



UNIVERSIDAD NACIONAL PEDRO RUIZ GALLO

**FACULTAD DE INGENIERÍA CIVIL,
SISTEMAS Y ARQUITECTURA**



ESCUELA PROFESIONAL DE INGENIERÍA DE SISTEMAS

TRABAJO DE SUFICIENCIA PROFESIONAL

**Desarrollo de un sistema web para la gestión de incidencias
dirigidas al área de sistemas en la empresa Solpack s.a.c
- Lima**

Para optar por el Título Profesional de :

Ingeniero de Sistemas

Presentado por :

Huamanchumo Bances, Jhony Alex

Autor

Ing. Llontop Cumpa Luis Alberto

Asesor

Lambayeque - Perú

2026



UNIVERSIDAD NACIONAL PEDRO RUIZ GALLO

**FACULTAD DE INGENIERÍA CIVIL,
SISTEMAS Y ARQUITECTURA**



ESCUELA PROFESIONAL DE INGENIERÍA DE SISTEMAS

TRABAJO DE SUFICIENCIA PROFESIONAL

**Desarrollo de un sistema web para la gestión de incidencias
dirigidas al área de sistemas en la empresa Solpack s.a.c
- Lima**

Para optar por el Título Profesional de :

Ingeniero de Sistemas

Aprobado por los miembros del Jurado :

**Dr. Ing. Diaz Plaza, Regis Jorge Alberto
Presidente**

**Dr. Ing. Olavarria Paz, Jesús Bernando
Secretario**

**Dr. Ing. Guzmán Valle, María de los Ángeles
Asesor**

Lambayeque – Perú

2026



UNIVERSIDAD NACIONAL PEDRO RUIZ GALLO

**FACULTAD DE INGENIERÍA CIVIL,
SISTEMAS Y ARQUITECTURA**



ESCUELA PROFESIONAL DE INGENIERÍA DE SISTEMAS

TRABAJO DE SUFICIENCIA PROFESIONAL

**Desarrollo de un sistema web para la gestión de incidencias
dirigidas al área de sistemas en la empresa Solpack s.a.c
- Lima**

Para optar por el Título Profesional de :

Ingeniero de Sistemas

**Huamanchumo Bances, Jhony Alex
Autor**

**Ing. Llontop Cumpa, Luis Alberto
Asesor**

Lambayeque – Perú

2026



ACTA DE SUSTENTACIÓN N° 002-2026-UI-FICSA

Siendo las 8:30 am del día 15 de mayo del 2026, se reunieron los miembros de jurado del Trabajo de Suficiencia Profesional titulada: “DESARROLLO DE UN SISTEMA WEB PARA LA GESTIÓN DE INCIDENCIAS DIRIGIDAS AL ÁREA DE SISTEMAS EN LA EMPRESA SOLPACK S.A.C - LIMA” con código N° IS_V_TSP_2025_012, y designado por Resolución Decanal N° 044-2026-UNPRG-FICSA con la finalidad de Evaluar y Calificar la sustentación del trabajo de suficiencia profesional antes mencionado, conformado por los siguientes docentes:

DR. ING. REGIS JORGE ALBERTO DIAZ PLAZA	PRESIDENTE
DR. ING. JESUS BERNARDO OLAVARRIA PAZ	SECRETARIO
DRA. ING. MARIA DE LOS ANGELES GUZMAN VALLE	VOCAL

Asesorado por ING. LUIS ALBERTO LLONTOP CUMPA.

El acto de sustentación fue autorizado por OFICIO VIRTUAL N° 97-2026-UIFICSA, el trabajo de suficiencia profesional fue presentado y sustentado por el Bachiller: **JHONY ALEX HUAMANCHUMO BANCES**, tuvo una duración de 60 minutos. Después de la sustentación, y absueltas las preguntas y observaciones de los miembros del jurado; se procedió a la calificación respectiva:

	NUMERO	LETRAS	CALIFICATIVO
JHONY ALEX HUAMANCHUMO BANCES	<u>18</u>	<u>DECIOCHO</u>	<u>MUY BUENO</u>

Por lo que queda APTO para obtener el Título Profesional de **INGENIERO DE SISTEMAS** de acuerdo con la Ley Universitaria 30220 y la normatividad vigente de la Facultad de Ingeniería Civil De Sistemas y de Arquitectura de la Universidad Nacional Pedro Ruiz Gallo.

Siendo las 9:30 am; del mismo día, se dio por concluido el presente acto académico, dándose conformidad al presente acto, con la firma de los miembros del jurado.

DR. ING. REGIS JORGE ALBERTO DIAZ PLAZA
PRESIDENTE

DR. ING. JESUS BERNARDO OLAVARRIA PAZ
SECRETARIO

DRA. ING. MARIA DE LOS ANGELES GUZMAN VALLE
VOCAL

ING. LUIS ALBERTO LLONTOP CUMPA
ASESOR





ANEXO 01

CONSTANCIA DE VERIFICACION DE ORIGINALIDAD DEL TRABAJO DE SUFICIENCIA PROFESIONAL

Yo, Luis Alberto Llontop Cumpa, Asesor de tesis modalidad de suficiencia profesional,
del estudiante, **Huamanchumo Bances Jhony Alex**

Titulada: "DESARROLLO DE UN SISTEMA WEB PARA LA GESTIÓN DE
INCIDENCIAS DIRIGIDAS AL ÁREA DE SISTEMAS EN LA EMPRESA SOLPACK
S.A.C - LIMA", luego de la revisión exhaustiva del documento constato que la misma
tiene un índice de similitud de 0 % verificable en el reporte de similitud del programa
Turnitin.

El suscrito analizó dicho reporte y concluyó que cada una de las coincidencias
detectadas no constituyen plagio. A mi leal saber y entender la tesis bajo informe de
suficiencia profesional cumple con todas las normas para el uso de citas y referencias
establecidas por la Universidad Nacional Pedro Ruiz Gallo.

Lambayeque, 10 de noviembre del 2025

LUIS ALBERTO LLONTOP CUMPA
DNI: 16628464
ASESOR

Se adjunta:

Resumen del Reporte (Con porcentaje y parámetros de configuración)

Recibo digital

Desarrollo De Un Sistema Web Para La Gestión De Incidencias Dirigidas Al Área De Sistemas En La Empresa Solpack S.A.C - Lima

INFORME DE ORIGINALIDAD

0%

INDICE DE SIMILITUD

%

FUENTES DE INTERNET

0%

PUBLICACIONES

0%

TRABAJOS DEL
ESTUDIANTE

FUENTES PRIMARIAS

Excluir citas

Apagado

Excluir bibliografía

Apagado

Excluir coincidencias

Apagado



Asesor
Asesor

DNI:16628464



Recibo digital

Este recibo confirma que su trabajo ha sido recibido por **Turnitin**. A continuación podrá ver la información del recibo con respecto a su entrega.

La primera página de tus entregas se muestra abajo.

