores “gestionan el flujo de datos dentro de una red” (Graglia, 2025, párr. 13); y también están los *firewalls*, estos se encargan de dar protección a los sistemas en caso de existir amenazas como virus o intentos de *hackers*.

Por otro lado, se encuentran los dispositivos externos que permiten la interacción humana con la computadora y sus sistemas, entre ellos se encuentran los teclados, el *mouse*, escáner, impresoras, parlantes, monitores, unidades USB, discos externos, monitores y demás. Es importante conocer estos equipos ya que son de suma importancia dentro de la ejecución de los sistemas.

Para comprender de una mejor manera el funcionamiento de una computadora, se debe saber cuáles son sus componentes: “placa madre, CPU, memoria RAM, dispositivos de almacenamiento, fuente de alimentación y la unidad de procesamiento gráfico” (Graglia, 2025). Estos elementos suelen estar en todas las computadoras o al menos son la base para su funcionamiento. También entran como dispositivos del *hardware*, los sistemas de refrigeración, los cuales son los encargados de que todos los demás componentes se mantengan en temperaturas adecuadas para su buen funcionamiento.

Así, el *hardware* es esencial dentro del desarrollo del sistema para Invet Diagnóstico, ya que será la herramienta física que permita el desarrollo del sistema; además, será el componente con el cual los usuarios podrán acceder al sistema de planillas de manera fácil y sencilla.

En el capítulo I, la tabla 1: *Listado de hardware*, contiene una lista de todos los componentes del *hardware* necesarios para la implementación del presente proyecto.

Gestión de Recursos Humanos

La gestión de recursos humanos (GRH) ha experimentado una transformación significativa en la última década, pasando de ser un departamento puramente operativo a convertirse en un socio estratégico del negocio. En el entorno actual, caracterizado por la competitividad y la necesidad de eficiencia, la administración del capital humano es fundamental para el éxito organizacional. Según Chiavenato (2020), "la gestión del talento humano es el conjunto de políticas y prácticas necesarias para dirigir los aspectos de los cargos gerenciales relacionados con las personas o recursos humanos, incluidos reclutamiento, selección, capacitación, recompensas y evaluación de desempeño" (p. 9). Esta visión integral es la que debe

adoptar la veterinaria para asegurar no solo la retención de su personal médico y administrativo, sino también su satisfacción laboral.

Un sistema de gestión de recursos humanos eficiente debe garantizar que los pagos se realicen con exactitud, puntualidad y transparencia; por tanto, la automatización de este proceso mediante un sistema web, elimina la subjetividad y los errores manuales que podrían generar desconfianza en el equipo de trabajo.

En el contexto específico de las pequeñas y medianas empresas (Pymes) en Costa Rica, la gestión de recursos humanos suele enfrentar el desafío de la informalidad en sus procesos. Muchas organizaciones carecen de herramientas tecnológicas adecuadas, dependiendo excesivamente de hojas de cálculo propensas a errores. Por ende, la implementación de un *software* especializado permite estandarizar los procesos de nómina, asegurando que cada cálculo de horas extra, vacaciones o incapacidades se realice bajo los mismos criterios para todos los colaboradores. Esto profesionaliza la gestión administrativa y permite a los líderes de la veterinaria enfocarse en la atención de los pacientes.

Además, la gestión moderna de recursos humanos implica un estricto cumplimiento del marco legal laboral, el cual es cada vez más complejo y fiscalizado. El sistema propuesto actuará como un garante de este cumplimiento, integrando las variables de ley en su lógica de programación. Como señala Porret Gelabert (2020), la dirección de personas exige un conocimiento profundo de la normativa vigente, pues el desconocimiento de la ley no exime de su cumplimiento, y los errores en la aplicación de los derechos laborales pueden derivar en costosas demandas para la empresa. El *software*, al estar actualizado con los parámetros de la CCSS y el Ministerio de Trabajo y Seguridad Social, mitiga este riesgo legal.

Finalmente, la centralización de la información del personal en una base de datos única facilita la toma de decisiones estratégicas. Al tener un registro histórico de los pagos, las ausencias y las horas laboradas, la administración puede analizar tendencias, como el aumento de costos en ciertas temporadas o la rotación de personal. Esta capacidad analítica transforma la planilla de ser un simple gasto operativo, a ser una fuente de información para la planificación financiera. En resumen, la tecnología aplicada a la gestión de recursos humanos es el puente que conecta las obligaciones administrativas con la estrategia de crecimiento de la veterinaria.

Seguridad de Datos

En el desarrollo de sistemas modernos, y particularmente en aquellos que gestionan información financiera sensible como una nómina, la seguridad de los datos deja de ser una característica opcional para convertirse en un requisito indispensable del proyecto. Se entiende por seguridad de datos, la disciplina encargada de proteger la información digital contra accesos no autorizados, corrupción o robo durante todo su ciclo de vida. Siguiendo la teoría clásica expuesta por Stallings y Brown (2012), el diseño de este sistema se fundamenta en la CIA: confidencialidad, integridad y disponibilidad. Estos autores definen la confidencialidad como "la preservación de las restricciones autorizadas sobre el acceso y la divulgación de la información, incluidos los medios para proteger la privacidad personal y la información propietaria". (p. 12)

Para el caso específico de la Veterinaria Invet Diagnóstico, la confidencialidad representa la primera línea de defensa. Dado que el sistema alojará datos críticos, como salarios, números de cuenta bancaria y direcciones domiciliarias, es de suma importancia mitigar riesgos de fraude o robo de identidad. Por esta razón, se han definido protocolos de encriptación robustos, tanto para los datos en reposo (la información almacenada en la base de datos *MySQL*) como para los datos en tránsito.

Más allá de mantener el secreto, el sistema debe garantizar la integridad de la información, es decir, asegurar que los datos no sean alterados sin autorización, ya sea por error humano o por ataques externos. Según establece el *National Institute of Standards and Technology* (NIST, 2017), la integridad se define como “una propiedad que garantiza que los datos no han sido alterados de manera no autorizada desde su creación, transmisión o almacenamiento” (p. 3). Para este proyecto, esto implica que, si el salario de un empleado sufre una modificación, el sistema dejará una huella inalterable (*log*) que justifique dicho cambio, aportando transparencia administrativa.

El tercer pilar, la disponibilidad, asegura la continuidad operativa de la clínica. Un fallo en el servidor o un ataque de Denegación de Servicio (DoS) podría impedir que la administración realice los pagos a tiempo, generando contingencias legales. Para contrarrestar este riesgo, la arquitectura del sistema contempla estrategias de redundancia, como la ejecución de copias de seguridad (*backups*) automatizadas y periódicas. Siguiendo las recomendaciones del Instituto Nacional de Ciberseguridad (INCIBE, 2022) en sus guías para la gestión de riesgos, se entiende que el respaldo de datos no es un lujo técnico, sino una salvaguarda esencial para

garantizar la continuidad del negocio y minimizar el impacto operativo ante incidentes críticos, como el *ransomware*.

Finalmente, es crucial reconocer que la seguridad no es puramente técnica, tiene un componente humano determinante. La ingeniería social sigue siendo una de las vulnerabilidades más explotadas en la actualidad. Resultaría ineficaz blindar el código del *software* si las credenciales de los administradores son débiles o compartidas. Por ello, el diseño de seguridad del proyecto incluye políticas de gestión de usuarios estrictas, exigiendo el uso de contraseñas complejas y, en la medida de lo posible, autenticación de doble factor. De esta forma, la seguridad se integra no solo en el sistema, sino en la cultura organizacional de la veterinaria.

Ley de Protección al Trabajador

La *Ley de Protección al Trabajador*, N° 7983 (Asamblea Legislativa, 2000), aprobada en el año 2000, marcó un antes y un después en la legislación laboral y social de Costa Rica, reformando el sistema de pensiones y el auxilio de cesantía. Su objetivo principal fue fortalecer el régimen de Invalidez, Vejez y Muerte (IVM) y crear cuentas de ahorro individual para los trabajadores. Según el Sistema Costarricense de Información Jurídica (2024), esta ley "tiene como finalidad desarrollar, regular y fiscalizar el sistema de pensiones complementarias y el ahorro laboral, garantizando una administración transparente y segura de los recursos de los trabajadores" (Art. 1). Para el sistema de planillas, esta ley introduce cálculos obligatorios que van más allá del salario base.

Uno de los componentes más críticos que introduce esta ley es el Fondo de Capitalización Laboral (FCL). Este es un aporte que realiza el patrono equivalente al 3 % del salario mensual del trabajador. De este aporte, un 1,5 % se dirige al FCL y el otro 1,5 % al Régimen Obligatorio de Pensiones. La Superintendencia de Pensiones (Supén, 2023) aclara que el FCL es un ahorro que el empleado puede retirar cada cinco años de relación laboral continua o al momento de la ruptura laboral, independientemente de la causa. El sistema web debe calcular este porcentaje exacto mes a mes para asegurar que el aporte patronal sea correcto.

El segundo componente es el Régimen Obligatorio de Pensiones (ROP). Este es un sistema de capitalización individual. Esto significa que cada trabajador tiene una "cuenta" a su nombre donde se acumulan los aportes y sus rendimientos. El *software* de nómina debe estar parametrizado para identificar y segregar estos montos en los reportes que se envían a la CCSS

(quien actúa como recaudador centralizado a través del sistema Sicere). Cualquier error en este cálculo afecta directamente la futura pensión del colaborador.

La Ley N° 7983 también establece sanciones severas para los patronos que evaden estos pagos o los realizan por montos inferiores a los reales. La morosidad en el pago del FCL y ROP genera intereses y multas, además de impedir que la empresa esté al día con la CCSS, lo cual es requisito para múltiples trámites comerciales.

En conclusión, la *Ley de Protección al Trabajador N° 7983* (Asamblea Legislativa, 2000) añade una capa de complejidad financiera a la gestión de nóminas que hace casi imposible su manejo manual eficiente. El sistema web para la Veterinaria Invet Diagnóstico debe integrar las tablas de porcentajes actualizadas y permitir la generación de los archivos planos compatibles con las plataformas de recaudación nacional. Esto garantiza que la empresa cumpla su rol social y legal, protegiendo los ahorros a largo plazo de su personal.

Deducciones de Ley

Las deducciones de ley en Costa Rica son las retenciones obligatorias que todo patrono debe realizar sobre el salario bruto de sus empleados antes de depositar el salario neto. Según la Caja Costarricense de Seguro Social (CCSS, 2024), el modelo se basa en una contribución tripartita (Estado, patrono y trabajador), donde la cotización es obligatoria para todo trabajador asalariado y constituye la base del financiamiento de los servicios de salud y las pensiones del régimen básico.

El primer gran rubro de deducción es el Seguro de Enfermedad y Maternidad. Este seguro cubre la atención médica, cirugías, medicamentos y el pago de incapacidades. El *software* debe ser capaz de tomar el salario bruto, aplicar el porcentaje que corresponde a este rubro y restarlo. Además, debe considerar las excepciones, como en el caso de las incapacidades, donde la relación laboral se suspende y el cálculo cambia, un detalle que es normal que genere errores con cálculos manuales.

El segundo rubro es el Seguro de Invalidez, Vejez y Muerte (IVM), administrado también por la CCSS. Este aporte financia las pensiones por vejez y las ayudas en caso de invalidez o viudez. El aporte del trabajador es de aproximadamente un 4,17 %. Sumado al aporte del 1 % al Banco Popular, las cargas sociales totales del trabajador suman un 10,83 % para febrero del 2026 (dato sujeto a cambios periódicos por la Junta Directiva de la CCSS). El sistema desarrollado en *Python* debe mantener estos porcentajes como variables configurables y no como datos estáticos

(*hardcoded*), permitiendo al administrador actualizarlos si la ley cambia sin necesidad de reescribir el código

Además de las cargas sociales, existe la deducción por impuesto sobre la renta al salario. A diferencia de las cuotas de la CCSS que son fijas, este es un impuesto progresivo y escalonado que aplica solo a los salarios que superan un cierto umbral exento. El Ministerio de Hacienda publica anualmente los tramos fiscales. El sistema debe realizar un cálculo complejo: identificar el excedente sobre la base exenta y aplicar tarifas del 10 %, 15 %, 20 % o 25 %, según corresponda, a cada tramo.

Finalmente, las deducciones de ley también incluyen los embargos judiciales y pensiones alimenticias. Aunque estas no aplican a todos los empleados, el sistema debe estar preparado para procesarlas. La ley establece límites sobre cuánto se puede embargar de un salario (respetando el salario mínimo inembargable). El *software* debe automatizar esta validación lógica para evitar que la empresa deduzca más de lo permitido legalmente, protegiendo, así, el mínimo vital del trabajador.

Ley de Protección de la Persona Frente al Tratamiento de sus Datos Personales (Ley N° 8968)

La Ley N° 8968 (Asamblea Legislativa, 2011) es el marco jurídico costarricense que garantiza el derecho fundamental a la autodeterminación informativa. Esta ley regula cómo las empresas privadas y públicas deben recolectar, almacenar y utilizar la información personal de los ciudadanos. En el contexto del proyecto, la base de datos de la planilla contiene "Datos Personales de Acceso Irrestricto" (como el nombre y la cédula) y "Datos Personales de Acceso Restringido" (como el salario, la dirección y cuentas bancarias).

El principio de Consentimiento Informado es medular en esta ley. Establece que la veterinaria no puede recopilar datos de sus empleados sin informarles previamente para qué se usarán. Aunque la relación laboral implica tácitamente el uso de datos para el pago, el sistema debe ser transparente. Tecnológicamente, esto implica que el *software* debe tener avisos de privacidad y que los datos recolectados no pueden usarse para fines distintos a la gestión de nómina (por ejemplo, no se pueden vender a terceros para publicidad). El incumplimiento de esto se considera una falta gravísima sancionable con multas altas.

Otro aspecto crucial es la Seguridad de los Datos. La Ley No 8968 obliga a quien administra la base de datos, a adoptar las medidas técnicas necesarias para garantizar la integridad y confidencialidad de la información. El artículo 10 de la ley especifica el "Deber de

Seguridad", esto valida legalmente la necesidad de implementar la arquitectura de seguridad y roles de usuario definida en el marco técnico del proyecto.

La ley también consagra los derechos ARCO (Acceso, Rectificación, Cancelación y Oposición). Esto significa que cualquier empleado de la Veterinaria Invet Diagnóstico tiene el derecho de solicitar ver qué datos tiene el sistema sobre él y pedir que se corrijan si están erróneos. El sistema web debe facilitar este proceso, es decir, debe permitir al administrador actualizar rápidamente la información personal de un colaborador (como un cambio de domicilio o cuenta bancaria) para mantener la "Calidad de los Datos", otro principio rector de la normativa.

Finalmente, el manejo de datos sensibles, como el estado de salud (relacionado con incapacidades médicas), requiere un nivel de protección reforzado. La ley prohíbe el tratamiento de estos datos a menos que sea estrictamente necesario para el cumplimiento de derechos y obligaciones laborales. El sistema debe restringir la visualización de diagnósticos médicos o detalles de salud, limitándose a procesar los días de incapacidad necesarios para el cálculo matemático del pago. Al diseñar el sistema bajo estos lineamientos legales, el proyecto de graduación demuestra no solo competencia técnica, sino responsabilidad ética y jurídica.

CAPÍTULO III: MARCO METODOLÓGICO

Enfoques de Investigación

Para el desarrollo del sistema web de gestión de planillas en la Veterinaria Invet Diagnóstico, es de suma importancia establecer un marco metodológico sólido que guíe el proceso de indagación y construcción técnica del proyecto. Este constituye el eje central de cualquier trabajo de graduación, ya que define las herramientas y procedimientos sistemáticos que se utilizarán para alcanzar los objetivos de automatización propuestos anteriormente.

La selección de un enfoque adecuado no es una decisión que se toma a la ligera, sino que responde directamente a la naturaleza de la problemática identificada en la empresa. En este caso, la transición de procesos manuales basados en Excel hacia un sistema programado requiere una visión que abarque tanto la exactitud matemática de los cálculos salariales como la experiencia del personal que interactuará con la interfaz. Es así como lo mencionan Hernández Sampieri *et al* (2017), “si el diseño está concebido cuidadosamente, el producto final de un estudio (sus resultados) tendrá mayores posibilidades de producir conocimiento. Además, no es lo mismo seleccionar un diseño que otro: cada uno tiene sus características”. (p. 97)

Por lo tanto, en esta sección se describen y analizan las tres rutas metodológicas fundamentales: la cuantitativa, la cualitativa y la mixta. Cada una de ellas ofrece una perspectiva distinta que permite validar no solo la funcionalidad del código fuente en *Python*, sino también el impacto positivo que la herramienta tendrá en la gestión administrativa y el clima organizacional de la veterinaria. Además, se tomará uno de estos enfoques para la presente investigación y se explicará el por qué de esa elección.

Enfoque Cuantitativo

En este tipo de enfoque se toman datos numéricos para la investigación, aquí entran datos estadísticos, mediciones numéricas y todo aquello que tenga que ver con números y resultados de estos. Resulta ser un enfoque no solo muy importante, sino también muy utilizado, ya que permite obtener datos objetivos para ejecutar programas y algoritmos con mayor precisión.

El enfoque cuantitativo buscaría procesar variables financieras con menor margen de error, ya que, como se han mencionado, en la creación de esta base es de suma importancia controlar los montos a pagar, deducciones que se deben aplicar por ley (las cuales pueden variar)

y montos como extras y aumentos salariales. El tener una base cuantitativa como parte del desarrollo del sistema, permitirá que se cumplan los objetivos del proyecto.

Para definir la idea sobre el enfoque cuantitativo se tomará lo dicho por Hernández Flórez y Klimenko (2025), “La metodología de la investigación cuantitativa destaca por su enfoque en la medición y el análisis estadístico, definiendo metas explicativas que describen fenómenos mediante relaciones causales o correlacionales entre variables, construyendo generalizaciones” (p. 9). En este enfoque se suelen usar herramientas como encuestas con preguntas cerradas, pruebas y cuestionarios que ayudan a garantizar la veracidad de los datos obtenidos.

Enfoque Cualitativo

Por otro lado, se encuentra el enfoque cualitativo, el cual, contrario al cuantitativo, no se mide en números o estadísticas, sino que son más datos conceptuales. En este caso, igualmente se pueden hacer entrevistas, grupos focales y encuestas, la diferencia es que estas suelen ser abiertas para que el o los individuos puedan ahondar en el tema que se está investigando y así obtener información sobre sus vivencias.

El enfoque cualitativo es importante para la presente investigación, ya que permite conocer la problemática de una manera más real. Por medio de entrevistas se puede determinar cuáles son las aristas que se deben mejorar a la hora de implementar el sistema de planilla. Para Hernández Flórez y Klimenko (2025), una de las principales características de este enfoque es “la inmersión del investigador en el campo de estudio y en el caso de los participantes. El investigador se convierte en el instrumento principal en la recolección de datos: observa, entrevista, analiza y expone”. (p. 7)

Así, este método está más enfocado en la recolección de datos, no tanto numéricos, sino más bien descriptivos; por medio del enfoque cualitativo se logra entender un tema, desglosarlo y llegar a una respuesta de un problema específico.

En el entendido del presente proyecto, este enfoque es importante, ya que el sistema puede estar perfectamente construido a nivel técnico, pero debe estarlo también a nivel cultural de la empresa y alineado a las necesidades de las personas que lo utilizarán.

Enfoque Mixto

Por último, el enfoque mixto es una combinación de los enfoques anteriores para un solo estudio. Esta mezcla de enfoques permite al investigador un nivel de comprensión del problema de investigación y sus resultados a un nivel más amplio o profundo. En la creación del presente

proyecto, esta sinergia entre ambos enfoques permitirá no solo que se cree un sistema de cálculos matemáticos exactos y reales, si no, también, que quienes vayan a interactuar con este sistema lo hagan de manera organizada y pueda ser exitosa en el área de recursos humanos.

Para Medina Romero *et al.* (2023), este enfoque “busca aprovechar las fortalezas de ambos métodos para obtener una comprensión más completa y enriquecedora de los fenómenos estudiados” (p. 16). Es un método que le dará una respuesta más amplia al problema, con perspectivas distintas, tanto cuantificables como descriptivas.

Según Medina Romero *et al.* (2023), este método posee cuatro características que son clave para una buena aplicación, estas son: la integración de datos, la triangulación de datos, la complementariedad y el diseño de investigación.

Este método suele ser muy completo; sin embargo, no todas las investigaciones van a requerirlo. El uso de este enfoque en una investigación que no lo requiere puede brindar datos innecesarios, además de generar un desgaste en el investigador que no será valioso. El enfoque mixto busca recopilar datos e información de una manera integral, pero es, sin duda, un trabajo más amplio, que requiere de más tiempo por parte de quien lo implemente por su sola naturaleza.

Uno de los principales desafíos de los métodos mixtos es la complejidad del diseño de investigación. La integración de diferentes enfoques y la combinación de diferentes métodos de recolección y análisis de datos requiere una cuidadosa planificación y consideración. Los investigadores deben asegurarse de que los diferentes componentes del estudio estén bien conectados y que haya una lógica clara en la forma en que se integran los enfoques cuantitativos y cualitativos.

Los métodos mixtos pueden requerir más recursos y tiempo en comparación con los enfoques de investigación tradicionales. La recopilación y análisis de datos cuantitativos y cualitativos puede ser más intensiva en términos de recursos humanos, financieros y temporales. Los investigadores deben estar preparados para invertir los recursos necesarios y planificar cuidadosamente los tiempos y las etapas del estudio para garantizar una implementación exitosa. (Medina Romero *et al.*, 2023, p. 25)

De este modo, se puede entender que este enfoque es más completo que solo usar enfoque cualitativo o enfoque cuantitativo, pero el investigador debe contar con los recursos necesarios, tanto en tiempo, como económicos y de participación de la población para que sea exitoso. Así se asegura una investigación rigurosa y completa, pero sobre todo valiosa.

Enfoque de Investigación Seleccionado

Para la presente investigación se seleccionó el enfoque mixto, ya que permite una investigación más profunda de los temas a tratar en el proyecto. Además, por la naturaleza de este se requiere de datos tanto cualitativos como cuantitativos para el correcto desarrollo e implementación del *software*.

Desde la perspectiva cuantitativa, la selección se justifica por la exigencia crítica de precisión en los cálculos de planillas, incapacidades y liquidaciones. Al ser un sistema que maneja fondos económicos y obligaciones legales, la validación estadística de que cada deducción se ajusta a los porcentajes vigentes de la CCSS es un requisito no negociable para la viabilidad técnica del *software*. Esta rigurosidad numérica garantiza que los reportes financieros generados posean la veracidad y calidad de datos exigida por el marco jurídico nacional.

Desde el método cualitativo, el enfoque mixto permite aprovechar la retroalimentación continua que propone la metodología Scrum. Al realizar entregas parciales de los módulos de "Seguridad" y "Consultas", el investigador puede observar cualitativamente cómo interactúan los usuarios con el sistema, permitiendo realizar ajustes en tiempo real que mejoren la efectividad de la transformación digital de la veterinaria. Al priorizar la percepción y experiencia del usuario, el sistema deja de ser visto como una carga adicional para convertirse en una herramienta facilitadora que profesionaliza la gestión del talento humano en la clínica.

Tipos de Investigación

La clasificación de los tipos de investigación se determina fundamentalmente por el nivel de profundidad con el que se desea abordar el objeto de estudio y el objetivo que persigue el investigador. Existen cuatro tipos: la investigación exploratoria, descriptiva, correlacional y explicativa. Definir el tipo de investigación es un paso esencial en el diseño del marco metodológico, ya que establece la estrategia de recolección de datos y el alcance de las conclusiones que se podrá obtener al finalizar el proceso.

A continuación, se dará una explicación de cada una para posteriormente seleccionar el que se usará para el presente proyecto.

Investigación Exploratoria

La investigación exploratoria es aquella que se realiza cuando el “objetivo es examinar un tema o problema de investigación poco estudiado, del cual se tienen muchas dudas o no se ha

abordado antes” (Hernández Sampieri *et al.*, 2017, p. 75). Se caracteriza por ser mucho más flexible en su metodología en comparación con los estudios descriptivos o explicativos, ya que busca identificar conceptos o variables promisorias en lugar de probar hipótesis rígidas.

En resumen, la investigación exploratoria actúa como un primer acercamiento al terreno de estudio, permitiendo que el investigador determine qué aspectos del fenómeno son más relevantes. Así lo explican Hernández Sampieri *et al.* (2017):

Los estudios exploratorios sirven para familiarizarnos con problemas que no se conocen o de los cuales se sabe poco, obtener información sobre la posibilidad de llevar a cabo una investigación más completa respecto de un contexto particular, indagar nuevos problemas, identificar conceptos o variables promisorias, establecer prioridades para investigaciones futuras o proponer afirmaciones y postulados. (p. 75)

La investigación exploratoria funciona en ámbitos donde hay poca información o conocimiento. Para este proyecto, este tipo de investigación podría funcionar en el entendido que la empresa nunca ha tenido un sistema de pago de planillas. No obstante, más adelante se determinará cuál será la seleccionada.

Investigación Descriptiva

En este tipo de investigación se considera un fenómeno y sus componentes. Su objetivo es recolectar información sobre un tema en específico para determinar cómo es que este se manifiesta y sus variables. Para Hernández Sampieri *et al.* (2017), este tipo de investigación “busca especificar las propiedades, características y perfiles de personas, grupos, comunidades, procesos, objetos o cualquier otro fenómeno que se someta a un análisis” (p. 76). Esto quiere decir que se toman datos sobre las variables del tema para el que se está creando el proyecto.

Este nivel de investigación es fundamental en las ciencias y la ingeniería, ya que permite documentar requerimientos, inventarios y procesos con un alto grado de precisión. Al enfocarse en el “qué” y el “cómo”, la investigación descriptiva proporciona la materia prima necesaria para que otros investigadores puedan realizar asociaciones o buscar causas en etapas más avanzadas del conocimiento.

Para este tipo de exploración, el investigador debe ser capaz de delimitar cuáles serán los conceptos o variables a estudiar. Suele ser usada en investigaciones donde se requieren datos sobre una población, por ejemplo, un censo nacional donde el objetivo es medir aspectos de

educación, salud y vivienda; en este caso se consideran variables como los grupos poblacionales, si serán niños, adolescentes, adultos o adultos mayores.

Investigación Correlacional

La investigación correlacional tiene como finalidad conocer la relación o grado de asociación que existe entre dos o más conceptos, categorías o variables en una muestra o contexto particular. En estos estudios, se busca entender cómo se comporta una variable conociendo el comportamiento de otra variable relacionada, lo que permite, en ciertos casos, realizar predicciones sobre el fenómeno estudiado.

A diferencia de los estudios descriptivos, donde se miden variables de forma aislada, en la investigación correlacional el interés se centra en la asociación entre dichas variables. El investigador analiza si el aumento o disminución en una de las variables coincide con un cambio sistemático en la otra, cuantificando la fuerza de esa relación mediante análisis estadísticos o lógicos.

En ocasiones, solo se analiza la relación entre dos conceptos o variables, pero con frecuencia se ubican en el estudio vinculaciones entre tres, cuatro o más variables. Los estudios correlacionales, al evaluar el grado de asociación entre las variables, miden cada una (presuntamente relacionadas) y después cuantifican y analizan la vinculación.

(Hernández Sampieri *et al.*, 2017, p. 77)

La investigación correlacional permitirá, en gran medida, que se realicen asociaciones; sin embargo, esto no está estrechamente cerrado a que estas variables sean la causa una de la otra. Muchas veces se podrán obtener resultados donde las correlaciones podrán ser negativas y esto seguirá siendo de importancia para el estudio, ya que brindará otras perspectivas que si sólo fuesen positivas siempre.

