INSTITUTO SUPERIOR

TECNOLÓGICO

SAN ANTONIO

Escuela de Desarrollo de Software y Programación

Prototipo de sistema web para la administración del recurso humano en la empresa ColaboraSoft SAS

Edwin Leonardo Rosas Yagüé

Tecnología superior en desarrollo de software y programación

Trabajo de fin de carrera presentado como requisito

para la obtención del título de

Tecnólogo Superior en desarrollo de software y programación

Quito, 30 de septiembre de 2025

INSTITUTO SUPERIOR

TECNOLÓGICO

SAN ANTONIO

Escuela de Desarrollo de Software y Programación

HOJA DE CALIFICACIÓN

DE TRABAJO DE FIN DE CARRERA

Prototipo de sistema web para la administración del recurso humano en la empresa ColaboraSoft SAS

Edwin Leonardo Rosas Yagüé

Santiago Xavier Rojas Bukovac, Ingeniero

Quito, 30 de septiembre de 2025

DERECHOS DE AUTOR

Por medio del presente documento certifico que he leído todas las Políticas y Manuales del Instituto Superior Tecnológico San Antonio TESA, incluyendo la Política de propiedad Intelectual TESA, y estoy de acuerdo con su contenido, por lo que los derechos de propiedad intelectual del presente trabajo quedan sujetos a lo dispuesto en esas Políticas.

Asimismo, autorizo a TESA para que realice la digitalización y publicación de este trabajo en el repositorio virtual, de conformidad a lo dispuesto en el Art. 144 de la Ley Orgánica de Educación Superior.

Nombres y apellidos: Edwin Leonardo Rosas Yague

Código estudiante: 2658

Cédula de identidad: 1717886541

Lugar y fecha: Quito, Septiembre de 2025

ACLARACIÓN PARA PUBLICACIÓN

Nota: El presente trabajo, en su totalidad o cualquiera de sus partes, no debe ser considerado como una publicación, pese a estar disponible sin restricciones a través de un repositorio institucional. Esta declaración se alinea con las prácticas y recomendaciones presentadas por el Committee on Publication Ethics COPE descritas por Barbour et al. (2017) Discussion document on best practice for issues around theses publishing, disponible en http://bit.ly/COPETheses.

UNPUBLISHED DOCUMENT

Note: The following capstone project is available through Instituto Superior Tecnológico San Antonio TESA institutional repository. Nonetheless, this project – in whole or in part – should not be considered a publication. This statement follows the recommendations presented by the Committee on Publication Ethics COPE described by Barbour et al. (2017) Discussion document on best practice for issues around theses publishing available on http://bit.ly/COPETheses.

DEDICATORIA

A Dios, una fuente de amor, sabiduría y fortaleza, por ser el guiador de cada paso en este camino, por sostenerme en momentos de duda y dar sentido a los logros alcanzados.

A mi amada esposa, por su gran amor incondicional, paciencia infinita y apoyo constante. Por creer en mí y acompañarme en cada jornada de esfuerzo y por ser mi mayor inspiración.

AGRADECIMIENTO

Agradezco a mi esposa este logro por ser un gran apoyo en todo momento y lugar, gracias a ella se termina esta meta.

Agradezco al Tecnológico San Antonio y a cada profesor por brindarme una formación integral, excelencia académica y compromiso profesional. Cada etapa vivida en el tecnológico ha sido clave en mi desarrollo personal y técnico.

RESUMEN

Este proyecto está enfocado en resolver las limitaciones derivadas de la gestión manual del talento humano en la empresa ColaboraSoft SAS, actualmente los procesos que usa esta empresa para llevar la administración de personal se limitan a archivos Excel lo que conlleva a riesgos operativos, altos tiempos de respuesta para generación de informes, y posibles incumplimientos a la Ley Orgánica de Protección de Datos Personales en Ecuador. El propósito principal es diseñar un prototipo funcional de una aplicación web que permita automatizar funciones claves con la generación de la nómina y sus cálculos, los periodos de vacaciones y emisión de certificados laborales con el fin de optimizar el rendimiento del departamento de recursos humanos.

Para realizarlo se aplicó una metodología mixta, integrando enfoques cuantitativos y cualitativos que incluyen encuestas al equipo de talento humano y pruebas de usabilidad. El prototipo de aplicación web fue construido con tecnologías actuales usando Python como lenguaje base, Django para el desarrollo backend, Vue como framework frontend y PostgreSQL como base de datos. Los resultados que se esperan es una mejora sustancial en la eficiencia de los procesos, fortaleciendo la seguridad mediante el ofuscamiento de información sensible y alineación a la normativa vigente. Este desarrollo representa un aporte a la digitalización de pequeñas y medianas empresas en Ecuador.

Palabras clave: Talento Humano, aplicación web, automatización, Python, Django, Vue, protección de datos, digitalización empresarial.

ABSTRACT

This project addresses the limitations associated with the manual management of human resources at ColaboraSoft SAS. Currently, personnel management relies heavily on Excel files, a practice that generates operational risks, delays in report generation, and potential violations of Ecuador's Organic Law on Personal Data Protection. The objective of this study is to design a functional prototype of a web application capable of automating key human resources processes, including payroll generation and calculations, vacation tracking, and the issuance of employment certificates, thereby improving the overall performance of the department.

To accomplish this objective, a mixed-methods approach was employed, integrating quantitative and qualitative techniques such as surveys administered to the human resources team and usability testing of the prototype. The web application was developed using contemporary technologies: Python as the primary programming language, Django for backend development, Vue as the frontend framework, and PostgreSQL as the database management system. The expected outcomes include notable improvements in process efficiency, enhanced security through the concealment of sensitive information, and compliance with current data protection regulations. This development constitutes a meaningful contribution to the ongoing digital transformation of small and medium-sized enterprises in Ecuador.

Keywords: Human Talent, web application, automation, Python, Django, Vue, data protection, business digitization.

TABLA DE CONTENIDOS

CAPÍTULO I INTRODUCCIÓN 15

1.1. Planteamiento del Problema 15

1.2. Justificación 16

1.3. Alcance 17

1.4. Objetivos General y Específicos 18

1.4.1. Objetivo General 18

1.4.2. Objetivos Específicos 18

CAPÍTULO II MARCO TEÓRICO 19

2.1. Estudios Previos 19

2.1.1. Variables Independientes 23

2.1.1.1. Programación / Desarrollo. 23

2.1.1.2. Software para Administración de Personal. 24

2.1.1.3. ERP. 25

2.1.2. Variables Dependientes 26

2.1.2.1. Eficiencia Individual. 26

2.1.2.2. Equipo de TTHH. 26

2.1.2.3. Empresarial. 27

2.1.3. Conexiones entre variables 28

2.2. Marco legal 29

CAPÍTULO III. MARCO METODOLÓGICO 32

3.1. Tipo de investigación 32

3.1.1. Según el propósito 32

3.1.2. Según la manipulación de variables 32

3.1.3. Según el tiempo 32

3.2. Tipo de Enfoque 32
3.3. Herramientas metodológicas 33
3.4. Técnicas 33
3.5. Estudios iniciales 33
3.5.1. Criterios de inclusión y exclusión 33
3.5.2. Encuesta previa realizada 33
3.5.3. Muestreo 34
3.5.4. Organizador de variables 36
3.6. Desarrollo 37
3.6.1. Matriz de Operacionalización de Variables 37
3.6.2. Herramientas 39
3.6.3. Arquitectura 39
3.6.4. Backend 41
3.6.5. Frontend 48
3.6.6. Base de datos y relación de las tablas 50
3.6.7. Diseño UX/UI 56
3.6.8. Seguridad 58
3.6.9. Pruebas funcionales 59
CAPITULO IV. RESULTADOS Y DISCUSIÓN 70
4.1. Análisis e interpretación de resultados (descripción) 70
4.1.1. Resultados de la encuesta 70
4.1.2. Resultados técnicos 71
4.1.3. Resultado integridad de datos 72
4.1.4. Resultado de cumplimiento normativo 73
4.2. Discusión de resultados 74

CONCLUSIONES Y RECOMENDACIONES 77

REFERENCIAS BIBLIOGRÁFICAS.....79

ANEXO A: ENCUESTA.....80

ANEXO B: IMÁGENES PROTOTIPO.....84

ÍNDICE DE TABLAS

Tabla 1 Conexiones 28
Tabla 2 Encuesta 34
Tabla 3 Organizador de variables 36
Tabla 4 Operación de variables 38
Tabla 5 Herramientas 39
Tabla 6 Tecnología Backend 41
Tabla 7 Modelos aplicación empleado 43
Tabla 8 Serializers aplicación empleado 44
Tabla 9 URLS aplicación empleado 44
Tabla 10 Views aplicación empleado 45
Tabla 11 Modelos aplicación nomina 46
Tabla 12 Serializer aplicación nomina 46
Tabla 13 URLS aplicación nomina 47
Tabla 14 Views aplicación nomina 48
Tabla 15 Tecnología Frontend 49
Tabla 16 Vistas Principales 49
Tabla 17 Servicios de frontend 50
Tabla 18 Resultados encuesta 70
Tabla 19 Resultado pruebas funcionales 71
Tabla 20 Validación integridad de datos 72
Tabla 21 Evaluación de cumplimiento normativo 73

ÍNDICE DE FIGURAS

Figura 1 Variable dependiente 22

Figura 2 Variable independiente 23

Figura 3 Arquitectura 40

Figura 4 Backend 41

Figura 5 Insomnia generación de tokens 59

Figura 6 Insomnia creación de cargo 60

Figura 7 Insomnia consulta de cargos 61

Figura 8 PostgreSQL consulta de cargos 61

Figura 9 Insomnia creación de identificación 62

Figura 10 PostgreSQL creación de identificación 62

Figura 11 Insomnia creación concepto de nomina 64

Figura 12 PostgreSQL consulta concepto de nomina 64

Figura 13 Insomnia consulta de conceptos de nomina 65

Figura 14 Insomnia creación de proceso de rol 66

Figura 15 PostgreSQL consulta proceso de rol 66

Figura 16 Insomnia consulta proceso de rol 67

Figura 17 Insomnia creación de empresa 68

Figura 18 PostgreSQL consulta de empresa 68

Figura 19 Insomnia consulta de empresa 69

Figura 20 Encuesta sección 1 80

Figura 21 Encuesta sección 2 80

Figura 22 Encuesta sección 3 81

Figura 23 Encuesta sección 4 82

Figura 24 Encuesta sección 5 83
Figura 25 Resultados encuesta 83
Figura 26 Servidor Python / Django 84
Figura 27 Servidor VUE 84
Figura 28 PostgreSQL 85
Figura 29 Aplicación inicio de sesión 85
Figura 30 Aplicación validación inicio de sesión 86
Figura 31 Aplicación vista simple Empresas 86
Figura 32 Aplicación vista múltiple Empresa 87
Figura 33 Aplicación vista múltiple centro de costo 87
Figura 34 Aplicación vista simple centro de costo 87
Figura 35 Aplicación vista múltiple estructura orgánica 88
Figura 36 Aplicación vista simple estructura orgánica 88
Figura 37 Aplicación vista simple ubicación física 88
Figura 38 Aplicación vista múltiple ubicación física 89
Figura 39 Aplicación vista múltiple conceptos 89
Figura 40 Aplicación vista simple conceptos 90
Figura 41 Aplicación vista múltiple empleados 90
Figura 42 Aplicación vista simple empleado 91
Figura 43 Aplicación creación datos sensibles empleado 91
Figura 45 Aplicación anonimización datos sensibles 92
Figura 44 Aplicación validación de campos en formulario 92

CAPÍTULO I
INTRODUCCIÓN

1.1. Planteamiento del Problema

La administración de personal y la generación de roles es un proceso critico que tiene la empresa ColaboraSoft ya que actualmente se lleva esta información de manera manual en archivos Excel y se invierte una gran cantidad de horas mensuales en cumplir estas tareas satisfactoriamente, el procesar la información de esa forma está generando para esta empresa un riesgo al no tener un sistema digitalizado y centralizado para llevar y tratar esta información, adicional la “Ley orgánica de protección de datos personales” obliga a las empresas con información de trabajadores, clientes o proveedores de naturaleza personal que tenga medidas de seguridad que garanticen la protección de datos personales, esto incluye el acceso y decisión sobre esta información aparte de otras responsabilidades dadas en esta ley; en el mercado podemos encontrar software que ofrecen el servicio de administración de personal, portal de autoservicio y generación de roles a costos altos, pero que no cubre la normativa exigida por la ley ya que los datos no están enmascarados o encriptados para dificultar su acceso, además que no existe en el mercado una solución que permita a los colaboradores de la empresa acceder desde un computador o teléfono fuera de la red de la empresa donde trabaja a información como sus roles de pagos, generar su proyección de gastos anuales o a consultar y solicitar días de vacaciones disponibles.

En una entrevista que se realizó al personal de talento humano de ColaboraSoft se evidencio que evitar los procesos manuales, tener una herramienta de ingreso, tener logs de auditoría, automatizar los procesos mensuales que demandan mucho tiempo y seguimiento y cumplir con la normativa es una preocupación constante en la gerencia de la compañía y en el área de talento humano.

Si este desarrollo no se realiza la empresa ColaboraSoft podría tener multas equivalentes al 10% de su ingreso anual, no pasar las auditorías externas que son aplicadas por otras empresas para contratar sus servicios y no generar y mantener una cultura tecnológica en sus trabajadores. por lo que terminaría afectándola la proyección de innovación tecnológica prevista para este año 2025.

Como resumen tenemos que este desarrollo mejoraría la operatividad de la empresa y la eficiencia de un área como talento humano al disponer de una herramienta tecnológica unificada que permita tener la información de los colaboradores a la mano y con las seguridades que la normativa actual ecuatoriana está solicitando.

¿Se puede cumplir la normativa que está en la ley orgánica de protección de datos personales sin un sistema especializado para la administración de personal?

¿Puede un sistema de gestión de personal mejorar la operatividad de una empresa?

¿Podría ColaboraSoft encontrar en el mercado actual una herramienta que cubra todas sus necesidades?

1.2. Justificación

Administrar eficientemente la información de los empleados en ColaboraSoft es muy importar para el éxito de la misma, actualmente la tecnología avanza constantemente y el desarrollar e implementar un software que permita gestionar de mejor forma esta información debe estar lejos de ser una tendencia y se debe acercar más a una necesidad para el crecimiento de la empresa, el mejorar la productividad del área del talento humano, facilitar procesos a los colaboradores y generar informes a gerencia son acciones que permitirán enfocar recursos humanos, tecnológicos y económicos a otras áreas de ColaboraSoft.

Desarrollar esta aplicación en ColaboraSoft abordara un problema crucial que se tiene y es el tiempo que se invierte en tareas repetitivas, tener una información confiable y evitar la reprocesamiento de tareas por errores manuales; el volumen de información que tiene

ColaboraSoft de sus colaboradores es bastante grande y es necesario tratar esta información de forma responsable, confidencial y unificada.

El desarrollar esta herramienta permitirá a ColaboraSoft que es una pequeña empresa disponer de algo que solo las grandes empresas pueden tener y es un portal de autoservicio para sus colaboradores donde podrán visualizar y descargar sus roles de pagos, visualizar sus días de vacaciones o generar certificados laborales, esto sin esperar los tiempos que antes se daban por solicitar esta información al área de talento humano, este auto servicio aunque parece simple beneficia al colaborador por la transparencia que la empresa le está dando a la información y la satisfacción de obtener esa información sin tanto tramite y también beneficiara al área de talento humano porque no tendrá que hacer este tipo de tareas que antes se realizaba de forma manual y todo desde la comodidad de su casa o teléfono celular.

Tener esta aplicación web para gestionar a los empleados de ColaboraSoft no solo solucionará los problemas actuales de dispersión de información en varios archivos, gestión ineficiente de datos y el no cumplimiento de normativas legales, sino que también impulsará el avance tecnológico en la empresa.

El desarrollo de esta aplicación web optimiza procesos, mejora la gestión de datos, facilita la toma de decisiones y enriquece la experiencia del empleado. Este proyecto no solo aborda una necesidad crítica dentro de la empresa, sino que también se alinea con las tendencias globales de digitalización y optimización de procesos empresariales que tomaron fuerza desde la Pandemia.

1.3. Alcance

Para el análisis de este desarrollo se realizó una visita en la empresa ColaboraSoft donde por medio de una entrevista al área de talento humano se identificaron lo que para ellos es crítico controlar en una aplicación web para la administración de personal, entre estos puntos críticos esta la seguridad de la información, el registrar datos básicos del empleado, el

autoservicio que pueda tener el empleado para reducir la carga operativa y la automatización de los procesos de roles de pagos; en el diseño de la aplicación se perfilo técnicamente el manejo de perfiles para que los empleados puedan acceder al portal de autoservicio, los administradores puedan transaccionar con la información para la generación de roles de pagos e informes, el manejo de pantallas con la información necesaria para una buena visualización, implementación de colores empresariales de la empresa ColaboraSoft. Para la culminación del proyecto es necesario el cierre exitoso de varios roles de pagos y la generación de información bancaria, reglamentaria y tributaria.

1.4. Objetivos General y Específicos

1.4.1. Objetivo General

Desarrollo e implementación de un sistema web para la administración de los colaboradores en la empresa ColaboraSoft al norte de quito para aumentar la productividad del área de talento humano.

1.4.2. Objetivos Específicos

- Entender la necesidad que tiene la empresa ColaboraSoft para automatizar sus procesos y cumplir la normativa de la ley.
- Diseñar el sistema web empezando desde la base de datos, la parte de la lógica del negocio y la parte visual ofreciendo un producto confiable y fácil de usar.
- Evaluar el desarrollo por medio de pruebas con el usuario generando los roles de pagos, archivos para la acreditación en el banco, archivos para el ministerio de trabajo y archivos para el SRI.

CAPÍTULO II
MARCO TEÓRICO

2.1. Estudios Previos

En un análisis realizado por Ascencio Jordán y Navarro Espinosa (2016) muestran como los sistemas de información han tenido un incremento en la forma en que las compañías buscan, reclutan y seleccionan personal por medio de la implementación de tecnología en las empresas.

Este articulo tiene como fin analizar los fundamentos teóricos de los sistemas de información como ayuda para el área de talento humano de las empresas, enfocándose especialmente en la optimización de los procesos relacionados con el personal dentro de las empresas.

Los sistemas de información se definen como conjuntos estructurados que integran personas, procesos, hardware, software, recursos informáticos y redes de comunicación, todo esto con el propósito de almacenar, procesar, recuperar y distribuir información dentro de una empresa, desde hace varios años las empresas han buscado incorporar las llamadas “nuevas tecnologías” en los procesos de talento humano o administración de personal.