Autor de la entrega: Jhony Alex Huamanchumo Bances
Título del ejercicio: Quick Submit
Título de la entrega: Desarrollo De Un Sistema Web Para La Gestión De Incidencias...
Nombre del archivo: TrabajoSuficienteProfesional_JhonyHuamanchumoF.docx
Tamaño del archivo: 2.02M
Total páginas: 45
Total de palabras: 6,176
Total de caracteres: 35,715
Fecha de entrega: 10-nov-2025 04:53p.m. (UTC-0500)
Identificador de la entrega: 2810271489


Luis Alberto Elontop Cumpa
Asesor
DNI:16628464



UNIVERSIDAD NACIONAL PEDRO RUIZ GALLO
FACULTAD DE INGENIERIA CIVIL, DE SISTEMAS Y DE
ARQUITECTURA
Escuela Profesional de Ingeniería de Sistemas

"DESARROLLO DE UN SISTEMA WEB PARA LA GESTIÓN DE
INCIDENCIAS DIRIGIDAS AL ÁREA DE SISTEMAS EN LA EMPRESA
SOLINEX S.A.C - LIMA"

Trabajo de suficiencia profesional para optar por el Título Profesional de Ingeniero de
Sistemas, que presenta

Bach. Jhony Alex Huamanchumo Bances

Asesor Ing. Luis Elontop Cumpa

LAMBAYEQUE - PERU
2025

Dedicatoria

A Dios, por ser mi guía espiritual, por darme salud y sabiduría para lograr concretar cada uno de mis proyectos.

A mi madre, Natividad, por ser mi apoyo vital; y a mis abuelos, Pedro y Nicolasa, quienes en vida me impulsaron siempre a alcanzar mis metas con esfuerzo y sacrificio.

A mi hermano Carlos por decidir seguir mis pasos y convertirse en mi colega profesional.

Estoy seguro de que logrará cada uno de sus proyectos.

A mí mismo, por la perseverancia demostrada en este camino y por mantener firme mi esencia profesional y personal, aun frente a las dificultades.

Agradecimiento

A Dios, por ser la luz en mi camino, por brindarme vida, salud y fortaleza para concretar esta etapa importante de mi formación profesional.

A mi madre, Natividad, por todo su esfuerzo, sacrificio y apoyo constante en cada momento de mi vida.

A mis abuelos, Pedro y Nicolasa, quienes en vida me brindaron apoyo, cariño y enseñanzas durante mi camino universitario.

A mi asesor, Ing. Luis Alberto Llontop Cumpa, por su apoyo incondicional, orientación académica y valiosos consejos para la elaboración del presente trabajo.

A todas las personas que, de una u otra manera, contribuyeron a que este logro sea posible.

RESUMEN

El presente trabajo de suficiencia profesional se basa en la experiencia desarrollada en la empresa Solpack S.A.C., donde se identificó la necesidad de mejorar la gestión de incidencias dirigidas al área de Sistemas. Las solicitudes de los usuarios internos se atendían por diferentes medios, como correos, mensajes de WhatsApp o comunicaciones verbales, lo que generaba demoras, pérdida de información y poca trazabilidad en el seguimiento de cada caso.

Ante esta situación, se propuso y desarrolló el sistema web INTRANET SOLPACK, una herramienta orientada a centralizar el registro, asignación, seguimiento y cierre de tickets. Su implementación permitió ordenar el proceso de atención, mejorar la comunicación entre los usuarios y el equipo técnico, y contar con información confiable para evaluar el servicio brindado.

El desarrollo de esta solución representó un aporte importante para la empresa, ya que fortaleció la trazabilidad de las incidencias, redujo la dependencia de canales informales y facilitó la generación de indicadores para la toma de decisiones. Asimismo, permitió aplicar conocimientos técnicos y profesionales en un contexto real de mejora continua.

En conclusión, la implementación de la INTRANET SOLPACK contribuyó a brindar una atención más ordenada, transparente y eficiente, evidenciando la capacidad de identificar problemas reales, proponer soluciones tecnológicas y generar valor dentro de la organización.

Palabras clave: Gestión de incidencias, Sistema web, Intranet Solpack.

ABSTRACT

This professional proficiency work is based on the experience developed at Solpack S.A.C., where the need to improve the management of incidents addressed to the Systems area was identified. Internal user requests were handled through different channels, such as emails, WhatsApp messages, or verbal communication, which caused delays, loss of information, and limited traceability in the follow-up of each case.

In response to this situation, the INTRANET SOLPACK web system was proposed and developed as a tool aimed at centralizing the registration, assignment, monitoring, and closure of tickets. Its implementation made it possible to organize the support process, improve communication between users and the technical team, and obtain reliable information to evaluate the service provided.

The development of this solution represented an important contribution to the company, as it strengthened incident traceability, reduced dependence on informal communication channels, and facilitated the generation of indicators for decision-making. Likewise, it allowed technical and professional knowledge to be applied in a real context of continuous improvement.

In conclusion, the implementation of INTRANET SOLPACK contributed to providing a more organized, transparent, and efficient service, demonstrating the ability to identify real problems, propose technological solutions, and generate value within the organization.

Keywords: Incident management, Web system, Intranet Solpack.

ÍNDICE

RESUMEN.....	10
ABSTRACT	11
INTRODUCCIÓN	16
I. CAPÍTULO I: GENERALIDADES DEL PROYECTO.....	17
1.1. Descripción del problema:	17
1.2. Objetivos.....	18
1.3. Limitaciones de la investigación	18
1.4. Justificación del informe	19
II. CAPÍTULO II: FUNDAMENTO TEÓRICO.....	20
2.1. Sistema web	20
2.2. Gestión de incidencias	20
2.3 Sistema de tickets	20
2.4. Mesa de ayuda (Helpdesk)	21
2.5. Arquitectura MVC	21
2.6. Tecnologías para el desarrollo web.....	22
2.6.1. PHP	22
2.6.2. MySQL.....	22
2.6.3 AJAX y JSON	23
2.7. Metodologías ágiles.....	23
2.7.1. SCRUM.....	24
2.7.2. KANBAN.....	25
III. CAPÍTULO III: DESARROLLO DE LA SOLUCIÓN	27
3.1. Descripción de la solución propuesta	27
3.2. Metodología de desarrollo aplicada.....	28
3.3. Análisis y definición de requerimientos	30
3.3.1. Identificación de usuarios y roles.....	30
3.3.2. Requerimientos funcionales	30
3.3.3. Estados de los tickets.....	32
3.3.4. Reglas de negocio.....	32
3.3.5. Evolución del formulario de registro.....	33
3.3.6. Requerimientos no funcionales	34
3.4. Diseño de la solución.....	35
3.4.1. Diseño de la arquitectura del sistema	35

3.4.2. Diseño de la base de datos.....	38
3.4.3. Diseño de los módulos del sistema.....	41
3.4.4. Diseño del flujo de atención de tickets.....	43
3.5. Desarrollo e implementación.....	45
3.5.1. Desarrollo del sistema	45
3.5.2. Implementación del módulo de usuarios y accesos.....	45
3.5.3. Implementación del módulo de gestión de tickets	47
3.5.4. Implementación del módulo de configuración de tickets.....	49
3.5.5. Despliegue del sistema	51
CONCLUSIONES.....	53
RECOMENDACIONES.....	54
ANEXOS.....	57
Anexo 01: Código SQL de la base de datos	57
Anexo 02: Estructura del proyecto	61
Anexo 03: Configuración del entorno de desarrollo.....	62
Anexo 04: Dashboard Administrativo.	64