Investigación Explicativa

Por último, la investigación explicativa es aquella cuyo interés principal es explicar por qué ocurre un fenómeno y en qué condiciones se manifiesta, o por qué se relacionan dos o más variables de una forma determinada. “Son más que la descripción de conceptos o fenómenos o el establecimiento de relaciones entre variables; más bien, están diseñados para determinar las causas de los eventos y fenómenos físicos o sociales”. (Hernández Sampieri *et al.*, 2017, p. 78)

Este nivel de investigación es considerado el más profundo y exigente, ya que requiere no solo describir y relacionar, sino encontrar el sentido de entendimiento sobre la raíz del problema

investigado. Los estudios explicativos buscan confirmar o descartar relaciones de causa y efecto, proporcionando una base lógica que permite comprender la estructura interna de los procesos.

En el ámbito tecnológico, la investigación explicativa permite sustentar la selección de métodos o herramientas específicas con base en el análisis de las causas que originan una necesidad técnica. Proporciona las pruebas necesarias para demostrar que los resultados obtenidos no son producto del azar, sino la consecuencia directa de una serie de factores técnicos, teóricos y legales debidamente articulados y analizados.

Tipo de Investigación Seleccionado

Para la selección del tipo de investigación se tomarán dos: la explicativa y la descriptiva. Estas serán seleccionadas debido a que permiten abordar la problemática de la empresa Invet Diagnóstico de manera integral: *software* y *hardware*, pero también se detallarán los instrumentos, características que fundamentan, por medio de la investigación explicativa, el por qué de las técnicas seleccionadas.

El componente descriptivo se aplica ante la necesidad natural del proyecto de describir, o especificar con la mayor exactitud, todos aquellos requerimientos específicos del sistema; por ejemplo, describir cada módulo y la funcionalidad dentro del proyecto. Por otro lado, el componente explicativo se implementa ante la necesidad de responder a la raíz del problema de la empresa. No es solo observar y describir los errores que esta presenta, es aplicar este conocimiento en el desarrollo del sistema.

Fuentes de Información

A lo largo de la historia, el ser humano se ha caracterizado por buscar información, tener conocimiento y razonar, estas características lo han diferenciado siempre de los animales. El tener información sobre un tema o conocer más sobre algo en específico ha sido vital para el desarrollo de la sociedad. Esta búsqueda de conocimiento ha hecho que el hombre recopile y guarde la información para que pueda ser difundida y consultada por otros.

En el presente, con el fácil acceso a Internet, las fuentes de información han variado mucho, ya no son solamente los libros de texto o archivos que se conseguían en bibliotecas, si no que se tiene acceso a una amplia gama de conocimientos.

Cruz García (2019) define las fuentes de conocimiento como:

Son instrumentos para el conocimiento, acceso y búsqueda de la información, su objetivo principal es el de buscar, fijar y difundir la fuente de información implícita en cualquier soporte físico, estas se pueden catalogar desde diferentes perspectivas; sin embargo, cada autor puede elaborar su propia clasificación dependiendo su grado de información. De acuerdo con el grado de información que proporcionan, las fuentes de información se dividen en primarias, secundarias y terciarias; esta división se utiliza generalmente en el ámbito académico. (p. 57)

Como señala la autora, las fuentes de información en el mundo académico han sido catalogadas en tres subgrupos: primarias, secundarias y terciarias; esta distinción se aplica porque es importante para los investigadores reconocer de dónde viene la información que está tomando, ya que muchas veces puede existir confusión sobre quién dijo qué. Para evitar esto, es se aplican estas tres categorías.

En resumen, las fuentes de información son las herramientas, como entrevistas, libros, periódicos, cuestionarios, páginas web, entre otros, que ayudan a las personas a compartir y difundir información. Es todo lo que permite a la sociedad estar informada, como desde la época antigua cuando se enviaban mensajes por medio de cartas, las cuales debían viajar miles de kilómetros para llegar a su destino, o como lo conocemos hoy, enviar un mensaje de texto por medio de una aplicación donde el destinatario estará recibiendo la información casi de manera inmediata.

A continuación, se explicarán las fuentes de información por su grado, desde la primaria hasta la terciaria.

Fuentes de Información Primaria

Las fuentes de información primaria son aquellas que brindan datos que no han sido analizados o interpretados por terceros. Suelen ser creadas a base de investigación, mientras se estudia un tema específico. Estas fuentes permiten al investigador obtener sus propias conclusiones, suelen ser más reveladoras, ya que poseen información de “primera mano” debido a su propia naturaleza. Estas fuentes son de suma importancia para trabajos académicos, ya que aportan nuevos conocimientos y revelan información adicional.

Las fuentes primarias son de suma importancia, pero también se deben trabajar o utilizar con cuidado, ya que sugieren al investigador información sobre un tema, pero este deberá

corroborar estas fuentes y contextualizarlas al trabajo que esté realizando para así obtener datos que no sean sesgados o sumamente objetivos de la fuente primaria.

Para Avello (2018), estas fuentes “son aquellas que contienen información nueva y original obtenida como resultado de la investigación científica, entre los que se encuentran: monografías, publicaciones seriadas, documentos oficiales de instituciones públicas, informes técnicos, patentes, normas, tesis doctorales, actas de congresos, entre otras” (párr.4).

Fuentes de Información Secundaria

Por otro lado, se encuentran las fuentes de información secundaria. Estas suelen ser muy comunes dentro de las investigaciones y no deben ser confundidas con las primarias, ya que suele suceder, esto por la complejidad de conocer de donde está partiendo la información. Las fuentes secundarias contienen información de las fuentes primarias, pero de manera organizada y sintetizada. A diferencia de las primarias, estas se crean posterior a lo que están describiendo y suelen ser elaboradas por personas que no necesariamente hayan sido partícipes del acontecimiento o situación que se está estudiando.

Las fuentes secundarias no buscan dar información inédita, sino más bien, recopilar, analizar o interpretar datos y fuentes ya existentes. Para Avello Martínez (2018), “son aquellas que contienen información organizada y elaborada, producto del análisis, síntesis y reorganización de las fuentes primarias, entre las que se encuentran: diccionarios, enciclopedias, antologías, directorios, anuarios, bibliografías, catálogos, obras de referencia, entre otras”. (párr. 5)

Estas fuentes no deben ser consideradas como menos valiosas porque se analicen otros puntos de vista de los que se pueden ver o discutir en una fuente primaria, ya que son de igual importancia en una investigación. Además, su estudio es muy valioso, ya que le permite al investigador contrastar diferentes conceptos. Sin estas fuentes solo se obtendría la información primaria y nunca se analizaría más allá de lo que ésta diga o plantee.

Fuentes de Información Terciaria

Por último, se encuentran las fuentes terciarias, las cuales funcionan como guías o tablas de contenido que organizan y dan forma a las fuentes primarias y secundarias para su fácil y rápida localización. Estas fuentes no ofrecen información nueva, ni un análisis profundo o interpretación crítica de los temas, su valor radica exclusivamente en su capacidad para agrupar el conocimiento existente de manera lógica y accesible. Son el punto de partida ideal para

cualquier proyecto de investigación, ya que proporcionan visiones generales amplias y dirigen al usuario hacia los materiales más detallados y específicos que realmente le servirán para desarrollar su estudio o análisis.

Para Cruz García (2019), “son guías visuales o físicas que contienen información sobre las fuentes secundarias. Facilitan el control y el acceso a repertorios de referencia” (p. 58). Esta autora las clasifica en “índices, directorios, guías, metabuscadores, artículos sobre encuestas, bibliografías” (p. 58). Estas fuentes siempre son usadas dentro de las investigaciones como materiales complementarios, ya que facilitan y brindan un acompañamiento al investigador; sin embargo, no suelen ser citadas como fuentes de información importante ya que no la poseen. No obstante, es de suma importancia que se conozcan y se pueda entender el valor que presentan para cualquier investigación.

Variables

Para continuar con el proceso de investigación, se debe definir el término “variable”. Para Hernández Sampieri *et al.* (2017), es “una propiedad que tiene una variación que puede medirse u observarse” (p. 105). Entonces, se entiende que las variables son todos aquellos atributos, características o propiedades que tenga un sujeto, objeto o fenómeno de estudio, que puedan ser medidos y analizados. No son elementos aislados del marco metodológico; son, más bien, el hilo conductor de la investigación.

La variable es de suma importancia dentro de la investigación, ya que es la guía que proviene desde los objetivos y el planteamiento del problema hasta el análisis de los datos estadísticos. Existen, además, distintos tipos de variables, las cuales deben ser aplicadas de manera correcta dentro de la investigación para que esta no carezca de lógica o sentido. “Las variables adquieren valor para la investigación científica cuando llegan a relacionarse con otras variables, es decir, si forman parte de una hipótesis o una teoría”. (Hernández Sampieri *et al.*, 2017, p. 105)

Al iniciar con el proceso de investigación, se hace la formulación del problema o bien, de la hipótesis. Es de suma importancia el papel de la variable dentro de este, ya que permitirá que el investigador cree el concepto, el valor operacional y el instrumento que se utilizará para su estudio. Definir la variable en estas tres áreas es indispensable para que quien busque información dentro de la presente investigación, pueda entender el significado de esta para el

investigador. Esto se debe a que su conceptualización podría variar de un autor a otro, es decir, para un autor, la variable “requerimientos de sistema web” puede significar algo completamente diferente que para otro.

En conclusión, las variables deben estar presentes dentro de una investigación para que esta pueda ser entendida y analizada desde diferentes puntos de vista. A partir de ellas se obtendrán datos que serán un gran aporte al investigador y que este podrá debatir y confrontar con otros estudios. “Sin definición de las variables no hay investigación. Las variables deben ser definidas de dos formas: conceptual y operacionalmente”. (Hernández Sampieri *et al.*, 2017, p. 119)

Variable Conceptual

Como se indicó, existen tres tipos de variables dentro del proceso de investigación, una de ellas es la variable conceptual, la cual es de suma importancia para definir cuál es el significado que tiene para el investigador.

La variable conceptual involucra conceptos que se toman para el estudio de la hipótesis. Al establecer claramente qué se entiende por cada variable desde la literatura, se construye la base indispensable para su posterior medición empírica. Esta delimitación teórica funciona como el marco de referencia fundamental, evitando ambigüedades en la interpretación de los términos a lo largo del documento.

Para Hernández Sampieri *et al.* (2017), estas “son necesarias pero insuficientes para definir las variables de la investigación, porque no nos vinculan directamente con ‘la realidad’ o con ‘el fenómeno, contexto, expresión, comunidad o situación’. Después de todo, continúan con su carácter de conceptos” (p. 119). Sin embargo, sin estas no se podría definir el concepto que el investigador quiera dar a entender en relación con lo que se está investigando. La variable conceptual funciona como guía, da una conceptualización teórica al lector para que pueda interpretar los términos de la misma manera que lo hace el investigador.

Sin embargo, como afirman Hernández Sampieri *et al.* (2017), esta variable no puede funcionar por sí sola, ya que no le proporciona al investigador las herramientas prácticas necesarias para la recolección de datos. Esta se limita a responder a la pregunta: ¿qué es el fenómeno que se está investigando? “Los científicos necesitan ir más allá, deben definir las variables que se utilizan en sus hipótesis, en forma tal que puedan ser comprobadas y

contextualizadas. Lo anterior es posible por medio de lo que se conoce como definiciones operacionales” (Hernández Sampieri *et al.*, 2017, p. 119)

Variable Operacional

La variable operacional consiste en tomar un concepto teórico y convertirlo en algo que sea posible observar, medir y contar. Es la parte del proyecto donde por fin se responde a la gran pregunta: ¿cómo voy a medir esto en la vida real? En pocas palabras, se trata de ir desarmando la idea principal en partes cada vez más pequeñas y prácticas para poder trabajarlas. “Casi siempre se dispone de varias definiciones operacionales (o formas de operacionalizar) de una variable”. (Hernández Sampieri *et al.*, 2017, p. 120)

Que una investigación salga bien depende muchísimo de esta etapa, porque de aquí emana la herramienta exacta que se usará para recolectar la información. Al armar el cuadro de variables, se decide si se aplicará encuestas, entrevistas, observación o si se buscará en bases de datos. Si este paso queda mal estructurado, los datos que se recolecten estarán sesgados y, al final, las conclusiones del trabajo no serán válidas. Este paso es indispensable porque es justo aquí donde se consiguen las herramientas reales para comprobar si la hipótesis inicial se cumple o no en la realidad.

Una correcta selección de las definiciones operacionales disponibles o la creación de la propia definición operacional se encuentran muy relacionadas con una adecuada revisión de la literatura. Cuando ésta ha sido cuidadosa, se tiene una gama más amplia de definiciones operacionales para elegir o más ideas para desarrollar una nueva. Asimismo, al contar con estas definiciones, el tránsito a la elección del o los instrumentos para recabar los datos es rápido, pues sólo debemos considerar que se adapten al diseño y a la muestra del estudio. (Hernández Sampieri *et al.*, 2017, p. 121)

Variable Instrumental

Por último, se tiene la variable instrumental, la cual define el instrumento que se utilizará para la recolección de la información. Es una herramienta de apoyo para comprobar una causa-efecto. Esta variable no debe afectar de manera directa el resultado o resultados del proyecto, ya que su función es trabajar por medio de la variable principal.

Esta variable no se debe confundir con la operacional, ya que se debe entender que la primera es la que menciona el método, mientras que la instrumental especifica la herramienta

que se aplicará en dicho método. Por ejemplo, la variable instrumental para la ejecución de pruebas del sistema web será *Visual Studio Code*.

La definición instrumental de las variables es también conocido como la tercera variable, de modo que es aquello en la que se aclara cómo se estudiará la variable que se acaba de definir, los medios o instrumentos para recoger la información. En mérito de ello es que deben definirse y elaborarse los instrumentos y medios con que se recolectará la información. Los instrumentos nacen de las variables y de los objetivos. Nunca deberá elaborarse un instrumento sin tener definida la variable o variables. (Moreno, 2018, párr. 1)

Tabla 5.

Cuadro de variables.

Objetivos Específicos	Variable	Variable Conceptual	Variable Operacional	Variable Instrumental
Analizar los requerimientos funcionales y no funcionales del sistema web.	Requerimientos de sistema web.	Este concepto lo define la ISO/IEC/IEEE 2018, como “Conjunto estructurado de requerimientos, funciones, rendimiento, restricciones de diseño y otros atributos del sistema, así como de sus entornos operativos e interfaces externas. (p. 5)	Entrevista. Cuestionario. Observación.	Encuesta. Guía de observación.
Construir el diseño de la base de datos y del sistema web mediante la herramienta <i>Workbench</i> .	Diseño de la base de datos. Diseño del sistema web.	Para IBM (2026), el diseño de base de datos es el “conjunto de entidades y sus características, o atributos, y las reglas o relaciones entre dichas entidades” (párr. 1). Para definir el diseño de un sistema web se utilizará el concepto brindado por Martínez <i>et al.</i> (2010), “La lógica de presentación que administra las interacciones	Diagrama entidad-relación.	<i>MySQL</i> <i>Workbench</i>

		entre el usuario y el <i>software</i> ”. (p. 178)		
Programar los diferentes módulos del sistema mediante la herramienta <i>Visual Studio Code</i> .	Desarrollo de módulos del sistema.	Para IBM (2026), “un módulo web representa una aplicación web. El módulo web se crea ensamblando <i>servlets</i> , archivos JSP y contenido estático como páginas HTML, en una sola unidad desplegable”. (párr. 3)	Documentos de diseño. Diagrama entidad-relación.	<i>Visual Studio Code</i> .
Probar los diferentes módulos de forma independiente y también de forma integral, garantizando los resultados correctos del sistema.	Prueba de sistema web.	Para Susnjara y Smalley (s. f.), “son el proceso de evaluar y verificar que un producto o aplicación de <i>software</i> funciona correctamente, de forma segura y eficiente, de acuerdo con sus requisitos específicos”. (párr. 1)	Ejecutar pruebas al sistema.	<i>Visual Studio Code</i> .

Fuente: Elaboración propia (2026).

Población

La población, dentro del ámbito de la metodología de la investigación, representa la totalidad de los sujetos, objetos o fenómenos que comparten características específicas y que serán el foco principal del estudio. Según explica Suárez (2021, citado por Mejía *et al.*, 2021), "en el lenguaje de la investigación, se le llama población a la cantidad de personas o conjunto de elementos totales que cuentan con las características de interés para las investigaciones" (p. 138). Esta definición resalta que la población conforma el máximo de los datos posibles que un investigador desearía analizar si los recursos monetarios y temporales fueran ilimitados.

Sin embargo, dentro de la investigación es de suma importancia que de delimite esta población, ya que es indispensable para hacer el proceso de análisis. Para lograr esta precisión, el investigador debe establecer criterios de elegibilidad rigurosos (como la ubicación geográfica, el rango de edad, las características socioeconómicas o el periodo de tiempo evaluado). De esta

forma, cualquier lector comprenderá exactamente a quiénes o a qué elementos aplican las conclusiones del proyecto, garantizando la solidez de la investigación. Así, se define a continuación la población seleccionada para el presente proyecto.

La población que se eligió para la investigación abarca a los dos funcionarios del área de Recursos Humanos, quienes son los encargados de gestionar la planilla, realizar los pagos y son ellos quienes más harán uso del sistema. Además, como población de estudio también estarán los siete funcionarios de Invet Diagnóstico de las diferentes áreas, ya que, al crear un sistema de marcas, estos deberán hacer uso de este, por lo cual los aportes que estos brinden serán de suma importancia para la realización de este módulo.

Muestra

Para la definición de este concepto hay que partir del entendido que muchos proyectos carecen de viabilidad si se tuviese que analizar al 100 % de la población, esto porque contemplan poblaciones muy grandes. Es por ello por lo que se crea la muestra, que es una porción de esa población que se estudia, principalmente cuando es muy amplia. “La muestra es, en esencia, un subgrupo de la población. Digamos que es un subconjunto de elementos que pertenecen a ese conjunto definido en sus características al que llamamos población”. (Hernández Sampieri *et al.*, 2017, p. 175)

Es importante señalar que, en este caso, al ser la población tan pequeña, de nueve empleados, se usará toda la población, por lo que no se hará por medio de muestra.

Instrumentos de Recolección de Datos

Los instrumentos de recolección de datos son todas aquellas herramientas que el investigador aplicará para obtener la información que requiere para la investigación. Hernández Sampieri *et al.* (2017) los define como “recurso que utiliza el investigador para registrar información o datos sobre las variables que tiene en mente”. (p. 199)

La validez y la confiabilidad son los dos pilares absolutos que deben sostener a cualquier instrumento, ya sea un cuestionario, una entrevista o la observación. Por esta razón, el diseño, calibración y validación de estas herramientas constituyen una de las fases de mayor cuidado para asegurar la legitimidad de las conclusiones. Según Hernández y Duana (2020), para que este proceso se lleve a cabo de una manera correcta, el investigador debe seguir una serie de pasos que se especifican a continuación:

Autorización: condición necesaria previa a la recolección de datos. Tiempo: es la estimación del tiempo que el investigador requiere para la realización de la investigación (recolección, tabulación y análisis de datos). Recursos: humanos, económicos y físicos. Capacitación: referente a los objetivos, selección de muestra, instrumentos y procedimientos para la recolección de datos. Supervisión y coordinación: manera en que se distribuirá el recurso humano para cubrir los componentes de información del estudio. (p. 52)

Observación

Como parte esencial para la creación del proyecto, el investigador debe estar empapado de todo lo que sucede dentro de la empresa con respecto al pago de planilla; por ello, la observación es uno de los instrumentos más útiles. “Este método de recolección de datos consiste en el registro sistemático, válido y confiable de comportamientos y situaciones observables, a través de un conjunto de categorías y subcategorías”. (Hernández Sampieri *et al.*, 2017, p. 252)

La observación no es solo ver lo que está ocurriendo, es un proceso mucho más estructurado en donde el investigador debe registrar comportamientos y actitudes, en este caso de las personas, para tener una visión real. En este proyecto es importante, ya que le brinda al investigador una realidad de cómo se maneja actualmente la empresa. Además, se puede triangular información, ya que muchas veces en entrevistas o cuestionarios, los colaboradores podrían olvidar información importante.

La observación es ideal que se aplique de manera estructurada, para que no se vaya a perder ningún dato en el proceso. Esto quiere decir que lo ideal es llevar una guía y hacer las anotaciones que corresponda para que la información recopilada no se vaya a perder. Este proceso de observación garantizará un sistema más ajustado a lo que requiere la empresa.

Entrevista

La entrevista es una técnica de recolección de datos que consiste en un diálogo estructurado o semiestructurado entre el investigador y los sujetos involucrados en el entorno de estudio. Su importancia principal radica en la capacidad de profundizar en las perspectivas, experiencias y conocimientos técnicos de las personas, permitiendo obtener información detallada que no siempre es observable a simple vista o cuantificable mediante un cuestionario.

Existen diferentes tipos de entrevistas y son usadas con distintos fines. Las entrevistas pueden ser: estructuradas, semiestructuradas y abiertas. Por su parte, las estructuradas llevan una

estructura específica y no suelen salir de esta. “El entrevistador realiza su labor siguiendo una guía de preguntas específicas y se sujeta exclusivamente a ésta (el instrumento prescribe qué cuestiones se preguntarán y en qué orden)”. (Hernández Sampieri *et al.*, 2017, p. 403)

En las semiestructuradas, hay un poco más de flexibilidad para hacer preguntas adicionales o repreguntas sobre algún tema que puede surgir. “Las entrevistas semiestructuradas se basan en una guía de asuntos o preguntas y el entrevistador tiene la libertad de introducir preguntas adicionales para precisar conceptos u obtener mayor información” (Hernández Sampieri *et al.*, 2017, p. 403). Estas permiten un poco más de apertura por parte de quien está siendo entrevistado, ya que se toma en consideración la información que va aportando para que surjan nuevas preguntas.

Por último, la entrevista abierta, esta es más una guía general de lo que se desea consultar al entrevistado. No son preguntas ya preestablecidas, pueden ir variando conforme la entrevista avance y el rumbo que esta pueda tomar. “Las entrevistas abiertas se fundamentan en una guía general de contenido y el entrevistador posee toda la flexibilidad para manejarla. Regularmente, en la investigación cualitativa, las primeras entrevistas son abiertas”. (Hernández Sampieri *et al.*, 2017, p. 403)

En resumen, la entrevista aporta un amplio conocimiento de los participantes, ya que permite conocer a fondo sus opiniones. Para el presente proyecto se aplicarán entrevistas a los colaboradores con el fin de obtener información sobre cómo se maneja actualmente la empresa y cuáles son las áreas de mejora más importantes a tomar en consideración con el sistema de planillas. Es por ello que se toma como último concepto lo dicho por Hernández Sampieri *et al.* (2017), “el propósito de las entrevistas es obtener respuestas en el lenguaje y perspectiva del entrevistado (‘en sus propias palabras’). El entrevistador debe escucharlo con atención e interesarse por el contenido y la narrativa de cada respuesta”. (p. 405)

Cuestionario

El cuestionario no debe ser confundido con la entrevista, ya que existen características propias de este que son de suma importancia en la realización de una investigación. Hernández Sampieri *et al.* (2017) lo definen como un “conjunto de preguntas respecto de una o más variables que se van a medir” (p. 217). Así, este puede ser aplicado a la población o la muestra en casos en que la población total sea muy grande. Es una manera de recolectar información, de manera masiva, mucho más sencilla que por medio de una entrevista.

Los cuestionarios se utilizan en encuestas de todo tipo (por ejemplo, para calificar el desempeño de un gobierno, conocer las necesidades de hábitat de futuros compradores de viviendas y evaluar la percepción ciudadana sobre ciertos problemas como la inseguridad). Pero también, se implementan en otros campos. Por ejemplo, un ingeniero en minas usó un cuestionario como herramienta para que expertos de diversas partes del mundo aportaran opiniones calificadas con el fin de resolver ciertas problemáticas de producción. (Hernández Sampieri *et al.*, 2017, p. 217)

En los cuestionarios se pueden aplicar preguntas cerradas y abiertas, las cerradas son aquellas a las que se responde con un sí o no, verdadero o falso o, bien, en escalas, como bueno, malo, ni bueno ni malo. Las preguntas abiertas permiten a la persona encuestada escribir opiniones, y estos cuestionarios suelen ser un poco más tardados, ya que el personal deberá tomar el tiempo de dar sus opiniones y pensar más detalladamente lo que quiere comunicar. De igual manera, los cuestionarios pueden poseer ambos tipos de preguntas, en unas donde la persona solo tenga una única opción y en otra, donde será necesario que aporte datos o información adicional.

CAPÍTULO IV: ANÁLISIS DE RESULTADOS

En el presente apartado se analizará la información obtenida por medio de las herramientas de recolección de datos, iniciando con la guía de observación, continuando con la guía de cuestionario y terminando con la guía de entrevista. A su vez, se mencionarán los aportes que dichas herramientas brindaron y qué requerimientos se pudieron determinar a raíz de ellas en el contexto de Invet Diagnóstico.

Primeramente, se desarrolló la guía de observación, la cual cuenta con cinco columnas con los siguientes títulos:

1. Aspectos por observar.
2. Cumple.
3. No cumple.
4. Oportunidad de mejora.
5. Detalle de observación.

La columna de aspectos por observar fue utilizada para detallar cada acción que realizaba la administración al llevar a cabo un proceso; luego, las columnas de cumple, no cumple y oportunidad de mejora, ayudaron a determinar los errores del sistema actual para así darse una idea de cómo se podría estandarizar el proceso.

Para demostrar los requerimientos y aportes que brindó esta herramienta, se hará énfasis en las diez actividades observadas:

1. Observar el método mediante el cual los empleados notifican su entrada y salida (vía *WhatsApp* o manual). Gracias a la observación de este proceso se logró analizar y determinar que la empresa no posee sistemas confiables para el registro de asistencia y dependen de dispositivos móviles personales. Como aporte, esto fundamenta la creación de un reloj marcador dentro del Módulo de Gestión de Horas Extras.
2. Observar dónde y cómo se consolidan los mensajes de *WhatsApp* para el cálculo final de horas. Se logró detectar que este proceso se consolida manualmente, lo cual demanda mucho tiempo y abre una puerta para que se den malas prácticas o errores en la digitación. El aporte que se brinda es la necesidad de una base de datos centralizada que calcule automáticamente los tiempos.

3. Observar el procedimiento manual para diferenciar y calcular jornadas extraordinarias según la legislación. A partir de esta observación se evidenció que los cálculos manuales no tienen en cuenta todas las variables de ley, generando pagos inconsistentes. El aporte es la implementación de algoritmos precisos en el sistema que separen automáticamente las jornadas ordinarias de las extraordinarias.
4. Observar si la información de los empleados (contratos, datos personales) está centralizada o dispersa. Se demostró que la información se encuentra desorganizada, lo cual dificulta la gestión de recursos humanos. Gracias a este análisis, se logró crear el aporte del Módulo de Mantenimiento para consolidar todos los expedientes de manera digital.
5. Observar cómo se archivan y se reflejan las boletas de la CCSS en el control de pagos. Se logró observar que el manejo físico o desordenado de estas boletas complica los rebajos salariales por incapacidad. Esto aporta la justificación para el Módulo de Gestión de Incapacidades, permitiendo cargar los certificados médicos directamente al sistema.
6. Observar el formato y el medio de entrega de los desgloses de salario a los colaboradores. Se constató la ausencia de un medio formal, rápido y transparente para la entrega de recibos. El aporte consiste en agregar la funcionalidad de notificaciones y generación de comprobantes electrónicos en el Módulo de Pago de Planilla.
7. Observar el método de seguimiento de saldos de vacaciones y la acumulación mensual de pasivos laborales. Esta observación evidenció la falta de seguimiento sistemático, lo cual dificulta la gestión de indemnizaciones. Como aporte, se establece el Módulo de Gestionar Vacaciones para rastrear automáticamente la acumulación de días.
8. Observar si existen formularios físicos o digitales para medir los KPI's de los empleados. Se observó que no existe una herramienta para medir el desempeño, aspecto que dificulta la asignación de bonificaciones. El aporte directo es la creación del Módulo para Evaluar Empleado, el cual incluye herramientas para establecer metas.
9. Observar cómo se transfiere la información de planilla desde las sedes de Escazú y Cartago hacia la administración central. Se detectó que el manejo entre sedes es deficiente debido a la falta de un sistema en red. El aporte de esta observación es la justificación de una arquitectura web (Modelo-Vista-Controlador) que conecte a todas las sucursales simultáneamente.