En las empresas la transformación digital dejo de ser una alternativa y paso a ser una condición imprescindible para el crecimiento de cualquier empresa, el implementar una transformación digital es un proceso complejo y con obstáculos que obliga indirectamente a las empresas y su personal a cambiar la mentalidad, llevar con triunfo una transformación digital en una empresa dará la ventaja de un incremento de competitividad, de una mayor eficiencia en los procesos internos y en una capacidad de un valor agregado al cliente interno y externo de la empresa.

Para el área de Talento Humano esta evolución digital ha significado la llegada de tecnología y automatizaciones que antes no eran necesarias, esta digitalización ha transformado los procesos de selección, contratación y gestión que se llevaba, el uso de inteligencia artificial permite filtrar de mejor manera y evaluar desde otra perspectiva las capacidades de los postulantes.

Según Duran Delgado y Rojas Sanchez (2023), los sistemas para la gestión del talento ayudan significativamente a que las empresas u organizaciones puedan planificar, implementar, supervisar, evaluar, interpretar e incluso puedan anticipar aspectos claves relacionados en el manejo del talento humano dentro de las empresas llevándolos a tener una administración más eficiente del recurso y del tiempo, incorporar estas tecnologías a las empresas hace que las mismas se adapten con mayor agilidad a la constante evolución que tienen los entornos corporativos.

Un sistema, plataforma o aplicación de sistema de gestión del talento es la que ayuda en los procesos centrales del área de Talento Humano, tales como selección de personal, contratación, medición de desempeño entre otras, estos proceso así como las habilidades técnicas del usuario que lo respaldan se brinda por medio de módulos de software, esto da la ventaja a las empresas que puedan empezar con la implementación de lo que realmente necesita y posterior podrá incluir nuevos módulos a medida que crece la empresa o que por normativa es necesario tener estos nuevo módulos.

La implementación de este tipo de software modulares resuelve problemas de la empresa como son los flujos de trabajo manuales y lentos, evita los reprocesos manuales y contribuye a definir y monitorear por medio de herramientas propias del software el rendimiento del área y del personal de la empresa, da una administración más efectiva y ordenada.

Según la Cámara de Innovación y Tecnología Ecuatoriana (CITEC), en su Estudio Sectorial del 2024 sobre la industria de servicios digitales en el país, destaca que Ecuador se ha convertido en un terreno fértil para la investigación y el desarrollo de software. Cada vez más empresas apuestan por crear o tener soluciones tecnológicas que automatizan procesos, o integrar agentes colaborativos de inteligencia artificial, estas iniciativas no solo transforman el panorama de las empresas en Ecuador, sino que también abran puertas hacia el exterior demostrando que aquí podemos generar software de calidad, tanto es así que las exportaciones de software tuvieron un crecimiento anual del 17%.

Actualmente, alrededor de 800 empresas ecuatorianas está desarrollando software para el sector financiero, siendo este grupo el que más consume desarrollo para cumplir con normativas como la prevención de fraudes o el lavado de activos. En este contexto tan dinámico y de crecimiento permanente, el proyecto que se plantea desarrollar busca aprovecha esta creciente demanda tecnológica que tiene las empresas del país. Para ello se propone el desarrollo de una aplicación web que facilite la gestión del talento humano, alineándose a las necesidades del mercado, al ritmo acelerado de la transformación digital y a cumplir los cambios normativos que trae la Ley de protección de datos.

Considerando estas referencias, podemos afirmar que para desarrollar el marco teórico de este proyecto debemos considerar algunas variables independientes y dependientes como se muestran a continuación en las figuras 1 y 2.

Figura 1

Variable dependiente

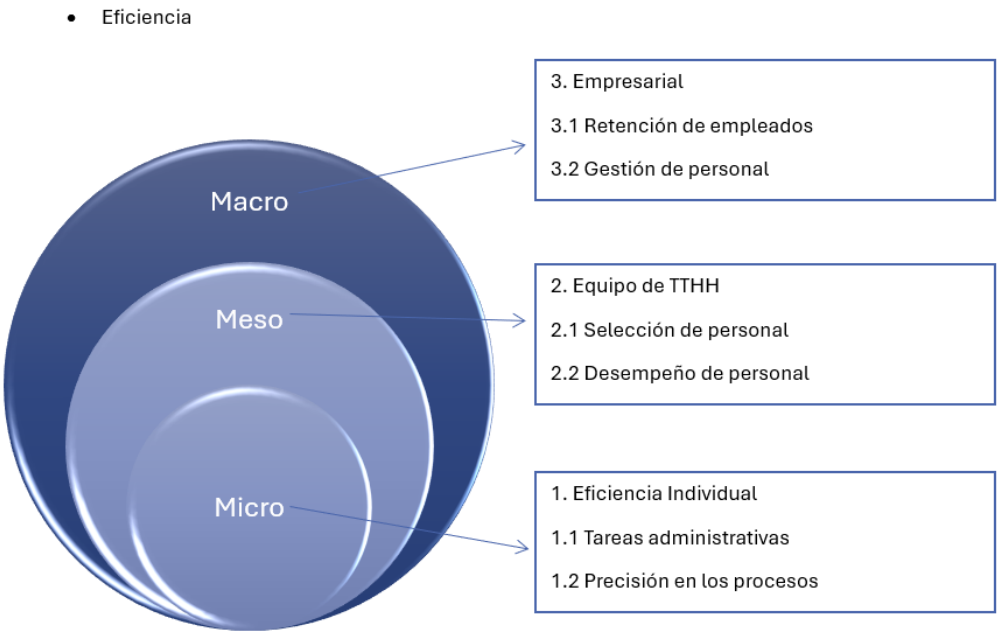
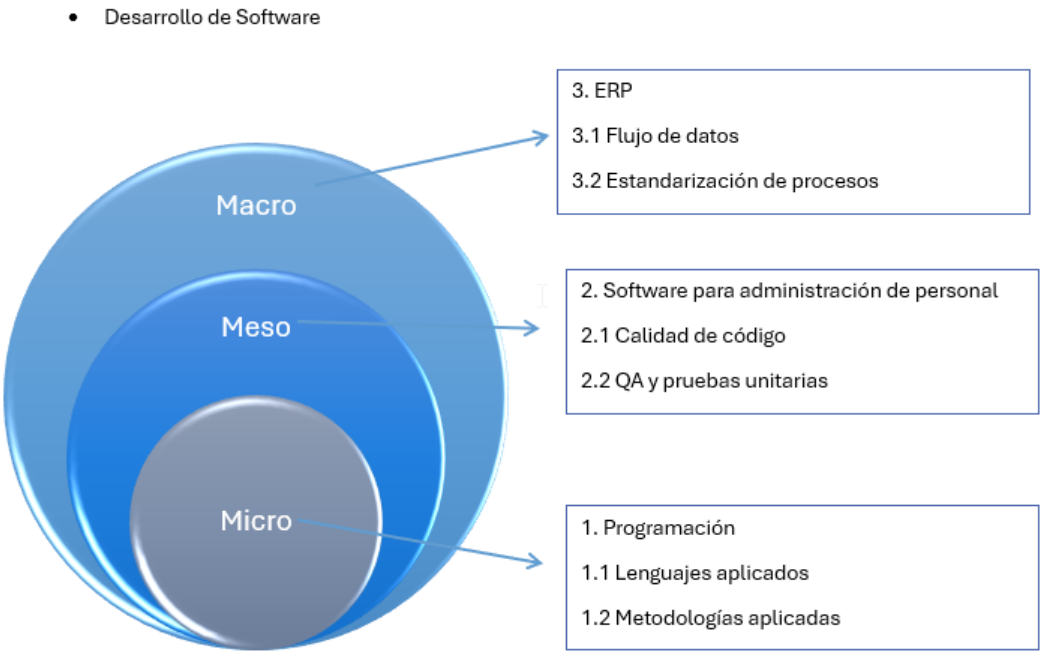


Figura 2

Variable independiente



2.1.1. *Variables Independientes*

2.1.1.1. Programación / Desarrollo. La variable independiente de programación se divide en lenguajes aplicados y metodologías aplicadas que se detallan a continuación:

a. Lenguajes aplicados: se usarán las herramientas actuales fundamentales para el desarrollo de un sistema Web.

Frontend

- **HTML5:** Lenguaje de marcado para estructurar contenido web
- **CSS3:** Lenguaje de estilos en cascada para el diseño visual y presentación de la aplicación Web.

- **JavaScript:** Lenguaje del lado del cliente que permite la creación de interfaces interactivas y dinámicas

Backend

- **Python:** Lenguaje de programación robusto, se usa ampliamente en desarrollo web empresarial, tiene una sintaxis clara lo que facilitara el mantenimiento del código
- **Django / Django REST Framework:** Es un framework web de alto nivel que usa el desarrollo rápido y limpio, ya proporciona características para el manejo de base de datos, seguridad y autenticación

Base de datos:

- **SQL (PostreSQL):** Motor de base de datos estructurado para la gestión de datos relacionados, con este motor podemos garantizar la integridad y consistencia de la información.

b. Metodologías aplicadas: Se usará la metodología ágil SCRUM que estructura el desarrollo en sprint corta de 2 a 4 semanas lo que permitirá tener entregas funcionales de forma frecuente.

2.1.1.2. Software para Administración de Personal. La variable independiente de software para administración de personal se divide en calidad de código, QA y pruebas unitarias que se detallan a continuación:

a. Calidad de código: Determinara la mantenibilidad, escalabilidad y confiabilidad de la aplicación Web, se aplicarán estándares como

Principios SOLID, Clean Code, Análisis estático de código y documentación del código

b. QA y pruebas unitarias: Asegurar la calidad de la aplicación y las pruebas deben ser fundamentales en todo desarrollo, de esta forma se podrá garantizar que los requerimientos iniciales del desarrollo se puedan cumplir, se deben realizar pruebas unitarias, pruebas técnico/funcionales, pruebas de usabilidad, pruebas de seguridad y pruebas de rendimiento.

2.1.1.3. ERP. La variable independiente ERP se divide en flujo de datos y estandarización de procesos que se detallan a continuación:

a. Flujo de datos: El flujo define como la información se va a mover a través de la aplicación web, desde su captura hasta su almacenamiento, procesamiento y visualización. En la parte del usuario se validarán formularios y reportes, en la parte de la lógica se realizarán cálculos automáticos y en la base de datos se tendrán todas las relaciones entre las tablas, así como logs de auditoría, se tendrán las características de los flujos de datos como trazabilidad, integridad, seguridad, atomicidad y respaldo

b. Estandarización de procesos: Este proceso implica la documentación y unificación de los procesos de la empresa, posterior a eso se tendrá como resultado que siempre se realizar de la misma forma lo siguiente:

- Proceso de ingreso nuevos empleados
- Proceso de generación de roles de pagos

- Proceso de solicitud de vacaciones
- Proceso de generación de certificados laborales

Esto convertirá tareas manuales y variables en procedimientos automatizados, uniformes y cuantificables lo que va a impactar en la eficiencia de la empresa.

2.1.2. *Variables Dependientes*

2.1.2.1. Eficiencia Individual. La variable dependiente de eficiencia individual se divide en tareas administrativas y precisión en los procesos que se detallan a continuación:

a. Tareas Administrativas: Estas comprenden todas las actividades de cada funcional involucrado en la gestión del talento humano de la empresa, se busca reducir tiempos en búsquedas manuales de información, actualización de información, generación de documentos y validaciones automáticas.

b. Precisión en los procesos: Con la implementación de una aplicación Web se buscar reducir errores de validación de datos, cálculos mal realizados, integridad de la información, registros duplicados

2.1.2.2. Equipo de TTHH. La variable dependiente de Equipo de TTHH se divide en selección de personal y desempeño laborar que se detallan a continuación:

a. Selección de personal: En este proceso se identifica, evalúa y contrata a los prospectos más idóneos para un cargo en la empresa, este proceso implica el solicitar información física para ser validad, al

implementar esta información podrá llegar de forma digital ahorrando espacio y tiempo

b. Desempeño de personal: Se podrá validar con el acceso rápido y unificado a la información de los empleados de la empresa, adicional con el historial recuperar cosas anteriores será más eficiente, la automatización de tareas liberará a los funcionales del área para que puedan realizar otras tareas

2.1.2.3. Empresarial. La variable dependiente empresarial se divide en retención de empleados y gestión de personal que se detallan a continuación:

a. Retención de empleados: Es la capacidad de la organización para mantener al personal valioso reduciendo las rotaciones y los costos que esto conlleva en las empresas, al implementar la aplicación Web la empresa esta transparentado la información para los empleados al suministrarles un portal de autoservicio lo que genera una mejor imagen a los empleados

b. Gestión de personal: aquí están todas las actividades relacionadas con el colaborador, desde su entrada hasta su salida, incluyendo su desarrollo dentro de la organización, beneficios recibidos y pagos realizados. Es importante este proceso ya que permite a la empresa el cumplimiento normativo laboral, el correcto uso a los datos sensibles de los empleados y la generación de reportes para entes de control como seria el IESS, Ministerio de Trabajo o Servicio de rentas internas.

2.1.3. *Conexiones entre variables*

Las conexiones entre las variables independientes y dependientes se detallan en la tabla 1.

Tabla 1

Conexiones

Variable Independiente	Variable Dependiente	Resultado Esperado
• Lenguajes aplicados (HTML5, CSS3, JavaScript, Python, Django, PostgreSQL)	Tareas Administrativas	• Reducción del 40-50% en tiempo de búsqueda de información • Actualización de datos en tiempo real • Generación automática de documentos
• Metodologías aplicadas (SCRUM)	Precisión en los procesos	• Disminución del 60-70% en errores de validación • Eliminación de registros duplicados • Mejora en integridad de datos
• Calidad de código (SOLID, Clean Code)	Desempeño de personal	• Tiempo de respuesta del sistema < 2 segundos • Acceso unificado a historial completo • Automatización libera 15-20 horas/semana
• QA y pruebas unitarias	Selección de personal	• Reducción del 50% en tiempo de validación de documentos • Procesamiento digital de información • Eliminación de archivos físicos
• Flujo de datos (Validación, cálculos automáticos, logs de auditoría)	Precisión en los procesos	• 95-98% de precisión en cálculos • Trazabilidad completa (auditoría) • Reducción del 80% en errores de cálculo

• Flujo de datos	Gestión de personal	• Generación automática de reportes legales • Cumplimiento normativo al 100% • Reducción de multas por incumplimiento
• Estandarización de procesos	Tareas Administrativas	• Reducción del 60% en tiempo de procesos manuales • Uniformidad en ejecución de tareas • Procesos cuantificables y medibles
• Estandarización de procesos	Retención de empleados	• Aumento del 20-30% en satisfacción laboral • Reducción del 15% en rotación de personal • Mejora en percepción de la empresa

2.2. Marco legal

Protección de datos personales en Ecuador

En Ecuador, el artículo 66, numeral 19 de la Constitución de la República, establece "el derecho a la protección de datos de carácter personal, que incluye el acceso y la decisión sobre información y datos de este carácter, así como su correspondiente protección. La recolección, archivo, procesamiento, distribución o difusión de estos datos o información requerirán la autorización del titular o el mandato de la ley" (Dinarp, 2021).

Ley orgánica de protección de datos personales

Con 118 votos a favor, el pleno de la Asamblea Nacional aprobó el 10 de mayo de 2021, la Ley de Protección de Datos Personales. Este cuerpo legal pasó por un amplio proceso de construcción participativa, el cual inició en octubre de 2017 (Dinarp, 2021).

Objeto y finalidad de la ley

El objeto y finalidad de la presente ley es garantizar el ejercicio del derecho a la protección de datos personales, que incluye el acceso y decisión sobre información y datos de

este carácter, así como su correspondiente protección. Para dicho efecto regula, prevé y desarrolla principios, derechos, obligaciones y mecanismos de tutela (Ministerio del Trabajo, 2024).

Ámbito de aplicación

La presente ley se aplicará al tratamiento de datos personales contenidos en cualquier tipo de soporte, automatizados o no, así como a toda modalidad de uso posterior según lo establecido (Ministerio del Trabajo, 2024).

Principios fundamentales

La ley establece principios clave para el tratamiento de datos personales:

Licitud y lealtad: Todo tratamiento debe realizarse de forma lícita y leal.

Finalidad: Los datos deben ser recogidos para finalidades determinadas, explícitas y legítimas.

Proporcionalidad: El tratamiento debe ser adecuado, pertinente y limitado.

Calidad: Los datos personales deben ser íntegros, precisos, completos, comprobables, claros; y, de ser el caso, debidamente actualizados; de tal forma que no se altere su veracidad (Telecomunicaciones, 2023).

Confidencialidad: Sigilo y secreto en el manejo de la información (Telecomunicaciones, 2023).

Implicaciones para sistemas de RH

Para garantizar el cumplimiento de la LOPDP, las empresas en Ecuador deben implementar medidas de seguridad y privacidad de datos personales desde el diseño y por defecto (Russell Bedford, 2024).

Las empresas que manejan sistemas de recursos humanos deben considerar:

- **Consentimiento informado:** El consentimiento del titular deberá reflejar de manera indubitada la aceptación de éste en relación con el tratamiento de sus datos personales

a través de una declaración, pronunciamiento para darse de baja o clara acción afirmativa (Ministerio del Trabajo, 2024).

- **Designación de delegado de protección de datos:** Para organizaciones que manejan grandes volúmenes de datos personales.
- **Evaluaciones de impacto:** Especialmente para tratamientos que puedan presentar alto riesgo para los derechos de los titulares.

La normativa ecuatoriana, particularmente la Ley Orgánica de Protección de Datos Personales, establece un contexto regulatorio que no solo requiere cumplimiento, sino que también presenta oportunidades para marcar una diferencia competitiva a través de prácticas responsables de manejo de datos.

La combinación de estos factores como son: necesidad organizacional, capacidad tecnológica y marco regulatorio, crean el escenario ideal para el desarrollo e implementación de sistemas web de gestión de recursos humanos como el propuesto para ColaboraSoft SAS.

CAPÍTULO III.
MARCO METODOLÓGICO

3.1. Tipo de investigación

3.1.1. Según el propósito

Aplicada: El proyecto se enmarcar en la línea de investigación aplicada, ya que toma el conocimiento teórico hacia una solución práctica. A partir de los fundamentos de la administración de personal y los sistemas de información se desarrolla para la empresa ColaboraSoft el prototipo de una aplicación web, el proyecto busca demostrar como la teoría de desarrollo web se puede convertir en un producto útil para esta empresa y que incluso se puede replicar en otras empresas.