ÍNDICE DE FIGURAS

Figura 1 Ciclo de vida de un ticket	21
Figura 2 Patrón Modelo - Vista - Controlador (MVC)	22
Figura 3 AJAX y JSON - Flujo de comunicación.....	23
Figura 4 Metodología ágil	24
Figura 5 Ciclo Simplificado de SCRUM	25
Figura 6 Tablero KANBAN.....	26
Figura 7 Flujo general de atención de incidencias.....	27
Figura 8 Enfoque ágil Iterativo – Incremental (Kanban / Scrum Ligero).....	29
Figura 9 Evolución del formulario de registro de tickets.....	34
Figura 10 MVC en PHP (Clásico) + Endpoints AJAX + SPA Ligera.....	37
Figura 11 Modelo simplificado de las principales entidades de la base de datos	39
Figura 12 Módulos principales del sistema web.....	42
Figura 13 Módulo de Tickets.....	42
Figura 14 Flujo de atención y gestión de tickets del sistema web.	44
Figura 15 Interfaz de autenticación del sistema web.....	46
Figura 16 Modulo de gestión de roles de acceso	46
Figura 17 Configuración de acceso a los módulos	47
Figura 18 Formulario de registro de tickets	48
Figura 19 Detalle y seguimiento de un ticket.....	48
Figura 20 Listado de los tickets	49
Figura 21 Módulo de tickets - Gestión de subcategorías.....	50
Figura 22 Módulo de tickets – Gestión de títulos.....	50
Figura 23 Módulo de tickets - Gestión de prioridades.....	51
Figura 24 Arquitectura de despliegue del sistema web.....	52

LISTA DE TABLAS

Tabla 1 Definición de roles de usuarios para su acceso	30
Tabla 2 Detalle de los requerimientos funcionales	31
Tabla 3 Estados de la gestión de tickets.	32
Tabla 4 Detalle de las reglas de negocio	33
Tabla 5 Detalles de los requerimientos no funcionales	34
Tabla 6 Estructura principal de tickets	40
Tabla 7 Módulos principales del sistema.....	41

INTRODUCCIÓN

En la actualidad, la tecnología desempeña un papel crucial en el crecimiento y funcionamiento de las empresas, ya que ayuda a optimizar procedimientos, facilitar el acceso a datos y aumentar la eficacia de las operaciones internas. En este marco, la atención rápida a los problemas y necesidades tecnológicas es esencial para asegurar el flujo continuo de las actividades y prevenir paradas que afecten el rendimiento de las distintas áreas de una organización. Por ello, una correcta gestión del soporte técnico debe centrarse no solo en solucionar los inconvenientes que surgen, sino también en mantener un registro estructurado que permita hacer seguimiento y evaluar la calidad del servicio ofrecido.

En las entidades, el departamento de Sistemas desempeña una función clave al ofrecer asistencia a los usuarios y ayudar a que los recursos tecnológicos operen eficientemente en las tareas cotidianas. En el caso de Solpack S. A. C. , una compañía en Lima que está en continuo crecimiento, el aumento de las operaciones y requerimientos tecnológicos ha hecho necesario establecer sistemas que faciliten la organización de los problemas reportados por las diversas áreas. Un manejo que dependa de canales no centralizados y de métodos manuales puede complicar el seguimiento de las peticiones, ocasionar demoras en su atención y restringir la disponibilidad de datos para analizar el rendimiento del servicio.

Frente a esta necesidad, se plantea la creación de una plataforma web destinada a la administración de incidentes en el área de sistemas. Su objetivo es mejorar el proceso de asistencia técnica interna a través de la centralización y automatización del registro, el seguimiento y la resolución de las incidencias que reportan los usuarios. Además, la iniciativa pretende facilitar la asignación y el control de las solicitudes, conservar un registro del servicio brindado y ofrecer datos que ayuden a analizar los tiempos de respuesta, reconocer problemas recurrentes y apoyar la toma de decisiones. Así, se pretende optimizar la organización, la trazabilidad y la eficacia de la gestión de soporte en el área de Sistemas.

I. CAPÍTULO I: GENERALIDADES DEL PROYECTO

1.1.Descripción del problema:

En el desarrollo de mis funciones, se identificó una problemática recurrente en la gestión de incidencias y requerimientos técnicos reportados por los usuarios internos de la empresa. Las solicitudes se comunicaban de forma verbal, por WhatsApp o mediante correo electrónico, sin contar con un canal oficial y centralizado para su registro y seguimiento. La atención se realizaba de manera informal, considerando la disponibilidad y urgencia percibida, sin un mecanismo establecido para asignar responsables, establecer prioridades o controlar los tiempos de atención. Esta situación ocasionaba pérdida de seguimiento, retrasos y la necesidad de que los usuarios volviesen a comunicarse para consultar el estado de sus solicitudes.

Asimismo, no existía un registro centralizado e histórico de las incidencias atendidas. La información estaba dispersa en correos electrónicos y mensajes de WhatsApp, lo que dificultaba consultar antecedentes, detectar incidencias recurrentes y registrar las soluciones implementadas. Además, no había datos confiables sobre los tiempos de respuesta y resolución, el número de solicitudes atendidas por cada técnico o las áreas que requerían mayor soporte. Estas limitaciones complicaban la distribución de tareas, la identificación de sobrecargas laborales y la toma de decisiones basada en información objetiva. Asimismo, afectaban la percepción de los usuarios sobre la organización y eficiencia del servicio de soporte.

Frente a esta problemática, se propuso el desarrollo de un sistema web para la gestión de incidencias dirigidas al área de sistemas. Este busca centralizar el registro de solicitudes, facilitar su asignación y seguimiento, mantener un historial organizado de las atenciones, y generar información que ayude a mejorar la eficiencia, trazabilidad y control del servicio de soporte técnico.

1.2.Objetivos

1.2.1. Objetivo principal

- Implementar un sistema web centralizado y automatizado que permita gestionar de manera eficiente y ordenada las incidencias, requerimientos y solicitudes técnicas de los usuarios internos, mejorando el control, la rapidez en las respuestas y la excelencia del servicio brindado por el área de Sistemas.

1.2.2. Objetivos específicos:

- Unificar los canales de atención mediante una plataforma web que centralice el registro de incidencias, requerimientos y solicitudes técnicas de los usuarios internos.
- Registrar y organizar las incidencias y requerimientos, permitiendo su consulta y filtrado según criterios como fecha, tipo, área y responsable.
- Facilitar la comunicación y consulta del estado de las solicitudes entre los usuarios internos y el área de Sistemas, proporcionando mayor transparencia durante el proceso de atención.
- Brindar visibilidad del estado actual de cada solicitud, tanto para el área de Sistemas como para el usuario solicitante.
- Generar estadísticas e indicadores sobre las atenciones realizadas, como cantidad de solicitudes, tiempos de respuesta y resolución e incidencias más frecuentes, con el fin de apoyar la toma de decisiones y la mejora continua del servicio.

1.3. Limitaciones de la investigación

El desarrollo de la investigación estuvo condicionado por la disponibilidad de información relacionada con la gestión de incidencias del área de Sistemas, debido a que anteriormente no se contaba con un registro centralizado de las atenciones realizadas. Asimismo, la información histórica disponible era limitada y se encontraba distribuida entre diferentes medios de comunicación y archivos utilizados por la empresa. El desarrollo y validación de la propuesta también estuvieron sujetos al tiempo disponible para su ejecución, a la disponibilidad de los usuarios y personal del área de Sistemas para proporcionar información y participar en las pruebas, así como a los recursos tecnológicos disponibles en la empresa.