10. Observar si el acceso a las hojas de cálculo de salarios está restringido o si es accesible a personal no autorizado. Al analizar este punto, se notó la vulnerabilidad de los datos sensibles en hojas de Excel. Esto deja como aporte un Módulo de Seguridad encargado de realizar la autenticación de contraseñas y definición de perfiles.

Por lo explicado, se estima que se puede continuar con el análisis y los resultados que brindó la herramienta de recolección de datos, llamada cuestionario. Este instrumento se conformó a partir de diez preguntas aplicadas a los colaboradores de la clínica para medir su percepción. Las preguntas y sus respectivos aportes se detallan seguidamente:

1. ¿Con qué frecuencia ha tenido que solicitar una corrección en su comprobante de pago debido a cálculos incorrectos de horas extra, feriados o deducciones?

Para esta pregunta se logra comprobar que hay un serio problema en el análisis y cálculos de horas extras y deducciones, ya que la mayoría los colaboradores expresa haber tenido, al menos una vez, errores en sus pagos. Para la empresa, además, es de suma importancia que esto no suceda, ya que se arriesga a enfrentar demandas de los mismos colaboradores por malos pagos o rebajos que no corresponden.

Es por ello que, para solucionar este error, se crea un módulo “pago de planillas”, el cual, por sí mismo, hace el cálculo para cada colaborador con base siempre en la ley vigente y los rebajos correspondientes que se deben aplicar. Al tener estas fórmulas automatizadas, se busca reducir el margen de error a cero.

2. ¿Entiende claramente cómo se calculan sus rebajos de ley (CCSS y Banco Popular) y el impuesto sobre la renta en su pago actual?

Al analizar las respuestas brindadas a esta interrogante, se hizo evidente que existe desconocimiento por parte de los colaboradores sobre cómo se aplican las deducciones obligatorias en su salario. La gran mayoría del personal expresó no tener claridad sobre los montos exactos que se les rebajan por concepto de la Caja Costarricense de Seguro Social (CCSS), el aporte al Banco Popular y el respectivo impuesto sobre la renta. Esta falta de comprensión genera confusión y dudas constantes al momento de recibir la remuneración, ya que los empleados no logran conciliar su salario bruto con el monto neto depositado.

Como aporte directo al desarrollo del proyecto, esta información justifica la necesidad de crear un sistema web que genere comprobantes de pago detallados. El nuevo *software* debe desglosar, de manera automática y comprensible, cada uno de los rubros descontados, mostrando

de forma amigable los porcentajes aplicados según la legislación vigente para la salud, pensiones y cargas tributarias.

3. ¿Qué tan difícil le resulta obtener una constancia de salario o revisar sus pagos de meses anteriores con el método administrativo actual?

Al evaluar las respuestas correspondientes a esta pregunta, la gran mayoría de los colaboradores coincidió en que el proceso actual resulta difícil y lento. Esta dificultad radica en que la solicitud de este tipo de documentación no se encuentra centralizada ni estandarizada, dependiendo por completo de canales informales para procesar la petición de forma manual. Esta dependencia administrativa crea un “cuello de botella” operativo que retrasa significativamente la entrega de constancias salariales, un documento que muchas veces los empleados requieren con urgencia para diversos trámites bancarios o personales.

Esta deficiencia justifica la creación e implementación de un apartado de solicitudes formales dentro del sistema web. A través de esta plataforma, el colaborador podrá generar una petición directa, organizada y trazable para su constancia salarial. De esta manera, el área administrativa recibirá la notificación en su panel de control, procesará el documento con los datos del sistema y, posteriormente, se encargará de enviarlo por correo electrónico o entregarlo de forma física al empleado. Este flujo no solo mejora el canal de comunicación interna, sino que permite llevar un seguimiento ordenado de las peticiones sin comprometer el control documental exclusivo de la administración.

4. ¿Considera que el registro de asistencia mediante *WhatsApp* es un método seguro y confiable para garantizar que sus horas laboradas se contabilicen con exactitud?

Los resultados de esta consulta reflejan que los empleados experimentan una profunda desconfianza hacia el uso de *WhatsApp* para el control de asistencia, percibiéndolo como un canal informal que facilita la pérdida de mensajes y genera errores al momento de trasladar los datos a la administración. Dicha vulnerabilidad metodológica compromete la exactitud en el cálculo de las jornadas y crea un ambiente de incertidumbre, pues el personal carece de un comprobante oficial que respalde su tiempo laborado.

En respuesta a esta problemática, el aporte central para el proyecto es la integración de un reloj marcador digital dentro de la plataforma web, diseñado para registrar la hora exacta del servidor e imposibilitar cualquier tipo de alteración. Mediante esta mejora tecnológica, se elimina por completo la dependencia de aplicaciones de mensajería externas y se centraliza la

información, permitiendo calcular las horas extras y gestionar las marcas de forma automatizada, segura y totalmente auditable

5. ¿Conoce usted con exactitud, cuántos días de vacaciones tiene disponibles en este momento sin tener que consultarlo directamente con administración?

Los hallazgos derivados de esta interrogante revelan que el personal carece de un medio autónomo para verificar sus saldos vacacionales, obligando a los trabajadores a depender exclusivamente de la administración para conocer este dato. Esta dinámica genera consultas constantes que interrumpen el flujo de trabajo administrativo y limita la capacidad del colaborador para planificar sus periodos de descanso con certeza.

Como solución a esta deficiencia, el aporte directo al proyecto consiste en integrar un panel informativo dentro del perfil de cada usuario, el cual proyectará en tiempo real el saldo exacto de sus días disponibles. Además de la visualización, el sistema permitirá que el colaborador solicite los días que desee disfrutar directamente desde la plataforma, automatizando la petición para la respectiva aprobación de su jefatura. Esta implementación tecnológica fomenta la autogestión y la transparencia para el empleado, a la vez que descongestiona significativamente las labores operativas de la gerencia al estandarizar y digitalizar todo el proceso.

6. ¿Le gustaría recibir su comprobante de pago de forma automática por correo electrónico o mediante una plataforma web apenas se procese la planilla?

La totalidad de las respuestas a esta pregunta evidenció una aceptación hacia la propuesta de digitalizar y agilizar la entrega de los desgloses salariales. Los colaboradores manifestaron un gran interés por recibir la información de su remuneración de manera inmediata.

Para satisfacer esta necesidad, el aporte estructurado para el proyecto radica en la integración de una funcionalidad de envío masivo automático dentro del motor de planillas. De esta forma, el sistema generará los comprobantes en un formato electrónico estandarizado y los enviará instantáneamente al correo de cada empleado. Esta implementación garantiza un flujo de comunicación rápido y confidencial, modernizando por completo la experiencia del colaborador dentro de la clínica.

7. Al solicitar un permiso o reportar una incapacidad, ¿cuánto tiempo tarda usualmente en recibir una confirmación oficial de que el movimiento fue aplicado correctamente?

Las respuestas ponen en evidencia una deficiencia significativa en los tiempos de respuesta y seguimiento ante trámites administrativos de gran importancia para el personal. Los colaboradores manifiestan experimentar lapsos prolongados de espera, o la ausencia total de confirmaciones oficiales tras presentar solicitudes de ausencia o reportar incapacidades, lo cual genera un alto nivel de incertidumbre sobre si sus justificaciones fueron procesadas correctamente por la gerencia. Para solventar esta preocupante brecha de comunicación, el aporte esencial al proyecto consiste en el desarrollo estructurado de un Módulo de Permisos e Incapacidades dentro del aplicativo web. A través de esta herramienta, en el instante en que el empleado registre una solicitud, el sistema notificará de forma inmediata a la jefatura encargada para su evaluación y, una vez autorizada, emitirá una alerta automática al perfil y correo del trabajador confirmando que el movimiento fue aplicado exitosamente, garantizando así un mejor flujo de información.

8. ¿Se le comunica formalmente el resultado de sus evaluaciones de desempeño y cómo estas afectan sus posibles bonificaciones o crecimiento en la clínica?

Los resultados de esta interrogante evidenciaron una carencia significativa en la comunicación formal dentro de la clínica, revelando que las evaluaciones de desempeño son prácticamente nulas o se transmiten de forma deficiente. Esta falta de retroalimentación estructurada genera un vacío de información, impidiendo que los colaboradores comprendan cómo su rendimiento diario impacta directamente en sus oportunidades de crecimiento profesional, ascensos o la posible asignación de bonificaciones económicas basadas en méritos.

Para mitigar esta problemática, el desarrollo del proyecto aportará un Módulo de Evaluación de Desempeño completamente estructurado, diseñado para estandarizar este proceso administrativo. Esta herramienta permitirá a las jefaturas registrar métricas precisas mediante matrices de calificación y comentarios cualitativos, consolidando la información en el expediente digital de cada empleado. Con esto, se garantizará la generación de informes de rendimiento transparentes, facilitando la toma de decisiones gerenciales fundamentadas y mejorando el clima organizacional.

9. ¿Siente incertidumbre sobre si el monto de su aguinaldo a fin de año será calculado correctamente bajo el sistema manual actual?

El análisis de las respuestas demostró un nivel alarmante de incertidumbre y desconfianza por parte del personal respecto a la precisión del cálculo de su aguinaldo. La dependencia actual

de un sistema basado en cálculos manuales y hojas de cálculo incrementa la percepción de vulnerabilidad ante errores humanos, lo que se traduce en ansiedad generalizada entre los trabajadores durante el cierre del período, al no tener seguridad sobre la exactitud de los montos que percibirán por derecho.

Como solución técnica y aporte directo de valor al proyecto, se establece la exigencia de integrar un módulo automatizado que garantice exactitud financiera. Este componente del sistema se encargará de recopilar el historial de los salarios netos y brutos devengados a lo largo del año, aplicando las normativas laborales vigentes de manera estricta y sin intervención manual. Además, generará reportes y colillas desglosadas que mostrarán, de forma clara, la trazabilidad de la matemática aplicada, devolviendo la tranquilidad al colaborador.

10. Del 1 al 5, ¿qué tan útil considera que sería tener un perfil de usuario donde pueda ver su historial de marcas, solicitar vacaciones y ver sus evaluaciones de desempeño?

Las métricas obtenidas en esta escala reflejaron una calificación altamente positiva, confirmando una excelente disposición y viabilidad tecnológica por parte del personal hacia un portal de autogestión. Los colaboradores manifestaron una clara necesidad de contar con mayor autonomía operativa para consultar su información laboral en tiempo real, interactuando con módulos, como el reloj marcador, los saldos de vacaciones disponibles y sus notas de desempeño, sin tener que depender de consultas constantes al área de Recursos Humanos.

A partir de esta validación de los usuarios finales, el aporte sustancial radica en el diseño de la capa de presentación del sistema. Utilizando el patrón de arquitectura MVC (Modelo-Vista-Controlador), se desarrollará interfaces gráficas enfocadas en la experiencia de usuario (UX). Esto resultará en un panel principal intuitivo que empoderará al colaborador para tramitar y gestionar sus propias solicitudes de manera fluida, reduciendo al mismo tiempo la carga operativa y los “cuellos de botella” administrativos de la clínica.

En esta última parte de la propuesta se ahondará específicamente de los resultados obtenidos por medio de la guía de entrevista realizada al personal administrativo y de recursos Humanos, la cual cuenta con 20 preguntas elaboradas para indagar con mayor profundidad en los procesos gerenciales de la empresa. Las preguntas son:

1. Considerando la sensibilidad de los datos salariales, ¿qué eventos específicos del sistema considera críticos para ser auditados?

En esta pregunta se logró determinar la necesidad de registrar quién, cuándo y por qué se modifica un salario. Como aporte, se establece la creación de tablas de *logs* y bitácoras de auditoría en la base de datos.

2. ¿Qué nivel de pérdida económica estima que ha sufrido la empresa debido a errores en pagos excesivos o cálculos incorrectos de aguinaldos y liquidaciones?

Se evidenció que los errores manuales han causado pérdidas tangibles y conflictos. El aporte al sistema es la integración de verificaciones de errores lógicos antes de proceder con la aprobación de cualquier pago masivo.

3. ¿Qué niveles de acceso específicos requiere? ¿Quiénes, además de usted, deben poder modificar salarios o eliminar registros de marcas?

Gracias a esta pregunta se logró determinar que se requiere extrema confidencialidad. El aporte brindado es la estricta definición de perfiles en el Módulo de Seguridad, limitando los permisos de escritura solo a la gerencia y Recursos Humanos.

4. Dado que es una veterinaria, ¿existen bonos por procedimientos específicos o ventas de productos? ¿Cómo es la fórmula exacta de cálculo para integrar este algoritmo en el módulo de planilla?

Se constató la existencia de bonificaciones variables sujetas al desempeño médico. El aporte radica en la parametrización de ingresos adicionales dentro de la lógica del cálculo salarial.

5. ¿Cuáles herramientas, según su experiencia, hacen falta para el control de marcas y asistencia que sustituyan el uso de *WhatsApp*?

El común denominador fue un sistema digital e inalterable. Se toma como aporte principal la creación de un registro automatizado dentro del Módulo de Horas Extras.

6. ¿Cuántos años de historial de pagos considera crítico que el sistema mantenga disponibles para consultas rápidas antes de ser archivados?

Se determinó la necesidad de mantener un amplio historial para fines legales y de auditoría. El aporte es la optimización de la base de datos *MySQL* para almacenar registros de manera eficiente a largo plazo.

7. ¿Cuáles herramientas, según su experiencia, hacen falta en el proceso de registro y control de expedientes de los colaboradores?

Los encargados manifestaron necesitar una ficha digital única. Gracias a estas respuestas se logró dar como aporte el Módulo de Mantenimiento para la inserción, modificación y actualización de datos personales.

8. ¿Qué tipos de empleados tiene la veterinaria (por ej.: médicos veterinarios, asistentes, recepcionistas, personal de limpieza)? ¿Tienen regímenes distintos de pago?

Se determinó una alta variedad de puestos operativos y administrativos con salarios diferentes. El aporte que brinda esta pregunta es la estructura de la base de datos donde se asignan cargos, salarios base y regímenes a cada colaborador.

9. ¿Cuáles son los niveles de acceso o tipos de usuarios que interactuarán con el sistema (por ej.: Administrador, Dueño, RRHH, Empleado estándar)?

Al analizar las respuestas, se logró establecer una jerarquía clara. Esto deja como aporte la creación de los roles específicos en el sistema para garantizar la seguridad de los datos sensibles.

10. ¿Qué tablas y qué datos específicos desea que se observen en el apartado de consultas de personal (por ej.: fecha de ingreso, historial de puestos, salarios anteriores)?

Se acordó que era vital visualizar el histórico salarial y fechas de ingreso para el cálculo de liquidaciones. Este aporte se materializa en las vistas de la interfaz del Módulo de Consultas.

11. ¿Se tendrá una forma diferente de cálculo de incentivos o comisiones según el tipo de puesto del empleado?

La mayoría de las respuestas indican que los médicos tienen estructuras distintas a las de los asistentes. Esta pregunta deja como aporte la flexibilidad que debe tener el código en *Python* para procesar diferentes fórmulas de incentivos.

12. ¿Cuáles son las jornadas laborales activas en la empresa (por ej.: diurna, mixta, nocturna, por turnos rotativos)?

Se confirmó la existencia de múltiples jornadas. Este aporte permite programar el controlador para que identifique los topes de horas ordinarias según el tipo de jornada antes de calcular horas extras.

13. ¿Cuáles son los tipos de pago que se manejan (por ej.: transferencia bancaria Sinpe, efectivo, cheque)?

Se evidenció el uso predominante de transferencias bancarias. El aporte es asegurar que los reportes de salida posean la información necesaria (cuentas IBAN) para agilizar las transferencias Sinpe.

14. ¿Cuáles son los plazos o fechas de corte en los que se debe realizar el cálculo y provisión del aguinaldo?

Los administradores confirmaron las fechas dictadas por la ley nacional (diciembre a noviembre). Esto aporta la lógica de fechas necesaria para programar el Módulo de Gestionar Aguinaldo.

15. ¿Cuáles funciones desea que se ingresen en el apartado de mantenimientos para la gestión de vacaciones (por ej.: cargar saldos iniciales, ajuste manual de días)?

Se solicitó la capacidad de ajustar días manualmente en caso de acuerdos internos. El aporte es la creación de permisos especiales en el Módulo de Mantenimiento para la sobreescritura justificada de saldos vacacionales.

16. ¿Qué métricas o KPI's desea que se observen en las consultas de rendimiento?

Se acordaron métricas como puntualidad, asistencia y cumplimiento de metas. Esto deja como aporte la generación de reportes específicos dentro del Módulo de Evaluar Empleado.

17. ¿Qué tipo de impuestos y cargas patronales específicas se deben reflejar en el resumen de costos de la empresa?

Se enfatizó la necesidad de separar la CCSS, ROP, FCL y el impuesto sobre la renta. Como aporte, se diseñarán tablas de configuración para mantener las deducciones de ley parametrizadas y actualizables.

18. En el flujo de trabajo, ¿cómo se validan las boletas de la CCSS? Resulta necesario definir si el sistema debe permitir la carga de archivos PDF o fotos de comprobantes para el Módulo de Gestión.

Se concluyó que la carga de archivos digitales optimizaría el proceso. Este aporte permite habilitar repositorios de archivos dentro del Módulo de Gestionar Incapacidades.

19. Describa el flujo actual desde que un empleado marca su entrada hasta que se emite su comprobante de pago. ¿Cuánto tiempo dedica semanalmente a estas tareas manuales en Excel?

Se determinó que se invierten numerosas horas semanales en un proceso manual tedioso y fragmentado. El aporte global de esta respuesta es la justificación absoluta de la viabilidad operativa y técnica del proyecto de automatización.

20. Según los módulos planteados (liquidaciones, aguinaldo, evaluación), ¿cuál considera que es el más urgente para reducir la carga operativa inmediata?

Al analizar las urgencias operativas, se concluyó que el control de asistencia y el pago regular de nómina son prioritarios. Como aporte, esta información define el orden del *Product Backlog* en la metodología Scrum, priorizando estos módulos para los primeros *sprints* de desarrollo.

El análisis de los datos obtenidos mediante la guía de observación, el cuestionario aplicado a los colaboradores y las entrevistas a la administración de Invet Diagnóstico, evidencia una dependencia crítica de procesos manuales y canales informales para la gestión del talento humano.

Estas deficiencias operativas no solo generan vulnerabilidades en la confidencialidad de la información y constantes errores en los cálculos salariales, sino que impactan negativamente la percepción del personal debido a la falta de transparencia y la lentitud en los trámites administrativos. Por consiguiente, se justifica, de manera absoluta, la viabilidad técnica y operativa para la automatización mediante el prototipo propuesto. La integración de módulos centralizados bajo una arquitectura web Modelo-Vista-Controlador, tales como el reloj marcador inalterable, el motor de planillas apegado a la legislación, el portal de autogestión y el control estricto de roles, permitirán erradicar los “cuellos de botella” actuales.

CAPÍTULO V: PROPUESTA

En este apartado se desglosa el diseño analítico y técnico de la propuesta de solución para la Veterinaria Invet Diagnóstico. La conceptualización de este sistema busca transformar la gestión operativa del Departamento de Recursos Humanos mediante la implementación de una arquitectura robusta, escalable y segura.

A continuación, se detallan los componentes lógicos, físicos y de red necesarios para el funcionamiento óptimo del prototipo.

Análisis Detallado del *Software* por Desarrollar

El sistema web propuesto se fundamenta en una arquitectura Modelo-Vista-Controlador (MVC), la cual permite segmentar la lógica de negocio, la base de datos y la interfaz gráfica. Para resolver las problemáticas identificadas en la empresa, el *software* se conforma a partir de los siguientes módulos integrales:

- **Módulo de Seguridad y Autenticación:** encargado de gestionar el acceso al sistema mediante el cruce de credenciales (correo y contraseña encriptada). Implementa una Matriz Dinámica de Permisos (RBAC) que restringe la visualización de los módulos según el rol del usuario.
- **Módulo de Mantenimiento de Expedientes:** permite centralizar la información de los colaboradores facilitando operaciones de inserción, actualización y cambio de estado (activo/inactivo) de las fichas técnicas y salarios base del personal médico y administrativo.
- **Módulo de Control de Asistencia y Horas Extras:** diseñado para sustituir los registros informales. Proporciona un reloj marcador digital y una interfaz para la solicitud de jornadas extraordinarias, las cuales deben ser aprobadas por la jefatura inmediata antes de sumarse a la nómina.
- **Módulo de Gestión de Ausencias e Incapacidades:** automatiza el flujo de solicitudes de vacaciones y permisos. Incluye un algoritmo que verifica el saldo disponible de vacaciones del empleado, omitiendo los fines de semana en el cálculo, y permite al Departamento de Recursos Humanos registrar boletas médicas oficiales de la CCSS.

- **Módulo Motor de Pago de Planillas:** constituye el núcleo del sistema. Se encarga de procesar los salarios base proporcionales, sumar horas extras y deducir automáticamente las cargas sociales obrero-patronales, el impuesto sobre la renta y las ausencias no justificadas (llegadas tardías), generando recibos electrónicos detallados.
- **Módulo de Liquidaciones y Aguinaldo:** facilita el cálculo matemático exigido por la legislación laboral costarricense. Automatiza la ponderación de preaviso, cesantía, aguinaldo proporcional y vacaciones no gozadas al momento de finalizar una relación laboral.
- **Módulo de Indicadores Gerenciales:** proporciona gráficas y reportes visuales en tiempo real sobre el ausentismo, el gasto histórico de planillas y la cantidad de empleados activos, apoyando la toma de decisiones estratégicas de la clínica.

Análisis Detallado del *Hardware* Requerido

Para asegurar la viabilidad del proyecto, se ha realizado una evaluación del *hardware* separando las necesidades de la fase de programación (prototipo) de los requerimientos para la puesta en producción del sistema a nivel local.

Para la fase de desarrollo y despliegue del prototipo, se utilizará la infraestructura con la que ya cuenta la empresa, la cual incluye monitores de 22" FHD, gabinetes de pared Lantek y periféricos estándar. Sin embargo, para poner en producción el sistema de bases de datos y asegurar un rendimiento óptimo al procesar cálculos matemáticos y reportes históricos, se determinó la necesidad de adquirir equipo de procesamiento y almacenamiento especializado.

Especificaciones Técnicas y Costos de Adquisición

- **Procesador:** AMD Ryzen 7 5700G, necesario para el procesamiento ágil del motor de *Python*. Costo: ₡102.000,00.
- **Memoria RAM:** TeamGroup Elite 32GB DDR4 3200Mhz – SO-DIMM. La tecnología es vital para servidores de bases de datos, ya que previene la corrupción de datos financieros. Costo: ₡149.000,00.
- **Almacenamiento:** Adata Mars 980 Blade - PCIe Gen5 x4 M.2 2280 SSD - 2 TB, el cual garantiza espacio suficiente para la retención a largo plazo de las bitácoras, recibos en PDF y expedientes de la clínica. Costo: ₡215.000,00.

- **Fuente de Poder:** Antec G850 - 850W - 80Plus Gold, para asegurar la estabilidad energética del servidor. Costo: ₡27.500,00.

La inversión total requerida en infraestructura de *hardware* asciende a ₡493.500,00.

Figura 5.
Proforma de Extreme Tech

The screenshot shows the Extreme Tech website interface. At the top, there's a navigation bar with the logo, a search bar, and links for 'ACCESO / REGISTRO', a heart icon, and a shopping bag icon showing a total of ₡493,500 (impuesto incluido). Below the navigation bar is a menu with 'VER CATEGORÍAS', 'PERSONALIZA TU PC', 'ARMA TU PC', 'NUESTRAS TIENDAS', 'METODOS DE PAGO', 'INFORMACIÓN', and 'DESCUENTOS'. The main content area displays a shopping cart with four items:

PRODUCTO	PRECIO	CANTIDAD	SUBTOTAL
AMD Ryzen 7 5700G	₡102,000	1	₡102,000 (impuesto incluido)
TeamGroup Elite 32GB DDR4 3200Mhz - SO-DIMM	₡149,000	1	₡149,000 (impuesto incluido)
Adata Mars 980 Blade - PCIe Gen5 X4 M.2 2280 SSD - 2 TB	₡215,000	1	₡215,000 (impuesto incluido)
Antec G850 - 850W - 80Plus Gold	₡27,500	1	₡27,500 (impuesto incluido)

On the right side of the cart, there's a summary box with a green checkmark and the text '¡ENVÍO GRATIS ACTIVADO!'. Below this, the 'Subtotal' is shown as ₡493,500, which includes ₡56,776 of taxes. A red button labeled 'FINALIZAR COMPRA' is at the bottom of the summary box. At the bottom of the cart area, there's a coupon code field and an 'APLICAR CUPÓN' button.

Análisis Detallado de los Elementos Relacionados con las Telecomunicaciones

Dado que la empresa Invet Diagnóstico opera con sucursales distribuidas en San Pedro, Escazú y Cartago, el sistema debe garantizar conectividad ininterrumpida entre estas sedes y el servidor central.

- **Tipo de Red y Uso de Internet:** el *software* está diseñado como una aplicación web, por lo que operará bajo una red WAN (*Wide Area Network*). Esto permite que el sistema sea consumido a través de Internet desde cualquier navegador web en las distintas clínicas, centralizando la información en una única base de datos sin necesidad de instalaciones locales en cada computadora.

- **Puertos de Comunicación:** para la puesta en producción, el sistema utilizará el puerto estándar HTTP (80) y requerirá un certificado SSL para cifrar el tráfico a través del puerto seguro HTTPS (443), protegiendo las credenciales y datos salariales durante su transmisión. Durante la fase de desarrollo del prototipo, la herramienta *Flask* utiliza por defecto el puerto local 5000.
- **Equipos y Costos:** la comunicación se gestionará a través de los *routers* y *switches* comerciales ya provistos por el Proveedor de Servicios de Internet (ISP) en cada sucursal. No se contemplan costos adicionales en equipos especializados de telecomunicaciones, ya que el ancho de banda y la infraestructura de red actual de Invet Diagnóstico cumplen con los requerimientos mínimos para soportar tráfico web transaccional.

Descripción Detallada de las Herramientas Técnicas Utilizadas

El desarrollo del proyecto se fundamenta en tecnologías de vanguardia que garantizan la integridad, escalabilidad y mantenibilidad del código fuente.

- **Lenguaje de Programación (*Backend*):** se utilizará *Python* en su versión 3.14. Este lenguaje de alto nivel orientado a objetos, fue seleccionado por su robustez matemática y su inmensa biblioteca de soporte para cálculos complejos.
- **Framework Controlador: *Flask*.** Actuará como el enrutador principal del sistema, manejando las sesiones de los usuarios, las conexiones a la base de datos y la seguridad de las peticiones web HTTP.
- **Motor de Base de Datos:** se implementará *MySQL* 8.0.44 mediante la herramienta de modelado *MySQL Workbench Community Edition*. Este motor relacional garantiza la Atomicidad, Consistencia, Aislamiento y Durabilidad (ACID) de los datos transaccionales de la planilla. Su capacidad de almacenamiento permite gestionar millones de registros, asegurando un ciclo de vida funcional a largo plazo para la veterinaria.
- **Entorno de Desarrollo (IDE):** *Visual Studio Code* (versión 1.105.1), utilizado para la codificación y depuración del sistema.
- **Herramientas *Frontend*:** el diseño visual se construirá utilizando HTML5, CSS3, *JavaScript* y el *framework Bootstrap*, asegurando que la interfaz sea responsiva y se adapte a dispositivos de escritorio y móviles.