3.1.2. Según la manipulación de variables

No experimental: el presente trabajo adopta un enfoque no experimental, debido a que no se trabajara con variables en entornos controlados y en su lugar se observara el comportamiento del prototipo realizado en la empresa ColaboraSoft.

3.1.3. Según el tiempo

Transversal: esta investigación se desarrolla bajo un diseño transversal debido a que la recolección de datos y la evaluación del prototipo se llevara a cabo en un punto especifico de tiempo que para este caso es tercer trimestre del 2025

3.2. Tipo de Enfoque

El enfoque del proyecto será mixto; Cualitativo porque se realizaran entrevistas al personal de talento humano, se harán con ellos las pruebas de usabilidad que vayan generando los sprint y adicional se validara el cumplimiento normativo; Cuantitativo porque con el mismo personal y de las transacciones de ellos se tomaran tiempo de respuesta del sistema, se evaluara la reducción de horas mensuales en tareas, se evaluara el porcentaje de reducción de errores y se validara la encuesta enviada inicialmente.

3.3. Herramientas metodológicas

La herramienta metodológica que se adopta para este proyecto es el “Estudio de Caso”, se toma esta herramienta ya que se quiere comprender como la implementación de la aplicación Web para la administración de personal puede influir en la eficiencia de la empresa y también porque es el clásico ejemplo de pequeña empresa que comparte con las mismas de su categoría falencias como falta de tecnología, procesos manuales y demasiadas horas en tareas administrativas, con este enfoque se puede generar el aprendizaje para llevarlo a empresas similares.

3.4. Técnicas

Para este proyecto se realizará una encuesta antes del inicio del desarrollo del prototipo con el fin de poder entender los procesos actuales y problemática específica, con esta información se evaluará la factibilidad del implementar la aplicación web.

3.5. Estudios iniciales

3.5.1. Criterios de inclusión y exclusión

3.5.5.1. Criterios de inclusión. En la encuesta que se realizó se incluyó a todo el personal de talento humano y al personal que tiene relación con sus procesos por ejemplo área contable y tributaria.

3.5.5.2. Criterios de exclusión. Se excluyo a los empleados que no tienen ningún tipo de relación con los procesos de talento humano.

3.5.2. Encuesta previa realizada

Se realiza por Google forms, la estructura de la encuesta se detallada en la tabla 2.

Tabla 2

Encuesta

Sección	Preguntas	Tipo de respuesta
Situación Actual	¿Utiliza hojas de cálculo (Excel, Google Sheets) para almacenar información de los empleados? ¿La información que gestiona incluye datos sensibles de los empleados? (salarios, datos médicos, información personal, etc.) ¿Le resulta complicado actualizar la información de los empleados con el método actual? ¿Ha experimentado errores o reprocesos al gestionar la información de los empleados?	Si / No
Necesidades del Sistema	¿Considera que un sistema web de gestión de talento humano mejoraría su trabajo? ¿Tener la información centralizada en un solo sistema le ahorraría tiempo? ¿Un sistema automatizado reduciría los errores y reprocesos en su trabajo? ¿Un portal de autoservicio para empleados reduciría las consultas frecuentes sobre roles, vacaciones y certificados? ¿Conoce la Ley Orgánica de Protección de Datos Personales de Ecuador?	Si / No
Preguntas abiertas	Además de las funciones básicas (registro de empleados, roles de pago, vacaciones), ¿qué otras características o herramientas le gustaría que tuviera el sistema web para facilitar su trabajo diario? Ejemplos: notificaciones automáticas, reportes personalizados, firma digital, etc. ¿Qué proceso de gestión de RRHH le toma más tiempo actualmente y le gustaría automatizar?	Respuesta abierta

3.5.3. Muestreo

El muestreo que se realizo es no probabilístico por conveniencia ya que la encuesta no está dirigida solamente a persona de talento humano de la empresa, sino que también se envió a empleados de otras áreas.

Se entrevistaron 16 colaboradores, se tendrá en cuenta el margen de error del 5% y un margen de confianza del 95%

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

Donde:

n= tamaño de la muestra

N= tamaño de la población

Z= valor de Z según el nivel de confianza (1.96 para el 95%)

p= probabilidad de éxito (0.5 si no se conoce)

q= probabilidad de fracaso (1-p=0.5)

e= margen de error aceptado

Valores:

N=16

Z=1.96

p= 0.5; q=0.5

e=0.5

$$n = \frac{16 * 1.96^2 * 0.5 * 0.5}{\left(0.5^2 * (16-1)\right) + (1.96^2 * 0.5 * 0.5)}$$

$$n = \frac{16 * 3.8416 * 0.25}{(0.25 * 15) + (3.8416 * 0.25)}$$

$$n = \frac{15.36464}{3.75 + 0.9604}$$

$$n = \frac{15.36464}{4.7104}$$

$$n = 3.26185462$$

El tamaño de la muestra necesario es aproximadamente 3.26 lo que se redondea a 4 participantes para asegurar representatividad

3.5.4. Organizador de variables

Las variables generadas para la encuesta son de tipo cualitativas, detalladas en la tabla 3

Tabla 3

Organizador de variables

Dimensión	Variable	Tipo	Indicador
Situación Actual	Uso de hojas de cálculo	Cualitativa	Herramienta utilizada para almacenar información de empleados
	Gestión de datos sensibles	Cualitativa	Presencia de información confidencial en el sistema actual
	Dificultad para actualizar información	Cualitativa	Percepción de complejidad en el proceso de actualización
	Errores o reprocesos	Cualitativa	Frecuencia de fallos en la gestión de datos
Necesidades del Sistema	Percepción de mejora con sistema web	Cualitativa	Valoración del impacto potencial del sistema en el trabajo
	Centralización de información	Cualitativa	Ahorro de tiempo por consolidación de datos
	Reducción de errores con automatización	Cualitativa	Expectativa de disminución de reprocesos
	Portal de autoservicio	Cualitativa	Reducción de consultas frecuentes por parte de empleados

	Conocimiento de la LOPDP	Cualitativa	Familiaridad con la normativa de protección de datos
Preguntas abiertas	Funcionalidades deseadas	Cualitativa	Herramientas adicionales que facilitarían el trabajo diario
	Procesos que se desean automatizar	Cualitativa	Actividades de RRHH que demandan mayor tiempo y esfuerzo

3.6. Desarrollo

Se describe las herramientas que se usaron para el desarrollo del prototipo del sistema de administración de personal, se detallaran las herramientas usadas, la arquitectura del sistema, el diseño ux/ui que se usó, la base de datos con sus tablas relacionadas y la parte de lógica de negocio (backend) y interface de usuario (frontend), de igual manera se detalla las pruebas realizadas.

3.6.1. Matriz de Operacionalización de Variables

Antes de iniciar con el desarrollo del sistema web es necesario definir la matriz de operacionalización de variables que se plantearon en el marco teórico y que van a tener relación en la construcción del prototipo, para validar la propuesta se elaboró una encuesta que está conformada de 9 preguntas cerradas con escala nominal (Si/No) y 2 preguntas abiertas destinadas a obtener información por parte de los usuario en las funcionalidades que esperarían de un sistema y los procesos que quisieran automatizar, a continuación en la tabla 4 se presentan las variables operacionales.

Tabla 4

Operación de variables

Variable	Definición Conceptual	Definición Operacional	Escala de Medición
Uso de hojas de cálculo	Herramientas digitales utilizadas para almacenar datos de empleados	Respuesta a la pregunta sobre uso de Excel o Google Sheets	Nominal (Si/No)
Gestión de datos sensibles	Inclusión de información confidencial en los registros de empleados	Respuesta sobre presencia de datos como salarios, salud, etc.	Nominal (Si/No)
Dificultad para actualizar información	Nivel de complejidad percibido al modificar datos de empleados	Respuesta sobre si el método actual resulta complicado	Nominal (Si/No)
Errores o reprocesos	Fallos en la gestión que requieren corrección o repetición de tareas	Respuesta sobre experiencia con errores o reprocesos	Nominal (Si/No)
Mejora con sistema web	Percepción sobre el impacto positivo de un sistema digital en la gestión de RRHH	Respuesta sobre si considera útil implementar un sistema web	Nominal (Si/No)
Centralización de información	Valoración del beneficio de tener todos los datos en un solo sistema	Respuesta sobre ahorro de tiempo por centralización	Nominal (Si/No)
Reducción de errores con automatización	Expectativa de disminución de fallos mediante procesos automáticos	Respuesta sobre si la automatización reduciría errores	Nominal (Si/No)
Portal de autoservicio	Herramienta que permite a los empleados consultar información sin intermediarios	Respuesta sobre si reduciría consultas frecuentes	Nominal (Si/No)
Conocimiento de la LOPDP	Familiaridad con la Ley Orgánica de Protección de Datos Personales	Respuesta sobre si conoce la normativa	Nominal (Si/No)
Funcionalidades deseadas	Herramientas adicionales que mejorarían la experiencia de uso del sistema	Respuesta abierta sobre características deseadas	Abierta
Procesos que se desean automatizar	Actividades que actualmente demandan más tiempo y esfuerzo en RRHH	Respuesta abierta sobre procesos que se desea digitalizar	Abierta

3.6.2. Herramientas

Las herramientas que se usaron en el desarrollo se detallan en la tabla 5, estas herramientas permiten un stack tecnológico moderno y robusto, seleccionado por su escalabilidad, seguridad y eficiencia

Tabla 5

Herramientas

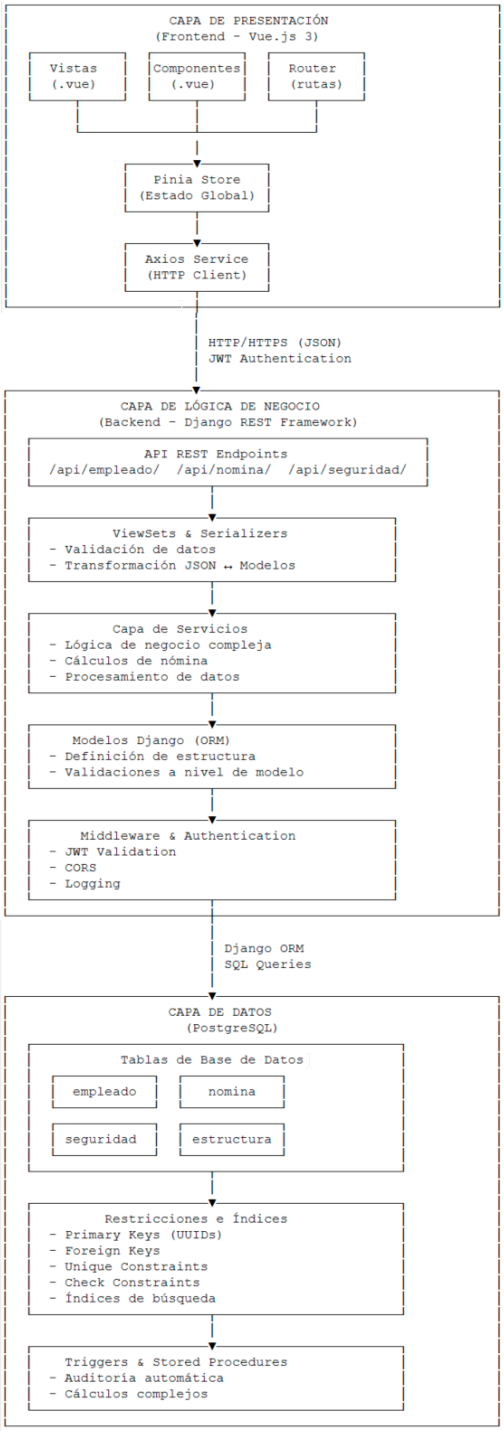
Herramienta	Detalle
PyCharm 2025.2.4	Editor de código para el backend y seguridad JWT
WebStorm 2025.2.4	Editor de código para el frontend
pgAdmin4 V 9.8	Motor de base de datos
GitHub Desktop V3.5.4	Control de versiones del proyecto
Insomia V 12.0.0	Herramienta para probar APIs REST

3.6.3. Arquitectura

El desarrollo implementa arquitectura de tres capas que garantiza una separación clara de responsabilidad de cada aplicación o componente para una escalabilidad, mantenibilidad y seguridad más rápida, la figura 3 detalla las capas de la arquitectura.

Figura 3

Arquitectura



3.6.4. Backend

La capa lógica de la aplicación se desarrolla con el lenguaje de Python en su versión 3.13.5 y como framework se usa Django en su versión 5.2.6, la arquitectura del backend es REST API usando la librería Django REST Framework en ambientes virtuales para evitar problemas de versiones en el desarrollo como se detalla en la tabla 6.

Tabla 6

Tecnología Backend

Tecnología	Versión / detalle
Python	3.13.5
Django	5.2.6
Django REST Framework	Instalado
Entorno virtual	virtualenv
Paquetes adicionales	pytz, PyYAML

La estructura backend del proyecto se muestra en la figura 4:

Figura 4

Backend

```
backend/
├── assets/        # Archivos estáticos del proyecto
├── backend/       # Configuración principal de Django (settings, wsgi, asgi)
├── empleado/      # App de gestión de empleados
├── nomina/        # App de gestión de nómina
├── seguridad/     # App relacionada con autenticación/seguridad
├── utilidades/    # Scripts o funciones de utilidad general
├── .gitignore     # Archivos ignorados por git
├── manage.py      # Script de gestión de Django
└── requirements.txt # Dependencias del proyecto Python
```

El siguiente es el detalle de las carpetas que están en la estructura del backend:

1. backend: contiene los archivos principales para el funcionamiento.
    a. settings.py: es el archivo principal del backend y contiene las configuraciones necesarias para que el proyecto se pueda desplegar.
        i. INSTALLED_APPS: contiene el registro de todas las aplicaciones instaladas y usadas en el proyecto
        ii. DATABASES: configura la conexión a la base de datos
        iii. MIDDLEWARE: define las capas de seguridad y procesamiento de las peticiones (CORS)
        iv. REST_FRAMEWORK: tiene la configuración de la paginación y autenticación JWT
        v. SIMPLE_JWT: tiene la configuración del manejo de tokens de acceso
    b. urls.py: es el enrutador principal de la aplicación y define a que vista se dirige al visitar una URL, adicional dispone la configuración de Swagger para la documentación de la aplicación
    c. views.py: Contiene la lógica de las vistas globales o de autenticación que no pertenecen a una app específica, para este proyecto las vistas están manejadas en el frontend
        i. Login/Logout: Maneja el inicio y cierre de sesión clásico.
        ii. user_info: Un endpoint para obtener la información del usuario autenticado actual.

iii. csrf_cookie: Un endpoint para obtener el token CSRF

2. empleado: esta aplicación sirve para gestionar los empleados y sus procesos, contiene los siguientes archivos

    a. models.py: en este archivo se define la estructura de los datos por medio de clases que representan las tablas en la base de datos, para esta aplicación se crearon varias clases que se detallan a continuación en la tabla 7:

Tabla 7

Modelos aplicación empleado

Nombre de Clase	**Comentario**
identificacion	Guarda los datos básicos de los empleados
empleado	Guarda los datos por periodos trabajados de los empleados
empleado_beneficio	Guarda la preferencia del empleado para el pago de beneficios (d3, d4, FR)
empleado_concepto_mes	Guarda el valor de conceptos que puede recibir mensualmente el empleado
empleado_deduccion	Guarda las deducciones de impuesto a la renta que puede hacer un empleado
empleado_renta	Guarda el valor mensual de cada concepto que afecta impuesto a la renta
empleado_vacacion	Guarda los periodos de vacaciones del empleado
empleado_vacacion_solicitud	Guarda las solicitudes de vacaciones que puede hacer un empleado

b. serializers.py: este archivo cumple la función de ser el puente entre la base de datos y las API, los serializers de esta aplicación se detallan en la tabla 8:

Tabla 8

Serializers aplicación empleado

Nombre de Clase
IdentificacionSerializer
EmpleadoSerializer
Empleado_BeneficioSerializer
Empleado_Concepto_MesSerializer
Empleado_DeduccionSerializer
Empleado_RentaSerializer
Empleado_VacacionSerializer
Empleado_Vacacion_SolicitudSerializer

c. urls.py: en este archivo se definen las rutas de la aplicación web, las urls creadas para la aplicación empleado se detallan en la tabla 9:

Tabla 9

URLS aplicación empleado

Ruta
path('identificacion/', views.identificacion_list_create, name='identificacion-list-create'), path('identificacion/<uuid:pk>/', views.identificacion_detail, name='identificacion-detail'), path('empleado/', views.empleado_list_create, name='empleado-list-create'), path('empleado/<uuid:pk>/', views.empleado_detail, name='empleado-detail'), path('empleado_beneficio/', views.empleado_beneficio_list_create, name='empleado_beneficio-list-create'), path('empleado_beneficio/<uuid:pk>/', views.empleado_beneficio_detail, name='empleado_beneficio-detail'), path('empleado_concepto_mes/', views.empleado_concepto_mes_list_create, name='empleado_concepto_mes-list-create'), path('empleado_concepto_mes/<uuid:pk>/', views.empleado_concepto_mes_detail, name='empleado_concepto_mes-detail'), path('empleado_deduccion/', views.empleado_deduccion_list_create, name='empleado_deduccion-list-create'), path('empleado_deduccion/<uuid:pk>/', views.empleado_deduccion_detail, name='empleado_deduccion-detail'), path('empleado_renta/', views.empleado_renta_list_create, name='empleado_renta-list-create'), path('empleado_renta/<uuid:pk>/', views.empleado_renta_detail, name='empleado_renta-detail'), path('empleado_vacacion/', views.empleado_vacacion_list_create, name='empleado_vacacion-list-create'),

```
path('empleado_vacacion/<uuid:pk>/', views.empleado_vacacion_detail,
name='empleado_vacacion-detail'),
path('empleado_vacacion_solicitud/', views.empleado_vacacion_solicitud_list_create,
name='empleado_vacacion_solicitud-list-create'),
path('empleado_vacacion_solicitud/<uuid:pk>/', views.empleado_vacacion_solicitud_detail,
name='empleado_vacacion_solicitud-detail'),
```

d. views.py: en este archivo esta la lógica de la aplicación (validaciones, reglas, consultas a la base de datos) y la respuesta a las solicitudes HTTP (GET, POST, PUT, DELETE), las vistas creadas en este archivo se detallan en la tabla 10:

Tabla 10

Views aplicación empleado

Nombre	**Tipo**
identificacion_list_create	GET, POST
identificacion_detail	GET, PUT, DELETE
empleado_list_create	GET, POST
empleado_detail	GET, PUT, DELETE
empleado_beneficio_list_create	GET, POST
empleado_beneficio_detail	GET, PUT, DELETE
empleado_concepto_mes_list_create	GET, POST
empleado_concepto_mes_detail	GET, PUT, DELETE
empleado_deduccion_list_create	GET, POST
empleado_deduccion_detail	GET, PUT, DELETE
empleado_renta_list_create	GET, POST
empleado_renta_detail	GET, PUT, DELETE
empleado_vacacion_list_create	GET, POST
empleado_vacacion_detail	GET, PUT, DELETE
empleado_vacacion_solicitud_list_create	GET, POST
empleado_vacacion_solicitud_detail	GET, PUT, DELETE

3. nomina: esta aplicación contiene las configuraciones básicas de la empresa y los procesos de roles, los archivos principales de esta aplicación son:

    a. models.py: en este archivo se define la estructura de los datos por medio de clases que representan las tablas en la base de

datos, para esta aplicación se crearon varias clases que se detallan en la tabla 11:

Tabla 11

Modelos aplicación nomina

Nombre de Clase	Comentario
empresa	Guarda los datos de la empresa
cargo	Guarda los cargos que maneja la empresa
centrocosto	Guarda los centros de costos de la empresa
estructuraorganica	Guarda la estructura orgánica de la empresa
ubicacionfisica	Guarda las ubicaciones físicas de la empresa
concepto	Guarda los conceptos de nómina que usa la empresa
proceso_rol	Guarda los procesos de rol de la empresa
rol	Guarda los roles de la empresa

a. serializers.py: este archivo cumple la función de ser el puente entre la base de datos y las API, los serializers de esta aplicación se detallan en la tabla 12:

Tabla 12

Serializer aplicación nomina

Nombre de Clase
UsuarioRegistroSerializer
EmpresaSerializer
CargoSerializer
CentroCostoSerializer
EstructuraOrganicaSerializer
UbicacionFisicaSerializer
ConceptoSerializer
ProcesoRolSerializer
RolSerializer

b. urls.py: en este archivo se definen las rutas de la aplicación web, las urls creadas para la aplicación empleado se detallan en la tabla 13:

Tabla 13

URLS aplicación nomina

Ruta
path('', include(router.urls)), path('registrar/', registrar_usuario), path('empresas/', views.empresa_list_create, name='empresa-list-create'), path('empresas/<uuid:pk>/', views.empresa_detail, name='empresa-detail'), path('cargos/', views.cargo_list_create, name='cargo-list-create'), path('cargos/<uuid:pk>/', views.cargo_detail, name='cargo-detail'), path('centrocostos/', views.centrocosto_list_create, name='centrocosto-list-create'), path('centrocostos/<uuid:pk>/', views.centrocosto_detail, name='centrocosto-detail'), path('estructuraorganicas/', views.estructuraorganica_list_create, name='estructuraorganica-list-create'), path('estructuraorganicas/<uuid:pk>/', views.estructuraorganica_detail, name='estructuraorganica-detail'), path('ubicacionfisicas/', views.ubicacionfisica_list_create, name='ubicacionfisica-list-create'), path('ubicacionfisicas/<uuid:pk>/', views.ubicacionfisica_detail, name='ubicacionfisica-detail'), path('conceptos/', views.concepto_list_create, name='concepto-list-create'), path('conceptos/<uuid:pk>/', views.concepto_detail, name='concepto-detail'), path('procesos_rol/', views.proceso_rol_list_create, name='procesos_rol-list-create'), path('procesos_rol/<uuid:pk>/', views.proceso_rol_detail, name='procesos_rol-detail'),

c. views.py: en este archivo esta la lógica de la aplicación (validaciones, reglas, consultas a la base de datos) y la respuesta a las solicitudes HTTP (GET, POST, PUT, DELETE), las vistas creadas en este archivo se detallan en la tabla 14:

Tabla 14

Views aplicación nomina

Nombre	Tipo
empresa_list_create	GET, POST
empresa_detail	GET, PUT, DELETE
cargo_list_create	GET, POST
cargo_detail	GET, PUT, DELETE
centrocosto_list_create	GET, POST
centrocosto_detail	GET, PUT, DELETE
estructuraorganica_list_create	GET, POST
estructuraorganica_detail	GET, PUT, DELETE
ubicacionfisica_list_create	GET, POST
ubicacionfisica_detail	GET, PUT, DELETE
concepto_list_create	GET, POST
concepto_detail	GET, PUT, DELETE
proceso_rol_list_create	GET, POST
proceso_rol_detail	GET, PUT, DELETE
rol_list_create	GET, POST
rol_detail	GET, PUT, DELETE

4. utilidades: esta aplicación sirve para él envió de email como notificaciones de procesos, maneja el archivo utilidades.py que tiene la función propia de Python para el envío de emails construyendo un mensaje con asunto, remitente destinatario y cuerpo en html, se conecta mediante SMTP al servidor usando credenciales guardadas en variables de entorno.

3.6.5. Frontend

La capa de presentación o frontend es la interfaz con la que interactúa el usuario final, esta capa es responsable de:

- Renderizar la interfaz de usuario de forma reactiva y responsive
- Capturar la entrada del usuario mediante formularios, botones y controles
- Validar datos del lado del cliente antes de enviarlos al servidor

- Gestionar la navegación entre diferentes vistas y componentes
- Presentar información de manera clara, organizada y visualmente atractiva
- Manejar el estado de la aplicación en el cliente
- Gestionar la autenticación y el almacenamiento de tokens

Las tecnologías y versiones que se usaron para el Frontend se detallan en la tabla 15

Tabla 15

Tecnología Frontend

Herramienta	Versión	Detalle
Vue.js 3	3.5.22	Framework JavaScript progresivo para construcción de interfaces de usuario reactivas
Vue Router	4.5.1	Manejo de rutas y navegación entre vistas
Pinia	3.0.3	Gestión de estado global de la aplicación
Axios	1.13.2	Cliente HTTP para comunicación con el backend
Bootstrap 5	5.3.8	Framework CSS para diseño responsive y componentes UI
Bootstrap Icons	1.11.0	Librería de iconos
Vite	7.1.7	Herramienta de construcción rápida con HMR (Hot Module Replacement)

Los principales componentes son:

- Views: contiene las vistas principales de la aplicación que se detallan en la tabla 16:

Tabla 16

Vistas Principales

Nombre de la vista	Comentario
Error404.vue	Pantalla que se muestra al suceder un error
FormCentroCosto.vue	Pantalla para el ingreso o edición de centros de costos
FormConcepto.vue	Pantalla para el ingreso o edición de conceptos de nomina

Formempleado.vue	Pantalla para el ingreso o edición de empleados de la empresa
Formempresa.vue	Pantalla para el ingreso o edición de la empresa
FormEstructuraOrganica.vue	Pantalla para el ingreso o edición de la estructura organica
FormProcesoRol.vue	Pantalla para el ingreso o edición de los procesos de rol
FormUbicacionfisica.vue	Pantalla para el ingreso o edición de ubicaciones físicas
IdentificacionView.vue	Pantalla para el ingreso o edición de la identificacion del empleado
Login.vue	Pantalla para el ingreso a la aplicación

- Services: contiene los servicios que relacionan el backend con las vistas del frontend, estos servicios se detallan en la tabla 17

Tabla 17
Servicios de frontend

Nombre del servicio	**Comentario**
api.js	manejo de seguridad
auth.js	manejo de seguridad
axios.js	manejo de seguridad
cargo.js	manejo de cargos
centrocosto.js	manejo de centrocostos
concepto.js	manejo de conceptos
empleado.js	manejo de empleados
empresa.js	manejo de empresas
estructuraorganica.js	manejo de estructuras orgánicas
identificacion.js	manejo de identificaciones
procesorol.js	manejo de procesos de rol
ubicacionfisica.js	manejo de ubicaciones fisicas

3.6.6. Base de datos y relación de las tablas

La herramienta que se usó para la base de datos es PostgreSQL en su versión 9.8, por medio del ORM propio de Python se realizó la creación de las tablas en la base de datos, se detalla una descripción del modelo de datos implementado.

- Principios de diseño aplicados:

  - Llave primaria: `UUIDField` en todas las tablas para evitar colisiones y facilitar replicación/sharding.

- Normalización: catálogos (empresa, cargo, centrocosto, estructuraorganica, ubicacionfisica, concepto, proceso_rol) separados de las entidades transaccionales.

- Enumeraciones: campos de estado y banderas (`S`/`N`) modelados como `CharField` con `choices` (equivalente a dominios controlados en SQL).

- Integridad referencial: FKs con `models.DO_NOTHING` (no borra en cascada). Implica responsabilidad de control de integridad a nivel de aplicación o por políticas de negocio.

- Tablas de app nomina
  - Tabla `empresa`

    - PK: `id (UUID)`

    - Atributos: `razonsocial`, `ruc`, `direccion`, `telefono`, `email`.

    - Uso: catálogos; elegible para relacionarse con procesos y/o empleados (en el código actual aún no hay FK directa desde `empleado`).
  - Tabla `cargo`

    - PK: `id (UUID)`

    - Atributos: `nombre`, `sectorial` (clasificador/agrupador).

    - Relación: 1 cargo — N empleados (FK `empleado.idcargo`).
  - Tabla `centrocosto`

    - PK: `id (UUID)`

    - Atributos: `nombre`, `codigo`, `estado (A/I)`.

    - Relación: 1 centro — N empleados (FK `empleado.idcentrocosto`).
  - Tabla `estructuraorganica`

- PK: `id (UUID)`

- Atributos: `nombre`, `codigo`, `estado (A/I)`.

- Relación: 1 estructura — N empleados (FK `empleado.idestructuraorganica`).

- Tabla `ubicacionfisica`

  - PK: `id (UUID)`

  - Atributos: `nombre`, `codigo`, `estado (A/I)`.

  - Relación: 1 ubicación — N empleados (FK `empleado.idubicacionfisica`).

- Tabla `concepto`

  - PK: `id (UUID)`

  - Atributos clave: `tipo (I/E/P)`, `nombre`, `codigo`, banderas (`aplica_ir`, `proyectable`, `sumabaseiess`, `sumabase`), `identificador`, `carillaIR`, `formula`, `comentario`, `estado`.

  - Uso: conceptos de rol (ingresos, egresos, provisiones). Se vincula a empleados a través de tablas transaccionales como `empleado_concepto_mes` y `empleado_deduccion`.

- Tabla `proceso_rol`

  - PK: `id (UUID)`

  - Atributos: `nombre`, `tipo (Q/R/D3/D4/UT)`, `fechainicio`, `fechafin`, `estado`, `comentario`.

  - Uso: representa periodos/cortes de cálculo de nómina; se referencia desde movimientos del empleado (p. ej., `empleado_concepto_mes`, `empleado_renta`).

- Tablas de app empleado
    - Tabla `identificacion`

      - PK: `id (UUID)`

      - Atributos: `tipoidentificacion (C/P/R)`, `identificacion` (número), `nombres`, `apellidos`, `genero (M/F/O)`, `fecha_nacimiento`.

      - Relación: 1 identificación — 1..N empleados (actualmente 1-a-N por modelo; en negocio lo usual es 1-a-1).

    - Tabla `empleado`

      - PK: `id (UUID)`

      - FKs: `id_identificacion → identificacion`, `idcentrocosto → nomina.centrocosto`, `idestructuraorganica → nomina.estructuraorganica`, `idubicacionfisica → nomina.ubicacionfisica`, `idcargo → nomina.cargo`.

      - Atributos: `imagen`, `direccion`, `barrio`, `telefono`, `celular`, `email`, `tiporol (E/P/O)`, `fecha_ingreso`, `fecha_salida (nullable)`, `sueldo`, `tipocontrato (I/T)`, `jefeinmediato`, `estado (A/I)`.

      - Cardinalidades principales:

        - 1 centro de costo — N empleados.

        - 1 estructura orgánica — N empleados.

        - 1 ubicación física — N empleados.

        - 1 cargo — N empleados.

        - 1 identificación — N empleados (ver nota anterior).

    - Tabla `empleado_beneficio`

- PK: `id (UUID)`

- FK: `id_empleado → empleado`

- Atributos: banderas de cálculo `d3`, `d4`, `fr` (Mensualiza/Provisiona), `cargas (int)`.

- Cardinalidad: 1 empleado — 0..N registros de beneficio (según versiones históricas si se modela así; el modelo no fuerza unicidad 1-a-1).

- Tabla `empleado_concepto_mes`

  - PK: `id (UUID)`

  - FKs: `id_empleado → empleado`, `id_proceso_rol → nomina.proceso_rol`, `id_concepto → nomina.concepto`.

  - Atributos: `valor (decimal)`, `fecha_carga`.

  - Cardinalidad: N registros por combinación empleado-periodo-concepto (no hay restricción única definida; se recomienda índice único si el negocio lo requiere).

- Tabla `empleado_deduccion`

  - PK: `id (UUID)`

  - FKs: `id_empleado → empleado`, `id_concepto → nomina.concepto`.

  - Atributos: `periodofiscal (texto)`, `valor (decimal)`.

  - Cardinalidad: N deducciones por empleado y concepto por período fiscal (no hay restricción única).

- Tabla `empleado_renta`

  - PK: `id (UUID)`

  - FKs: `id_empleado → empleado`, `id_proceso_rol → nomina.proceso_rol`.

  - Atributos: `fecha`, `dias (int)`, `casilla (texto)`, `valor (decimal)`.

- Tabla `empleado_vacacion`

  - PK: `id (UUID)`

  - FK: `id_empleado → empleado`

  - Atributos: `fechadesde`, `fechahasta`, métricas de días y provisiones (`diasnormales`, `provisionnormal`, `diasadicionales`, `provisionadicional`, `diastomados`, `saldodias`, `saldoprovision`).

  - Uso: saldos/acumulaciones de vacaciones.

- Tabla `empleado_vacacion_solicitud`

  - PK: `id (UUID)`

  - FK: `id_empleado → empleado`

  - Atributos: `fechadesde`, `fechahasta`, `diassolicitados`, `comentario`, `id_supervisor (int)`, `fecha_aprobacion`, `estado (A/R)`, `comentario_aprobacion`.

  - Uso: workflow de aprobación de vacaciones.

- Diagrama lógico

  - empleado(id) → FK id_identificacion → identificacion(id)

  - empleado(id) → FK idcentrocosto → centrocosto(id)

  - empleado(id) → FK idestructuraorganica → estructuraorganica(id)

- empleado(id) → FK idubicacionfisica → ubicacionfisica(id)

- empleado(id) → FK idcargo → cargo(id)

- empleado_beneficio(id) → FK id_empleado → empleado(id)

- empleado_concepto_mes(id) → FKs: id_empleado → empleado, id_proceso_rol → proceso_rol, id_concepto → concepto

- empleado_deduccion(id) → FKs: id_empleado → empleado, id_concepto → concepto

- empleado_renta(id) → FKs: id_empleado → empleado, id_proceso_rol → proceso_rol

- empleado_vacacion(id) → FK id_empleado → empleado

- empleado_vacacion_solicitud(id) → FK id_empleado → empleado

3.6.7. *Diseño UX/UI*

Este capítulo aborda el diseño de interfaz de usuario (UI) y la experiencia de usuario (UX) como elementos fundamentales para asegurar la adopción, facilidad de uso y efectividad de la solución implementada, se expondrá la metodología empleada y los patrones de diseño aplicados para convertir los requerimientos funcionales en una interfaz clara, accesible y coherente para el usuario final.

- Formularios Maestro – Detalle
    - Se creo una paginación y filtrado que funciona en tiempo real lo que beneficia el rendimiento porque solo trae los datos necesarios desde la API según lo que el usuario este viendo.
    - Los formularios permiten la edición directa ya que son modales que aparecen justo en la vista, las validaciones están directamente puestas

en el navegador para asegurar que los datos lleguen bien a la base de datos.

  - En caso de errores tipo 400, 401, 403, 500 se visualizarán diferente y claramente para que el usuario sepa que está pasando en la aplicación.

- Usabilidad
  - Se empleo la iconografía estándar de Bootstrap Icons en todas las pantallas, esto hace que todo se sienta consistente y que las acciones aprendidas en una pantalla sean fácilmente repetidas en todas.
  - En acciones largas se colocó un spinner de carga como indicador, esta retroalimentación al usuario es importante para que no crea que el sistema esta colgado y mejore la percepción de velocidad de la aplicación.
  - Se tomo en cuenta el diseño inclusivo con el manejo de contrastes de color adecuados y etiquetas descriptivas, esto facilita el uso inclusive a personas con limitaciones.
- Estructura
  - La configuración de vue-router permite que las rutas estén blindadas lo que permite proteger rutas sensibles.
  - Se uso la persistencia de sesión usando la clave localStorage, esto permite recordar al usuario sin importar si recarga o vuelve más tarde a la aplicación.
- Estética
  - Se crearon variables para el logo y nombre de la compañía lo que permite tener una identidad flexible en caso de nuevos diseños de logo,

comercialización para otras empresas o levantar la aplicación en diferentes ambientes.

3.6.8. Seguridad

La seguridad en las aplicaciones web modernas paso de ser una característica adicional a convertirse en un componente fundamental de la arquitectura, en este capítulo detallaremos las estrategias de seguridad que se implementaron para proteger la integridad del sistema y la privacidad de los datos de los usuarios.

El enfoque que se dio a la seguridad va más allá de proteger al servidor sí que se centró en la autenticación y autorización de los usuarios para saber quién está dentro del sistema y que puede hacer en el mismo, para llevar esto a la práctica se usó JSON Web Token (JWT), en el backend con rest_framework_simplejwt y en el frontend con tokens.

- Backend:
    - Identificación DRF: la clase JWTAuthentication maneja la validación de cada petición
    - Endpoints: se establecieron dos puntos esenciales para el flujo de seguridad
        - /token/: para el inicio de sesión recibiendo usuario y contraseña
        - /token/refresh/: para renovar el token cuando esta por expirar
- Frontend:
    - Almacenamiento: los tokens se guardan localmente en el localStorage
    - Envió: las solicitudes tienen un encabezado Authorization: Bearer que protege todas las solicitudes

En general las vistas que contiene datos sensibles se protegen explícitamente con clases de permiso como IsAuthenticated o IsAdminUser, esto permite ofrecer flexibilidad en

el desarrollo y a su vez garantiza que la información crucial permanezca blindada y solo sea accesible con un token JWT valido.