1.4. Justificación del informe

El presente proyecto se justifica por la necesidad de mejorar la gestión de las incidencias, requerimientos y solicitudes técnicas del área de Sistemas de Solpack S.A.C., mediante una solución que permita centralizar y organizar el proceso de atención. La implementación del sistema web contribuirá a mejorar el registro, seguimiento y control de las solicitudes, facilitando la atención de los usuarios internos y proporcionando información que apoye la toma de decisiones y la mejora del servicio.

Desde el ámbito académico y profesional, el desarrollo del proyecto permite evidenciar la aplicación de los conocimientos y competencias adquiridos durante la formación profesional en el análisis de una problemática real y en el desarrollo de una solución tecnológica. Asimismo, el presente informe documenta las etapas, procedimientos y resultados obtenidos durante la ejecución del proyecto, por lo que podrá servir como material de consulta y referencia para futuros estudiantes que desarrollen trabajos relacionados con la gestión de incidencias, sistemas web o la implementación de soluciones informáticas orientadas a mejorar procesos empresariales.

II. CAPÍTULO II: FUNDAMENTO TEÓRICO

2.1. Sistema web

Valente (2024) señala que las aplicaciones web constituyen sistemas de software que utilizan tecnologías de la Web para proporcionar funcionalidades a los usuarios mediante un navegador. Estas aplicaciones permiten desarrollar soluciones orientadas al procesamiento y gestión de información, facilitando el acceso a los servicios del sistema desde diferentes dispositivos.

MDN Contributors (2026) indica que las aplicaciones web funcionan mediante una comunicación entre el cliente y el servidor, donde el navegador realiza solicitudes HTTP y el servidor procesa dichas solicitudes para devolver las respuestas correspondientes. Este modelo permite gestionar información y recursos de manera centralizada mediante una plataforma accesible desde un navegador.

2.2. Gestión de incidencias

Ghazizadeh et al. (2019) señalan que la gestión de incidencias constituye una de las actividades principales de la gestión de servicios de tecnologías de la información (ITSM), orientada a atender los inconvenientes, errores o fallas reportados por los usuarios durante el uso de un servicio. Su adecuada gestión permite mejorar la eficiencia de los procesos de atención y contribuir a la continuidad del servicio.

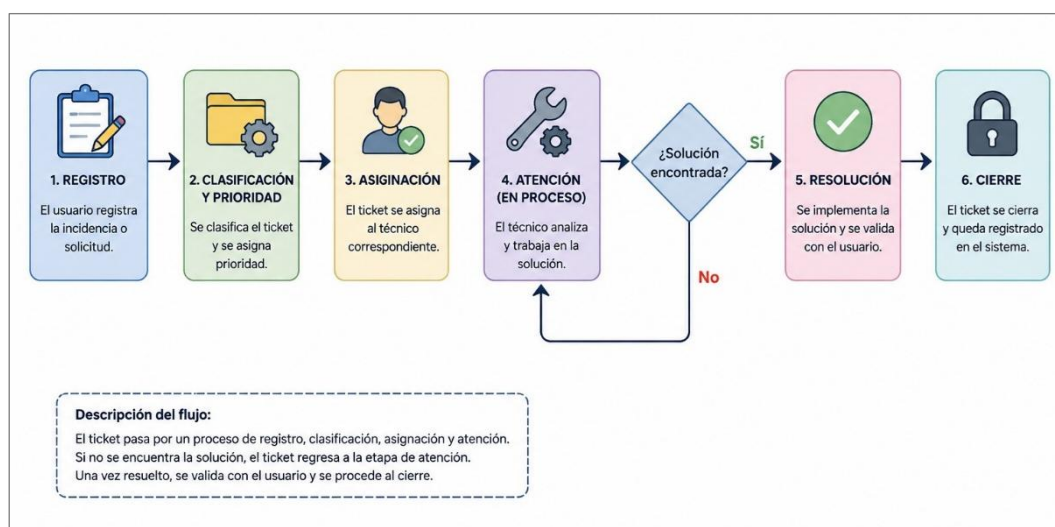
PeopleCert (2026) establece que la gestión de incidencias tiene como propósito restablecer rápidamente el funcionamiento normal de un servicio después de una interrupción, considerando actividades de atención, seguimiento, resolución y medición del desempeño del proceso.

2.3 Sistema de tickets

Salah et al. (2016) señalan que un sistema de tickets de incidentes constituye una herramienta utilizada en la gestión de servicios de tecnologías de la información para realizar el seguimiento, diagnóstico y coordinación de la resolución de los incidentes reportados durante la operación de los servicios. Asimismo, una adecuada organización de los tickets puede contribuir a mejorar la eficiencia del proceso de gestión de incidencias.

Figura 1

Ciclo de vida de un ticket



Nota. Elaboración propia basada en Salah et al.(2016).

2.4. Mesa de ayuda (Helpdesk)

Jäntti et al. (2012) señalan que la mesa de ayuda constituye un punto único de contacto entre los usuarios y el área de tecnologías de la información para la atención de solicitudes de soporte. Su función comprende la recepción y gestión de las solicitudes, así como la comunicación y seguimiento de los casos durante el proceso de atención.

Agir et al. (2018) sostienen que una mesa de ayuda permite centralizar la atención de los usuarios mediante un punto único de contacto, facilitando el control de los problemas reportados y contribuyendo a una atención más rápida y organizada.

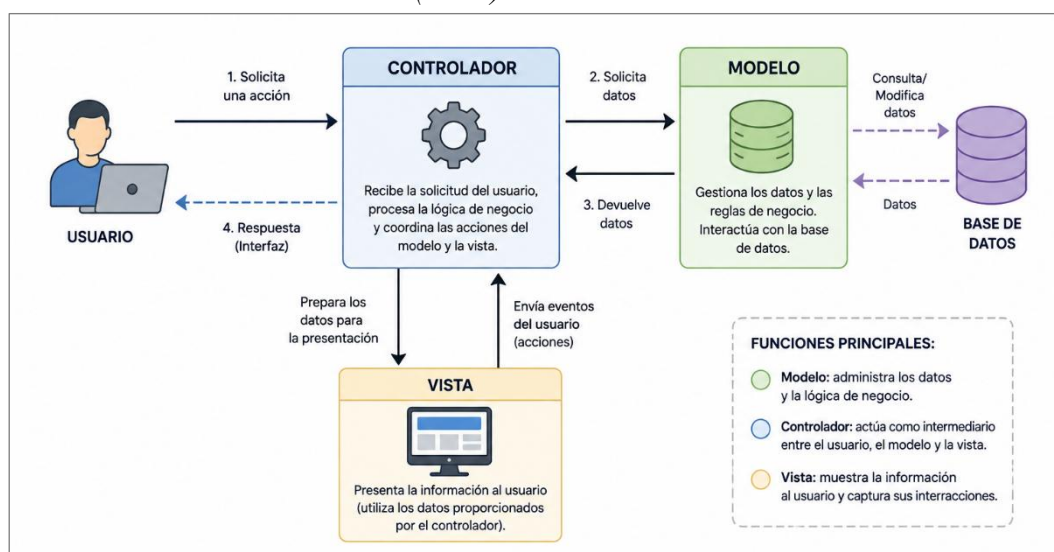
2.5. Arquitectura MVC

Krasner y Pope (1988) plantean el patrón Modelo-Vista-Controlador (MVC) como una forma de organizar una aplicación mediante la separación de los componentes encargados de gestionar los datos, presentar la información y controlar la interacción con el usuario. Esta separación permite distribuir las responsabilidades de la aplicación y facilita su mantenimiento y evolución.