- **Licenciamiento y Costos:** todas las herramientas técnicas y *frameworks* seleccionados para este proyecto operan bajo licencias de *software* libre u *Open Source* (GPL, BSD, MIT). Por consiguiente, la empresa no deberá incurrir en costos de licenciamiento, pagos de suscripción, ni anualidades por el uso del *software* de desarrollo y gestión de bases de datos.

Descripción Detallada del Conocimiento Básico del Recurso Humano

El sistema ha sido estructurado con un enfoque prioritario en la usabilidad y la experiencia de usuario (UX), buscando reducir drásticamente la curva de aprendizaje del personal de Invet Diagnóstico.

- **Personal Operativo y Médico:** para los colaboradores generales, veterinarios y asistentes, el conocimiento requerido se limita a la ofimática básica y la navegación estándar en Internet. Deben ser capaces de ingresar credenciales en un navegador web, navegar por menús interactivos y emplear formularios simples para marcar su asistencia o solicitar vacaciones. No se requiere ningún conocimiento técnico especializado.
- **Personal Administrativo y Jefaturas:** se requiere familiaridad básica con la lectura de gráficas de barras e indicadores gerenciales, así como un conocimiento claro de los procesos internos de aprobación del talento humano.
- **Departamento de Recursos Humanos (RRHH):** el usuario que administre el sistema debe contar con un entendimiento sólido de la legislación laboral costarricense. Aunque el sistema automatiza las fórmulas matemáticas, el operador debe comprender conceptos como las deducciones de la CCSS, tramos del impuesto sobre la renta, cálculo de cesantía, preaviso y jornadas extraordinarias para poder auditar adecuadamente las planillas generadas y mantener actualizados los parámetros de ley en la configuración del *software*. No se exige ningún conocimiento en lenguajes de programación, ni consultas SQL.

Elaboración de los Casos de Uso

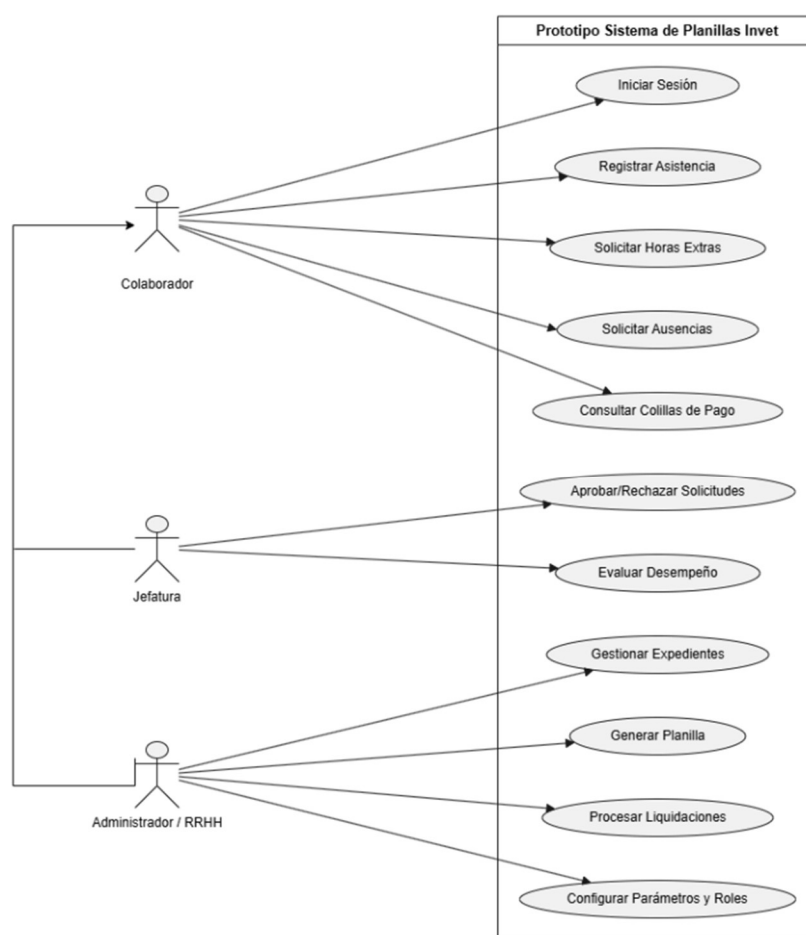
El modelado de casos de uso constituye una herramienta fundamental en la fase de análisis de requerimientos y diseño de la arquitectura de *software*. Este estándar gráfico permite estructurar, visualizar y delimitar el comportamiento funcional del sistema desde la perspectiva

de los usuarios finales, garantizando que el diseño atienda las necesidades operativas de la organización.

A continuación, se presenta el diagrama general de casos de uso del Prototipo Sistema de Planillas Invet, el cual ilustra las interacciones directas entre la plataforma y los tres actores principales definidos para el control de accesos: el *Colaborador* (con privilegios estándar de autoservicio), la *Jefatura* (encargada de la revisión y evaluación) y el *Administrador / RRHH* (responsable del procesamiento financiero y la configuración paramétrica).

Figura 6.

Diagrama de casos de uso.



Fuente: Elaboración propia (2026).

Tabla 6.*Caso de uso: Iniciar Sesión en el Sistema.*

Prototipo Sistema de Planillas Invet	
Número de caso de uso: CU-001.	Nombre de caso de uso: Iniciar Sesión en el Sistema.
Fecha de elaboración:	09/06/2026.
Descripción del caso de uso:	Validar las credenciales de la persona usuaria para permitir el acceso seguro a los módulos correspondientes según su rol.
Autor del caso de uso:	Fabián Chaves Delgado.
Actores relacionados:	Persona usuaria (aplica para todos los perfiles).
Precondiciones:	La persona usuaria debe tener un expediente creado en la base de datos y su estado debe ser "Activo".
1.1 La persona inicia el sistema. 1.2 El sistema carga la pantalla principal de inicio de sesión. 1.3 La persona usuaria digita su correo electrónico y contraseña. FA-01 1.4 El sistema consulta la base de datos para validar las credenciales y el estado del empleado. FA-02 1.5 El sistema genera la sesión de forma segura y extrae la matriz de permisos. SF-1 1.6 El sistema redirige a la persona usuaria al panel principal.	
Subflujos	
SF-1	El sistema consulta la tabla de permisos asociadas al perfil del usuario para cargar en memoria únicamente los módulos autorizados.
Flujos Alternos	
FA-1	El sistema detecta que la contraseña o el correo no coinciden con la base de datos, deniega el acceso y muestra el mensaje: “Credenciales incorrectas o usuario no registrado”.
FA-2	El sistema valida las credenciales, pero detecta el estado de la persona usuaria es “Inactivo”. Deniega el acceso y notifica la restricción.
Requerimientos especiales	
El acceso genera un registro implícito en el manejo de sesiones del <i>framework</i> para mantener la seguridad.	
Poscondiciones	
El sistema crea un entorno de sesión seguro cargando el identificador, perfil y los permisos dinámicos asignados al usuario.	

Fuente: Elaboración propia (2026).

Tabla 7.*Consultar panel principal y notificaciones.*

Prototipo Sistema de Planillas Invet	
Número de caso de uso: CU-002.	Nombre de caso de uso: Consultar Panel Principal y Notificaciones.
Fecha de elaboración:	09-06-2026.
Descripción del caso de uso:	Visualizar el estado general del sistema, acceder a los módulos y consultar alertas.
Autor del caso de uso:	Fabián Chaves Delgado.
Actores relacionados:	Persona usuaria, Administrador, RRHH, Supervisor, Sistema.
Precondiciones:	Debe existir una sesión activa válida en el sistema (resultado del CU-01).
1.1. La persona usuaria accede a la ruta principal del sistema. 1.2. El sistema valida los permisos asociados a la sesión. FA-01 1.3. El sistema consulta la tabla de notificaciones y carga el historial del usuario. SF-1 1.4. El sistema despliega las tarjetas de acceso rápido correspondientes a los módulos autorizados. SF-2 1.5. La persona usuaria interactúa con los accesos rápidos o revisa sus alertas en la barra de navegación.	
Subflujos	
SF-1	El sistema cuenta automáticamente las alertas con estado "no leídas" e inyecta un indicador visual dinámico en la campana de notificaciones.
SF-2	Si el perfil de la persona usuaria es de Jefatura (RRHH/Admin), el sistema ejecuta consultas adicionales y despliega las listas de solicitudes pendientes de revisión.
Flujos Alternos	
FA-01	El sistema detecta que no hay una variable de sesión activa en el navegador y redirige automáticamente a la pantalla de <i>Login</i> .
Requerimientos especiales	
Las consultas para jefaturas implican el cruce de múltiples tablas (JOIN) en <i>MySQL</i> para traer los nombres completos de los solicitantes.	
Poscondiciones	
La persona usuaria se ubica en la interfaz central, habilitada para ejecutar transacciones hacia otras vistas del sistema.	

Fuente: Elaboración propia (2026).

Tabla 8.*Registrar marca de asistencia.*

Prototipo Sistema de Planillas Invet	
Número de caso de uso: CU-003.	Nombre del caso de uso: Registrar Marca de Asistencia.
Fecha de elaboración:	09/06/2026.
Descripción del caso de uso:	Registrar la hora exacta de inicio, descansos y finalización de la jornada laboral del colaborador.
Autor del caso de uso:	Fabián Chaves Delgado.
Actores relacionados:	Persona usuaria, Sistema.
Precondiciones:	Debe existir una sesión activa.
1.1. La persona usuaria ingresa al módulo de Reloj Marcador desde el menú principal. 1.2. El sistema muestra la hora en tiempo real y los botones de registro. 1.3. La persona usuaria selecciona el tipo de marca a ejecutar (Entrada, Salida, etc.). 1.4. El sistema captura la fecha y hora de la transacción. SF-1 1.5. El sistema valida la secuencia lógica contra las marcas históricas del día. FA-01, FA-02 y FA-03 1.6. El sistema guarda el registro en la base de datos. 1.7. El sistema recarga la pantalla y muestra un mensaje de éxito.	
Subflujos	
SF-01	El sistema captura el tiempo (fecha y hora) directamente desde el servidor <i>backend (Python)</i> para evitar manipulaciones de hora en el dispositivo cliente.
Flujos Alternos	
FA-01	El sistema detecta una alteración de secuencia (por ej. intentar marcar salida sin tener entrada) y bloquea la transacción mostrando un mensaje de error.
FA-02	El sistema detecta que ya existe una marca idéntica registrada para la fecha actual y bloquea la inserción duplicada.
FA-03	El sistema detecta que ya existe una marca de "Salida" previa en el día, asume la jornada por finalizada y bloquea de forma definitiva nuevos registros.
Requerimientos especiales	
El diseño requiere alta precisión en la captura de marcas de tiempo para el posterior cálculo automático de llegadas tardías.	
Poscondiciones	
Se inserta un nuevo registro inmutable en la tabla Registro Asistencia de la base de datos.	

Fuente: Elaboración propia (2026).

Tabla 9.*Solicitar aprobación de horas extras.*

Prototipo Sistema de Planillas Invet	
Número de caso de uso: CU-004.	Nombre del Caso de Uso: Solicitar Aprobación de Horas Extras.
Fecha de elaboración:	09/06/2026.
Descripción del caso de uso:	Permitir a la persona usuaria reportar tiempo extra laborado para someterlo a revisión.
Autor del caso de uso:	Fabián Chaves Delgado.
Actores relacionados:	Persona usuaria, Sistema.
Precondiciones	Sesión activa y contar con el permiso Módulo de Horas Extras en la matriz de seguridad.
1.1. La persona usuaria accede al Módulo de Horas Extras. 1.2. El sistema carga el formulario de solicitud. 1.3. La persona usuaria completa el formulario ingresando fecha, cantidad de horas, tarifa y justificación. 1.4. La persona usuaria presiona el botón para enviar la solicitud. 1.5. El sistema procesa la información y valida las reglas del negocio de la clínica. FA-01 y FA-02 1.6. El sistema verifica que el usuario no cuente con trámites previos en proceso. FA-03 1.7. El sistema inserta el registro indicando el estado "Pendiente". SF-1 1.8. El sistema muestra la pantalla de confirmación de envío.	
Subflujos	
SF-01	El sistema extrae el ID del empleado directamente de la sesión activa en el servidor, garantizando que el usuario no pueda alterar a quién pertenece la solicitud mediante el HTML.
Flujos Alternos	
FA-01	La fecha seleccionada corresponde a un día en el pasado. El sistema rechaza el proceso por políticas de auditoría.
FA-02	La cantidad de horas excede el límite máximo parametrizado por la empresa (4 horas). El sistema bloquea el envío y muestra una alerta.
FA-03	El sistema detecta mediante una consulta SQL que ya existe una solicitud en estado "Pendiente" a nombre del usuario. Se bloquea el formulario temporalmente para evitar transacciones duplicadas.
Requerimientos especiales	
El límite de horas (4) es un parámetro estricto de regla de negocio que se valida a nivel de <i>backend</i> .	
Poscondiciones	
La solicitud se almacena en la tabla Solicitudes Horas Extras en estado Pendiente.	

Fuente: Elaboración propia (2026).

Tabla 10.*Gestionar solicitud de ausencias.*

Prototipo Sistema de Planillas Invet	
Número de caso de uso: CU-005.	Nombre del Caso de Uso: Gestionar Solicitud de Ausencias.
Fecha de elaboración:	09/06/2026.
Descripción del caso de uso:	Tramitar la petición de días libres, validando los saldos y omitiendo fines de semana.
Autor del caso de uso:	Fabián Chaves Delgado.
Actores relacionados:	Persona usuaria, Sistema.
Precondiciones:	Para tramitar vacaciones, el empleado debe tener un saldo acumulado mayor a cero en su expediente.
1.1. La persona usuaria ingresa al módulo de Ausencias. 1.2. El sistema carga el formulario de petición de tiempo libre. 1.3. La persona usuaria selecciona las fechas, el tipo (Vacaciones o Permiso) y digita el motivo. 1.4. La persona usuaria presiona el botón "Solicitar". 1.5. El sistema evalúa la cronología lógica de las fechas brindadas. FA-01 1.6. El sistema ejecuta el algoritmo para calcular los días hábiles reales solicitados. SF-1 y FA-02 1.7. El sistema valida el cálculo contra el saldo del empleado. FA-03 1.8. El sistema almacena la transacción en la base de datos en estado "Pendiente". 1.9. El sistema recarga la pantalla notificando el éxito de la solicitud.	
Subflujos	
SF-01	El sistema itera sobre el rango de fechas seleccionado y cuenta exclusivamente los días laborables, omitiendo los sábados y domingos de la sumatoria total a rebajar.
Flujos Alternos	
FA-01	La fecha final es anterior a la fecha de inicio. El sistema devuelve un error lógico y aborta el guardado.
FA-02	Las fechas ingresadas abarcan únicamente un fin de semana (días no hábiles resultan en cero). El sistema rechaza la solicitud de forma automática.
FA-03	La sumatoria de días solicitados es mayor a la cantidad de vacaciones disponibles en la base de datos del empleado. El sistema deniega la transacción mostrando el límite permitido.
Requerimientos especiales	
El motor de validación temporal requiere evaluar la solicitud mediante un bucle iterativo (<i>While</i>) procesado en el servidor.	
Poscondiciones	
La solicitud se inserta en Solicitudes Ausencias. El saldo del empleado se mantiene intacto hasta la aprobación formal.	

Fuente: Elaboración propia (2026).

Tabla 11.*Registro de incapacidades.*

Prototipo Sistema de Planillas Invet	
Número de caso de uso: CU-006.	Nombre del Caso de Uso: Registro de Incapacidades.
Fecha de elaboración:	09/06/2026.
Descripción del caso de uso:	Ingreso de una incapacidad médica para un colaborador por parte de la jefatura.
Autor del caso de uso:	Fabián Chaves Delgado.
Actores relacionados:	Administrador, RRHH, Supervisor, Sistema.
Precondiciones:	El actor debe tener un rol autorizado (Jefatura/RRHH) y el empleado a incapacitar debe existir en estado “Activo”.
1.1. El actor ingresa al apartado de incapacidades en el menú principal. 1.2. El sistema carga la página y despliega un menú con el listado de empleados. SF-1 1.3. El actor selecciona al empleado correspondiente desde la lista. 1.4. El actor ingresa la fecha de inicio y la fecha de finalización de la incapacidad. FA-01 1.5. El actor ingresa el motivo o justificación médica brindada. 1.6. El actor presiona el botón de guardar incapacidad. 1.7. El sistema registra la ausencia y le asigna el estado correspondiente. SF-2 y FA-02 1.8. El sistema redirige al panel principal y muestra un mensaje de éxito.	
Subflujos	
SF-01	El sistema ejecuta una consulta a la tabla de Empleados para cargar únicamente aquellos con estado "Activo", filtrando a los exempleados.
SF-02	Por políticas de negocio, el sistema asigna por defecto el estado "Aprobado" sin requerir flujo de validaciones extras, al tratarse de un registro de jefatura.
Flujos Alternos	
FA-01	El actor ingresa una fecha de inicio mayor a la de finalización, el sistema detecta la inconsistencia lógica y solicita corrección.
FA-02	Falla la conexión con la base de datos, el sistema ejecuta un <i>rollback</i> y envía un mensaje de error indicando que la transacción falló.
Requerimientos especiales	
Se inyecta de forma implícita (<i>hardcode</i>) el tipo de ausencia como "Incapacidad" para su posterior tratamiento en el Motor de Planillas.	
Poscondiciones	
La incapacidad queda registrada en la tabla Solicitudes Ausencias, lista para aplicarle el porcentaje de deducción (50 % o 100 %) en el pago.	

Fuente: Elaboración propia (2026).

Tabla 12.*Cálculo de liquidaciones.*

Prototipo Sistema de Planillas Invet	
Número de caso de uso: CU-007	Nombre del Caso de Uso: Cálculo de Liquidaciones.
Fecha de elaboración:	09/06/2026.
Descripción del caso de uso:	Cálculo automático de los extremos laborales de un empleado tras su salida de la organización.
Autor del caso de uso:	Fabián Chaves Delgado.
Actores relacionados:	Administrador, RRHH, Sistema.
Precondiciones:	El actor debe poseer el permiso de acceso al módulo y el empleado a liquidar debe estar en estado "Activo".
1.1. El actor ingresa al apartado de liquidaciones desde el menú principal. 1.2. El sistema carga el formulario y despliega la lista de empleados activos. 1.3. El actor selecciona al empleado e ingresa las fechas de ingreso y salida. FA-01 1.4. El actor ingresa el motivo de la salida laboral (por ej. Despido con Responsabilidad). 1.5. El actor presiona el botón para calcular la liquidación. 1.6. El sistema extrae el salario bruto y procesa el tiempo laborado. SF-1 1.7. El sistema ejecuta el cálculo de aguinaldo proporcional, vacaciones, preaviso y cesantía, según corresponda. SF-2 1.8. El sistema registra la transacción y actualiza el estado del expediente. SF-3 y FA-02 1.9. El sistema muestra en pantalla el comprobante desglosado con los totales a pagar.	
Subflujos	
SF-1	El sistema convierte y evalúa las fechas proporcionadas para calcular matemáticamente los días y meses totales laborados por el colaborador.
SF-2	El algoritmo aplica condicionales basados en el Código de Trabajo para determinar si el motivo de salida amerita el pago de los extremos de preaviso y cesantía.
SF-3	El sistema ejecuta una actualización asíncrona (<i>UPDATE</i>) en la tabla Empleados para cambiar el estado del colaborador de "Activo" a "Inactivo".
Flujos Alternos	
FA-01	El actor ingresa una fecha de salida que es cronológicamente anterior a la fecha de ingreso. El sistema detiene el proceso y muestra un error lógico de validación.
FA-02	Falla la inserción de los datos financieros en la base de datos. El sistema aplica un <i>rollback</i> , detiene la inhabilitación del usuario y notifica el error.
Requerimientos especiales	
Las fórmulas matemáticas inyectadas en el controlador deben respetar estrictamente los factores de cálculo de la legislación laboral vigente de Costa Rica.	
Poscondiciones	
El empleado queda deshabilitado para ingresar al sistema y los montos calculados se registran como pasivos en la tabla Liquidaciones.	

Fuente: Elaboración propia (2026).

Tabla 13.*Reversión de pagos de planilla.*

Prototipo Sistema de Planillas Invet	
Número de caso de uso: CU-008.	Nombre del Caso de Uso: Reversión de Pagos de Planilla.
Fecha de elaboración:	09/06/2026.
Descripción del caso de uso:	Eliminación en cascada de una planilla generada erróneamente para limpiar los registros financieros del sistema.
Autor del caso de uso:	Fabián Chaves Delgado.
Actores relacionados:	Administrador, Sistema.
Precondiciones:	El actor debe tener permisos de auditoría y debe existir al menos una planilla procesada en el historial.
1.1. El actor ingresa al historial de planillas del sistema. 1.2. El sistema consulta y muestra la lista de planillas procesadas ordenadas por fecha. 1.3. El actor ubica la planilla a reversar y presiona el botón "Reversar". 1.4. El sistema captura el ID unívoco de la planilla seleccionada. 1.5. El sistema elimina los recibos de pago individuales asociados. SF-1 1.6. El sistema elimina el registro maestro de la planilla. SF-2 y FA-01 1.7. El sistema registra la acción destructiva en la bitácora de auditoría. SF-3 1.8. El sistema recarga el historial mostrando el mensaje de éxito	
Subflujos	
SF-1	El sistema ejecuta un <i>DELETE</i> en la tabla Detalle Planilla para borrar todos los registros "hijos" vinculados al identificador de la planilla anulada.
SF-2	El sistema ejecuta un <i>DELETE</i> en la tabla Pagos Planilla para borrar al registro "padre".
SF-2	El sistema invoca el motor de <i>logs</i> transaccionales para registrar la identidad del usuario que ejecutó la reversión.
Flujos Alternos	
FA-01	Si ocurre un fallo de integridad referencial al eliminar los detalles o el maestro, el sistema ejecuta un <i>rollback</i> para deshacer los borrados parciales y evitar datos corruptos.
Requerimientos especiales	
Este proceso es destructivo y requiere un control transaccional estricto desde el <i>backend</i> .	
Poscondiciones	
Los registros de pago son eliminados permanentemente, desapareciendo del historial gerencial y de los portales de autogestión de los colaboradores afectados.	

Fuente: Elaboración propia (2026).

Tabla 14.*Consulta y descarga de colilla.*

Prototipo Sistema de Planillas Invet	
Número de caso de uso: CU-009.	Nombre del Caso de Uso: Consulta y Descarga de Colilla.
Fecha de elaboración:	09/06/2026.
Descripción del caso de uso:	Visualización del desglose salarial y exportación del documento en formato PDF.
Autor del caso de uso:	Fabián Chaves Delgado.
Actores relacionados:	Persona usuaria (Colaborador), Sistema.
Precondiciones:	La persona usuaria debe contar con una sesión activa y poseer historial de pagos registrados a su nombre.
1.1. La persona usuaria ingresa al apartado "Mis Recibos" en el menú. 1.2. El sistema consulta la base de datos y carga el historial de pagos. SF-1 1.3. La persona usuaria presiona el botón de "Ver Recibo" en la quincena requerida. 1.4. El sistema valida los permisos de acceso y carga el desglose completo de ingresos y deducciones. FA-01 1.5. La persona usuaria presiona el botón "Descargar PDF". 1.6. El sistema renderiza y procesa el documento en segundo plano. SF-2 1.7. El sistema envía el archivo al navegador forzando su descarga automática.	
Subflujos	
SF-1	El sistema utiliza el ID del empleado almacenado en la sesión de red para filtrar la consulta SQL, garantizando que el usuario solo visualice su propia información.
SF-2	El sistema utiliza la librería interna de transformación para renderizar la vista HTML en un documento PDF, construyéndolo en la memoria volátil del servidor.
Flujos Alternos	
FA-01	La persona usuaria intenta alterar la URL (método GET) para visualizar un recibo que pertenece a un tercero. El sistema detecta la violación de políticas de privacidad y deniega el acceso.
Requerimientos especiales	
Requiere procesamiento en memoria RAM (BytesIO) para generar el documento descargable sin almacenar archivos residuales en el disco duro del servidor.	
Poscondiciones	
La persona usuaria obtiene su comprobante salarial en un formato digital inmutable.	

Fuente: Elaboración propia (2026).

Tabla 15.*Auditoría de marcas y puntualidad.*

Prototipo Sistema de Planillas Invet	
Número de caso de uso: CU-010.	Nombre del Caso de Uso: Auditoría de Marcas y Puntualidad.
Fecha de elaboración:	09/06/2026.
Descripción del caso de uso:	Generación de un reporte que evalúa la puntualidad y cuantifica las horas efectivas laboradas por el personal de la clínica.
Autor del caso de uso:	Fabián Chaves Delgado.
Actores relacionados:	Administrador, RRHH, Supervisor, Sistema.
Precondiciones:	Deben existir registros de marcas en el reloj institucional correspondientes a las fechas analizadas.
1.1. El actor ingresa al módulo "Auditor de Asistencia" desde la barra lateral. 1.2. El sistema ejecuta una consulta agrupada sobre los registros de marcas. 1.3. El sistema evalúa la primera hora de entrada registrada contra la hora oficial de inicio de jornada. SF-1 1.4. El sistema calcula las horas netas trabajadas. SF-2 y FA-01 1.5. El sistema asigna etiquetas y modificadores de interfaz según la puntualidad. SF-3 1.6. El sistema renderiza la tabla consolidada mostrando los resultados al actor.	
Subflujos	
SF-1	El motor de base de datos compara la hora mínima registrada (MIN(hora_marca)) contra el parámetro de tolerancia institucional preestablecido.
SF-2	El sistema calcula la diferencia temporal en minutos entre la entrada y la salida final, restando automáticamente el bloque de tiempo correspondiente a descansos de ley.
SF-3	El sistema inyecta variables condicionales para resaltar visualmente las llegadas tardías en color rojo, mejorando el reconocimiento de patrones de ausentismo.
Flujos Alternos	
FA-01	Un empleado omitió registrar su marca de salida. El cálculo matemático de horas efectivas devuelve un valor nulo o incompleto y el sistema lo refleja explícitamente en la tabla de auditoría para su seguimiento.
Requerimientos especiales	
El procesamiento de este reporte recae directamente sobre el motor SQL (GROUP BY e IF anidados) para maximizar la velocidad de carga de la vista.	
Poscondiciones	
El actor gerencial obtiene una radiografía del comportamiento de puntualidad de la organización.	

Fuente: Elaboración propia (2026).