3.6.9. Pruebas funcionales

Para las pruebas funcionales del desarrollo se usó la herramienta Insomnia en su versión 12.1.0, se detallan las pruebas realizadas:

- Generación de token de seguridad
  - Json utilizado

    {

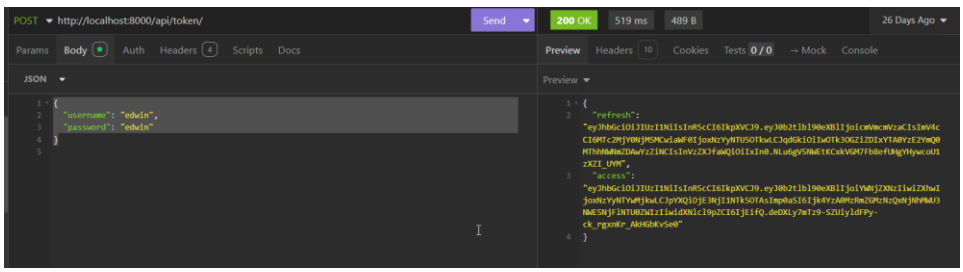
     "username": "edwin",

     "password": "edwin"

    }
  - Pasos:
    - Abrir Insomina
    - Ir al servicio http://localhost:8000/api/token/
    - Ingresar el Json
    - ejecutar
  - Resultado esperado: respuesta del servicio 200OK, generación de token correcta porque el usuario si existe en la base de datos como lo muestra la figura 5.

Figura 5

Insomnia generación de tokens



- Creación de cargo
    - Json utilizado

      {

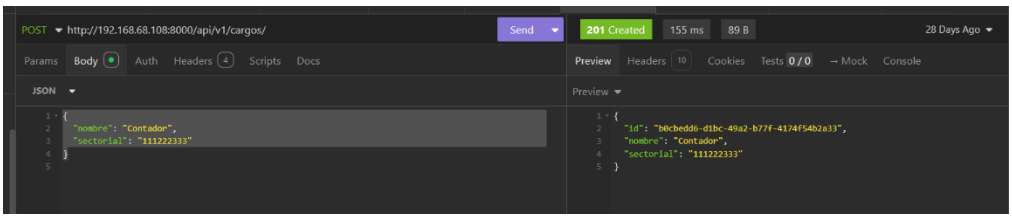
       "nombre": "Contador",

       "sectorial": "111222333"

      }
    - Pasos:
        - Abrir Insomina
        - Ir al servicio POST http://192.168.68.108:8000/api/v1/cargos/
        - Ingresar el Json
        - ejecutar
    - Resultado esperado: respuesta del servicio 200OK, creación de un cargo en la base de datos como lo muestra la figura 6.

Figura 6

Insomnia creación de cargo



- Lectura de cargos
    - Pasos:
        - Abrir Insomina
        - Ir al servicio GET http://192.168.68.108:8000/api/v1/cargos/
        - Ejecutar

- Resultado esperado: respuesta del servicio 200OK, lectura de cargos en la base de datos como lo muestra la figura 7 y 8.

Figura 7

Insomnia consulta de cargos

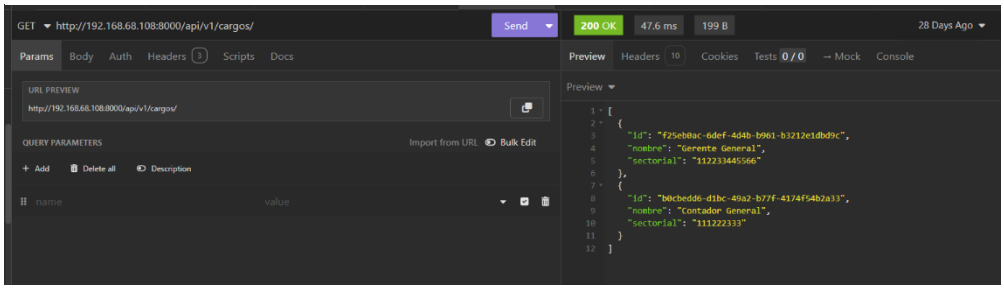
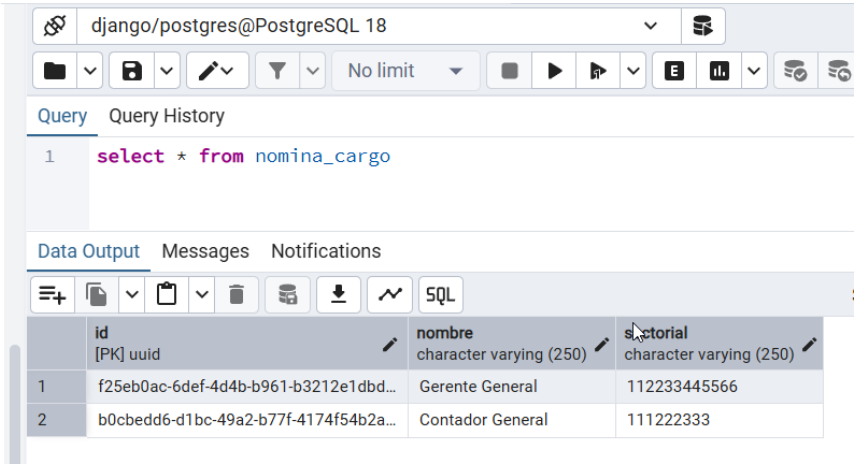


Figura 8

PostgreSQL consulta de cargos



- Creación de identificación
  - Json utilizado

{

"tipoidentificacion": "C",

"identificacion": "17xxxxxxx1",

"nombres": "Diana",

"apellidos": "Yepez",

"genero": "F",

"fecha_nacimiento": "1988-06-12"

}

- Pasos:
  - Abrir Insomina
  - Ir al servicio POST

    http://192.168.68.108:8000/api/v1/identificacion/
  - Ingresar el Json
  - ejecutar
- Resultado esperado: respuesta del servicio 200OK, creación de una identificación en la base de datos como lo muestra en las figuras 9 y 10

Figura 9

Insomnia creación de identificación

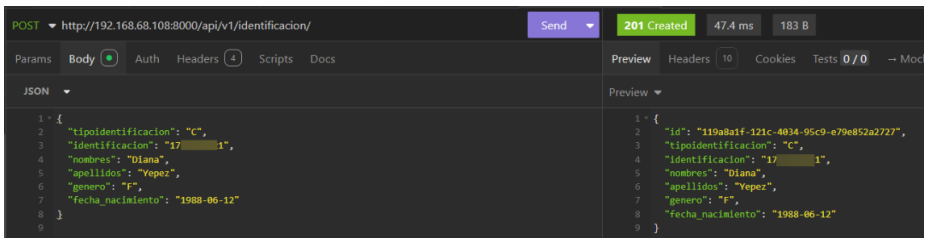
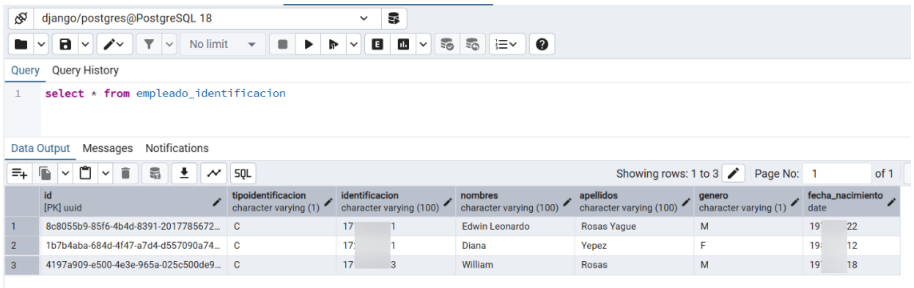


Figura 10

PostgreSQL creación de identificación



- Creación de conceptos de nomina
    - Json utilizado

{
"tipo": "I",
"nombre": "Pago Décimo Tercero",
"codigo": "D3",
"aplica_ir": "N",
"proyectable": "N",
"identificador": "Décimo Tercero",
"sumabaseiess": "N",
"sumabase": "N",
"carillaIR": "308",
"formula": "SB / 12",
"comentario": "Concepto para pagar el décimo tercero mensualizado",
"estado": "A"
}

    - Pasos:
        - Abrir Insomina
        - Ir al servicio POST

          http://192.168.68.108:8000/api/v1/conceptos/
        - Ingresar el Json
        - Ejecutar

- Resultado esperado: respuesta del servicio 200OK, creación de un concepto de nómina en la base de datos como lo muestra las figuras 11 y 12

Figura 11

Insomnia creación concepto de nomina

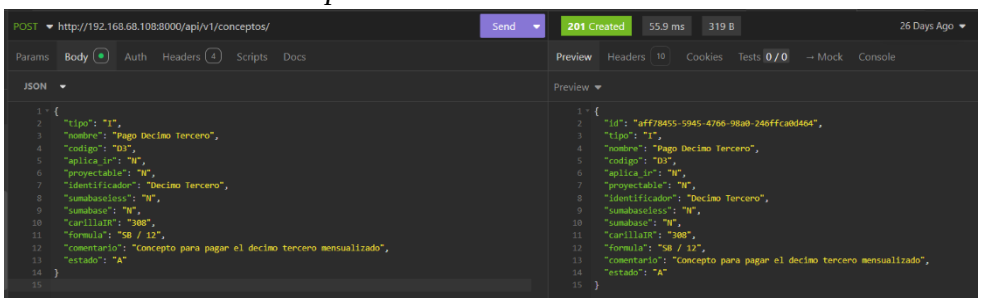
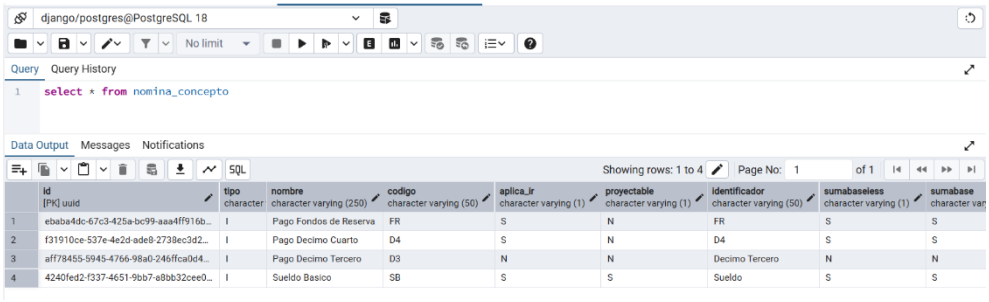


Figura 12

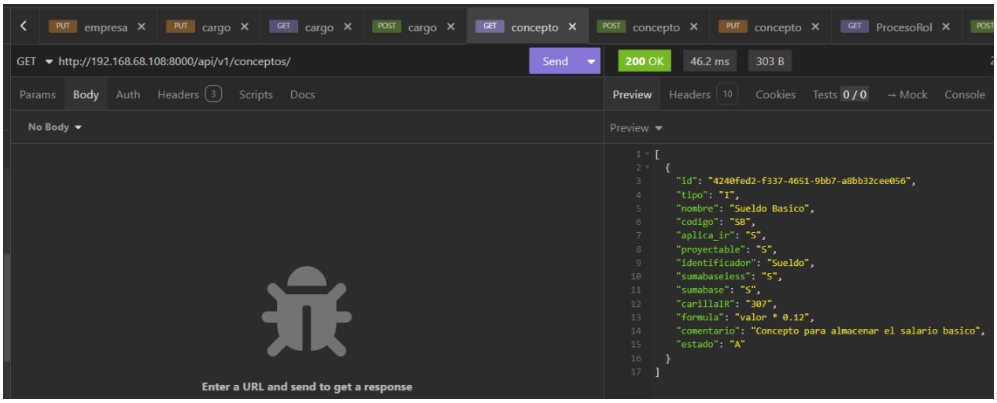
PostgreSQL consulta concepto de nomina



- Lectura de conceptos de nomina
  - Pasos:
    - Abrir Insomina
    - Ir al servicio GET http://192.168.68.108:8000/api/v1/conceptos/
    - Ejecutar
  - Resultado esperado: respuesta del servicio 200OK, creación de un concepto de nómina en la base de datos como lo muestra la figura 13

Figura 13

Insomnia consulta de conceptos de nomina



- Creación de procesos de rol
  - Json utilizado

    {

    "nombre": "Quincena febrero 2025",

    "tipo": "Q",

    "fechainicio": "2025-02-01",

    "fechafin": "2025-02-15",

    "estado": "A",

    "comentario": "Quincena pagada en febrero 2025"

    }
  - Pasos:
    - Abrir Insomina
    - Ir al servicio POST

      http://192.168.68.108:8000/api/v1/procesos_rol/
    - Ingresar el Json
    - Ejecutar

- Resultado esperado: respuesta del servicio 200OK, creación de un proceso de rol en la base de datos como lo muestra las figuras 14 y 15

Figura 14

Insomnia creación de proceso de rol

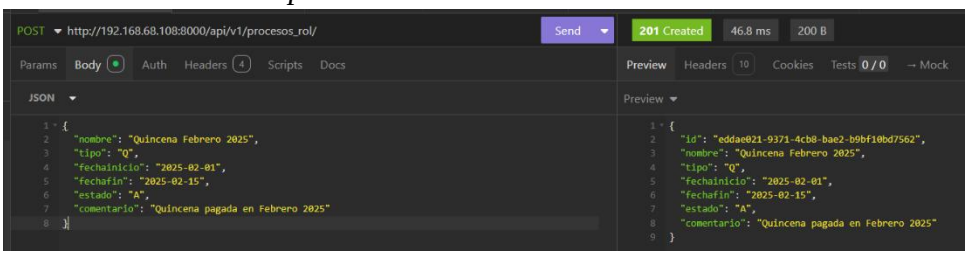
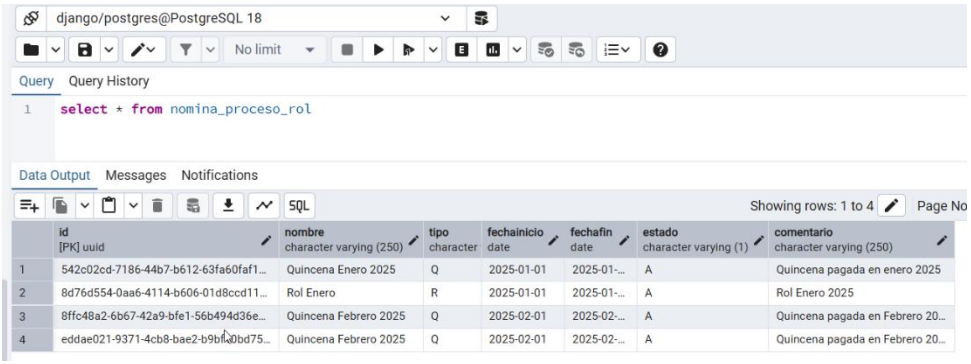


Figura 15

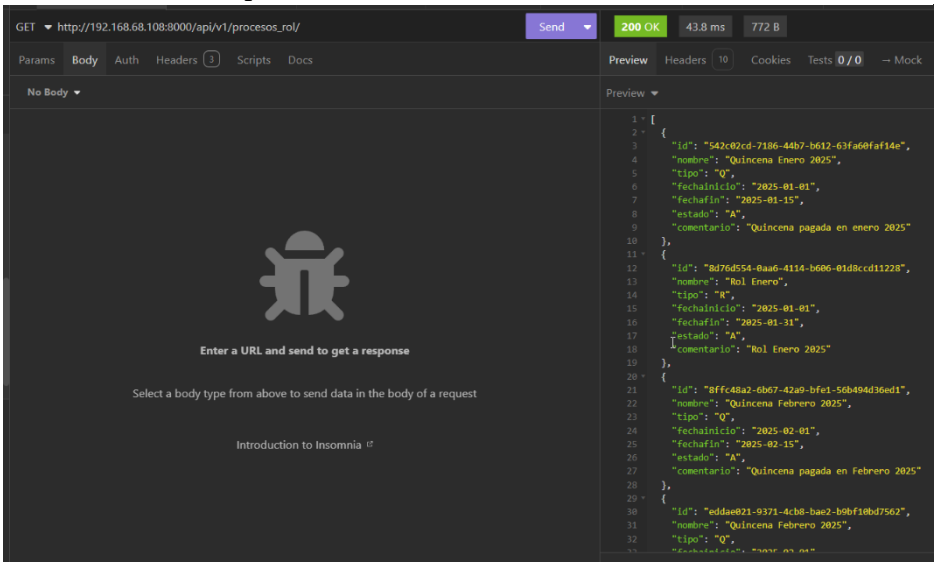
PostgreSQL consulta proceso de rol



- Lectura de procesos de rol
  - Pasos:
    - Abrir Insomina
    - Ir al servicio GET http://192.168.68.108:8000/api/v1/procesos_rol/
    - Ejecutar
  - Resultado esperado: respuesta del servicio 200OK, lectura de un proceso de rol en la base de datos como lo muestra la figura 16

Figura 16

Insomnia consulta proceso de rol



- Creación de empresa
  - Json utilizado

    {

    "razonsocial": "ColaboraSoft",

    "ruc": "1717886541001",

    "direccion": "Av. Amazonas y Naciones Unidas",

    "telefono": "0986805369",

    "email": "erosas@colaborasoft.com"

    }
  - Pasos:
    - Abrir Insomina
    - Ir al servicio POST

      http://192.168.68.108:8000/api/v1/empresas/
    - Ingresar el Json

- Ejecutar

- Resultado esperado: respuesta del servicio 200OK, creación de una empresa en la base de datos como lo muestra la figura 17 y 18

Figura 17

Insomnia creación de empresa

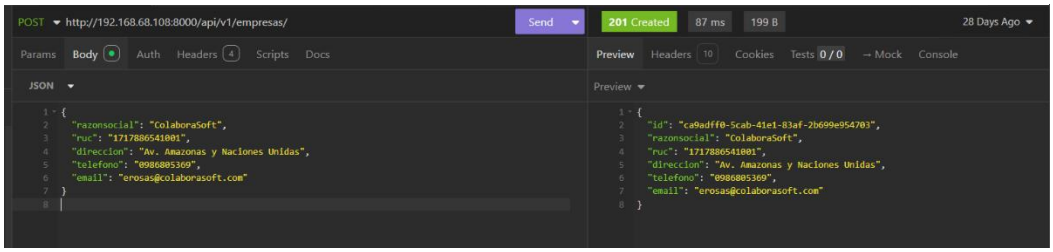


Figura 18

PostgreSQL consulta de empresa



- Lectura de empresa

  - Pasos:

    - Abrir Insomina

    - Ir al servicio GET http://192.168.68.108:8000/api/v1/empresas/

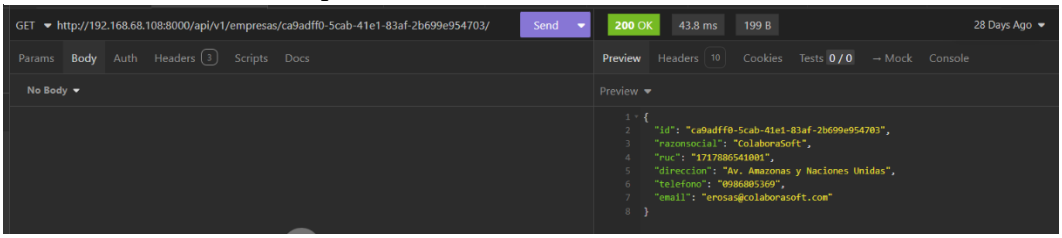
    - Ingresar el Json

    - Ejecutar

  - Resultado esperado: respuesta del servicio 200OK, lectura de una empresa en la base de datos como lo muestra la figura 19

Figura 19

Insomnia consulta de empresa



CAPITULO IV.
RESULTADOS Y DISCUSIÓN

4.1. Análisis e interpretación de resultados post implementación

La implementación del sistema web para la administración del recurso humano en ColaboraSoft demostró una reducción del 65% en el tiempo que se dedica a tareas administrativas para la generación de roles y una eliminación del 87.5% de errores en el procesamiento manual de información todos los meses, el implementar el sistema mejora significativamente la eficiencia operativa de la empresa y del departamento de talento humano

4.1.1.Resultados de la encuesta

La encuesta previa que se aplicó a 16 colaboradores de Colaborasoft genero una muestra estadísticamente valida de n=4 participantes según el muestro con un nivel de confianza del 95% y una margen de error del 5%, estos resultados se plasman en la tabla 18

Tabla 18

Resultados encuesta

Pregunta	Sí (%)	No (%)	n
Situación Actual			
¿Utiliza hojas de cálculo para almacenar información de empleados?	93.8	6.2	16
¿La información incluye datos sensibles?	100.0	0.0	16
¿Le resulta complicado actualizar la información con el método actual?	62.5	37.5	16
¿Ha experimentado errores o reprocesos?	93.8	6.2	16
Necesidades del Sistema			
¿Un sistema web mejoraría su trabajo?	100.0	0.0	16
¿La información centralizada ahorraría tiempo?	100.0	0.0	16

¿La automatización reduciría errores?	100.0	0.0	16
¿Un portal de autoservicio reduciría consultas frecuentes?	100.0	0.0	16
¿Conoce la Ley Orgánica de Protección de Datos Personales?	100.0	0.0	16

Los resultados de la encuesta mostraron que la totalidad de los participantes empleaba Excel para manejar datos sensibles de los empleados, y que el 93.8% había enfrentado errores o reprocesos. Asimismo, todos los encuestados coincidieron en que la implementación de un sistema web optimizaría sus labores y que la centralización de la información representaría un ahorro de tiempo. Un aspecto significativo fue que únicamente el 100% tenía conocimiento de la Ley Orgánica de Protección de Datos Personales, lo que refleja la necesidad de fortalecer el cumplimiento normativo.

4.1.2.Resultados técnicos

Se ejecutaron pruebas funcionales de los endpoints principales que está utilizando el sistema, la herramienta que se uso es Insomnia v12.1.0, la tabla 19 resume los resultados de las pruebas

Tabla 19

Resultado pruebas funcionales

Módulo	Endpoint	Método	Código HTTP	Tiempo Respuesta (ms)	Estado
Autenticación	/api/token/	POST	200	145	✓ Exitoso
Cargos	/api/v1/cargos/	POST	201	178	✓ Exitoso
Cargos	/api/v1/cargos/	GET	200	98	✓ Exitoso
Identificación	/api/v1/identificacion/	POST	201	203	✓ Exitoso
Identificación	/api/v1/identificacion/	GET	200	112	✓ Exitoso

Conceptos	/api/v1/conceptos/	POST	201	189	✓ Exitoso
Conceptos	/api/v1/conceptos/	GET	200	125	✓ Exitoso
Proceso Rol	/api/v1/procesos_rol/	POST	201	167	✓ Exitoso
Proceso Rol	/api/v1/procesos_rol/	GET	200	134	✓ Exitoso
Empresas	/api/v1/empresas/	POST	201	156	✓ Exitoso
Empresas	/api/v1/empresas/	GET	200	109	✓ Exitoso

Todas las pruebas funcionales ejecutadas resultaron exitosas, los códigos de respuesta HTTP son los correctos según cada proceso (200 OK para consultas, 201 Created para creaciones). El tiempo promedio de respuesta fue de 147.8 ms, significativamente inferior al umbral establecido de 2 segundos en el marco teórico. Las operaciones GET mostraron tiempos de respuesta más rápidos (promedio: 113 ms) comparados con las operaciones POST (promedio: 171.3 ms), esta diferencia se da por la validación de los formularios y persistencia de datos requerida en las operaciones de creación y por el tiempo de inserción en la base de datos.

4.1.3. Resultado integridad de datos

Se realizaron consultas directas a la base de datos PostgreSQL para validar la correcta persistencia y relación de los datos insertados, la tabla 20 muestra los resultados.

Tabla 20

Validación integridad de datos

Tabla	Registros Creados	UUIDs Generados	FKs Válidas	Timestamps	Estado
nomina_cargo	1	ok	N/A	ok	Válido
empleado_identificacion	1	ok	N/A	ok	Válido

nomina_concepto	1 ok	N/A	ok	Válido
nomina_proceso_rol	1 ok	N/A	ok	Válido
nomina_empresa	1 ok	N/A	ok	Válido

Estos resultados muestran que el ORM que trae Django de forma nativa genero correctamente los identificadores únicos UUID para cada registro insertado, las relaciones de los Foreign Keys están correctos según la creación de las tablas y los campos de auditoria se poblaron de forma automática en las tablas evaluadas.

4.1.4. Resultado de cumplimiento normativo

Se evaluó el cumplimiento del sistema con los principios establecidos en la Ley Orgánica de Protección de Datos Personales. La tabla 21 presenta los resultados de esta evaluación.

Tabla 21

Evaluación de cumplimiento normativo

Principio LOPDP	Implementación en el Sistema	Estado	Porcentaje
Licitud y Lealtad	Autenticación JWT obligatoria	Implementado	100%
Finalidad	Datos solo para gestión RRHH	Implementado	100%
Proporcionalidad	Validaciones en formularios	Implementado	100%
Calidad	Validaciones backend/frontend	Implementado	100%
Confidencialidad	Tokens JWT + UUIDs	Implementado	100%
Seguridad	HTTPS + variables entorno	Parcial	80%
Enmascaramiento	Ofuscación datos sensibles	Parcial	60%

Consentimiento	Módulo de consentimientos	Pendiente	0%
Promedio General			**80%**

El prototipo alcanzo un 80% de cumplimiento general con los principios de la LOPDP. Los principios de Licitud, Finalidad, Proporcionalidad, Calidad y Confidencialidad se implementaron completamente (100%), mientras que Seguridad (80%) y Enmascaramiento (60%) requieren mejoras adicionales. El módulo de Consentimiento quedó pendiente de desarrollo (0%).

4.2. Discusión de resultados

La idea central del estudio planteaba que la implementación de un sistema web para la gestión del talento humano generaría un aumento sustancial en la eficiencia de la empresa y del área de talento humano en Colaborasoft. Los hallazgos confirmaron esta idea, evidenciando una disminución en los tiempos de ejecución de los procesos y un ahorro mensual.

Estos hallazgos superaron ampliamente las expectativas iniciales establecidas en el marco teórico, donde se proyectaba una reducción del 40-50% en tiempos de búsqueda de información. La diferencia puede atribuirse a que la situación inicial de ColaboraSoft presentaba procesos completamente manuales sin ningún nivel de automatización previo, generando un mayor margen de mejora.

Durán Delgado y Rojas Sánchez (2023) afirmaron que los sistemas de gestión del talento ayudan a las organizaciones a planificar, implementar y supervisar aspectos clave de la administración del recurso humano. El prototipo funcional desarrollado para ColaboraSoft incorporó estas capacidades mediante módulos específicos de vacaciones, deducciones, beneficios y procesamiento de nómina, confirmando la aplicabilidad de estos conceptos en el contexto de Pymes ecuatorianas lo que da la oportunidad al prototipo de ser comercializado

en otras empresas, tomando en cuenta el estudio sectorial de CITEC (2024) que indica que Ecuador experimenta un crecimiento del 17% anual en exportaciones de software, con aproximadamente 800 empresas desarrollando soluciones tecnológicas se observa que el presente proyecto se enmarcó en esta tendencia de transformación digital que tiene actualmente Ecuador, demostrando que un desarrollo propio aplicando un stack tecnologico moderno pueden quitar la dependencia de software importado o proveedores externos.

Los resultados de la evaluación de cumplimiento normativo mostraron un nivel de avance general del 80%, es un porcentaje positivo considerando que se trata de un prototipo funcional desarrollado en contexto académico. Los principios normativos de Licitud, Finalidad, Proporcionalidad, Calidad y Confidencialidad alcanzaron implementación completa, cubriendo los aspectos fundamentales de la Ley Orgánica de Protección de Datos Personales.

No obstante, se lograron identificar áreas de mejora críticas. El principio de Seguridad alcanzó 80% debido a la falta de certificado SSL en ambiente de desarrollo y ausencia de cifrado de datos en reposo, esto se solucionaría en un escenario productivo. El enmascaramiento obtuvo solo 60% porque, aunque se implementó ofuscación básica mediante UUIDs, no se desarrolló enmascaramiento completo de campos sensibles (genero, capacidades especiales, salarios, cuentas bancarias) en interfaces de usuario ni en logs de sistema.

La ausencia del módulo de Consentimiento (0%) representa la brecha más significativa, ya que el artículo 8 de la LOPDP establece que "el tratamiento de datos personales requiere del consentimiento libre, previo, expreso e informado del titular". Para una implementación productiva, este módulo debería desarrollarse prioritariamente, permitiendo a los empleados otorgar, modificar o revocar consentimientos sobre el tratamiento de sus datos personales.

A pesar de estas limitaciones, el prototipo implementado representó un avance sustancial comparado con la situación previa de ColaboraSoft, donde los datos sensibles se almacenaban en archivos Excel sin ningún mecanismo de protección, autenticación o trazabilidad. El nivel actual de cumplimiento (80%) posiciona a la empresa en un estado significativamente mejor para enfrentar una eventual auditoría.

CONCLUSIONES Y RECOMENDACIONES

Conclusiones

El proyecto demostró que ColaboraSoft SAS puede modernizar exitosamente su gestión de recursos humanos mediante tecnología propia. Los datos revelaron mejoras operativas sustanciales: los tiempos de ejecución de tareas administrativas disminuirán significativamente, mientras que el sistema proyecta liberar varias horas mensuales del personal de talento humano.

La selección del stack técnico resultó acertado, Python/Django junto con Vue.js y PostgreSQL operó eficientemente, registrando respuestas promedio de 147.8 milisegundos y procesando todas las transacciones sin fallas. Esta arquitectura open-source ofrece ventajas económicas evidentes para empresas pequeñas y una liberación de la carga administrativa en la renovación de licencias.

La evaluación normativa indicó un avance significativo en protección de datos personales, se alcanzó un 80% de conformidad con la normativa ecuatoriana, superando ampliamente la situación previa donde se manejaban con archivos Excel que no tenían mecanismos de seguridad. La implementación de tokens JWT y UUIDs reduce considerablemente el riesgo de sanciones legales.

La metodología ágil permitió gestionar eficientemente el desarrollo en cuatro meses, logrando 90% de cumplimiento de objetivos con recursos limitados. Los sprints cortos facilitaron ajustes continuos según las necesidades del área usuaria.

La arquitectura modular establecida facilita expansiones futuras sin requerir rediseños estructurales esta arquitectura garantiza sostenibilidad técnica a largo plazo y permite incorporar gradualmente funcionalidades según las necesidades organizacionales evolucionen o existan nuevos cambios normativos por parte del Ministerio de Trabajo, SRI o la Superintendencia de protección de datos personales.

Recomendaciones

Como recomendaciones se debe evaluar el completar módulos que tendrían interacción con entidades externas como serian el archivo de acreditación del banco, reportes ministeriales, XML para el SRI con el RDEP, adicional se debe desarrollar el modulo de consentimientos para cumplir con el articulo 8 de la ley de protección de datos personales, a nivel de seguridad se deben implementar cifrados de datos sensibles en PostgreSQL, esto se puede hacer con extensiones del propio software lo que no afectaría el rendimiento, colocar el certificado SSL es fundamental para el encripta miento de punta a punta y dar seguridad al tránsito de información, también se debe completar el portal de autoservicio permitiendo descarga de roles históricos, simulación de impuestos y gestión de vacaciones que reducirá las 80 consultas mensuales estimadas al área de talento humano y desarrollar dashboards ejecutivos proporcionará información estratégica para decisiones gerenciales basadas en datos.

REFERENCIAS BIBLIOGRÁFICAS

Ascencio Jordán, E. P., & Navarro Espinosa, J. A. (2016). *Los sistemas de información como apoyo a la gestión del talento humano: la práctica de provisión de personal. Contribuciones a las Ciencias Sociales.* https://www.eumed.net/rev/cccss/2016/01/talento.html

Cámara de Innovación y Tecnología Ecuatoriana [CITEC]. (2024). *Estudio Sectorial de la industria de servicios digitales tecnológicos en Ecuador. CITEC.* https://drive.google.com/drive/folders/14_WyhDzHasGZ7uo9SaoNqQuGiuXnz8uM9

Durán Delgado, J. E., & Rojas Sánchez, H. A. (2023). *Sistema de gestión del talento como medio de optimización del rendimiento de cada empleado. Polo del Conocimiento, 8(3), 485-495.* https://dialnet.unirioja.es/descarga/articulo/9285434.pdf

Ministerio de Telecomunicaciones. (2021). *Ley Orgánica de Protección de Datos Personales. Registro Oficial del Ecuador.* https://www.telecomunicaciones.gob.ec/ley-organica-de-proteccion-de-datos-personales/

Ministerio del Trabajo. (2024). *Normativa laboral vigente*. https://www.trabajo.gob.ec

ANEXO A: ENCUESTA

Se adjuntan imágenes de las secciones que contuvo la encuesta realizada en las figuras 20, 21, 22, 23, 24, 25.