Seshadri (2007) señala que MVC es un patrón ampliamente utilizado en el desarrollo de aplicaciones web, debido a que permite separar la lógica de negocio de la presentación y del control de las solicitudes, favoreciendo una estructura organizada del código.

Figura 2

Patrón Modelo - Vista - Controlador (MVC)



Nota. Elaboración propia basada en Krasner y Pope (1988).

2.6. Tecnologías para el desarrollo web

Según MDN Contributors (s. f.), HTML permite definir la estructura y el contenido de las páginas web, mientras que CSS se utiliza para establecer su presentación y apariencia. Por su parte, JavaScript permite incorporar interactividad y comportamiento dinámico a las aplicaciones web, facilitando la interacción del usuario con los elementos de la interfaz. (developer.mozilla.org).

Bootstrap (s. f.) señala que es un conjunto de herramientas de frontend orientado al desarrollo de interfaces web responsivas, proporcionando componentes y utilidades que facilitan la construcción de diseños adaptables a diferentes tamaños de pantalla. (getbootstrap.com).

2.6.1. PHP

Según PHP Documentation Group (s. f.), PHP es un lenguaje de programación de código abierto orientado principalmente al desarrollo de aplicaciones web del lado del servidor. Su ejecución permite procesar solicitudes, generar contenido dinámico e interactuar con diferentes sistemas de bases de datos, por lo que constituye una herramienta adecuada para desarrollar aplicaciones web dinámicas.

2.6.2. MySQL

Según Oracle (s. f.), MySQL es un sistema de gestión de bases de datos relacionales que utiliza SQL para almacenar, organizar y consultar información. Los datos se estructuran

mediante elementos como bases de datos, tablas, filas y columnas, permitiendo establecer relaciones entre diferentes conjuntos de información y mantener su integridad.

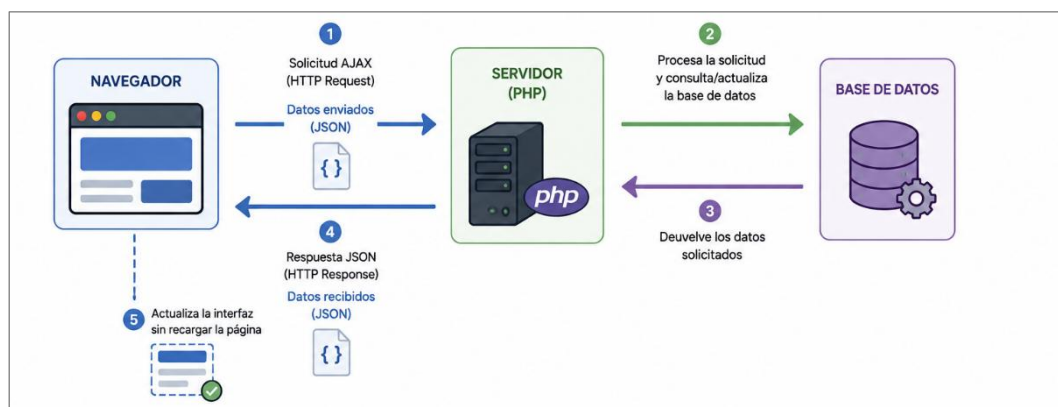
2.6.3 AJAX y JSON

Según MDN Contributors (2025), AJAX es una técnica de desarrollo web que permite realizar solicitudes HTTP de manera asíncrona para obtener información del servidor y actualizar determinadas partes de una página sin necesidad de realizar una carga completa. Esta característica permite desarrollar interfaces más dinámicas y mejorar la interacción del usuario con las aplicaciones web.

Asimismo, MDN Contributors (2025) señala que JSON es un formato de texto utilizado para representar datos estructurados y facilitar su intercambio entre diferentes componentes de una aplicación. En las aplicaciones web es utilizado con frecuencia para transmitir información entre el cliente y el servidor.

Figura 3

AJAX y JSON - Flujo de comunicación

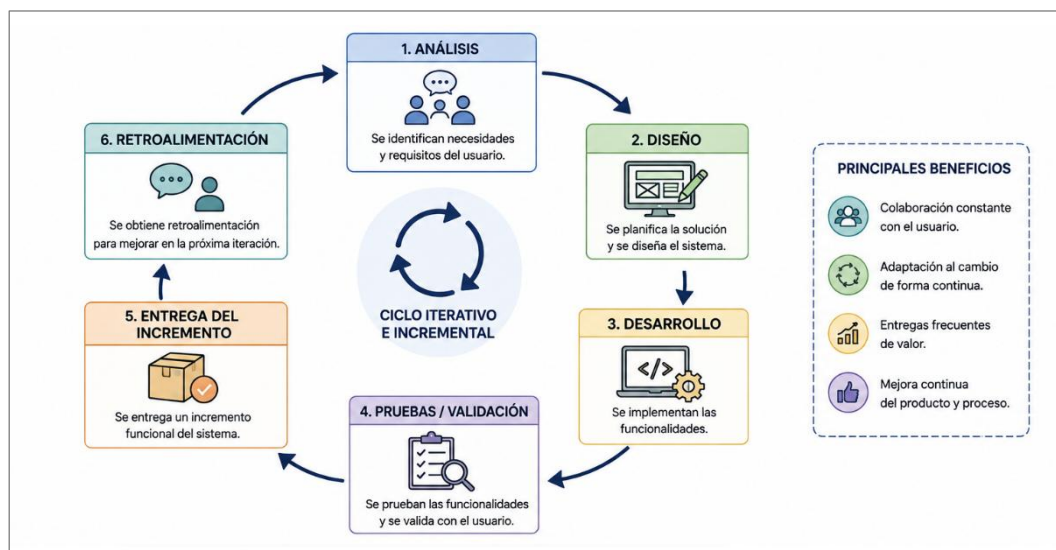


Nota. Elaboración propia basada en MDN Contributors (2025).

2.7. Metodologías ágiles

Beck et al. (2001) establecen en el Manifiesto por el Desarrollo Ágil de Software un enfoque basado en la colaboración con el cliente, la respuesta al cambio, la entrega frecuente de software funcional y la interacción entre las personas involucradas en el desarrollo. Estos principios buscan que el desarrollo de software pueda adaptarse progresivamente a las necesidades que surjan durante el proyecto.