Tabla 16.*Registro de nuevos expedientes.*

Prototipo Sistema de Planillas Invet	
Número de caso de uso: CU-011.	Nombre del Caso de Uso: Registro de Nuevos Expedientes.
Fecha de elaboración:	09/06/2026.
Descripción del caso de uso:	Creación formal de un nuevo expediente laboral y la respectiva asignación de credenciales de acceso al sistema.
Autor del caso de uso:	Fabián Chaves Delgado.
Actores relacionados:	Administrador, RRHH, Sistema.
Precondiciones:	El actor debe contar con los permisos correspondientes a gestión de planillas y recursos humanos.
1.1. El actor ingresa al apartado de "Mantenimiento de Empleados". 1.2. El sistema carga la interfaz de registros y la matriz de perfiles de seguridad disponibles. 1.3. El actor completa el formulario ingresando los datos personales, salario base y credenciales de acceso del nuevo colaborador. 1.4. El actor presiona el botón "Guardar Expediente". 1.5. El sistema ejecuta validaciones de formato de datos e integridad sobre la información recibida. FA-01 y FA-02 1.6. El sistema inserta los metadatos laborales en la tabla de empleados y captura el identificador generado. SF-1 1.7. El sistema inserta las credenciales de sesión en la tabla de usuarios vinculándolas al expediente recién creado. SF-2 1.8. El sistema recarga la vista notificando la incorporación exitosa.	
Subflujos	
SF-1	El sistema asigna automáticamente un saldo inicial de 0 días de vacaciones disponibles y etiqueta al registro con el estado "Activo".
SF-2	Se garantiza la integridad referencial inyectando el ID del empleado como llave foránea dentro del registro de credenciales.
Flujos Alternos	
FA-01	El actor ingresa un monto salarial equivalente o inferior a cero. El sistema detiene la transacción y devuelve una alerta por violación de validación financiera.
FA-02	El sistema detecta mediante una expresión regular, que el número de cédula posee caracteres inválidos y aborta la inserción en la base de datos.
Requerimientos especiales	
Las transacciones de escritura se ejecutan de manera concurrente (<i>commit</i> único) para prevenir escenarios de orfandad de datos entre la tabla Empleados y la tabla Usuarios.	
Poscondiciones	
El nuevo colaborador se integra plenamente a la lógica del sistema, habilitado para iniciar sesión y ser procesado en la próxima ejecución de nómina.	

Fuente: Elaboración propia (2026).

Tabla 17.*Calculo y Generación de Planillas.*

Prototipo Sistema de Planillas Invet	
Número de caso de uso: CU-012.	Nombre del Caso de Uso: Cálculo y Generación de Planillas.
Fecha de elaboración:	09/06/2026.
Descripción del caso de uso:	Procesamiento matemático, visualización y registro oficial de la nómina quincenal para todos los colaboradores activos, aplicando ingresos, deducciones de ley y afectaciones operativas (extras, ausencias y tardías).
Autor del caso de uso:	Fabián Chaves Delgado.
Actores relacionados:	Administrador, RRHH, Sistema.
Precondiciones:	El actor debe contar con los permisos correspondientes a gestión de planillas y recursos humanos.
2. El actor ingresa al apartado de Planillas en el menú principal. 3. El sistema carga la vista desplegando los parámetros de ley vigentes y preselecciona la fecha actual. 4. El actor ingresa la "Fecha de Inicio" y la "Fecha de Fin" correspondientes al periodo a calcular. FA-01 5. El actor presiona el botón de "Calcular Nómina". 6. El sistema valida que el periodo solicitado no haya sido procesado anteriormente. SF-1 y FA-02 7. El sistema extrae de la base de datos el padrón de empleados habilitados. SF-2 8. El motor matemático del sistema procesa el salario base, evalúa la regla de antigüedad y calcula todas las sumas (horas extras) y restas (ausencias, incapacidades, tardías, CCSS, Renta). SF-3 9. El sistema redirige a la pantalla de "Pre-aprobación" mostrando una tabla detallada con los cálculos individuales y los totales globales de la nómina. 10. El actor revisa los montos y presiona "Guardar Planilla". 11. El sistema guarda el encabezado general (Tabla Padre) y el desglose individual por empleado (Tabla Hija). SF-4 12. El sistema actualiza los saldos de vacaciones de cada colaborador. SF-5 13. El sistema redirige al historial de planillas y muestra un mensaje de éxito. FA-03	
Subflujos	
SF-1	Antes de iniciar la matemática, el sistema ejecuta una consulta en la tabla PagosPlanilla buscando coincidencias exactas con las fechas ingresadas para garantizar la unicidad del periodo.
SF-2	El sistema consulta la tabla Empleados extrayendo únicamente a los colaboradores con estado "Activo", excluyendo a personal liquidado o inactivo.
SF-3	El motor evalúa si el empleado tiene menos de 15 días laborando; de ser así, lo excluye del pago actual. Si en el corte anterior el empleado fue excluido por esta misma regla, el sistema le suma retroactivamente esos días pendientes a la quincena actual. Además, cruza las fechas con los módulos de Horas Extras, Ausencias y Reloj Marcador para inyectar los montos variables.
SF-4	El sistema empaqueta la inserción del pago maestro y sus líneas de detalle en un bloque estructurado (try/except) para asegurar que los datos se guarden en su totalidad o no se guarde ninguno.

SF-5	Tras guardar el recibo, el sistema calcula la proporción de vacaciones generadas en el periodo y actualiza de forma asíncrona la columna <code>dias_vacaciones_disponibles</code> del empleado.
Flujos Alternos	
FA-01	El actor ingresa una fecha de finalización menor a la fecha de inicio. El sistema detiene la ejecución por inconsistencia lógica.
FA-02	El sistema detecta que las fechas ingresadas ya existen en el historial de planillas. Se interrumpe el cálculo, se devuelve un código HTTP 400 y se muestra una alerta roja indicando que debe reversar la planilla anterior si desea recalcularla.
FA-3	Ocurre una falla de conexión o un error de integridad de llaves al guardar los registros. El sistema ejecuta una instrucción <code>rollback()</code> deshaciendo cualquier inserción a medias, mostrando el error exacto y protegiendo la base de datos.
Requerimientos especiales	
El objeto con los resultados del cálculo se serializa en formato JSON y se oculta en la vista de pre-aprobación, optimizando el rendimiento al evitar que el servidor deba recalcular la matemática completa al momento de confirmar el guardado.	
Poscondiciones	
Se genera un registro maestro en la tabla <code>PagosPlanilla</code> y múltiples registros en <code>DetallePlanilla</code> . Los recibos electrónicos quedan listos para su visualización o distribución masiva vía correo SMTP (Módulo de Pagos). Los empleados reciben su aumento proporcional en el saldo de vacaciones.	

Fuente: Elaboración propia (2026).

Arquitectura del Sistema

La arquitectura del sistema para el prototipo se fundamenta en un modelo Cliente-Servidor estructurado en tres niveles lógicos de infraestructura. Esta disposición garantiza la centralización de la lógica de negocio y la seguridad de la información, permitiendo que los colaboradores interactúen con el sistema sin necesidad de instalar *software* especializado en sus dispositivos.

- **Nivel de Cliente (Capa de Presentación Externa):** corresponde a los dispositivos físicos (computadoras de escritorio, terminales o dispositivos móviles) utilizados por el personal de la clínica. La interacción se realiza de forma exclusiva a través de un navegador web estandarizado, el cual procesa las interfaces gráficas y envía las peticiones HTTP (GET y POST) hacia el servidor central.
- **Nivel de Servidor de Aplicaciones (Capa Lógica):** representa el entorno donde se hospeda el núcleo del sistema, desarrollado en el lenguaje *Python* bajo el marco de trabajo *Flask*. Este servidor actúa como el intermediario principal: recibe las solicitudes

de los clientes, procesa los algoritmos matemáticos (como el cálculo de planillas y auditoría de asistencia) y gestiona el enrutamiento de las vistas.

- **Nivel de Servidor de Base de Datos (Capa de Persistencia):** ubicado en el extremo más seguro de la topología, este servidor aloja el motor relacional *MySQL*. Su función exclusiva es el almacenamiento, resguardo y control transaccional de los expedientes, configuraciones de ley y registros de asistencia, respondiendo a las consultas directas (SQL) emitidas por el servidor de aplicaciones.

Figura 7.

Arquitectura del sistema.



Fuente: Elaboración propia (2026).

Arquitectura del *Software*

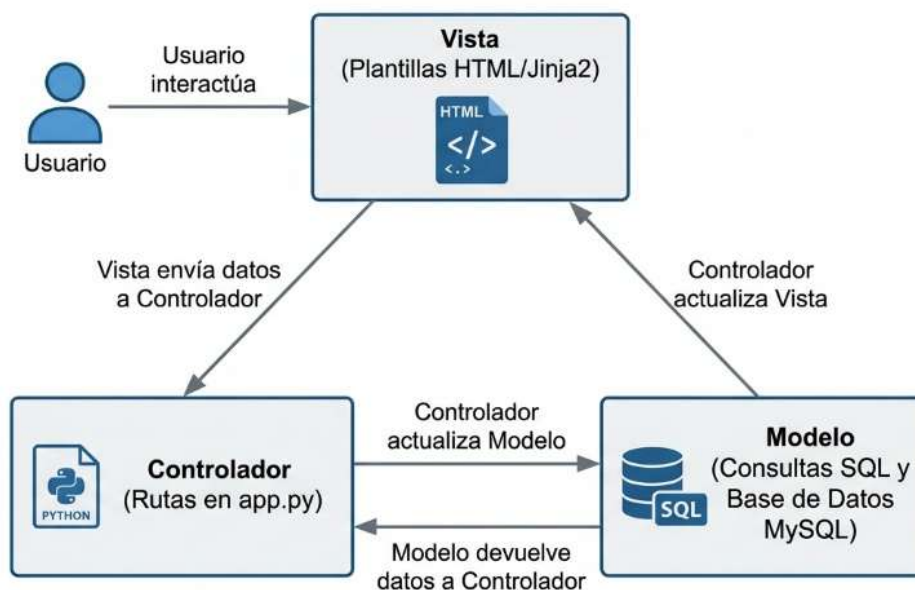
Para el desarrollo interno del prototipo, se implementó una arquitectura centrada en los datos, complementada por el patrón de diseño Modelo-Vista-Controlador (MVC). Esta decisión de ingeniería obedece a la naturaleza transaccional del sistema de planillas, donde la precisión matemática, el control algorítmico y la velocidad de respuesta sobre los datos almacenados son críticos para el éxito del proyecto.

Al emplear una arquitectura centrada en los datos en conjunto con el patrón MVC de forma procedural (sin la intervención de un mapeador objeto-relacional o ORM), se elimina la sobrecarga de procesamiento en memoria, permitiendo que el motor de base de datos asuma la ejecución de operaciones complejas. Los componentes se dividen de la siguiente manera:

- **El Modelo:** está constituido por el esquema relacional en *MySQL* y las consultas SQL nativas (*Raw SQL*) incrustadas en el código. Esta capa es la responsable de garantizar la integridad de los datos, aplicar restricciones de negocio a nivel de base y resolver funciones de agregación para el cálculo salarial y deducciones legales.
- **La Vista:** representa la interfaz de usuario, desarrollada utilizando HTML5, CSS (*Bootstrap*) y el motor de plantillas Jinja2. Esta capa no posee lógica de negocio; su función es recibir los paquetes de datos enviados por el servidor y renderizarlos dinámicamente para que el colaborador o la jefatura puedan consumirlos de forma intuitiva y accesible.
- **El Controlador:** representado por el archivo principal de enrutamiento (*app.py*), desarrollado en *Python*. Actúa como el orquestador del sistema: intercepta las peticiones HTTP entrantes, valida la seguridad y los permisos de sesión (RBAC), ejecuta la lógica de validación de entradas y determina qué consultas SQL deben enviarse al Modelo y qué Vista debe retornarse al usuario final.

Figura 8.

Arquitectura del software.



Fuente: Elaboración propia (2026).

Diseño de Entradas

El diseño de entradas del sistema Invet tiene como objetivo principal la captura ágil, segura y validada de la información proveniente de los usuarios. Para garantizar la integridad de la base de datos, todas las interfaces de entrada aplican el principio de "cero confianza" (*Zero-Trust*), combinando validaciones de formato en el lado del cliente (HTML5) con filtros estrictos de reglas de negocio en el controlador (*Python*), minimizando así la inyección de datos corruptos o errores humanos.

A continuación, se presentan cinco interfaces representativas del diseño de entradas del prototipo:

- 1. Formulario de Registro de Nuevos Expedientes:** esta interfaz está diseñada para la captura de datos maestros del personal. Incluye validaciones de expresiones regulares para la cédula de identidad y bloqueos matemáticos para evitar el ingreso de salarios negativos. Permite al Departamento de Recursos Humanos crear simultáneamente el expediente físico y las credenciales de acceso al sistema en una sola transacción.

Figura 9.

Formulario de registro de nuevos expedientes.

Nuevo Contrato Laboral ✕

1. Datos Personales y Salariales

Cédula de Identidad

Nombre

Apellidos

Salario Bruto (€)

Fecha Oficial de Ingreso 📅

2. Accesos al Sistema Web

Correo Electrónico (Usuario)

Contraseña Temporal

Nivel de Privilegios (Rol)

▼

Fuente: Elaboración propia (2026).

- 2. Interfaz del Reloj Marcador:** representa una entrada de datos de alta criticidad temporal. Su diseño es minimalista para facilitar la rápida interacción del colaborador. El sistema anula la posibilidad de manipulación de la hora por parte del usuario, ya que la interfaz únicamente captura el "tipo de marca" (Entrada, Salida) y delega la captura del tiempo exacto (*Timestamp*) al servidor *backend*.

Figura 10.*Interfaz del reloj marcador.*
Fuente: Elaboración propia (2026).

- 3. Formulario de Solicitud de Ausencias:** diseñado para la gestión de tiempo libre y permisos. La entrada de datos utiliza calendarios dinámicos que restringen la selección de fechas en el pasado y evalúan cronológicamente que la fecha de finalización no sea anterior a la de inicio.

Figura 11.*Formulario de solicitud de ausencias.*
Fuente: Elaboración propia (2026).

- 4. Matriz de Evaluación de Desempeño:** interfaz de captura de datos cualitativos y cuantitativos. Permite a las jefaturas ingresar calificaciones parametrizadas (del 1 al 5) en cinco criterios específicos de rendimiento, además de incluir un área de texto validada para retroalimentación formal que se adjuntará al expediente del colaborador.

Figura 12.*Matriz de evaluación de desempeño.*

Registrar Nueva Evaluación

1. Datos del Colaborador

Seleccionar Colaborador: -- Elige un empleado activo --

Fecha de Evaluación: 23/06/2026

2. Matriz de Calificación (Escala 1 al 5)

1: Deficiente | 2: Regular | 3: Bueno | 4: Muy Bueno | 5: Excelente

Puntualidad y Asistencia: Seleccione nota...

Disponibilidad (Apoyo/Horas Extras): Seleccione nota...

Productividad y Eficiencia: Seleccione nota...

Trabajo en Equipo: Seleccione nota...

Iniciativa y Resolución: Seleccione nota...

CALIFICACIÓN FINAL DEL SISTEMA

0.00%

3. Retroalimentación Cualitativa

Comentarios Finales: Redacte el resumen de la reunión con el colaborador...

Cancelar | Guardar Evaluación Oficial

Fuente: Elaboración propia (2026).

- 5. Configuración de Parámetros de Ley:** interfaz de entrada de nivel gerencial. Permite la actualización de los modificadores porcentuales y topes financieros (como la CCSS o el tramo exento de renta). Su diseño tipo listado facilita la rápida modificación de múltiples variables fiscales en una sola petición al servidor.

Figura 13.*Configuración de parámetros de ley.*

Mantenimiento: Variables del Ministerio de Hacienda y CCSS Volver al Panel

Advertencia de Auditoría
 Modificar estos valores alterará inmediatamente la forma en que el sistema calcula las deducciones de nómina, aguinaldos y finiquitos. Asegúrese de ingresar los valores exactos publicados en el diario oficial.

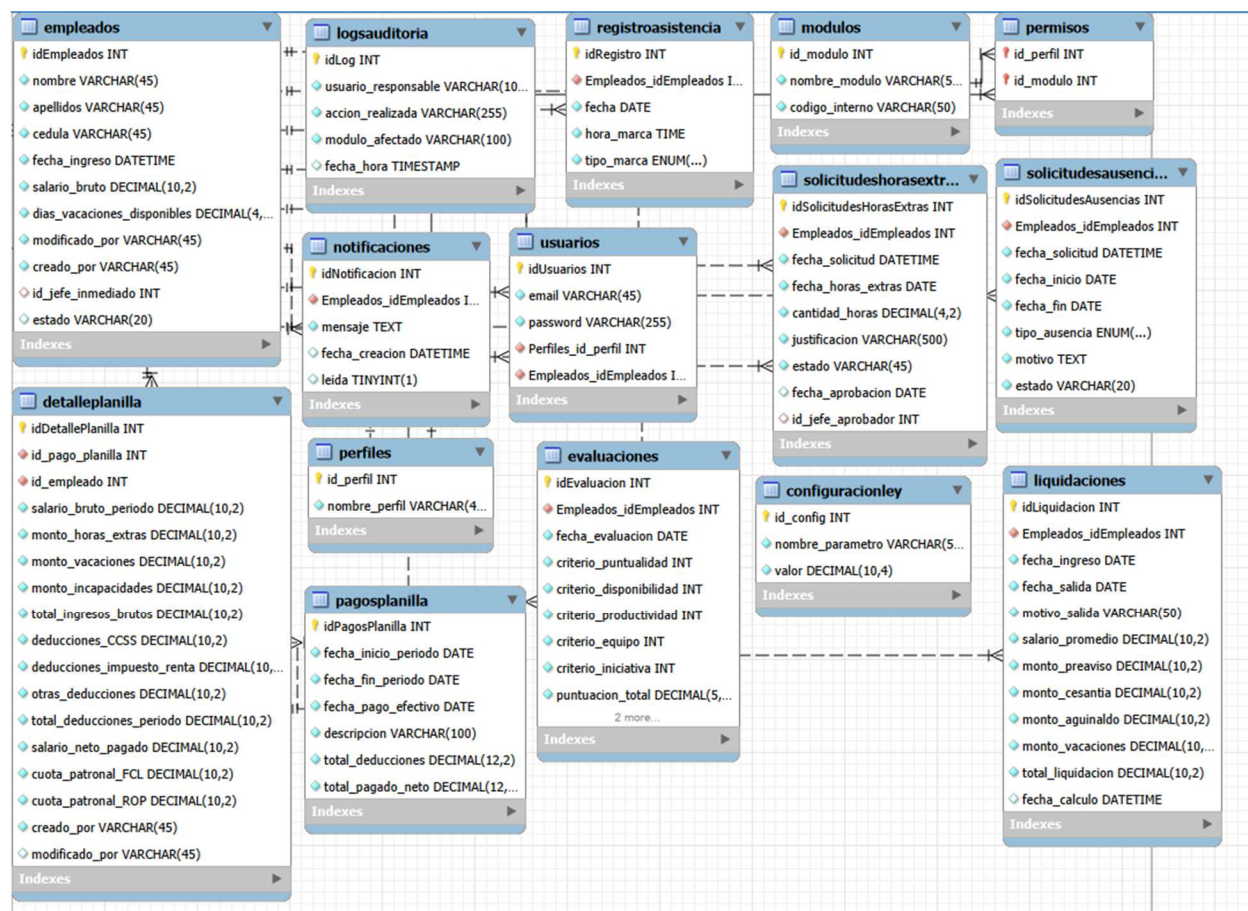
Configuración Global del Sistema

CCSS OBRERO % 0.1083	FCL PATRONAL % 0.0150
ROP PATRONAL % 0.0150	TOPE RENTA EXENTO € 918000.0000

Guardar Cambios Oficiales

Fuente: Elaboración propia (2026).

Diseño Físico de la Base de Datos

Figura 14.*Diseño físico de la base de datos.**Fuente: Elaboración propia (2026).*

Diccionario de Datos

Tabla 17.

Configuración de ley.

Campo	Tipo de Dato	Tamaño	Llave	Nulo	Descripción / Atributos
id_config	INT	-	PK	No	Identificador único (AI).
Nombre_parametro	VARCHAR	50	-	No	Nombre descriptivo del parámetro.
Valor	DECIMAL	10,4	-	No	Valor numérico o porcentual.

Fuente: Elaboración propia (2026).

Descripción: almacena las variables y porcentajes de ley (como deducciones de CCSS o renta) parametrizados por el sistema.

Tabla 18.

Empleados.

Campo	Tipo de Dato	Tamaño	Llave	Nulo	Descripción / Atributos
idEmpleados	INT	-	PK	No	Identificador único (AI).
nombre	VARCHAR	45	-	No	Nombre del colaborador.
apellidos	VARCHAR	45	-	No	Apellidos del colaborador.
cedula	VARCHAR	45	-	No	Documento de identidad.
fecha_ingreso	DATETIME	-	-	No	<i>Feechi</i> de contratación.
salario_bruto	DECIMAL	10,2	-	No	Salario base mensual.
dias_vacaciones_disponibles	DECIMAL	4,1	-	No	Saldo actual de vacaciones.
modificado_por	VARCHAR	45	-	No	Usuario que realizó la última edición.
creado_por	VARCHAR	45	-	No	Usuario que registró el expediente.
id_jefe_inmediado	INT	-	FK	Sí	ID del supervisor asignado (Autorreferencia).
estado	VARCHAR	20	-	Sí	Estado lógico (<i>Default</i> : 'Activo').

Fuente: Elaboración propia (2026).

Descripción: catálogo principal o expediente maestro del personal de la clínica.

Tabla 19.*Pagos planilla.*

Campo	Tipo de Dato	Tamaño	Llave	Nulo	Descripción / Atributos
idPagosPlanilla	INT	-	PK	No	Identificador de la planilla (AI).
fecha_inicio_periodo	DATE	-	-	No	Fecha inicial del corte.
fecha_fin_periodo	DATE	-	-	No	Fecha final del corte.
fecha_pago_efectivo	DATE	-	-	No	Fecha de transferencia al personal.
descripcion	VARCHAR	100	-	No	Nombre o justificación de la planilla.
total_deducciones	DECIMAL	12,2	-	No	Sumatoria global de deducciones.
total_pagado_netto	DECIMAL	12,2	-	No	Sumatoria global del líquido a pagar.

Fuente: Elaboración propia (2026).

Descripción: registro maestro (cabecera) de cada ejecución de nómina generada por el sistema.

Tabla 20.*Detalle planilla.*

Campo	Tipo de Dato	Tamaño	Llave	Nulo	Descripción / Atributos
idDetallePlanilla	INT	-	PK	No	Identificador del recibo (AI).
id_pago_planilla	INT	-	FK	No	Referencia a pagos planilla.
id_empleado	INT	-	FK	No	Referencia a empleados.
salario_bruto_periodo	DECIMAL	10,2	-	No	Salario base proporcional de la quincena.
monto_horas_extras	DECIMAL	10,2	-	No	Dinero total por horas extras.
monto_vacaciones	DECIMAL	10,2	-	No	Dinero pagado por vacaciones.
monto_incapacidades	DECIMAL	10,2	-	No	Subsidio por incapacidad.
total_ingresos_brutos	DECIMAL	10,2	-	No	Sumatoria de ingresos.
deducciones_CCSS	DECIMAL	10,2	-	No	Monto retenido por cargas sociales.
deducciones_impuesto_renta	DECIMAL	10,2	-	No	Monto retenido por tributación.
otras_deducciones	DECIMAL	10,2	-	No	Rebajos por llegadas tardías o permisos.
total_deducciones_periodo	DECIMAL	10,2	-	No	Sumatoria de rebajos.
salario_netto_pagado	DECIMAL	10,2	-	No	Monto final depositado al empleado.
cuota_patronal_FCL	DECIMAL	10,2	-	No	Provisión patronal FCL.

cuota_patronal_ROP	DECIMAL	10,2	-	No	Provisión patronal ROP.
creado_por	VARCHAR	45	-	No	Autor del cálculo.
modificado_por	VARCHAR	45	-	Sí	Autor de reversiones o cambios.

Fuente: Elaboración propia (2026).

Descripción: registro transaccional (hijo) que almacena la colilla de pago desglosada por cada empleado.

Tabla 21.

Evaluaciones.

Campo	Tipo de Dato	Tamaño	Llave	Nulo	Descripción / Atributos
idEvaluacion	INT	-	PK	No	Identificador de la evaluación (AI).
Empleados_idEmpleados	INT	-	FK	No	Referencia al empleado evaluado.
fecha_evaluacion	DATE	-	-	No	Fecha en que se aplicó el instrumento.
criterio_puntualidad	INT	-	-	No	Nota de puntualidad (1-5).
criterio_disponibilidad	INT	-	-	No	Nota de disponibilidad (1-5).
criterio_productividad	INT	-	-	No	Nota de productividad (1-5).
criterio_equipo	INT	-	-	No	Nota de trabajo en equipo (1-5).
criterio_iniciativa	INT	-	-	No	Nota de iniciativa (1-5).
puntuacion_total	DECIMAL	5,2	-	No	Promedio final porcentual.
comentarios	TEXT	-	-	Sí	Retroalimentación cualitativa.
evaluado_por	VARCHAR	100	-	No	Nombre de la jefatura que evalúa.

Fuente: Elaboración propia (2026).

Descripción: matriz de calificación para el registro del desempeño de los colaboradores.

Tabla 22.

Liquidaciones.

Campo	Tipo de Dato	Tamaño	Llave	Nulo	Descripción / Atributos
idLiquidacion	INT	-	PK	No	Identificador del cálculo (AI).
Empleados_idEmpleados	INT	-	FK	No	Referencia al empleado.
fecha_ingreso	DATE	-	-	No	Fecha original de contratación.
fecha_salida	DATE	-	-	No	Fecha de finalización laboral.

motivo_salida	VARCHAR	50	-	No	Justificación legal del cese.
salario_promedio	DECIMAL	10,2	-	No	Base de cálculo financiero.
monto_preaviso	DECIMAL	10,2	-	No	Monto adeudado por preaviso.
monto_cesantia	DECIMAL	10,2	-	No	Monto adeudado por cesantía.
monto_aguinaldo	DECIMAL	10,2	-	No	Aguinaldo proporcional al corte.
monto_vacaciones	DECIMAL	10,2	-	No	Vacaciones no gozadas.
total_liquidacion	DECIMAL	10,2	-	No	Monto final a pagar.
fecha_calculo	DATETIME	-	-	Sí	<i>Timestamp</i> de la ejecución.

Fuente: Elaboración propia (2026).

Descripción: registro de extremos laborales e indemnizaciones tras la salida de un trabajador.

Tabla 23.

Logs auditoría.

Campo	Tipo de Dato	Tamaño	Llave	Nulo	Descripción / Atributos
idLog	INT	-	PK	No	Identificador del <i>log</i> (AI).
usuario_responsable	VARCHAR	100	-	No	Nombre de quien ejecutó la acción.
accion_realizada	VARCHAR	255	-	No	Detalle del proceso ejecutado.
modulo_afectado	VARCHAR	100	-	No	Módulo o controlador impactado.
fecha_hora	TIMESTAMP	-	-	Sí	Marca temporal automática.

Fuente: Elaboración propia (2026).

Descripción: bitácora silenciosa de transacciones críticas para seguridad del sistema.

Tabla 24.

Módulos.

Campo	Tipo de Dato	Tamaño	Llave	Nulo	Descripción / Atributos
id_modulo	INT	-	PK	No	Identificador del módulo (AI).
nombre_modulo	VARCHAR	50	-	No	Nombre visible en la UI.
codigo_interno	VARCHAR	50	-	No	Llave única (<i>Unique</i>) utilizada en código.

Fuente: Elaboración propia (2026).

Descripción: catálogo de secciones del sistema Invet para la matriz de seguridad RBAC.

Tabla 25.*Notificaciones.*

Campo	Tipo de Dato	Tamaño	Llave	Nulo	Descripción / Atributos
idNotificacion	INT	-	PK	No	Identificador del mensaje (AI).
Empleados_idEmpleados	INT	-	FK	No	Destinatario de la alerta.
mensaje	TEXT	-	-	No	Cuerpo del mensaje descriptivo.
fecha_creacion	DATETIME	-	-	Sí	<i>Timestamp</i> automático.
leida	TINYINT	1	-	Sí	Booleano (0 = No leída, 1 = Leída).

Fuente: Elaboración propia (2026).

Descripción: mensajes de retroalimentación interna desplegados en la campana del *dashboard*.