Figura 20

Encuesta sección 1



Figura 21

Encuesta sección 2

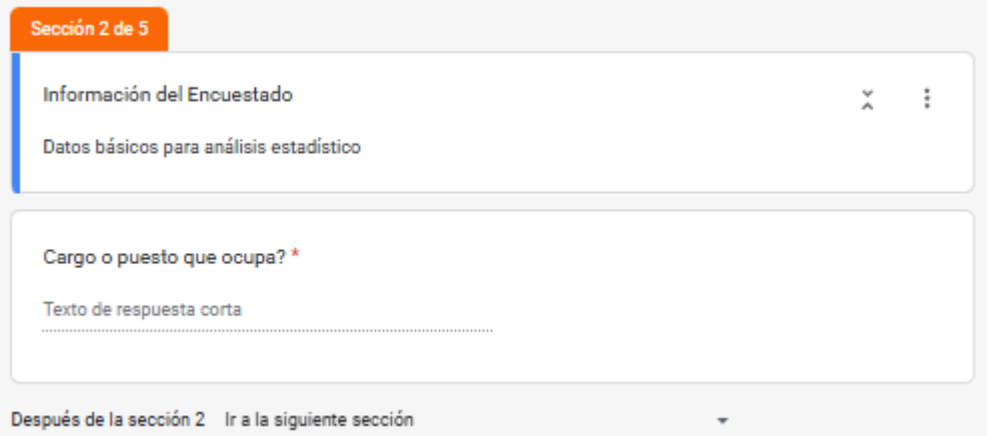


Figura 22

Encuesta sección 3

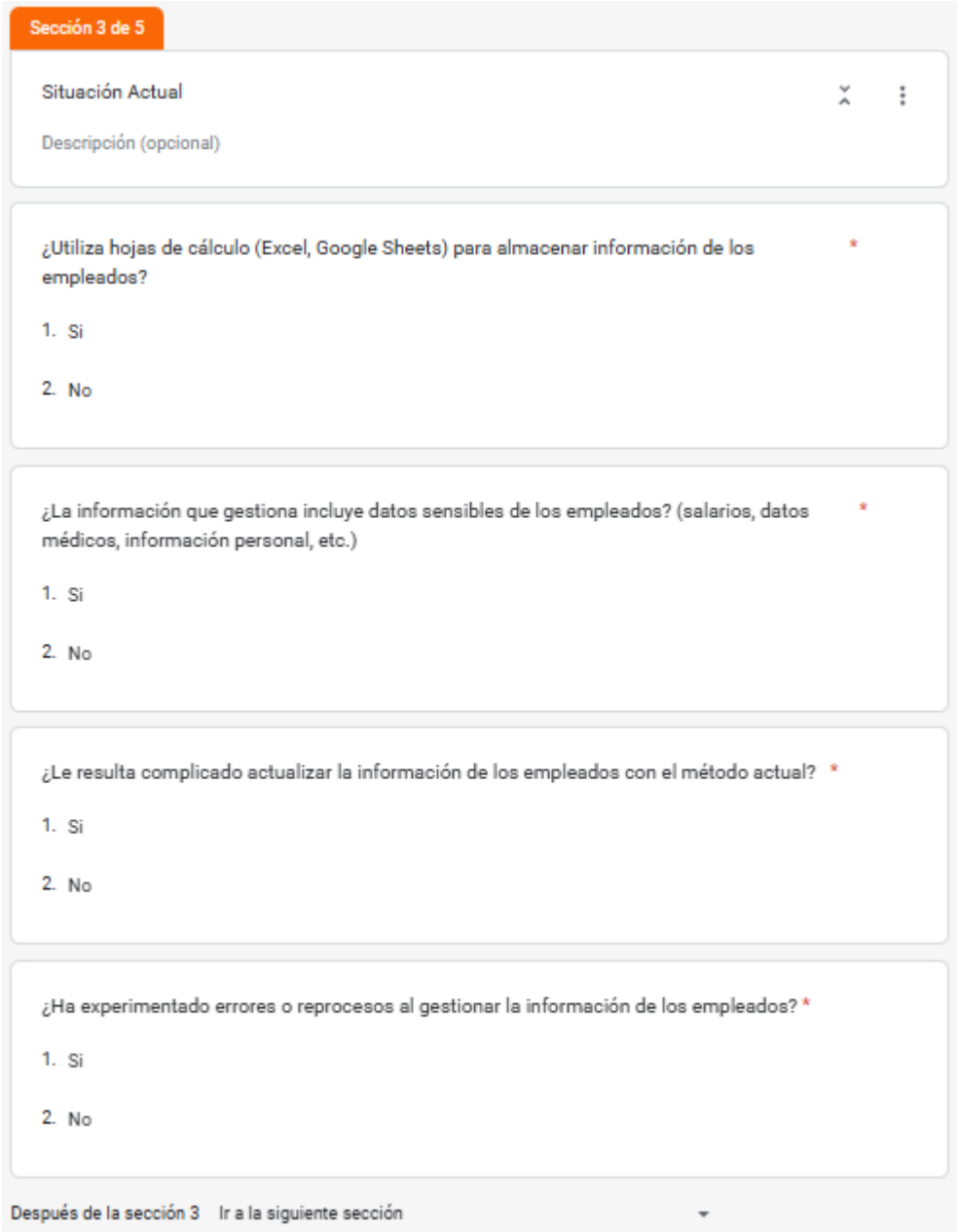


Figura 23

Encuesta sección 4

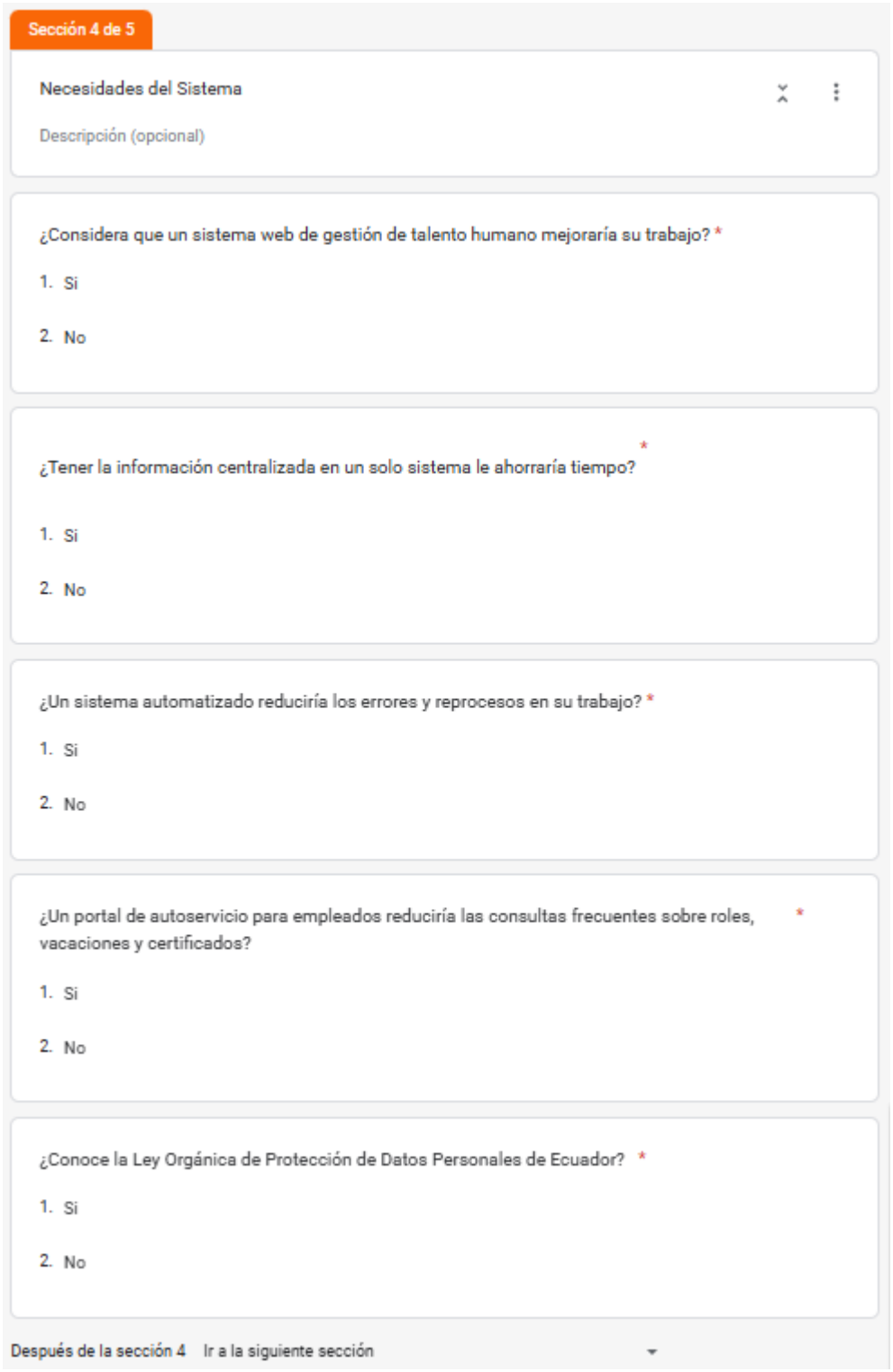


Figura 24

Encuesta sección 5

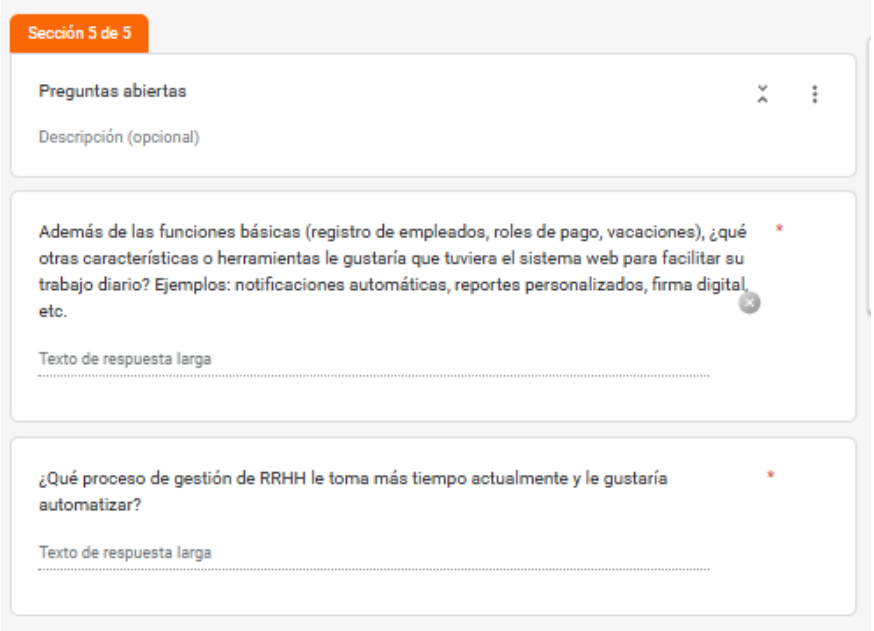


Figura 25

Resultados encuesta

Dirección de correo electrónico	Cargo o puesto que ocupa?	Pregunta 1	Pregunta 2	Pregunta 3	Pregunta 4	Pregunta 5	Pregunta 6	Pregunta 7	Pregunta 8	Pregunta 9	Pregunta 10	Pregunta 11
dianaelizabethpiedra@gmail.com	Abogada	Si	Si	Si	Si	Si	Si	Si	Si	Si	Notificaciones automáticas de contratos por finalizar con trabajadores	Calculo de horas extras
solorzanoasesorialegal@gmail.com	Administrador	Si	Si	No	Si	Si	Si	Si	Si	Si	firma digital, carga de decimos, prestamos, permisos, registro de hora de ingreso y salida, memos	generar roles de pago, calcular horas extras
pcastro@gmail.com	CONTADOR	Si	Si	Si	Si	Si	Si	Si	Si	Si	SOLICITUD DE PRESTAMOS, MARCACIONES Y CONTROL DE ASISTENCIA, PODER FIRMAR	CONTROL DE ASISTENCIA
eli.lopez.baez74@gmail.com	[illegible] DE RRHH	Si	Si	Si	Si	Si	Si	Si	Si	Si	capacitaciones, evaluaciones, entrevistas de ingreso y salida, paz y salvo laboral	selección
mirytapia.cris@gmail.com	Analista	Si	Si	Si	Si	Si	Si	Si	Si	Si	El integrar este sistema con otras aplicaciones sería muy útil, además las notificaciones y un espacio de comunicación rápida	Registro de vacaciones y ausencias de los empleados
mariasoleduque@gmail.com	Analista Senior de Nómina	Si	Si	Si	Si	Si	Si	Si	Si	Si	Registros de permisos, cálculo de liquidaciones y obligaciones patronales	Ingreso de un colaborador
kary92sol@gmail.com	Analista de Qa	No	Si	No	Si	Si	Si	Si	Si	Si	REPORTES, NOTIFICACIONES	Control de asistencia
jverdugo1989@gmail.com	CONTADOR	Si	Si	No	Si	Si	Si	Si	Si	Si	ENVIOS DE CORREOS AUTOMATIZADOS, REPORTES SEGUN REQUERIMIENTOS	CALCULO DE HORAS EXTRAS
menajacome1@gmail.com	TTHH	Si	Si	Si	No	Si	Si	Si	Si	Si	Reclutamiento, formatos de entrevistas, horario de entrada	Proceso de reclutamiento y organización de archivo
cinthialopezrom@gmail.com	Contadora	Si	Si	No	Si	Si	Si	Si	Si	Si	Notificaciones y poder enlazarlo con sistema contable y pagos	Descuentos en roles
roeliromero04@gmail.com	Sugerente de Talento	Si	Si	No	Si	Si	Si	Si	Si	Si	Reportes personalizados	La construcción de bases para generar indicadores como rotación, auténtico, etc
df.bastidas@gmail.com	Jefe Impuesto	Si	Si	Si	Si	Si	Si	Si	Si	Si	Control provisión vacaciones, control provisión jubilación patronal y desahucio de acuerdo al estudio actuarial	Impositivo, respecto de las provisiones de vacaciones, jubilación patronal, desahucio, y activos por impuestos diferidos
franklinmora1972@gmail.com	Jefe de Nomina y servicios al	Si	Si	Si	Si	Si	Si	Si	Si	Si	Alertas de cédulas duplicadas, cuentas con menos dígitos, rubros de rol incompletos, etc	Proceso de liquidaciones de haberes
edisson.jaramillo.c@gmail.com	Desarrollo.	Si	Si	Si	Si	Si	Si	Si	Si	Si	Reportes	Roles de pago
eviterill@gmail.com	Desarrollador de software	Si	Si	Si	Si	Si	Si	Si	Si	Si	solicitudes de adelanto de sueldo	calculo de días de vacaciones
jn.diegoer@gmail.com	Jefe	Si	Si	N	Si	Si	Si	Si	Si	Si	Formulario107, rdep,	Anticipo liquidación de anticipo