Figura 4
Metodología ágil



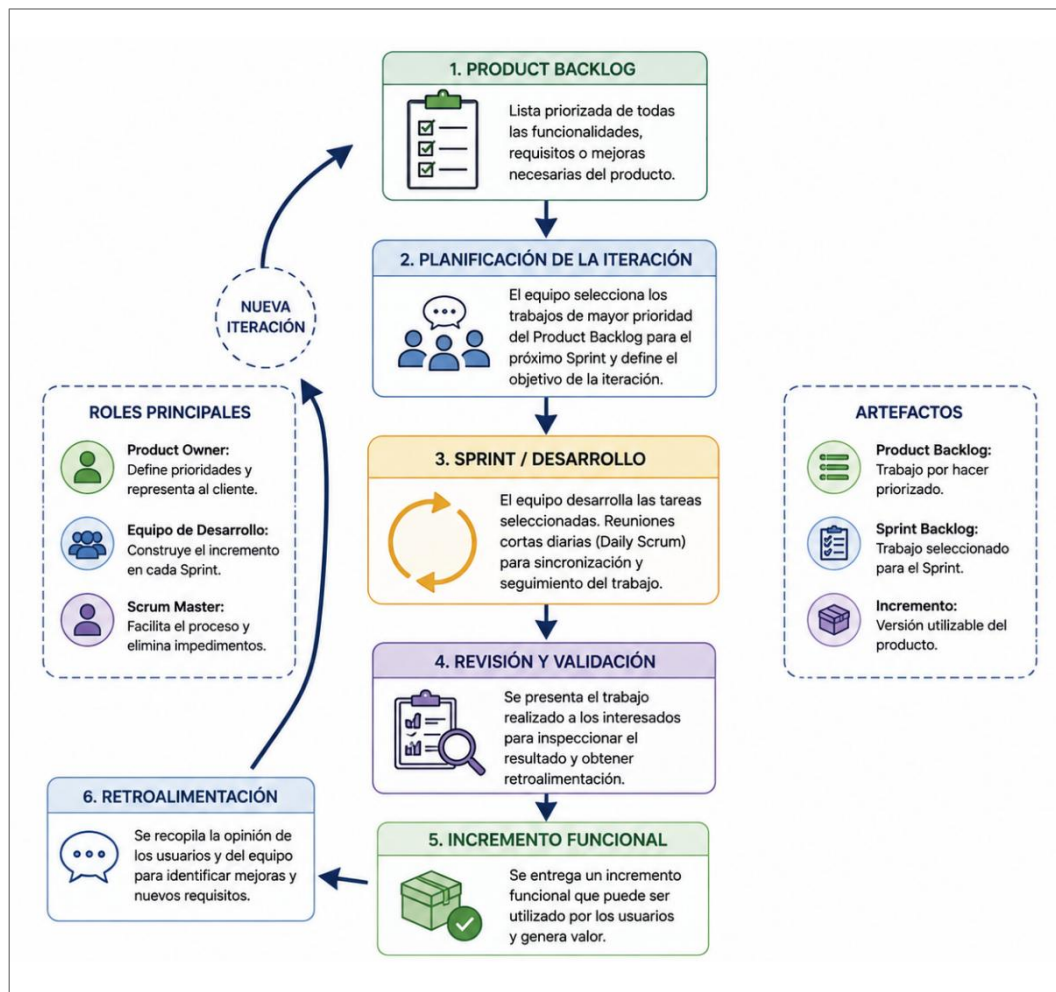
Nota. Elaboración propia basada en Beck et Al. (2001)

2.7.1. SCRUM

Schwaber y Sutherland (2020) definen Scrum como un marco de trabajo ligero que permite generar valor mediante soluciones adaptativas para problemas complejos. Este marco se basa en un proceso iterativo e incremental, en el que el trabajo se organiza en periodos denominados Sprints, permitiendo inspeccionar los resultados obtenidos y realizar ajustes de manera continua.

Asimismo, Scrum establece responsabilidades, eventos y artefactos que permiten organizar el trabajo del equipo. Entre sus principales elementos se encuentran el Product Backlog, que contiene el trabajo necesario para desarrollar el producto; el Sprint, periodo en el que se desarrolla un incremento; y el incremento que representa una versión utilizable del producto que cumple con los criterios establecidos.

Figura 5
Ciclo Simplificado de SCRUM



Nota. Elaboración propia basada en Schwaber y Sutherland (2020).

2.7.2. KANBAN

Anderson, D.J. (2010) señala que Kanban es un método orientado a la gestión del trabajo y la mejora del flujo de actividades, mediante la visualización del trabajo, la limitación del trabajo en progreso y la gestión continua del flujo. Estas prácticas permiten identificar obstáculos, controlar la cantidad de tareas que se encuentran en proceso y favorecer una entrega más continua.

Asimismo, Kanban permite representar visualmente el estado de las actividades mediante un tablero, facilitando que el equipo conozca qué tareas están pendientes, cuáles se encuentran en desarrollo y cuáles han sido finalizadas. De esta manera, el trabajo puede ser priorizado y supervisado de forma sencilla, promoviendo la mejora continua del proceso.

Figura 6
Tablero KANBAN



Nota. Elaboración propia basada en Anderson, D.J. (2010).

III. CAPÍTULO III: DESARROLLO DE LA SOLUCIÓN

3.1. Descripción de la solución propuesta

Como solución a la problemática identificada en la gestión de incidencias del área de Sistemas de Solpack S.A.C., se desarrolló un sistema web orientado a centralizar y organizar el registro, seguimiento y resolución de las incidencias reportadas por los usuarios internos. El sistema se encuentra desarrollado y operativo en un entorno local, y permite el acceso de los usuarios mediante una URL desde un navegador web, de acuerdo con los permisos asignados a cada rol.

La solución contempla tres tipos principales de usuarios: administrador, técnico de Sistemas y usuario solicitante, quienes intervienen en las diferentes etapas del proceso de atención. El flujo inicia cuando el usuario registra un ticket indicando la información correspondiente a la incidencia; posteriormente, el área de Sistemas recibe y asigna el caso al técnico responsable, quien realiza la atención, registra comentarios, actualiza el estado y puede adjuntar evidencias durante el proceso. Una vez solucionado el caso, se valida la atención y se procede con el cierre del ticket, permitiendo además registrar la satisfacción del usuario.

Figura 7
Flujo general de atención de incidencias



Nota. Elaboración propia.

El sistema incorpora funcionalidades para la gestión de tickets, categorías y subcategorías, títulos, prioridades, estados, asignación de técnicos, comentarios, evidencias, notificaciones, cierre de solicitudes, encuestas de satisfacción, reportes, estadísticas y dashboard. Estas funcionalidades permiten centralizar la información de las atenciones,

mantener un historial de cada caso y disponer de información organizada para el seguimiento y análisis de la gestión realizada por el área de Sistemas.