Tabla 26.*Perfiles.*

Campo	Tipo de Dato	Tamaño	Llave	Nulo	Descripción / Atributos
id_perfil	INT	-	PK	No	Identificador del rol (AI).
nombre_perfil	VARCHAR	45	-	No	Nombre descriptivo del puesto.

Fuente: Elaboración propia (2026).

Descripción: catálogo de roles de la clínica (por ej. Administrador, RRHH, Colaborador).

Tabla 27.*Permisos.*

Campo	Tipo de Dato	Tamaño	Llave	Nulo	Descripción / Atributos
id_perfil	INT	-	PK, FK	No	Referencia a tabla perfiles.
id_modulo	INT	-	PK, FK	No	Referencia a tabla módulos.

Fuente: Elaboración propia (2026).

Descripción: tabla intermedia (Muchos a Muchos) que mapea qué perfil tiene acceso a qué módulo.

Tabla 28.*Registro de asistencia.*

Campo	Tipo de Dato	Tamaño	Llave	Nulo	Descripción / Atributos
idRegistro	INT	-	PK	No	Identificador de la marca (AI).
Empleados_idEmpleados	INT	-	FK	No	Referencia al empleado.
fecha	DATE	-	-	No	Fecha física del servidor.
hora_marca	TIME	-	-	No	Hora capturada por <i>Python</i> .
tipo_marca	ENUM	-	-	No	('Entrada', 'Inicio Almuerzo', 'Fin Almuerzo', 'Salida').

Fuente: Elaboración propia (2026).

Descripción: captura inmutable de los pulsos del reloj marcador.

Tabla 29.*Solicitudes, ausencias.*

Campo	Tipo de Dato	Tamaño	Llave	Nulo	Descripción / Atributos
idSolicitudesAusencias	INT	-	PK	No	Identificador de solicitud (AI).
Empleados_idEmpleados	INT	-	FK	No	Empleado solicitante.
fecha_solicitud	DATETIME	-	-	No	Fecha de creación del registro.
fecha_inicio	DATE	-	-	No	Inicio del tiempo libre.
fecha_fin	DATE	-	-	No	Fin del tiempo libre.
tipo_ausencia	ENUM	-	-	No	('Vacaciones', 'Permiso con goce', 'Permiso sin goce', 'Incapacidad').
motivo	TEXT	-	-	No	Justificación redactada por el usuario.
estado	VARCHAR	20	-	No	Estado de aprobación (Pendiente, etc.).

Fuente: Elaboración propia (2026).

Descripción: almacena las peticiones de tiempo libre, vacaciones o permisos.

Tabla 30.*Solicitudes, horas extras.*

Campo	Tipo de Dato	Tamaño	Llave	Nulo	Descripción / Atributos
idSolicitudesHorasExtras	INT	-	PK	No	Identificador (AI).
Empleados_idEmpleados	INT	-	FK	No	Empleado que laboró la jornada extra.
fecha_solicitud	DATETIME	-	-	No	Momento en que se envió el formulario.
fecha_horas_extras	DATE	-	-	No	Fecha laborada a reclamar.
cantidad_horas	DECIMAL	4,2	-	No	Horas a facturar (Máx. 4.0).
justificacion	VARCHAR	500	-	No	Detalles y etiqueta de tarifa (Doble/Sencilla).
estado	VARCHAR	45	-	No	(Pendiente/Aprobado/Denegado).
fecha_aprobacion	DATE	-	-	Sí	Fecha de intervención de la jefatura.
id_jefe_aprobador	INT	-	FK	Sí	Referencia al usuario administrador.

Fuente: Elaboración propia (2026).

Descripción: bandeja de reporte de horas adicionales para aprobación gerencial.

Tabla 31.*Usuarios.*

Campo	Tipo de Dato	Tamaño	Llave	Nulo	Descripción / Atributos
idUsuarios	INT	-	PK	No	Identificador de credencial (AI).
email	VARCHAR	45	-	No	Correo institucional (<i>Username</i>).
password	VARCHAR	255	-	No	Contraseña de acceso.
Perfiles_id_perfil	INT	-	FK	No	Referencia al rol de seguridad.
Empleados_idEmpleados	INT	-	FK	No	Vínculo 1:1 con su expediente físico.

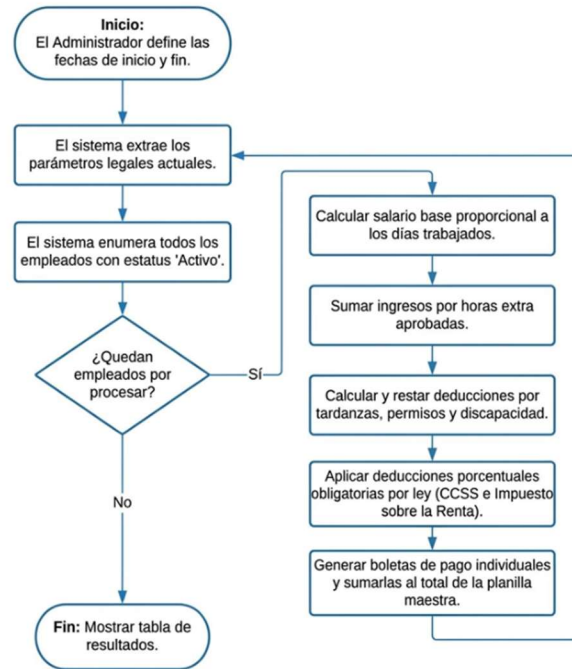
Fuente: Elaboración propia (2026).Descripción: entidad de autenticación (*Login*) separada del expediente por motivos de seguridad.**Diseño de Procesos**

El diseño de procesos describe la secuencia lógica e interna de las transacciones más complejas del prototipo. Debido al enfoque procedimental y centrado en los datos de la

arquitectura, estos procesos detallan la interacción secuencial entre las validaciones del controlador y la manipulación en la base de datos.

Figura 15.

Diseño de procesos.



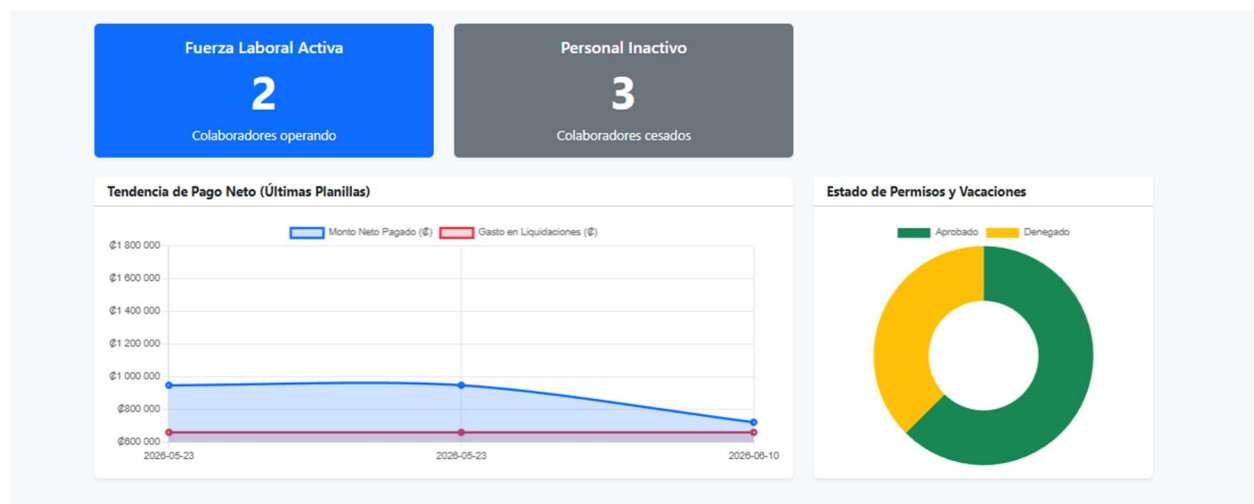
Fuente: Elaboración propia (2026).

Diseño de Salidas

El diseño de salidas corresponde a la forma en que el sistema procesa, consolida y presenta la información almacenada para facilitar la toma de decisiones administrativas y operativas. Las salidas del sistema Invet priorizan la claridad visual, la exportabilidad de los datos y el uso de indicadores clave de rendimiento (KPI's).

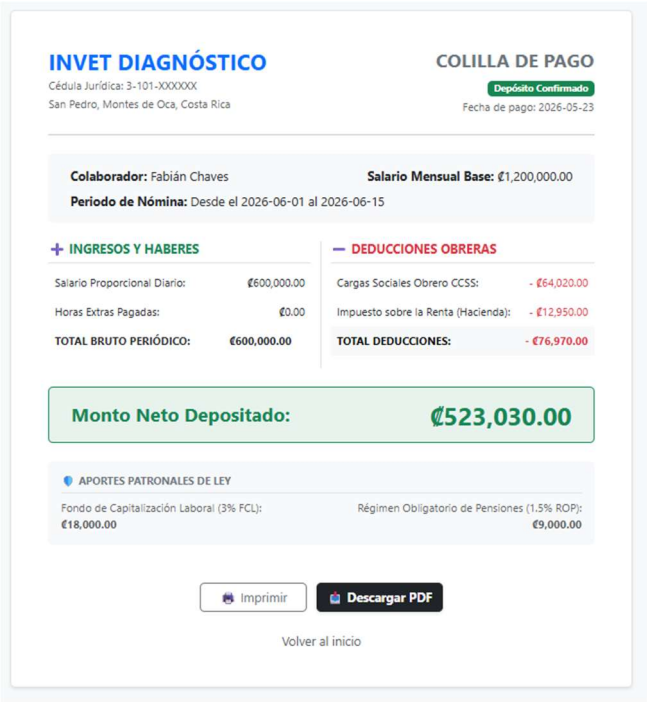
A continuación, se presentan cinco formatos representativos de las salidas del sistema:

1. ***Dashboard de Indicadores Gerenciales (Gráficas):*** representa una salida visual de alto nivel. Transforma los datos brutos de planillas, liquidaciones y ausentismo en gráficas interactivas mediante la librería Chart.js, permitiendo a la gerencia evaluar la salud financiera y operativa de la clínica en tiempo real.

Figura 16.*Dashboard de indicadores gerenciales.**Fuente:* Elaboración propia (2026).

2. **Colilla de Pago en Formato PDF:** salida de carácter legal y oficial. El sistema renderiza un documento inmutable y descargable que desglosa detalladamente los ingresos, las horas extras aplicadas y las deducciones obrero-patronales retenidas, cumpliendo con la necesidad de comprobantes físicos o digitales del colaborador.

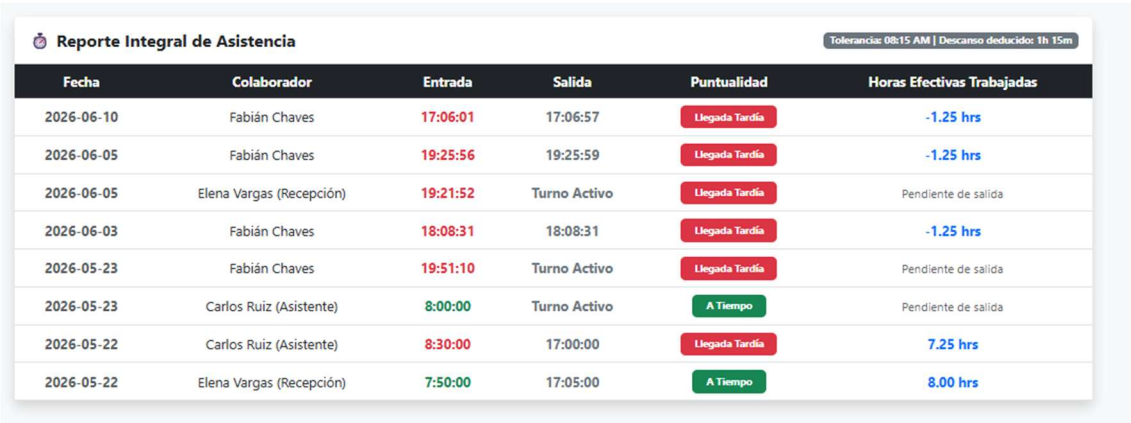
Figura 17.
Colilla de pago en formato PDF.



Fuente: Elaboración propia (2026).

- 3. **Reporte del Auditor de Asistencia:** salida analítica tabular. El sistema cruza las marcas de tiempo contra los parámetros de tolerancia institucionales y devuelve una tabla procesada que cuantifica las horas efectivas laboradas y resalta visualmente (en color rojo) los patrones de llegadas tardías.

Figura 18.
Reporte del auditor de asistencia.



Fuente: Elaboración propia (2026).

4. **Comprobante de Liquidación Laboral:** salida financiera precalculada. Una vez ejecutado el proceso de inhabilitación, el sistema despliega una pantalla de resumen con el desglose monetario proyectado para aguinaldo, vacaciones adeudadas, preaviso y cesantía, permitiendo al Departamento de Recursos Humanos extender el cheque de liquidación de forma inmediata.

Figura 19.

Comprobante de liquidación laboral.

 **Comprobante de Liquidación**

Colaborador: Elena Vargas (Recepción)

Motivo de Salida: Despido con Responsabilidad

Tiempo Laborado: 498 días

Concepto	Monto (€)
Aguinaldo Proporcional:	450,000.00
Vacaciones No Gozadas:	208,800.00
Preaviso:	450,000.00
Cesantía:	622,500.00

Total a Pagar: €1,731,300.00

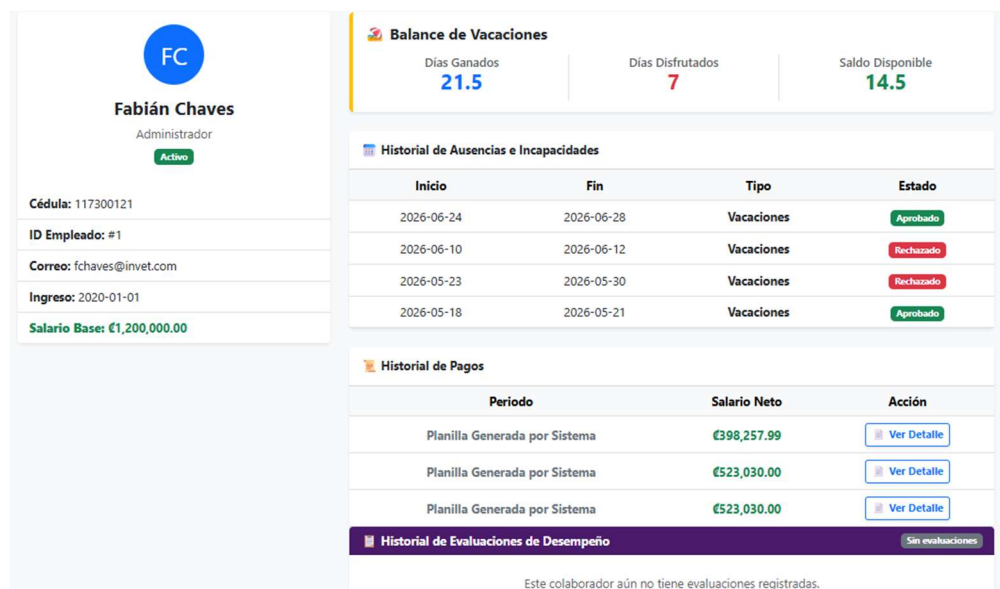
[Aceptar y Volver al Panel](#)

Fuente: Elaboración propia (2026).

5. **Vista Integral del Expediente del Colaborador:** salida de tipo "maestro-detalle". Agrupa y consolida toda la información distribuida del empleado en una sola vista, incluyendo su saldo en tiempo real de vacaciones, el promedio histórico de sus evaluaciones de desempeño y el historial completo de sus comprobantes de pago.

Figura 20.

Vista integral del expediente del colaborador.



Fuente: Elaboración propia (2026).

Diagramas UML

Dado el enfoque arquitectónico del sistema Invet, el cual emplea un patrón Modelo-Vista-Controlador (MVC) basado en funciones procedimentales y consultas SQL nativas, el modelado UML se enfoca en la implementación de Diagramas de Secuencia. Estos diagramas permiten representar, de manera gráfica y estructurada, la interacción cliente-servidor, ilustrando cómo la capa de presentación (Vista), la lógica de negocio (Controlador) y la capa de persistencia (Base de Datos) intercambian mensajes a lo largo del tiempo para concretar las transacciones y aplicar las reglas de validación del sistema.

A continuación, se detallan los diagramas de secuencia para tres de los procesos transaccionales más críticos del prototipo.

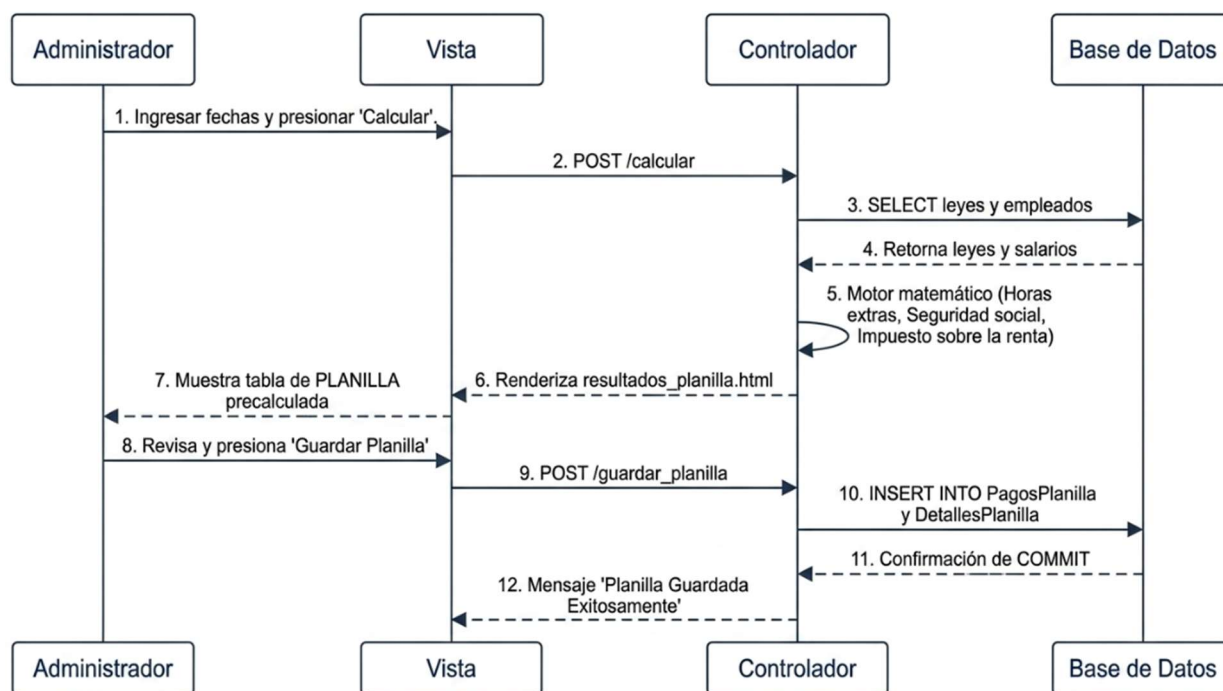
Generación de Planilla

El siguiente diagrama ilustra el flujo de control para el cálculo de la nómina. Este proceso representa la transacción matemática más compleja del sistema. El diagrama demuestra cómo actúa el controlador como orquestador al solicitar los parámetros de ley y la lista de empleados a la base de datos, para luego procesar los cálculos en la memoria del servidor (aplicando deducciones, sumando horas extras y validando incapacidades). Finalmente, tras la confirmación

de la jefatura en la vista previa, el controlador ejecuta una inserción estructurada almacenando el registro maestro y los detalles individuales de cada colaborador mediante un control transaccional seguro (*Commit*).

Figura 21.

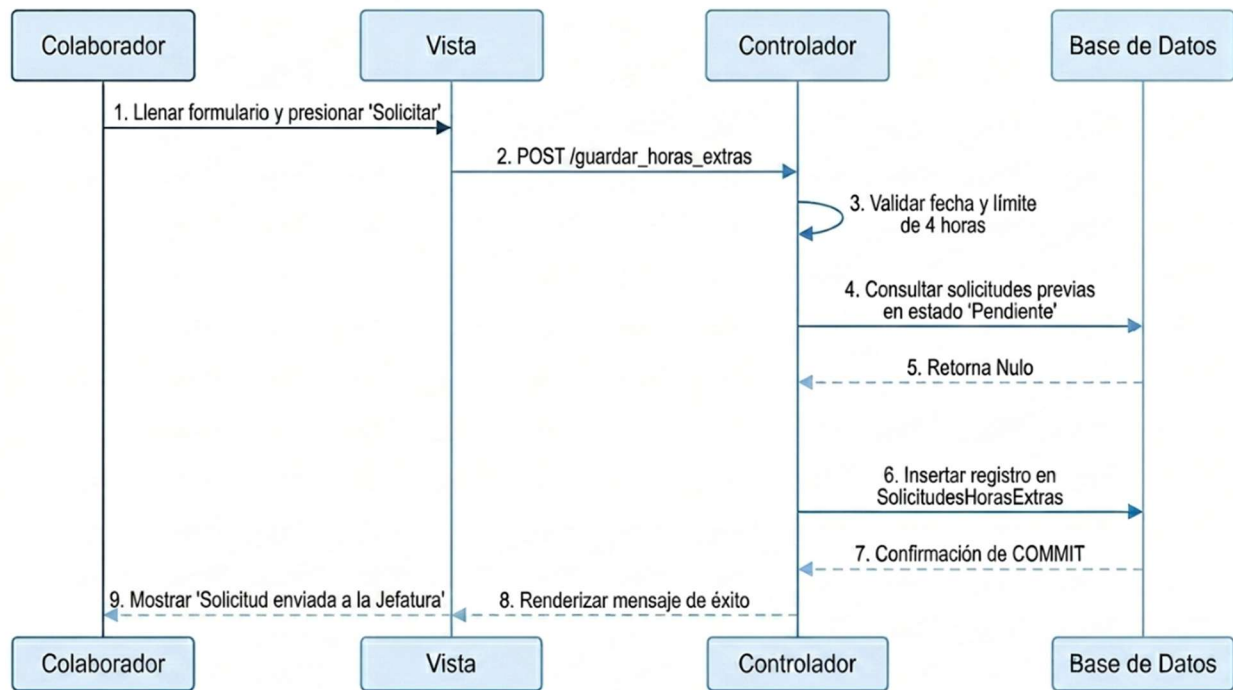
Diagrama de generación de planilla.



Fuente: Elaboración propia (2026).

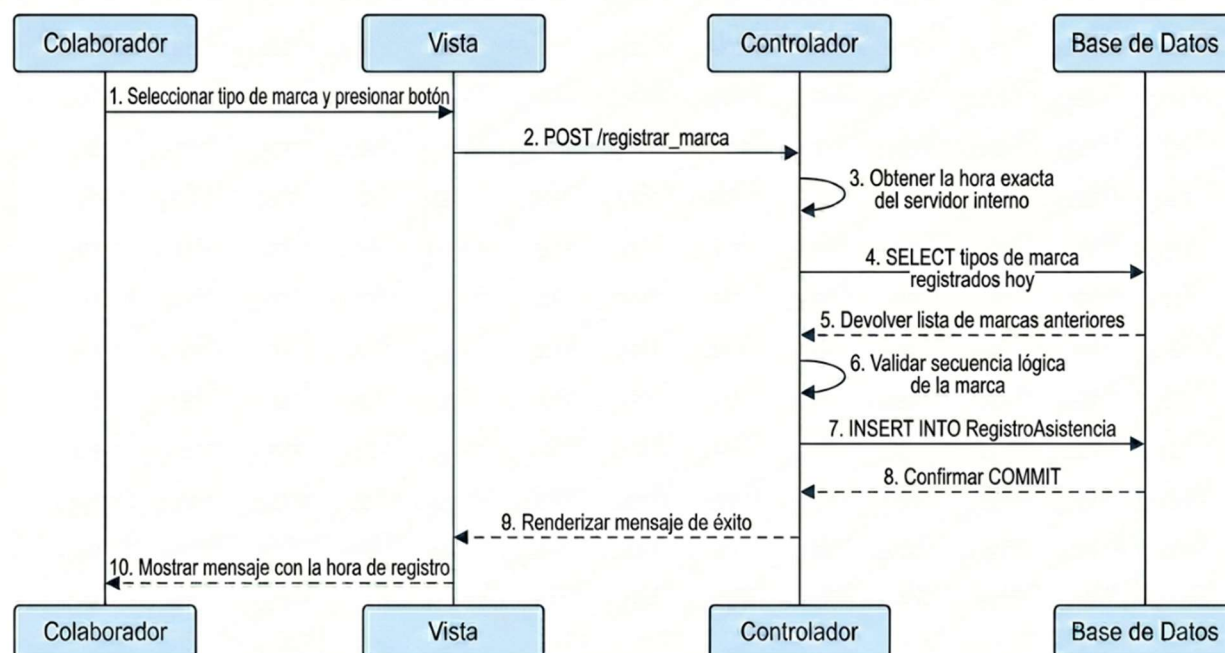
Solicitud de Horas Extras

El presente diagrama detalla el proceso de captura de solicitudes de tiempo extraordinario. El flujo destaca la aplicación del principio de seguridad perimetral (*Zero-Trust*) implementado en el controlador. Antes de interactuar con la base de datos para ejecutar la inserción, el sistema evalúa estrictas reglas de negocio en la memoria del servidor, tales como el bloqueo de fechas retroactivas y la restricción del límite máximo de cuatro horas diarias. Asimismo, demuestra la validación de integridad referencial al consultar la existencia de solicitudes previas en estado "Pendiente" para evitar la duplicidad de transacciones.

Figura 22.*Diagrama de solicitud de horas extras.**Fuente:* Elaboración propia (2026).

Registro de Asistencia

Este diagrama modela la lógica detrás del reloj marcador institucional. El diseño del flujo evidencia los controles de auditoría implementados para garantizar la integridad de las marcas de tiempo. Se ilustra cómo el controlador ignora la hora del dispositivo cliente y extrae el "timestamp" directamente del reloj del servidor para evitar manipulaciones. Adicionalmente, se observa el proceso de validación lógica donde el sistema consulta el historial de marcas de la jornada actual para asegurar la secuencia correcta (por ejemplo, impedir el registro de una "Salida" sin una "Entrada" previa) antes de consolidar el registro en la base de datos.

Figura 23.*Diagrama de registro de asistencia.**Fuente:* Elaboración propia (2026).

Programación

Una vez definida la arquitectura de la base de datos y estructurado el patrón de diseño Modelo-Vista-Controlador (MVC), la fase de construcción técnica del sistema se centró en la materialización de los requerimientos operativos de la clínica. Para lograr esto, el entorno de programación se apoyó sólidamente en el *framework Flask*, utilizando el lenguaje *Python* para la construcción de toda la lógica matemática y de transacciones en el *backend*. De manera paralela, las interfaces gráficas con las que interactuarán los diferentes perfiles de usuario fueron maquetadas empleando HTML5, CSS3, *JavaScript* y el entorno de trabajo *Bootstrap*, garantizando así un portal de autogestión responsivo y de alta usabilidad.

Durante esta etapa de codificación, el esfuerzo de ingeniería se enfocó en traducir la legislación laboral y las normativas administrativas internas en algoritmos computacionales precisos. Esto implicó la programación exhaustiva de controladores capaces de procesar de forma automatizada, las deducciones obrero-patronales, el cálculo de horas extras, el control de las vacaciones y la liquidación de extremos laborales. Asimismo, se implementó a nivel de código el motor de seguridad de Control de Acceso Basado en Roles (RBAC), asegurando

mediante validaciones estrictas en las rutas del servidor, que la navegación y la alteración de datos respeten la matriz de privilegios definida por la gerencia.