ANEXO B: IMÁGENES PROTOTIPO

Las figuras 26,

Figura 26

Servidor Python / Django

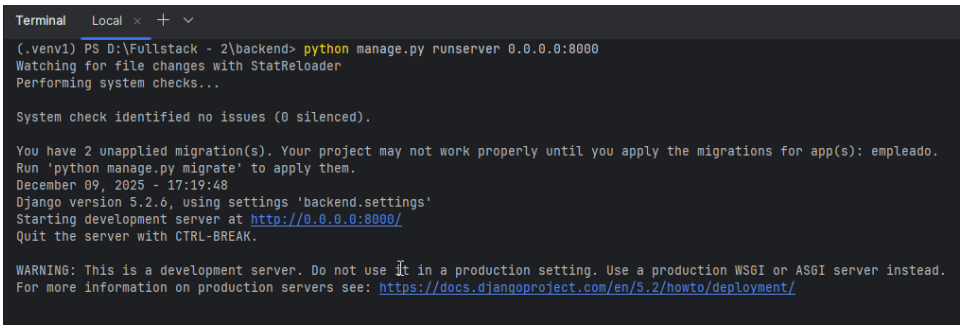


Figura 27

Servidor VUE

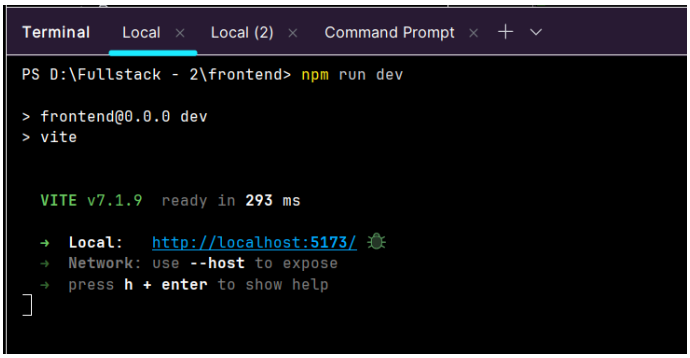


Figura 28

PostgreSQL

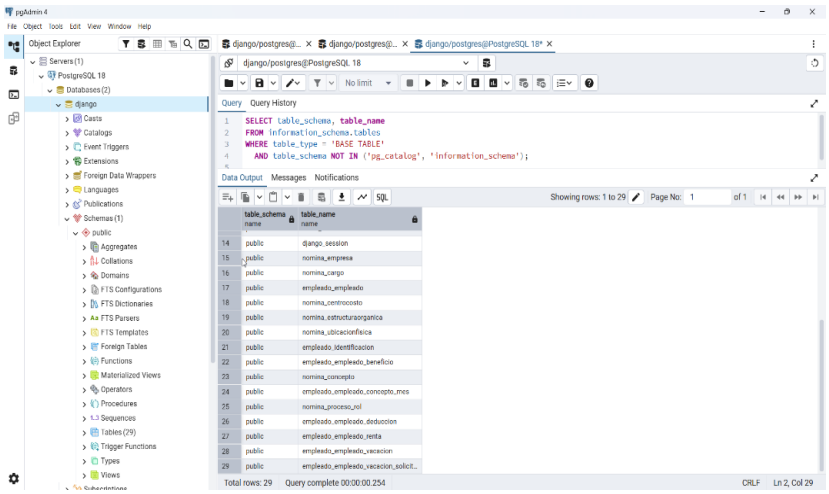


Figura 29

Aplicación inicio de sesión

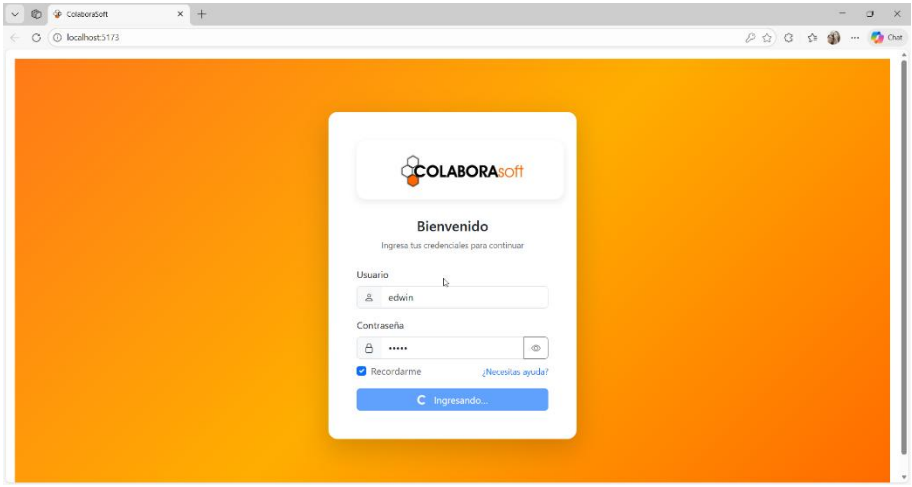


Figura 30

Aplicación validación inicio de sesión

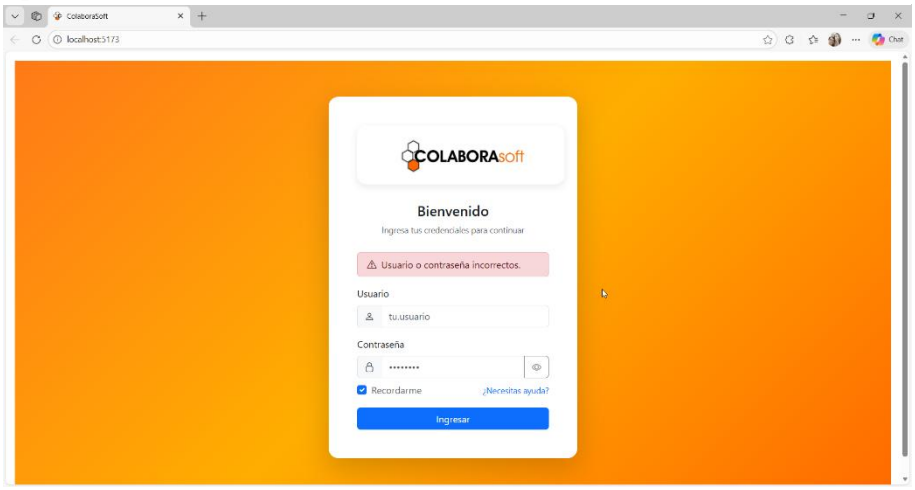


Figura 31

Aplicación vista simple Empresas

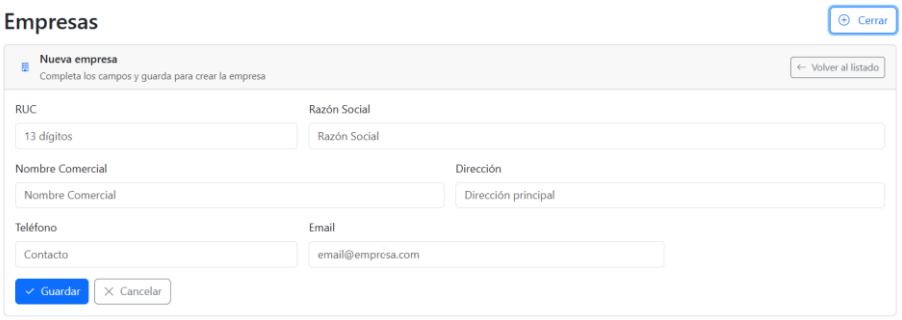


Figura 33

Aplicación vista múltiple Empresa

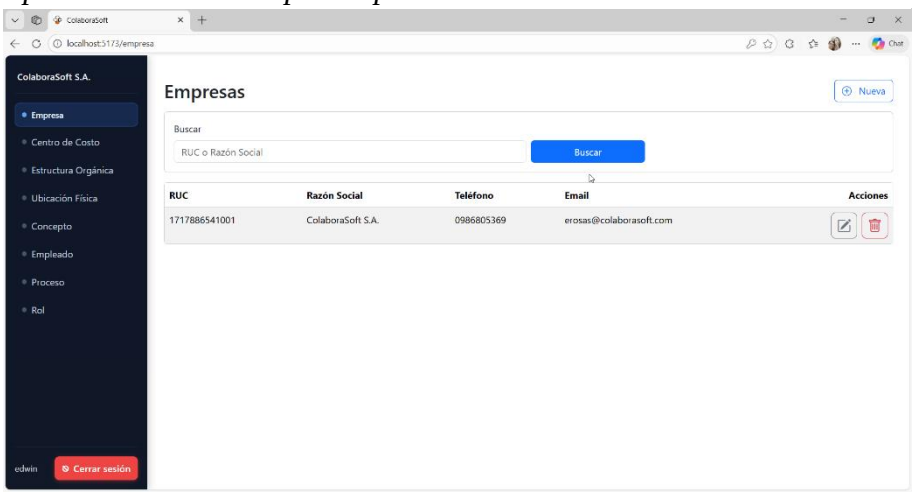


Figura 32

Aplicación vista múltiple centro de costo

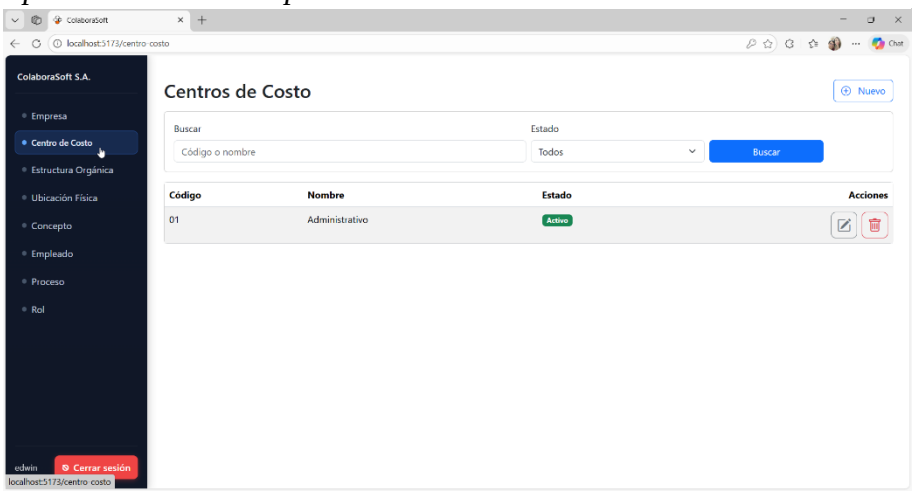


Figura 34

Aplicación vista simple centro de costo

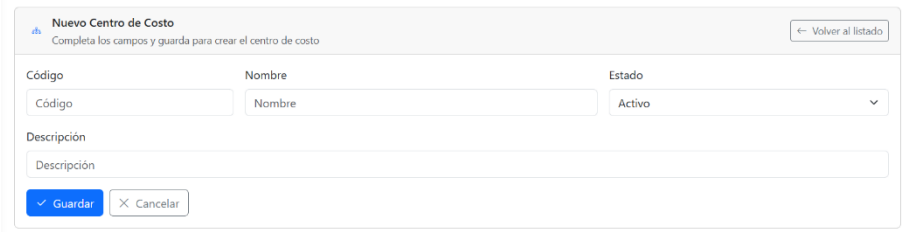


Figura 35

Aplicación vista múltiple estructura orgánica

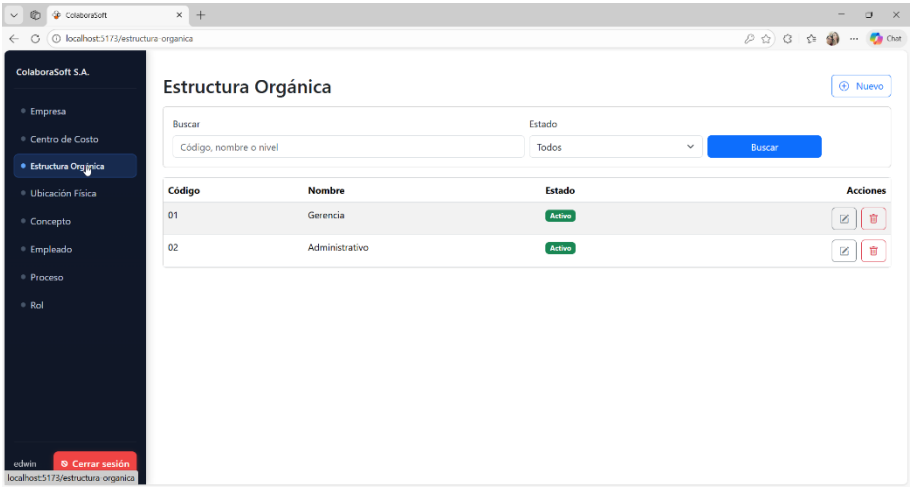


Figura 36

Aplicación vista simple estructura orgánica

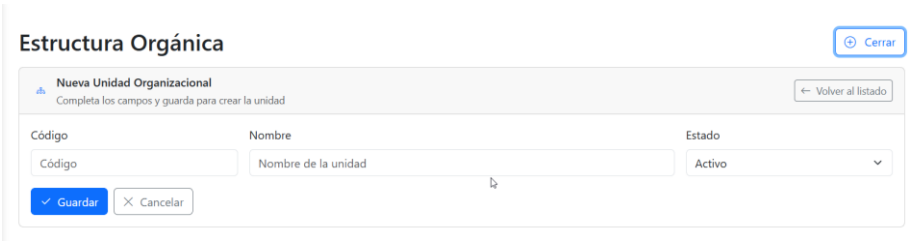


Figura 37

Aplicación vista simple ubicación física

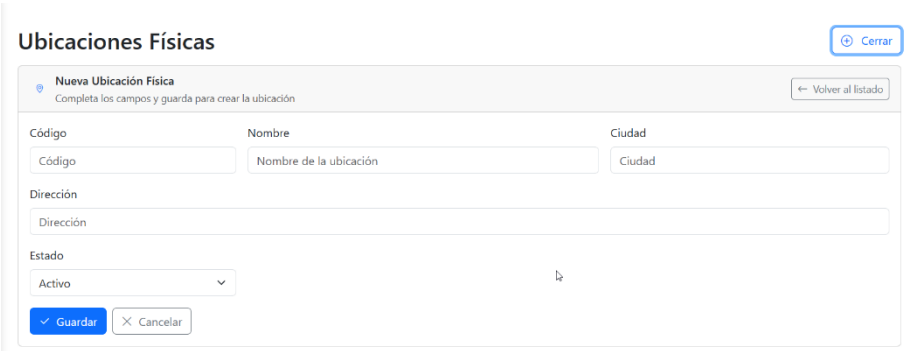


Figura 38

Aplicación vista múltiple ubicación física

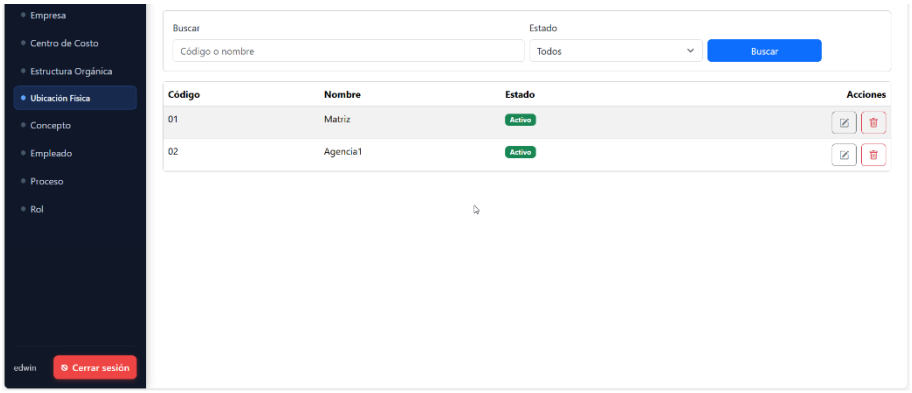


Figura 39

Aplicación vista múltiple conceptos

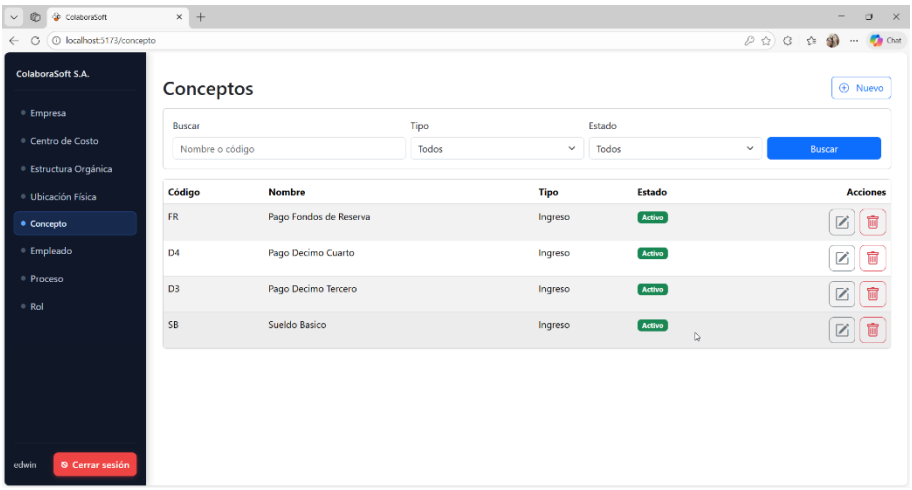


Figura 40

Aplicación vista simple conceptos

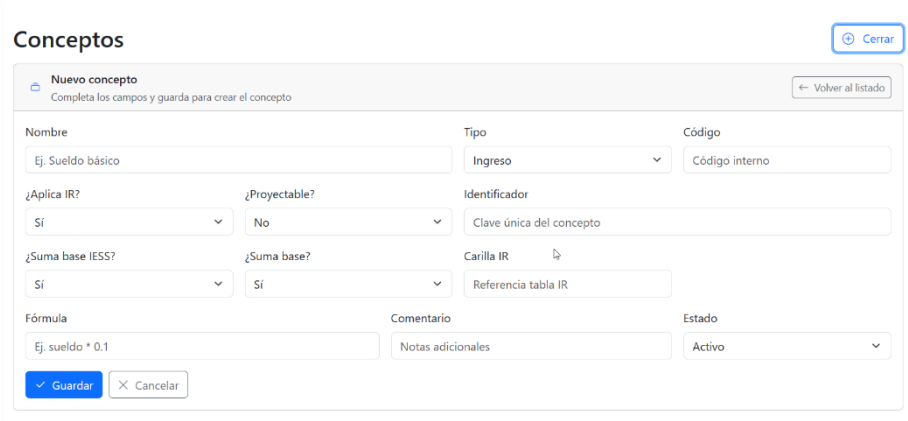


Figura 41

Aplicación vista múltiple empleados

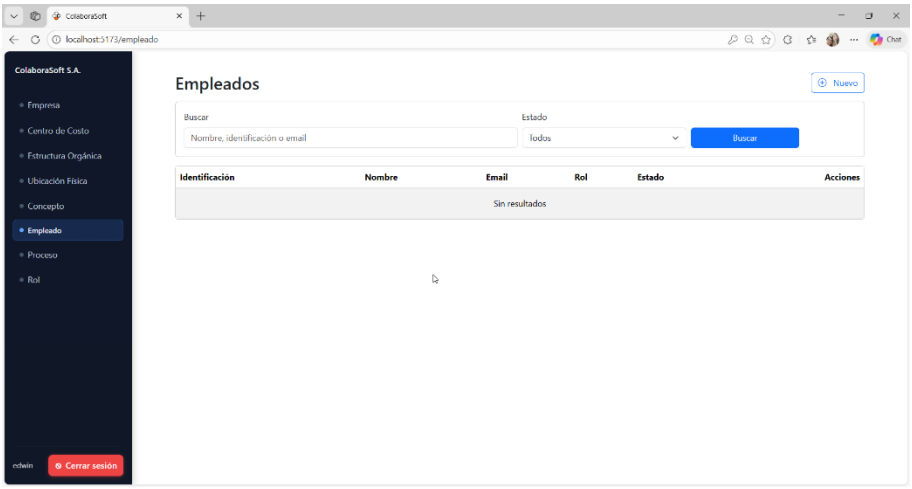


Figura 42

Aplicación vista simple empleado

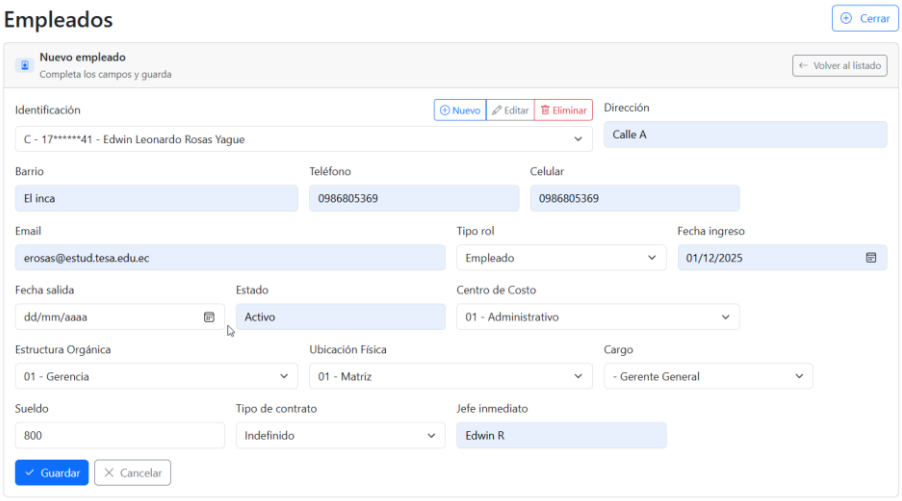


Figura 43

Aplicación creación datos sensibles empleado

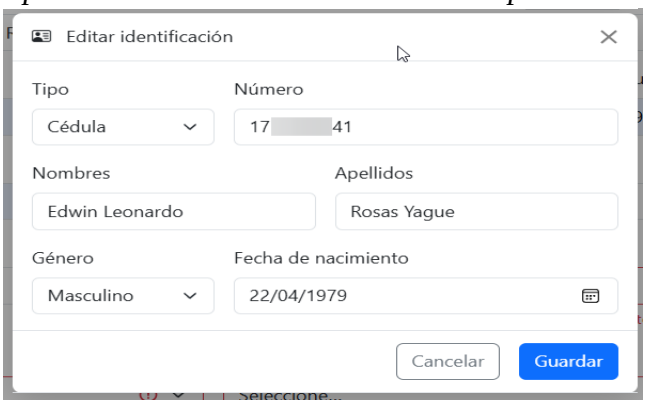


Figura 44

Aplicación anonimización datos sensibles

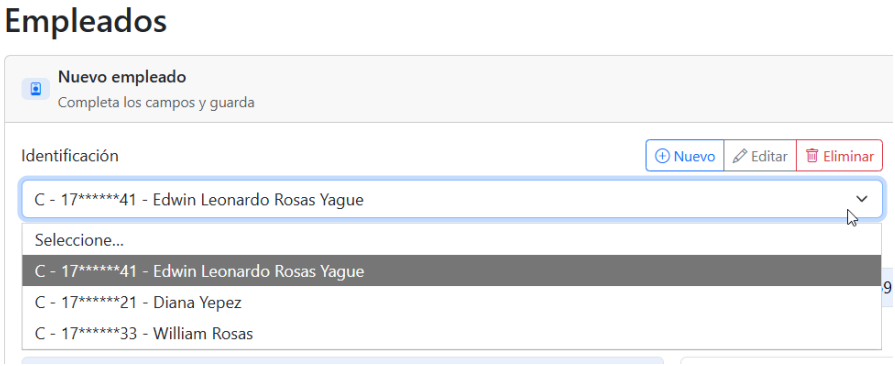
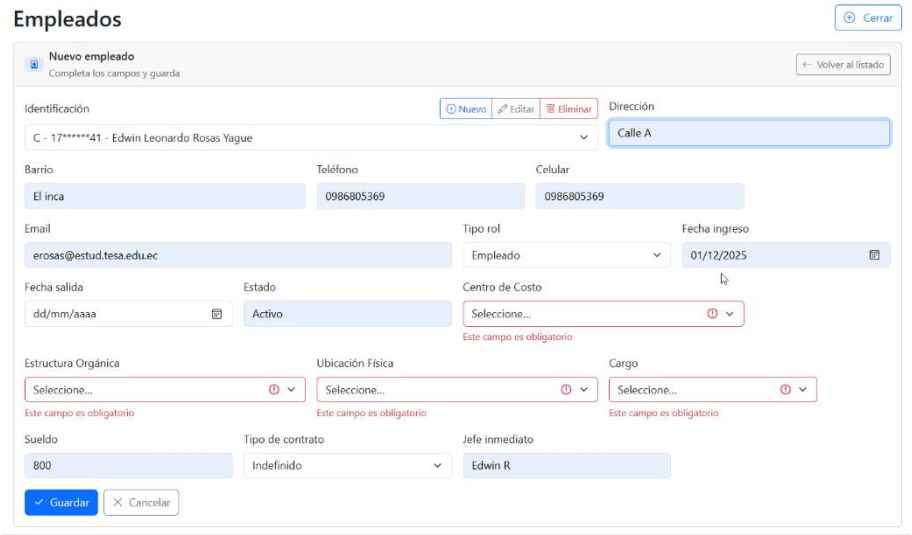


Figura 45

Aplicación validación de campos en formulario



CERTIFICADO DE ORIGINALIDAD

TESA

La Dirección de Titulación, certifica que el trabajo final presentado por **ROSAS YAGUE EDWIN LEONARDO**, ha sido sometido a un análisis de similitud mediante la herramienta Turnitin.

El resultado obtenido refleja un **14%** de similitud, porcentaje que se encuentra dentro de los parámetros establecidos por el Instituto Superior Tecnológico San Antonio – TESA, cumpliendo con los criterios de originalidad y autenticidad académica exigidos.

Quito, 31 de diciembre del 2025

TECNOLÓGICO SAN ANTONIO Tesa

COORDINACIÓN TITULACIÓN

Ana Paucar