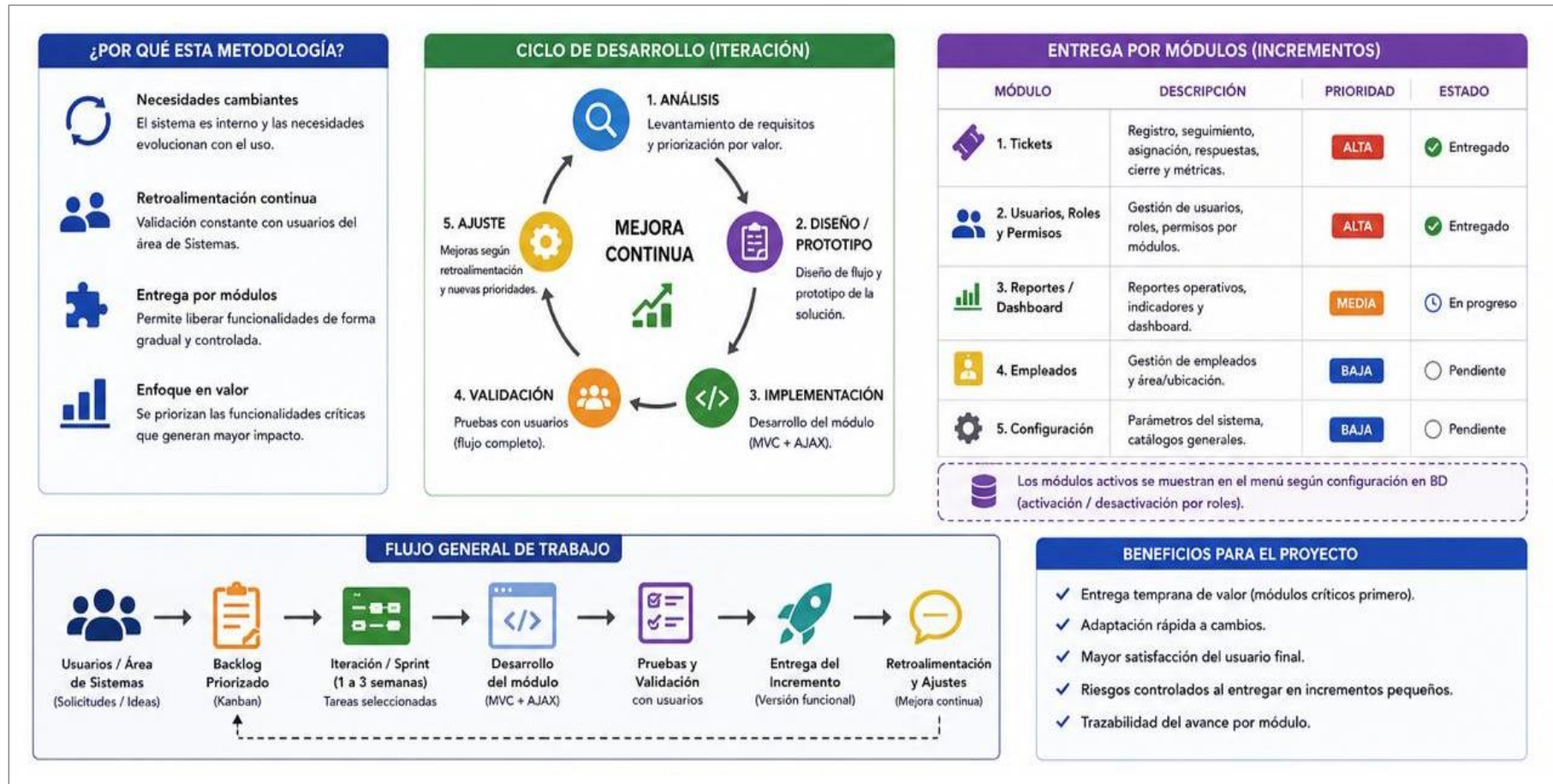
3.2. Metodología de desarrollo aplicada

Para el desarrollo del sistema web se adoptó un enfoque ágil iterativo-incremental, combinando prácticas de Scrum ligero y Kanban para organizar y priorizar las actividades de desarrollo. El proyecto fue desarrollado de manera individual por el autor, bajo la supervisión de la jefatura del área de Sistemas, considerando la participación de los usuarios durante las etapas de prueba y validación de las funcionalidades implementadas.

El desarrollo se realizó progresivamente mediante módulos, priorizando inicialmente las funcionalidades necesarias para establecer el acceso al sistema, la gestión de usuarios, roles y permisos, y posteriormente las funcionalidades relacionadas con la administración y atención de tickets. A partir de las necesidades identificadas, se incorporaron progresivamente elementos como categorías, subcategorías, títulos, prioridades, SLA, asignación de técnicos, estados, comentarios, evidencias, notificaciones, cierre de tickets, encuestas de satisfacción, reportes y dashboard.

Durante el proceso se realizaron pruebas con los usuarios del área de Sistemas, quienes proporcionaron observaciones sobre el funcionamiento y la facilidad de uso del sistema. Como resultado de esta retroalimentación, se desarrollaron dos versiones principales. En la primera versión, el registro de tickets se realizaba mediante un formulario general en el que el usuario ingresaba la información correspondiente a su solicitud. Posteriormente, en la segunda versión, se mejoró el proceso de registro mediante la incorporación de títulos predefinidos asociados a configuraciones de prioridad y SLA, permitiendo estandarizar determinados criterios de atención y facilitar el registro de las solicitudes.

Figura 8
Enfoque ágil Iterativo – Incremental (Kanban / Scrum Ligero)



Nota. Elaboración propia.

De esta manera, el desarrollo permitió incorporar mejoras de forma progresiva a partir de las observaciones obtenidas durante las pruebas, manteniendo un proceso de construcción orientado a la entrega de funcionalidades y a la mejora continua del sistema.

3.3. Análisis y definición de requerimientos

A partir del análisis del proceso de atención de incidencias y de las necesidades identificadas en el área de Sistemas, se determinaron los requerimientos necesarios para el desarrollo del sistema web. Estos requerimientos permitieron establecer las funcionalidades, restricciones y reglas que debía cumplir la solución para facilitar el registro, asignación, seguimiento y cierre de los tickets.

3.3.1. Identificación de usuarios y roles

El sistema contempla tres perfiles de usuario, cada uno con funciones y niveles de acceso diferenciados según las actividades que desarrolla dentro del proceso de atención.

Tabla 1

Definición de roles de usuarios para su acceso

Usuario	Función principal
Administrador	Corresponde al responsable o jefe del área. Gestiona la configuración del sistema y puede realizar la asignación de tickets a los técnicos.
Técnico de sistemas	Atiende los tickets asignados y puede autoasignarse nuevos tickets disponibles. Durante la atención actualiza el estado, registra comentarios, adjunta evidencias y registra la solución.
Usuario solicitante	Registra sus incidencias y puede consultar únicamente los tickets que ha generado, realizar el seguimiento correspondiente y validar la solución recibida.

Nota. Elaboración propia.

Esta distribución permite establecer diferentes niveles de acceso y mantener una separación de responsabilidades dentro del proceso de atención.

3.3.2. Requerimientos funcionales

Los requerimientos funcionales definen las operaciones que debe realizar el sistema para soportar el proceso de gestión de incidencias.

Tabla 2*Detalle de los requerimientos funcionales*

Código	Requerimiento funcional
RF01	Permitir el inicio de sesión de los usuarios registrados.
RF02	Permitir gestionar usuarios y sus respectivos roles y permisos.
RF03	Permitir gestionar categorías y subcategorías de las incidencias.
RF04	Permitir gestionar títulos predefinidos para los tickets.
RF05	Permitir configurar la prioridad y el SLA asociado a los títulos de atención.
RF06	Permitir al usuario solicitante registrar un ticket seleccionando un título, ingresando la descripción y adjuntando una evidencia cuando sea necesario.
RF07	Permitir registrar archivos adjuntos en diferentes formatos como evidencia del ticket.
RF08	Permitir al administrador asignar tickets a los técnicos de Sistemas.
RF09	Permitir que un técnico se autoasigne un ticket nuevo disponible para su atención.
RF10	Permitir actualizar el estado de los tickets durante el proceso de atención.
RF11	Permitir registrar comentarios y observaciones en el detalle del ticket.
RF12	Permitir al técnico registrar información relacionada con la solución de la incidencia.
RF13	Permitir notificar al usuario sobre los cambios relevantes relacionados con su ticket.
RF14	Permitir validar la atención y realizar el cierre del ticket.
RF15	Permitir registrar la satisfacción del usuario después de la atención.
RF16	Permitir consultar y generar reportes y estadísticas de los tickets gestionados.
RF17	Mostrar información consolidada de la gestión mediante un dashboard.

R18	Permitir al usuario solicitante consultar únicamente sus propios tickets.
R19	Permitir al técnico consultar los tickets que se encuentran disponibles para atención y aquellos que se encuentran bajo su gestión.

Nota. Elaboración propia.