Para proveer una comprensión detallada de la arquitectura de *software* construida y evidenciar la complejidad de la lógica de negocio implementada, a continuación, se expone una serie de extractos visuales del código fuente. Estas capturas documentan la estructura sintáctica de los componentes más críticos del aplicativo web, ilustrando cómo se enlazan las vistas del *frontend* con los procesos del controlador y la interacción directa con el motor relacional *MySQL*.

Entradas

Las entradas del sistema capturan la información proveniente de los usuarios a través de peticiones HTTP, garantizando la correcta extracción y conversión de tipos de datos antes de su procesamiento.

Ejemplo 1.1: Captura Múltiple y Asignación de Variables de Sesión en Mantenimiento de Empleados.

En el registro de nuevos expedientes, el controlador extrae datos de diferentes naturalezas directamente del formulario y, adicionalmente, captura metadatos invisibles generados por el servidor, como la sesión activa del usuario que registra el movimiento.

Figura 24.

Captura múltiple y asignación de variables de sesión en Mantenimiento de Empleados.

```
@app.route('/guardar_empleado', methods=['POST'])
def guardar_empleado():
    # Evaluacion en la matriz de permisos para la insercion de datos
    if 'modulo_expedientes' not in session.get('permisos', []):
        return "Acceso denegado.", 403

    # Extraccion de la estructura enviada en el body del request POST
    cedula = request.form['cedula']
    nombre = request.form['nombre']
    apellidos = request.form['apellidos']
    salario_bruto = float(request.form['salario_bruto'])
    fecha_ingreso = request.form['fecha_ingreso']
    email = request.form['email']
    password = request.form['password']
    id_perfil = request.form['id_perfil']
```

Fuente: Elaboración propia (2026).

Ejemplo 1.2: Captura Dinámica de Múltiples Variables Para Parámetros de Ley.

El controlador es capaz de iterar dinámicamente sobre un diccionario de entradas variables utilizando el método `.items()`, lo cual permite procesar actualizaciones masivas de configuración fiscal sin importar cuántos parámetros se envíen desde la interfaz.

Figura 25.

Captura dinámica de múltiples variables para Parámetros de Ley.

```
@app.route('/actualizar_parametros', methods=['POST'])
def actualizar_parametros():
    if 'modulo_parametros' not in session.get('permisos', []):
        return "Acceso denegado.", 403

    conexion = conectar_bd()
    if conexion:
        cursor = conexion.cursor()
        try:
            # Iteracion sobre el diccionario de payload del POST request
            for parametro, nuevo_valor in request.form.items():
                cursor.execute("UPDATE ConfiguracionLey SET valor = %s WHERE nombre_parametro = %s",
                               (nuevo_valor, parametro))

            accion = "Modificó los valores porcentuales de los parámetros de ley"
            registrar_log(session['nombre_completo'], accion, "Parámetros de Ley")

        conexion.commit()
```

Fuente: Elaboración propia (2026).

Salidas

Las salidas gestionan cómo la información procesada en el *backend* es empaquetada, serializada o convertida para ser devuelta y consumida por el navegador web del usuario final.

Ejemplo 2.1: Serialización de Datos JSON Para Indicadores Gerenciales.

Para la construcción del *Dashboard* Gerencial, los arreglos extraídos de la base de datos son convertidos a formato JSON. Esto permite que las librerías gráficas de *JavaScript* (como Chart.js) en la Vista puedan interpretar correctamente los ejes y valores históricos.

Figura 26.

Serialización de datos JSON para Indicadores Gerenciales.

```
import json
return render_template('indicadores.html',
                       fechas_json=json.dumps(fechas_planilla),
                       montos_json=json.dumps(montos_planilla),
                       fechas_liq_json=json.dumps(fechas_liq),
                       montos_liq_json=json.dumps(montos_liq),
                       etiq_aus_json=json.dumps(etiquetas_ausencias),
                       tot_aus_json=json.dumps(totales_ausencias),
                       empleados_activos=total_empleados_activos,
                       empleados_inactivos=total_empleados_inactivos)
```

Fuente: Elaboración propia (2026).

Ejemplo 2.2: Generación de Salidas Inmutables (Descarga de Colillas en PDF).

El sistema procesa una plantilla HTML renderizada en memoria (utilizando BytesIO) y la convierte a un documento PDF. Posteriormente, modifica las cabeceras HTTP de respuesta para forzar la descarga del archivo en el equipo del usuario.

Figura 27.

Generación de salidas inmutables (descarga de colillas en PDF).

```
# Integración lógica que procesa el HTML del template renderizado para conversión final
html = render_template('ver_recibo.html', r=recibo, modo_pdf=True)

# Buffer de memoria intermedia (BytesIO) evita la creación física del elemento en el server
pdf_file = BytesIO()

# Transformador por medio de módulo externo para generación dinámica del PDF
pisa_status = pisa.CreatePDF(BytesIO(html.encode('utf-8')), dest=pdf_file)

# Constructor de la respuesta HTTP final incluyendo el mimetype adecuado y forzando archivo de descarga adjunto
if not pisa_status.err:
    respuesta = make_response(pdf_file.getvalue())
    respuesta.headers['Content-Type'] = 'application/pdf'
    respuesta.headers['Content-Disposition'] = f'attachment; filename=Colilla_Invet_PLN_{recibo["id_pago_planilla"]}.pdf'
    return respuesta
```

Fuente: Elaboración propia (2026).

Procesos

Esta sección evidencia la implementación de la lógica de negocio, los algoritmos matemáticos en memoria y las sentencias SQL orientadas al procesamiento de datos.

Ejemplo 3.1: Algoritmo de Cálculo de Liquidaciones.

Implementación en *Python* de la matemática estipulada en el Código de Trabajo de Costa Rica. El sistema calcula proporciones de fechas y determina, mediante condicionales, si el colaborador es acreedor de extremos laborales, como preaviso y cesantía.

Figura 28.*Algoritmo de cálculo de liquidaciones.*

```

# Inicio del subproceso matematico de derecho laboral
salario_mensual = float(empleado['salario_bruto'])
salario_diario = salario_mensual / 30

# Estimacion de factores fraccionarios en base a los metadatos suministrados
from datetime import datetime
formato = "%Y-%m-%d"
d_ingreso = datetime.strptime(fecha_ingreso, formato)
d_salida = datetime.strptime(fecha_salida, formato)
dias_totales = (d_salida - d_ingreso).days
meses_totales = dias_totales / 30

# Algoritmo Base 1: Calculo estandar simplificado del extremo del aguinaldo
monto_aguinaldo = (salario_mensual / 12) * min(meses_totales, 12)

# Algoritmo Base 2: Proyeccion de vacaciones no gozadas segun factor constitucional
dias_vacaciones_adeudados = min(meses_totales, 12) * 1.16
monto_vacaciones = dias_vacaciones_adeudados * salario_diario

# Algoritmo Base 3: Validacion condicional orientada a las politicas de despido
monto_preaviso = 0.0
monto_cesantia = 0.0

if motivo_salida == 'Despido con Responsabilidad':
    if meses_totales >= 12:
        monto_preaviso = salario_mensual
        monto_cesantia = salario_mensual * (meses_totales / 12)
    elif meses_totales >= 3:
        monto_preaviso = salario_diario * 15
        monto_cesantia = salario_diario * 7

```

Fuente: Elaboración propia (2026).**Ejemplo 3.2: Delegación de Procesos Matemáticos a la Base de Datos (Auditoría).**

Para optimizar el rendimiento, el procesamiento de las marcas de asistencia se delegó al motor relacional mediante agrupaciones lógicas y evaluaciones condicionales anidadas (IF), calculando variables complejas directamente desde el lenguaje SQL.

Figura 29.

Delegación de procesos matemáticos a la base de datos (auditoría).

```

consulta = """
SELECT
    r.fecha,
    e.nombre,
    e.apellidos,
    MIN(CASE WHEN r.tipo_marca = 'Entrada' THEN r.hora_marca END) AS hora_entrada,
    MAX(CASE WHEN r.tipo_marca = 'Salida' THEN r.hora_marca END) AS hora_salida,

    -- Clasificación según tiempo oficial del parametro
    IF(MIN(CASE WHEN r.tipo_marca = 'Entrada' THEN r.hora_marca END) > '08:15:00', 'Llegada Tardía', 'A Tiempo') AS estado_auditoria,
    IF(MIN(CASE WHEN r.tipo_marca = 'Entrada' THEN r.hora_marca END) > '08:15:00', 'danger', 'success') AS color_alerta,

    -- Operaciones en BD: diferencia en minutos deduciendo margenes de descanso|
    ROUND(
        (TIMESTAMPDIFF(MINUTE,
            MIN(CASE WHEN r.tipo_marca = 'Entrada' THEN r.hora_marca END),
            MAX(CASE WHEN r.tipo_marca = 'Salida' THEN r.hora_marca END)
        ) - 75) / 60.0,
        2) AS horas_efectivas

FROM RegistroAsistencia r
JOIN Empleados e ON r.Empleados_idEmpleados = e.idEmpleados
GROUP BY r.fecha, r.Empleados_idEmpleados, e.nombre, e.apellidos
ORDER BY r.fecha DESC
"""

```

Fuente: Elaboración propia (2026).

Validaciones

Las validaciones aplican el principio de seguridad "*Zero-Trust*", rechazando solicitudes lógicamente inconsistentes, manipuladas o fraudulentas a nivel de *backend* antes de comprometer la integridad del modelo de datos.

Ejemplo 4.1: Validación de Reglas de Negocio Mediante Iteración (Módulo de Ausencias).

El algoritmo evalúa un rango de fechas y ejecuta un proceso matemático que excluye los días no laborables (sábados y domingos). Posteriormente, audita si el saldo acumulado en la base de datos es suficiente para cubrir la solicitud enviada por el colaborador.

Figura 30.*Validación de reglas de negocio mediante iteración (módulo de ausencias).*

```

# Validación del tipo de solicitud para la aplicación de filtros lógicos orientados a vacaciones
if tipo_ausencia == 'Vacaciones':
    formato = "%Y-%m-%d"
    d_inicio = datetime.strptime(fecha_inicio, formato)
    d_fin = datetime.strptime(fecha_fin, formato)

    # Algoritmo de filtrado de días no laborales durante el rango solicitado
    dias_pedidos = 0
    dia_actual = d_inicio
    while dia_actual <= d_fin:
        # Modificador indexado de días; se descartan valores mayores o iguales a 5 (Sábado y Domingo)
        if dia_actual.weekday() < 5:
            dias_pedidos += 1
            dia_actual += timedelta(days=1)

    if dias_pedidos == 0:
        conexion.close()
        return "Error Lógico: Las fechas seleccionadas solo abarcan días no hábiles (fines de semana).", 400

    # Extracción del flotante que representa el acumulado del empleado según la BD
    cursor.execute("SELECT dias_vacaciones_disponibles FROM Empleados WHERE idEmpleados = %s", (id_empleado,))
    empleado_db = cursor.fetchone()
    saldo_actual = float(empleado_db['dias_vacaciones_disponibles']) if empleado_db else 0.0

    # Evaluación del disponible vs el solicitado para evitar saldos negativos
    if dias_pedidos > saldo_actual:
        conexion.close()
        return f"""
        <div style="text-align: center; margin-top: 50px; font-family: sans-serif;">
        <h1 style="color: #dc3545;">✖ Solicitud Rechazada</h1>
        <p style="font-size: 18px;">Estás intentando solicitar <b>{dias_pedidos}</b> días de vacaciones, pero tu saldo actual es de solo <b>{saldo_actual:.2f}</b> días.</p>
        <br>
        <a href="/ausencias" style="padding: 10px 20px; background: #6c757d; color: white; text-decoration: none; border-radius: 5px;">Volver a intentarlo</a>
        </div>
        """, 400

```

Fuente: Elaboración propia (2026).**Ejemplo 4.2: Validación Secuencial de Estado (Reloj Marcador).**

El sistema intercepta la intención del usuario y extrae su historial de la jornada en tiempo real. Mediante un árbol de decisiones (if/elif), previene el registro de transacciones que violan el flujo laboral natural (como marcar salida de almuerzo sin haberlo iniciado).

Figura 31.*Validación secuencial de estado (reloj marcador).*

```

# Query para mapear las acciones que el empleado ya ha registrado en la fecha presente
cursor.execute("""
    SELECT tipo_marca
    FROM RegistroAsistencia
    WHERE Empleados_idEmpleados = %s AND fecha = %s
""", (id_empleado, fecha_actual))

marcas_hoy = [fila['tipo_marca'] for fila in cursor.fetchall()]

print(f"Marcas de hoy: {marcas_hoy} | Intentando marcar: {tipo_marca}")

# Estructuras condicionales de validacion secuencial para integridad de las marcas
if 'Salida' in marcas_hoy:
    return render_template('reloj.html', mensaje_error="✗ Tu jornada ya ha finalizado. No puedes registrar más marcas hoy.")

if tipo_marca in marcas_hoy:
    return render_template('reloj.html', mensaje_error=f"✗ Ya registraste tu '{tipo_marca}' el día de hoy.")

if tipo_marca == 'Inicio Almuerzo' and 'Entrada' not in marcas_hoy:
    return render_template('reloj.html', mensaje_error="✗ Debes marcar tu 'Entrada' antes de salir a almorzar.")

if tipo_marca == 'Fin Almuerzo' and 'Inicio Almuerzo' not in marcas_hoy:
    return render_template('reloj.html', mensaje_error="✗ No puedes finalizar tu almuerzo si no lo has iniciado.")

if tipo_marca == 'Salida':
    if 'Entrada' not in marcas_hoy:
        return render_template('reloj.html', mensaje_error="✗ No puedes marcar tu 'Salida' si nunca registraste la 'Entrada'.")
    if 'Inicio Almuerzo' in marcas_hoy and 'Fin Almuerzo' not in marcas_hoy:
        return render_template('reloj.html', mensaje_error="✗ Tienes un almuerzo en curso. Debes finalizarlo antes de marcar tu salida.")

```

Fuente: Elaboración propia (2026).

Módulos Señalados en el Alcance

A continuación, se presentan fragmentos del código fuente que corresponden a los principales módulos transaccionales definidos en el alcance del proyecto. Cada extracto está acompañado de una justificación técnica que explica la lógica de negocio implementada, el manejo de datos en el servidor y las sentencias relacionales utilizadas para garantizar la integridad del sistema.

5.1. Módulo: Pagar Planilla.

Este módulo representa el núcleo financiero del sistema. A nivel de ingeniería, destaca por la implementación de transacciones relacionales de tipo "Maestro-Detalle". Para garantizar el principio de atomicidad en la base de datos, el controlador deserializa un paquete JSON precalculado, genera un registro maestro (padre) y, utilizando el identificador autogenerado por el motor SQL (`cursor.lastrowid`), vincula iterativamente todas las colillas individuales (hijos).

Figura 32.*Módulo: Pagar planilla.*

```

@app.route('/guardar_planilla', methods=['POST'])
def guardar_planilla():
    # Validación de seguridad para evitar ejecuciones externas
    if session.get('perfil') != 'Administrador':
        return "Acceso denegado.", 403

    fecha_inicio = request.form['fecha_inicio']
    fecha_fin = request.form['fecha_fin']
    total_deducciones = float(request.form.get('total_deducciones') or 0.0)
    total_neto = float(request.form.get('total_neto') or 0.0)

    # Deserialización del payload recibido desde el input hidden
    import json
    paquete_recibos = request.form['recibos_json']
    lista_recibos = json.loads(paquete_recibos)

    # Cálculo de la provisión de días de vacaciones (14 días / 360 al año comercial)
    from datetime import datetime
    d_inicio = datetime.strptime(fecha_inicio, "%Y-%m-%d")
    d_fin = datetime.strptime(fecha_fin, "%Y-%m-%d")
    dias_trabajados = (d_fin - d_inicio).days + 1
    fraccion_vacaciones = dias_trabajados * (14.0 / 360.0)

    conexion = conectar_bd()
    if conexion:
        cursor = conexion.cursor()
        try:
            # Inserción del encabezado en la tabla padre
            consulta_maestro = """
                INSERT INTO PagosPlanilla
                (fecha_inicio_periodo, fecha_fin_periodo, fecha_pago_efectivo, descripcion, total_deducciones, total_pagado_neto)
                VALUES (%s, %s, CURDATE(), %s, %s, %s)
            """
            cursor.execute(consulta_maestro, (fecha_inicio, fecha_fin, "Planilla Generada por Sistema", total_deducciones, total_neto))
            id_planilla = cursor.lastrowid

```

Fuente: Elaboración propia (2026).

5.2. Módulo: Gestionar Horas Extras.

Este módulo opera bajo un esquema de autoservicio. El diseño arquitectónico aplica el principio de "Zero-Trust" (Cero Confianza) al extraer la identidad del usuario directamente desde la variable de sesión cifrada del *backend* (`session['id_empleado']`), mitigando riesgos de suplantación de identidad. Adicionalmente, aplica reglas de validación temporal y de límites paramétricos antes de ejecutar la inserción en la base de datos.

Figura 33.*Módulo: Gestionar horas extras.*

```

@app.route('/guardar_horas_extras', methods=['POST'])
def guardar_horas_extras():
    if 'modulo_horas_extras' not in session.get('permisos', []):
        return "Acceso denegado.", 403

    # Extracción del identificador del empleado usando la variable backend protegida
    id_empleado = session['id_empleado']

    fecha_registro = request.form['fecha_registro']
    cantidad_horas = float(request.form['cantidad_horas'])
    tipo_hora = request.form['tipo_hora']
    motivo_usuario = request.form['justificacion']

    from datetime import date
    hoy = date.today().strftime('%Y-%m-%d')

    # Validación de integridad referencial: bloqueo de solicitudes asíncronas retroactivas
    if fecha_registro < hoy:
        return "Error de Seguridad: No se permite registrar horas extras con fechas en el pasado.", 400

    # Regla de restricción de límites máximos parametrizados para las jornadas extra
    if cantidad_horas <= 0 or cantidad_horas > 4:
        return f"""
        <div style="text-align: center; margin-top: 50px; font-family: sans-serif;">
        <h1 style="color: #d35455;">✖ Error de Validación</h1>
        <p style="font-size: 18px;">Estás intentando solicitar <b>{cantidad_horas}</b> horas, pero la política de la empresa solo permite un máximo de <b>4</b> horas extras diarias</p>
        <br>
        <a href="/horas_extras" style="padding: 10px 20px; background: #6c757d; color: white; text-decoration: none; border-radius: 5px;">Volver a intentarlo</a>
        </div>
        """
        return 400

```

Fuente: Elaboración propia (2026).

5.3. Módulo: Gestionar Permisos (Ausencias).

El gestor de permisos funciona como la capa de resolución administrativa. Este fragmento demuestra la recepción del veredicto emitido por la jefatura y la actualización de la entidad en la base de datos. Destaca la inclusión de controles lógicos que previenen cambios de estado en transacciones que ya fueron previamente aprobadas.

Figura 34.*Módulo: Gestionar permisos (ausencias).*

```

@app.route('/procesar_ausencia', methods=['POST'])
def procesar_ausencia():
    # Restricción a roles administrativos
    if session.get('perfil') not in ['Administrador', 'RRHH', 'Supervisor']:
        return "Acceso denegado.", 403

    id_solicitud = request.form['id_solicitud']
    nuevo_estado = request.form['nuevo_estado']

    conexion = conectar_bd()
    if conexion:
        cursor = conexion.cursor(dictionary=True)
        try:
            cursor.execute("SELECT Empleados_idEmpleados, tipo_ausencia, fecha_inicio, fecha_fin, estado FROM SolicitudesAusencias WHERE idSolicitudesAusencias = %s", (id_solicitud,))
            solicitud = cursor.fetchone()

            # Regla de negocio: Si se aprueban vacaciones, ejecutar el debito en el saldo del empleado
            if nuevo_estado == 'Aprobado' and solicitud['estado'] != 'Aprobado' and solicitud['tipo_ausencia'] == 'Vacaciones':

```

5.4. Módulo: Gestionar Vacaciones.

Íntimamente ligado al módulo de permisos, la gestión de vacaciones implica matemáticas dinámicas sobre calendarios. Cuando un permiso tipificado como "Vacaciones" es aprobado, el controlador activa un iterador que recorre el rango de fechas solicitado, evaluando

el índice del día (weekday()) para excluir matemáticamente los fines de semana del rebajo, garantizando la equidad laboral.

Figura 35.

Módulo: Gestionar vacaciones.

```
from datetime import timedelta

dias_a_rebajar = 0
dia_actual = solicitud['fecha_inicio']

# Iteracion iterativa evaluando cada día del rango para excluir sabados(5) y domingos(6)
while dia_actual <= solicitud['fecha_fin']:
    if dia_actual.weekday() < 5:
        dias_a_rebajar += 1
        dia_actual += timedelta(days=1)

    cursor.execute("""
        UPDATE Empleados
        SET dias_vacaciones_disponibles = dias_vacaciones_disponibles - %s
        WHERE idEmpleados = %s
    """, (dias_a_rebajar, solicitud['Empleados_idEmpleados']))

# Actualizacion del estado definitivo del registro
cursor.execute("UPDATE SolicitudesAusencias SET estado = %s WHERE idSolicitudesAusencias = %s", (nuevo_estado, id_solicitud))
conexion.commit()
```

Fuente: Elaboración propia (2026).

5.5. Módulo: Gestionar Liquidaciones.

Este controlador automatiza el cierre del ciclo laboral. A nivel lógico, procesa conversiones de tipo de datos para calcular los extremos legales proporcionales y, en paralelo, previene bloqueos del sistema interceptando secuencias cronológicas imposibles (como una fecha de salida anterior a la de ingreso).

Figura 36.*Módulo: Gestionar liquidaciones.*

```

@app.route('/calcular_liquidacion', methods=['POST'])
def calcular_liquidacion():
    if 'modulo_liquidaciones' not in session.get('permisos', []):
        return "Acceso denegado.", 403

    id_empleado = request.form['id_empleado']
    fecha_ingreso = request.form['fecha_ingreso']
    fecha_salida = request.form['fecha_salida']
    motivo_salida = request.form['motivo_salida']

    # Manejo de error para secuencias cronologicas imposibles
    if fecha_salida <= fecha_ingreso:
        return "Error Lógico: La fecha de salida debe ser mayor a la fecha de ingreso.", 400

    conexion = conectar_bd()
    if conexion:
        cursor = conexion.cursor(dictionary=True)
        cursor.execute("SELECT nombre, apellidos, salario_bruto FROM Empleados WHERE idEmpleados = %s", (id_empleado,))
        empleado = cursor.fetchone()

        # Inicio del subproceso matematico de derecho laboral
        salario_mensual = float(empleado['salario_bruto'])
        salario_diario = salario_mensual / 30

```

Fuente: Elaboración propia (2026).

5.6. Módulo: Gestionar incapacidades.

A diferencia de las ausencias estándar, las incapacidades legales requieren un *bypass* administrativo. El código demuestra cómo el controlador, reservado exclusivamente para perfiles de Recursos Humanos, inyecta la solicitud forzando (*hardcoding*) la variable del estado inicial "Aprobado", asegurando que la incapacidad se descuenta, sin demoras, en la próxima ejecución de planillas.

Figura 37.*Módulo: Gestionar incapacidades.*

```

@app.route('/guardar_incapacidad', methods=['POST'])
def guardar_incapacidad():
    if session.get('perfil') not in ['Administrador', 'RRHH', 'Supervisor']:
        return "Acceso denegado.", 403

    id_empleado = request.form['id_empleado']
    fecha_inicio = request.form['fecha_inicio']
    fecha_fin = request.form['fecha_fin']
    motivo = request.form['motivo']

    # Asignacion del estado de forma explicita en la transaccion debido a la intervencion de RRHH
    tipo_ausencia = "Incapacidad"
    estado_inicial = "Aprobado"

    conexion = conectar_bd()
    if conexion:
        cursor = conexion.cursor()
        try:
            consulta = """
                INSERT INTO SolicitudesAusencias
                (Empleados_idEmpleados, fecha_solicitud, fecha_inicio, fecha_fin, tipo_ausencia, motivo, estado)
                VALUES (%s, NOW(), %s, %s, %s, %s, %s)
            """
            cursor.execute(consulta, (id_empleado, fecha_inicio, fecha_fin, tipo_ausencia, motivo, estado_inicial))
            conexion.commit()

```

Fuente: Elaboración propia (2026).

5.7. Módulo: Gestionar Aguinaldo.

El cálculo del aguinaldo se procesa de manera dinámica integrándose como una subrutina matemática dentro de los flujos financieros de liquidación. El algoritmo utiliza la librería *datetime*, nativa de *Python*, para transformar los días efectivos laborados en un factor proporcional, aplicando la fórmula legal establecida.

Figura 38.*Módulo: Gestionar aguinaldo.*

```

# Estimacion de factores fraccionarios en base a los metadatos suministrados
from datetime import datetime
formato = "%Y-%m-%d"
d_ingreso = datetime.strptime(fecha_ingreso, formato)
d_salida = datetime.strptime(fecha_salida, formato)
dias_totales = (d_salida - d_ingreso).days
meses_totales = dias_totales / 30

# Algoritmo Base 1: Calculo estandar simplificado del extremo del aguinaldo
monto_aguinaldo = (salario_mensual / 12) * min(meses_totales, 12)

```

Fuente: Elaboración propia (2026).

5.8. Módulo: Evaluar empleado.

Este controlador aborda el procesamiento de métricas cualitativas. Captura desde la vista web, una matriz de cinco criterios ponderados y su respectiva retroalimentación de texto, mapeando y transformando las entradas alfanuméricas en enteros para consolidar un *scoring* final persistente en el modelo de datos.

Figura 39.

Módulo: Evaluar empleado.

```
# Mapeo y extraccion del sistema matricial enviado en el formulario
c_puntualidad = int(request.form['c_puntualidad'])
c_disponibilidad = int(request.form['c_disponibilidad'])
c_productividad = int(request.form['c_productividad'])
c_equipo = int(request.form['c_equipo'])
c_iniciativa = int(request.form['c_iniciativa'])

# Recoleccion del scoring consolidado general para registro persistente
puntuacion = float(request.form['puntuacion'])
comentarios = request.form['comentarios']
evaluador = session.get('nombre_completo', 'Sistema')

conexion = conectar_bd()
if conexion:
    cursor = conexion.cursor()
    try:
        consulta = """
        INSERT INTO Evaluaciones
        (Empleados_idEmpleados, fecha_evaluacion, criterio_puntualidad, criterio_disponibilidad,
        criterio_productividad, criterio_equipo, criterio_iniciativa, puntuacion_total, comentarios, evaluado_por)
        VALUES (%s, %s, %s, %s, %s, %s, %s, %s, %s, %s)
        """
        cursor.execute(consulta, (id_empleado, fecha_evaluacion, c_puntualidad, c_disponibilidad,
        c_productividad, c_equipo, c_iniciativa, puntuacion, comentarios, evaluador))
        conexion.commit()
```

Fuente: Elaboración propia (2026).

5.9. Módulo: Mantenimiento.

En cumplimiento de las mejores prácticas de integridad referencial, el mantenimiento de perfiles evita el uso de sentencias destructivas (*DELETE*). Este bloque de código implementa un "Soft Delete" o borrado lógico mediante un operador ternario de *Python*, el cual invierte el estado actual del empleado en la base de datos sin destruir su historial financiero.

Figura 40.*Módulo: Mantenimiento.*

```
# Endpoint funcional que actua como toggle o flip switch para las habilitaciones
@app.route('/cambiar_estado_empleado', methods=['POST'])
def cambiar_estado_empleado():
    if session.get('perfil') not in ['Administrador', 'RRHH']:
        return "Acceso denegado.", 403

    id_empleado = request.form['id_empleado']
    estado_actual = request.form['estado_actual']

    # Asignacion ternaria del nuevo estado dependiente del valor preexistente
    nuevo_estado = "Inactivo" if estado_actual == "Activo" else "Activo"

    conexion = conectar_bd()
    if conexion:
        cursor = conexion.cursor()
        try:
            cursor.execute("UPDATE Empleados SET estado = %s WHERE idEmpleados = %s", (nuevo_estado, id_empleado))
            accion = f"Cambió el estado del empleado ID #{id_empleado} a '{nuevo_estado}'"
            registrar_log(session['nombre_completo'], accion, "Expedientes de Personal")
            conexion.commit()
        except:
```

Fuente: Elaboración propia (2026).

5.10. Módulo: Consultas (Expediente Integral).

Para consolidar la vista panorámica del empleado, el sistema interroga a la base de datos empleando sentencias relacionales nativas (JOIN). Este fragmento muestra cómo el controlador une las entidades de Empleados, Usuarios y Perfiles en una sola petición al servidor para armar la cabecera del expediente.

Figura 41.*Módulo: Consultas (expediente integral).*

```
@app.route('/expediente/<int:id_empleado>')
def ver_expediente(id_empleado):
    if 'modulo_expedientes' not in session.get('permisos', []):
        return redirect('/')

    conexion = conectar_bd()
    if conexion:
        cursor = conexion.cursor(dictionary=True)

        # Bloque Consultivo 1: Entidad de Perfil Maestro
        cursor.execute("""
            SELECT e.*, u.email, p.nombre_perfil
            FROM Empleados e
            LEFT JOIN Usuarios u ON e.idEmpleados = u.Empleados_idEmpleados
            LEFT JOIN Perfiles p ON u.Perfiles_id_perfil = p.id_perfil
            WHERE e.idEmpleados = %s
            """, (id_empleado,))
        empleado = cursor.fetchone()
```

Fuente: Elaboración propia (2026).

5.10. Módulo: Reportes (Auditoría).

La generación de reportes delega la carga computacional pesada al motor de base de datos relacional. El siguiente extracto de código demuestra el uso de funciones de agregación (MIN, MAX) y sentencias lógicas integradas (IF, CASE) en lenguaje SQL para determinar y etiquetar dinámicamente las llegadas tardías del personal.

Figura 42.

Modulo: Reportes (auditoría).

```
@app.route('/auditor')
def auditor_asistencia():
    # El perfil determina la visualizacion de datos criticos
    if session.get('perfil') not in ['Administrador', 'RRHH', 'Supervisor']:
        return redirect('/')

    conexion = conectar_bd()
    if conexion:
        cursor = conexion.cursor(dictionary=True)

        # Query de alto nivel para extraccion pivotada:
        # Cruza los timestamps, agrupa por empleado y fecha, y determina tardancias (condicional de > 08:15)
        consulta = """
        SELECT
            r.fecha,
            e.nombre,
            e.apellidos,
            MIN(CASE WHEN r.tipo_marca = 'Entrada' THEN r.hora_marca END) AS hora_entrada,
            MAX(CASE WHEN r.tipo_marca = 'Salida' THEN r.hora_marca END) AS hora_salida,

            -- Clasificacion segun tiempo oficial del parametro
            IF(MIN(CASE WHEN r.tipo_marca = 'Entrada' THEN r.hora_marca END) > '08:15:00', 'Llegada Tardía', 'A Tiempo') AS estado_auditoria,
```

Fuente: Elaboración propia (2026).

5.11. Módulo: Seguridad (Roles y Permisos).

La gestión de matrices de Control de Acceso Basado en Roles (RBAC), emplea una técnica de reestructuración iterativa. Ante un cambio de directrices en la vista, el *backend* borra transaccionalmente la configuración antigua (*DELETE*) y, mediante un *bucle for*, siembra en la base de datos la nueva matriz recibida vía *POST*.

Figura 43.*Módulo: Seguridad (roles y permisos).*

```

@app.route('/actualizar_permisos_rol', methods=['POST'])
def actualizar_permisos_rol():
    if 'modulo_llaves_roles' not in session.get('permisos', []):
        return "Acceso denegado.", 403

    id_perfil = request.form['id_perfil']
    # Analisis del payload por form.getlist para atrapar una matriz de elementos provenientes de checkboxes
    modulos_seleccionados = request.form.getlist('modulos')

    con (variable) conexion: PooledMySQLConnection | MySQLConnectionAbstract | None
    if conexion:
        cursor = conexion.cursor()
        try:
            # Limpieza: Borrado transaccional inicial de los privilegios correspondientes al perfil a modificar
            cursor.execute("DELETE FROM Permisos WHERE id_perfil = %s", (id_perfil,))

            # Bucle de re-asignacion de accesos utilizando los ids seleccionados en el frontend
            for id_modulo in modulos_seleccionados:
                cursor.execute("""
                    INSERT INTO Permisos (id_perfil, id_modulo)
                    VALUES (%s, %s)
                """, (id_perfil, id_modulo))

            accion = f"Modificó la matriz de accesos para el rol ID #{id_perfil}"
            registrar_log(session['nombre_completo'], accion, "Seguridad y Roles")
            conexion.commit()

```

Fuente: Elaboración propia (2026).

CAPÍTULO VI: CONCLUSIONES Y RECOMENDACIONES

Conclusiones

A partir del desarrollo del Trabajo Final de Graduación y la culminación del sistema web para la gestión de planillas, se derivan conclusiones fundamentales estructuradas en correspondencia directa con los objetivos de la investigación. En primer lugar, el análisis exhaustivo de los requerimientos funcionales y no funcionales del sistema web permitió identificar las brechas críticas en el modelo operativo manual de la organización. Esta correcta delimitación de necesidades sentó las bases para justificar funcionalidades clave, como el reloj marcador digital y la automatización salarial, al mismo tiempo que definió los estándares de seguridad y tiempos de respuesta óptimos para el uso fluido de la aplicación.

Posterior a este levantamiento, la construcción del diseño de la base de datos y del sistema web mediante la herramienta *Workbench* resultó esencial para garantizar la exactitud y la integridad referencial de los datos financieros y personales de los colaboradores. La elaboración de este modelo relacional centralizó la información de manera escalable, eliminando la redundancia de registros y proveyendo una arquitectura de almacenamiento sólida. Esto asegura que la plataforma sea capaz de soportar el crecimiento futuro de la nómina corporativa sin comprometer el rendimiento general.

Con la estructura de datos claramente definida, se procedió a programar los diferentes módulos del sistema mediante el entorno de desarrollo *Visual Studio Code*, integrando lenguajes y herramientas como *HTML5*, *CSS3*, *JavaScript* y *Python*. Esta fase de codificación culminó exitosamente en la materialización de un ecosistema web robusto. La implementación de estas tecnologías bajo el patrón arquitectónico adecuado permitió desarrollar una interfaz de usuario altamente intuitiva y un motor lógico capaz de procesar cálculos complejos, asegurando que las horas extras, rebajos, vacaciones y aguinaldos se ejecuten con estricto apego a la legislación laboral.

Para culminar el proceso de desarrollo, el procedimiento de probar los diferentes módulos de forma independiente y, posteriormente, de forma integral, comprobó la exactitud algorítmica y la estabilidad operativa de la plataforma. El software superó con éxito todas las validaciones lógicas, garantizando resultados precisos y confiables en cada transacción procesada y en la emisión de comprobantes electrónicos. El cumplimiento escalonado de estos hitos técnicos

certifica la consecución del objetivo general del proyecto, logrando entregar a la empresa una herramienta tecnológica que moderniza, asegura y automatiza por completo su gestión administrativa.

Recomendaciones

Para garantizar el éxito en la adopción del nuevo sistema web de planillas, se recomienda a la administración de Invet Diagnóstico planificar una transición operativa mediante un enfoque por fases en el corto plazo durante los primeros tres meses posteriores a la entrega. Es fundamental ejecutar un periodo inicial de marcha blanca de al menos un mes, durante el cual la nueva plataforma opere en paralelo con los controles manuales existentes. Esta estrategia permitirá que los usuarios se familiaricen con la interfaz sin comprometer la integridad de la nómina. Asimismo, se sugiere ejecutar durante el primer mes de implementación un cronograma de capacitación integral con una duración total estimada de 15 horas teóricas y prácticas. Este proceso, acompañado por la elaboración de un manual de usuario y procedimientos documentados para estandarizar la instrucción, se brindará a toda la planilla operativa dividida en dos grupos: por un lado, el personal de Recursos Humanos y Jefaturas, quienes se capacitarán a fondo para administrar las evaluaciones y aprobaciones; y por otro, los colaboradores estándar, abarcando a todo el personal veterinario y administrativo, enfocándose en el uso correcto del portal de autogestión y del reloj marcador digital.

Desde el punto de vista de la infraestructura y la seguridad de la información, se recomienda establecer de manera inmediata políticas estrictas para la realización de respaldos periódicos de la base de datos relacional. Dado el volumen y la alta sensibilidad de los datos financieros, expedientes médicos y cálculos laborales almacenados, es vital mantener actualizados los protocolos de seguridad y auditar los accesos según los roles definidos con una frecuencia al menos trimestral. En esta misma línea, se aconseja proceder en un plazo máximo de 30 a 60 días con la adquisición del equipo físico detallado en la proforma de requerimientos técnicos, garantizando que el motor de Python y la arquitectura web cuenten con los recursos de procesamiento y almacenamiento óptimos para operar de manera continua, estable y libre de interrupciones.

Finalmente, en cuanto a la escalabilidad tecnológica del proyecto, se sugiere contemplar a mediano plazo en un periodo de 12 a 18 meses el diseño y desarrollo de nuevos módulos que

complementen la arquitectura actual. Entre estas futuras expansiones, la empresa podría considerar la integración del sistema con servicios de reportes gerenciales avanzados mediante herramientas de inteligencia de negocios para el análisis predictivo de costos laborales y rotación de personal. Del mismo modo, se recomienda evaluar a partir del segundo año de estabilización del sistema la creación de una aplicación móvil nativa que consuma los datos del sistema web, lo cual facilitaría aún más el registro de asistencia y la consulta de colillas de pago, permitiendo que la plataforma evolucione en paralelo con el crecimiento institucional de la clínica.

REFERENCIAS

Asamblea Legislativa de la República de Costa Rica. (14 de octubre, 1982). *Ley Sobre Derechos de Autor y Derechos Conexos, Ley N° 6683*.

<https://www.asamblea.go.cr/sd/SiteAssets/Lists/Consultas%20Biblioteca/EditForm/Ley%206683%20Derechos%20de%20autor%20y%20derechos%20conexos.pdf>

Asamblea Legislativa de la República de Costa Rica. (18 de febrero de 2000). *Ley de Protección al Trabajador, Ley N° 7983*. Publicada en La Gaceta N° 35, Alcance 11.

Asamblea Legislativa de la República de Costa Rica. (24 de octubre, 2001). *Adición de los artículos 196 BIS, 217 BIS y 229 BIS al Código Penal, Ley N° 4573 para reprimir y sancionar los delitos informáticos, Ley N° 8148*. Sistema Costarricense de Información Jurídica, Procuraduría General de la República.

https://pgrweb.go.cr/scij/Busqueda/Normativa/Normas/nrm_texto_completo.aspx?param1=NRTC&nValor1=1&nValor2=47430&nValor3=50318&strTipM=TC

Asamblea Legislativa de la República de Costa Rica. (5 de setiembre, 2011). *Ley de Protección de la Persona frente al Tratamiento de sus Datos Personales, Ley N° 8968*. Sistema Costarricense de Información Jurídica, Procuraduría General de la República.

https://pgrweb.go.cr/scij/Busqueda/Normativa/Normas/nrm_texto_completo.aspx?param1=NRTC&nValor1=1&nValor2=70975&nValor3=85989&strTipM=TC

Avello Martínez, Raidell. (19 marzo, 2018). *Las fuentes de información y su evaluación*. Blog, Comunicar: Escuela de Autores. <https://www.grupocomunicar.com/wp/escuela-de-autores/las-fuentes-de-informacion-y-su-evaluacion/>

- Bermúdez León, Max José. (Diciembre, 2020). *Introducción a las bases de datos*. Costa Rica: Universidad San Marcos.
<https://repositorio.usam.ac.cr/xmlui/bitstream/handle/11506/2120/lec%20ing%20sist%200064%202020.pdf?sequence=1&isallowed=y>
- Caja Costarricense de Seguro Social [CCSS]. (2024). *Instructivo para el cálculo de planillas y cargas sociales*. Dirección de Inspección de la CCSS. <https://www.ccss.sa.cr>
- Casals, A. (19 de diciembre, 2025). *¿Qué es un software? Definición, para qué sirve, tipos y beneficios*. Blog, Taclia.com. <https://www.taclia.com/blog/que-es-un-software#%C2%BFqu%C3%A9-es-un-software>
- Chiavenato, I. (2020). *Gestión del talento humano: El nuevo papel de los recursos humanos en las organizaciones*. (5ª Ed.). McGraw-Hill.
<https://gc.scalahed.com/recursos/files/r161r/w24756w/gestion-del-talento-humano-chiavenato.pdf>
- Coursera Staff. (29 de noviembre, 2023). *¿Qué es Python y para qué se usa? Guía para principiantes*. <https://surl.li/sigloo>
- Cruz García, Margarita. (2019). Fuentes de información. *Boletín Científico de las Ciencias Económico Administrativas del ICEA*, 8(15), Repository UAEH. [Fuentes de Información | Boletín Científico de las Ciencias Económico Administrativas del ICEA](#)
- Fernández Iglesias, M. J. (7 de marzo, 2023). *Pequeña introducción a las bases de datos (1.1)*. https://www.researchgate.net/publication/369039369_Pequena_introduccion_a_las_bases_de_datos
- Grinberg, M. (2018). *Flask Web Development: Developing Web Applications with Python*. (2th Ed.). O'Reilly Media. [Flask Web Development, 2nd Edition \[Book\]](#)
- Hernández Flórez, Nubia y Klimenko, Olena. (2025). Metodología de la investigación, enfoque cualitativo, cuantitativo y mixto. En: *Nexus, Revista de Investigación Multidisciplinaria*, Vol. 2, N° 3, pp. 1-19. <https://dialnet.unirioja.es/servlet/articulo?codigo=10622809>
- Hernández Sampieri, Roberto; Fernández Collado, Carlos y Baptista Lucio, Pilar. (2014). *Metodología de la investigación*. (6ª. Ed.). España: McGraw Hill.
<https://dialnet.unirioja.es/servlet/libro?codigo=775008>
- Hernández Sampieri, Roberto; Méndez Valencia, Sergio; Mendoza Torres, Paulina y Cuevas Romo, Ana. (2017). *Fundamentos de Investigación*. (1ª Ed.). McGraw Hill Educación.

- <https://es.scribd.com/document/748654581/Fundamentos-de-Investigacion-2017-Sampieri-R-Mendez-S-Mendoza-C-y-Cuevas-A>
- Hernández, S. y Duana, D. (2020). Técnicas e instrumentos de recolección de datos. *Boletín Científico de las Ciencias Económico Administrativas del ICEA*, 9(17), Repository UAEH-51-53. <https://repository.uaeh.edu.mx/revistas/index.php/icea/issue/archive>
- IBM. (6 de enero, 2026). *Diseño de bases de datos*.
<https://www.ibm.com/docs/es/db2/12.1.x?topic=databases-designing>
- Instituto Nacional de Ciberseguridad [INCIBE]. (2022). *Guía de seguridad de las TIC para la gestión de riesgos*. <https://www.incibe.es/empresas/blog/gestion-de-riesgos-una-guia-practica-para-tomar-mejores-decisiones-en-ciberseguridad>
- International Standard ISO/IEC/IEEE 29148. (2018). *Systems and software engineering — Life cycle processes — Requirements engineering*.
<https://ieeexplore.uned.elogim.com/stamp/stamp.jsp?tp=&arnumber=8559686&tag=1>
- Invgate. (2 de enero, 2025). *¿Qué es el hardware?: Definición, tipos y componentes*.
<https://blog.invgate.com/es/que-es-hardware>
- ISO/IEC/IEEE. (2017). *Systems and Software Engineering Vocabulary*.
info_isoiecieee24765{ed2.0}en.pdf
- Lomelí, Luis. (26 de mayo, 2023). *Metodología Scrum: Roles, Procesos y Artefactos*. Blog: Sugeris.com. <https://innevo.com/blog/metodologia-scrum>
- Lutz, Mark. (2013). *Learning Python*. (5th Ed.). O'Reilly Media.
<https://www.oreilly.com/library/view/learning-python-5th/9781449355722/>
- Martínez, G., Camacho, G. y Biancha, D. (2010). Diseño de *framework* Web para el desarrollo dinámico de aplicaciones. *Scientia Et Technica*, XVI (44), Redylac-178-183.
<https://www.redalyc.org/pdf/849/84917316032.pdf>
- Medina Romero, M. Á., Hurtado Tiza, D. R., Muñoz Murillo, J. P., Ochoa Cervantez, D. O. y Izundegui Ordoñez, G. (2023). *Método Mixto de Investigación Cuantitativo y Cualitativo*. (1^a Ed.). Perú: Inudi. <https://doi.org/10.35622/inudi.b.105>
- Mejía, J., Díaz, E. y Fuentes, L. (2021). *Metodología de la Investigación en Enfermería Elementos para elaborar un proyecto de investigación*. (1^a Ed.). <https://surl.li/nwdvwd>
- Ministerio de Trabajo y Seguridad Social [MTSS]. (2026). *Lista de Salarios Mínimos, 2026*.
lista_salarios_minimos_2026.pdf

- Moreno, E. (9 de marzo, 2018). Metodología de investigación, pautas para hacer tesis. *Metodología de Investigación Científica*. <https://tesis-investigacion-cientifica.blogspot.com/2018/03/definicion-instrumental-de-las-variables.html>
- National Institute of Standards and Technology [NIST]. (2017). *An Introduction to Information Security (NIST Special Publication 800-12, Rev. 1)*. U.S. Department of Commerce. <https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-12r1.pdf>
- NCQ Technologies. (15 de diciembre, 2021). *Aspectos imprescindibles de un sistema de planillas para administración de personal*. Blog, Qupos.com. <https://blog.qupos.com/aspecto-sistema-planillas-administracion-personal>
- Porret Gelabert, Miquel. (2020). *Gestión de personas: Manual para la gestión del capital humano*. (6ª Ed.). ESIC Editorial. <https://www.esic.edu/editorial/producto/gestion-de-personas>
- Pressman, Roger. (2020). *Ingeniería del Software: un enfoque práctico*. (7ª ed.) [PDF, obra original publicada 2010]. <https://profesorezequielruizgarcia.wordpress.com/wp-content/uploads/2015/01/ingenieria-del-software-un-enfoque-practico-roger-spressman.pdf>
- Python Software Foundation. (2023). *General Python FAQ*. <https://docs.python.org/3/faq/general.html>
- Schwaber, K. y Sutherland, J. (Noviembre, 2020). *La Guía Scrum: La Guía Definitiva de Scrum: Las Reglas del Juego*. <https://scrumguides.org/docs/scrumguide/v2020/2020-Scrum-Guide-Spanish-European.pdf>
- Sistema Costarricense de Información Jurídica. (2024). *Ley de Protección al Trabajador N.º 7983*. Procuraduría General de la República.
- Softland Inversiones S. L. (2 septiembre, 2024). *Guía de Software de Planillas para Empresarios en Costa Rica*. <https://softland.com/cr/guia-software-planillas-empresarios/>
- Stallings, W., & Brown, L. (2012). *Computer Security: Principles and Practice*. (5th Ed.). Pearson. https://unidel.edu.ng/focelibrary/books/Computer%20Security%20_%20Principles%20-%20WILLIAM%20STALLINGS_2089.pdf
- Superintendencia de Pensiones [Supén]. (2023). *Estructura del Sistema Nacional de Pensiones*.
- Susnjara, S. y Smalley, I. (s. f.). *¿Qué son las pruebas de software?* IBM. <https://www.ibm.com/es-es/think/topics/software-testing>

ANEXOS

Anexo 1: Guía de Entrevista.

Entidad:	
Nombre del entrevistado:	
Puesto del entrevistado:	
Nombre del estudiante:	
Fecha de la entrevista:	
Lugar o medio de la entrevista:	

Preguntas:

Pregunta 1: Considerando la sensibilidad de los datos salariales, ¿qué eventos específicos del sistema considera críticos para ser auditados? Por ejemplo, ¿necesita que el sistema guarde un registro histórico de quién, cuándo y por qué se modificó el salario de un empleado o se alteró una marca de asistencia?

Pregunta 2: ¿Qué nivel de pérdida económica estima que ha sufrido la empresa debido a errores en pagos excesivos o cálculos incorrectos de aguinaldos y liquidaciones?

Pregunta 3: ¿Qué niveles de acceso específicos requiere? ¿Quiénes, además de usted, deben poder modificar salarios o eliminar registros de marcas?

Pregunta 4: Dado que es una veterinaria, ¿existen bonos por procedimientos específicos o ventas de productos? ¿Cómo es la fórmula exacta de cálculo para integrar este algoritmo en el módulo de planilla?

Pregunta 5: ¿Cuáles herramientas, según su experiencia, hacen falta para el control de marcas y asistencia que sustituyan el uso de *WhatsApp*?

Pregunta 6: ¿Cuántos años de historial de pagos considera crítico que el sistema mantenga disponibles para consultas rápidas antes de ser archivados?

Pregunta 7: ¿Cuáles herramientas, según su experiencia, hacen falta en el proceso de registro y control de expedientes de los colaboradores?

Pregunta 8: ¿Qué tipos de empleados tiene la veterinaria (por ej.: médicos veterinarios, asistentes, recepcionistas, personal de limpieza)? ¿Tienen regímenes de pago distintos?

Pregunta 9: ¿Cuáles son los niveles de acceso o tipos de usuarios que interactuarán con el sistema (por ej.: Administrador, Dueño, RH, Empleado estándar)?

Pregunta 10: ¿Qué tablas y qué datos específicos desea que se observen en el apartado de consultas de personal (por ej.: fecha de ingreso, historial de puestos, salarios anteriores)?

Pregunta 11: ¿Se tendrá una forma diferente de cálculo de incentivos o comisiones según el tipo de puesto del empleado?

Pregunta 12: ¿Cuáles son las jornadas laborales activas en la empresa (por ej.: diurna, mixta, nocturna, por turnos rotativos)?

Pregunta 13: ¿Cuáles son los tipos de pago que se manejan (por ej.: transferencia bancaria Sinpe, efectivo, cheque)?

Pregunta 14: ¿Cuáles son los plazos o fechas de corte en los que se debe realizar el cálculo y provisión del aguinaldo?

Pregunta 15: ¿Cuáles funciones desea que se ingresen en el apartado de mantenimientos para la gestión de vacaciones (por ej.: cargar saldos iniciales, ajuste manual de días)?

Pregunta 16: ¿Qué métricas o KPI's desea que se observen en las consultas de rendimiento?

Pregunta 17: ¿Qué tipo de impuestos y cargas patronales específicas se deben reflejar en el resumen de costos de la empresa?

Pregunta 18: En el flujo de trabajo, ¿cómo se validan las boletas de la CCSS? Resulta necesario definir si el sistema debe permitir la carga de archivos PDF o fotos de comprobantes para el Módulo de Gestión.

Pregunta 19: Describa el flujo actual desde que un empleado marca su entrada hasta que se emite su comprobante de pago. ¿Cuánto tiempo dedica semanalmente a estas tareas manuales en Excel?

Pregunta 20: Según los módulos planteados (liquidaciones, aguinaldo, evaluación), ¿cuál considera que es el más urgente para reducir la carga operativa inmediata?

Anexo 2: Guía de Cuestionario

En el marco de una investigación sobre sistema web para la gestión de pago de planilla en la Veterinaria Invet Diagnóstico, ubicada en San Pedro Montes de Oca, 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.

Pregunta 1: ¿Con qué frecuencia ha tenido que solicitar una corrección en su comprobante de pago debido a cálculos incorrectos de horas extra, feriados o deducciones?

☐ Nunca ☐ Rara vez ☐ Frecuentemente ☐ Siempre

Pregunta 2: ¿Entiende claramente cómo se calculan sus rebajos de ley (CCSS y Banco Popular) y el impuesto sobre la renta en su pago actual?

☐ Totalmente claro ☐ Poco claro ☐ Nada claro

Pregunta 3: ¿Qué tan difícil le resulta obtener una constancia de salario o revisar sus pagos de meses anteriores con el método administrativo actual?

☐ Muy difícil ☐ Difícil ☐ Fácil ☐ Muy fácil

Pregunta 4: ¿Considera que el registro de asistencia mediante *WhatsApp* es un método seguro y confiable para garantizar que sus horas laboradas se contabilicen con exactitud?

☐ Sí ☐ No ☐ Indiferente

Pregunta 5: ¿Conoce usted con exactitud cuántos días de vacaciones tiene disponibles en este momento sin tener que consultarlo directamente con administración?

☐ Sí, lo sé con certeza ☐ Tengo una idea aproximada ☐ No lo sé

Pregunta 6: ¿Le gustaría recibir su comprobante de pago de forma automática por correo electrónico o mediante una plataforma web apenas se procese la planilla?

☐ Sí, es necesario ☐ Me es indiferente ☐ No lo considero necesario

Pregunta 7: Al solicitar un permiso o reportar una incapacidad, ¿cuánto tiempo tarda usualmente en recibir una confirmación oficial de que el movimiento fue aplicado correctamente?

☐ Menos de 24 horas ☐ De 2 a 3 días ☐ Más de una semana ☐ Nunca recibo confirmación

Pregunta 8: ¿Se le comunica formalmente el resultado de sus evaluaciones de desempeño y cómo estas afectan sus posibles bonificaciones o crecimiento en la clínica?

☐ Sí, siempre ☐ A veces ☐ Nunca me evalúan formalmente

Pregunta 9: ¿Siente incertidumbre sobre si el monto de su aguinaldo a fin de año será calculado correctamente bajo el sistema manual actual?

☐ Mucha incertidumbre ☐ Poca incertidumbre ☐ Ninguna incertidumbre

Pregunta 10: Del 1 al 5, ¿qué tan útil considera que sería tener un perfil de usuario donde pueda ver su historial de marcas, solicitar vacaciones y ver sus evaluaciones de desempeño?

(1 - Nada útil / 5 - Muy útil)

Anexo 3: Guía del Proceso de Observación

Entidad:	INVET DIAGNÓSTICO
Dirección física de la entidad:	San Pedro Montes de Oca
Fecha de la actividad de observación:	
Nombre del estudiante:	Fabián Chaves

Tabla de control de aspectos observados:

Nº	Aspectos por observar	Cumple	No Cumple	Oportunidad de mejora	Detalle de observación
1	Observar el método en que los empleados notifican su entrada y salida (vía <i>WhatsApp</i> o manual).				
2	Observar dónde y cómo se consolidan los mensajes de <i>WhatsApp</i> para el cálculo final de horas.				
3	Observar el procedimiento manual para diferenciar y calcular jornadas extraordinarias según la legislación.				
4	Observar si la información de los empleados (contratos, datos personales) está centralizada o dispersa.				
5	Observar cómo se archivan y se reflejan las boletas de la CCSS en el control de pagos.				
6	Observar el formato y el medio de entrega de los desgloses de salario a los colaboradores.				

Nº	Aspectos por observar	Cumple	No Cumple	Oportunidad de mejora	Detalle de observación
7	Observar el método de seguimiento de saldos de vacaciones y la acumulación mensual de pasivos laborales.				
8	Observar si existen formularios físicos o digitales para medir los KPI's de los empleados.				
9	Observar cómo se transfiere la información de planilla desde las sedes de Escazú y Cartago hacia la administración central.				
10	Observar si el acceso a las hojas de cálculo de salarios está restringido o si es accesible para personal no autorizado.				