3.3.3. Estados de los tickets

El ciclo de vida de un ticket se encuentra definido mediante estados que permiten identificar su situación durante la atención.

Tabla 3

Estados de la gestión de tickets.

Estado	Descripción
Nuevo	El ticket ha sido registrado por el usuario y se encuentra disponible para ser atendido.
Asignado	El ticket ha sido asignado a un técnico responsable.
En proceso	El técnico se encuentra realizando las actividades necesarias para atender la incidencia.
Resuelto	El técnico ha finalizado la atención y ha registrado la solución correspondiente.
Cerrado	La atención ha sido validada y el ticket ha finalizado su ciclo de gestión.

Nota. Elaboración propia.

3.3.4. Reglas de negocio

A partir del funcionamiento definido para el sistema, se establecieron las siguientes reglas de negocio:

Tabla 4*Detalle de las reglas de negocio*

Código	Regla de negocio
RN01	Todo ticket debe registrarse mediante un título y una descripción de la incidencia o requerimiento.
RN02	El usuario solicitante puede adjuntar una evidencia al momento de registrar el ticket cuando esta sea necesaria para explicar o sustentar la incidencia.
RN03	Los títulos predefinidos pueden tener configuradas una categoría, subcategoría, prioridad y un SLA determinado.
RN04	Al seleccionar un título predefinido, el sistema utiliza la configuración asociada para establecer los parámetros correspondientes del ticket.
RN05	El administrador puede asignar un ticket a un técnico de Sistemas.
RN06	Los técnicos pueden autoasignarse tickets que se encuentren disponibles para atención.
RN07	El usuario solicitante únicamente puede consultar los tickets que ha registrado.
RN08	El técnico puede consultar los tickets nuevos disponibles para atención y aquellos que se encuentran bajo su gestión.
RN09	El ticket debe avanzar mediante los estados definidos para controlar su ciclo de atención: Nuevo, Asignado, En proceso, Resuelto y Cerrado.
RN10	La información registrada durante la atención debe conservarse como parte del historial del ticket.

Nota. Elaboración propia.

3.3.5. Evolución del formulario de registro

Durante el desarrollo se realizaron ajustes al formulario de registro de tickets a partir de las observaciones obtenidas durante las pruebas con los usuarios. En la primera versión, el registro requería que el usuario proporcionara directamente la información correspondiente a la solicitud mediante un formulario general. Posteriormente, en la segunda versión, se incorporó un mecanismo basado en títulos predefinidos, los cuales se encuentran asociados a una categoría, subcategoría, prioridad y SLA previamente configurados. De esta manera, el

usuario únicamente debe seleccionar el título correspondiente, ingresar la descripción de su solicitud y adjuntar una evidencia cuando sea necesario. Esta modificación permitió simplificar el registro y estandarizar determinados criterios de clasificación y atención.

Figura 9
Evolución del formulario de registro de tickets

Versión 1 (Formulario general)	Versión 2 (Con título predefinido)
Numero Ticket: S250803085241 ÁREA DE DESTINO Sistemas TÍTULO (*) DESCRIPCION (*) ADJUNTO Sin archivos seleccionados Elige archivos Puede adjuntar imágenes y documentos menores a 2MB X CANCELAR + CREAR Características principales: - El usuario debía completar el título manualmente. - No existían parámetros predefinidos de categoría, subcategoría, prioridad ni SLA. - Mayor carga de información para el usuario.	Numero Ticket: S250823114929 TÍTULO (*) Selecciona un título Ventas - Cancelar entrega (Bula) Ventas - Cerrar orden de venta Operaciones de stock - Cancelar solicitud de traslado Compras - Cancelar entrada de mercancía Ventas - Error al crear oferta de ventas Datos maestros de artículo - Activar - Desactivar SKUs por puesto DESCRIPCION (*) ADJUNTO Ningún archivo seleccionado Elige archivos Puede adjuntar imágenes y documentos menores a 2MB X CANCELAR + CREAR Características principales: - El usuario selecciona un título predefinido. - El sistema carga automáticamente la categoría, subcategoría, prioridad y el SLA asociado al título. - El usuario solo completa la descripción y adjunta evidencia si es necesario. - Registro más rápido, estandarizado y con menos errores.

Nota. Elaboración propia.

3.3.6. Requerimientos no funcionales

Los requerimientos no funcionales establecen las características relacionadas con la seguridad, facilidad de uso, integridad de la información y mantenimiento del sistema. Estos aspectos complementan las funcionalidades definidas y permiten establecer condiciones para el correcto funcionamiento de la solución propuesta.

Tabla 5
Detalles de los requerimientos no funcionales

Código	Requerimiento no funcional
RNF01	Seguridad: el sistema debe controlar el acceso mediante autenticación y permisos asociados al rol del usuario.
RNF02	Control de acceso: cada usuario debe visualizar y utilizar únicamente las funcionalidades correspondientes a los permisos establecidos para su perfil.
RNF03	Usabilidad: la interfaz debe permitir que los usuarios registren, consulten y gestionen los tickets de manera clara y sencilla.

RNF04	Integridad de la información: el sistema debe mantener la consistencia de la información registrada durante las diferentes etapas del ciclo de vida del ticket.
RNF05	Trazabilidad: el sistema debe conservar la información generada durante la atención de cada ticket, permitiendo realizar el seguimiento de las acciones y cambios efectuados.
RNF06	Accesibilidad: el sistema debe permitir el acceso mediante un navegador web utilizando la URL correspondiente.
RNF07	Mantenibilidad: la estructura del sistema debe facilitar la modificación y ampliación de sus funcionalidades, permitiendo incorporar nuevos módulos o realizar ajustes posteriormente.

Nota. Elaboración propia.

En conjunto, los requerimientos funcionales y no funcionales establecidos sirvieron como base para el diseño y desarrollo del sistema web. Los requerimientos funcionales permitieron determinar las operaciones que debía realizar cada módulo, mientras que los requerimientos no funcionales definieron condiciones relacionadas con la seguridad, acceso, usabilidad, integridad y mantenimiento de la solución.

3.4. Diseño de la solución

El diseño de la solución comprende la definición de la arquitectura, componentes, módulos y mecanismos de interacción que conforman el sistema web desarrollado para la gestión de incidencias dirigidas al área de Sistemas. Para ello, se estableció una arquitectura basada en el patrón Modelo-Vista-Controlador (MVC), complementada con un Front Controller, un mecanismo de autorización mediante middleware y comunicación asíncrona mediante AJAX y respuestas en formato JSON.

La solución fue desarrollada utilizando PHP como tecnología principal del lado del servidor, MySQL para el almacenamiento de información y tecnologías del lado del cliente como HTML, CSS, JavaScript y Bootstrap para la construcción de la interfaz y la interacción con el sistema.

3.4.1. Diseño de la arquitectura del sistema

La arquitectura implementada se encuentra organizada en diferentes componentes que permiten separar la presentación, la lógica de negocio y el acceso a los datos. Esta organización

facilita el mantenimiento del sistema y permite que las funcionalidades sean desarrolladas de manera modular.

La interacción comienza cuando el usuario accede al sistema mediante un navegador web y solicita una determinada ruta. Esta solicitud es recibida por el archivo `index.php`, que funciona como punto de entrada principal de la aplicación. Posteriormente, la solicitud es procesada por el controlador correspondiente, donde se realizan las validaciones necesarias y se determina la operación que debe ejecutarse.

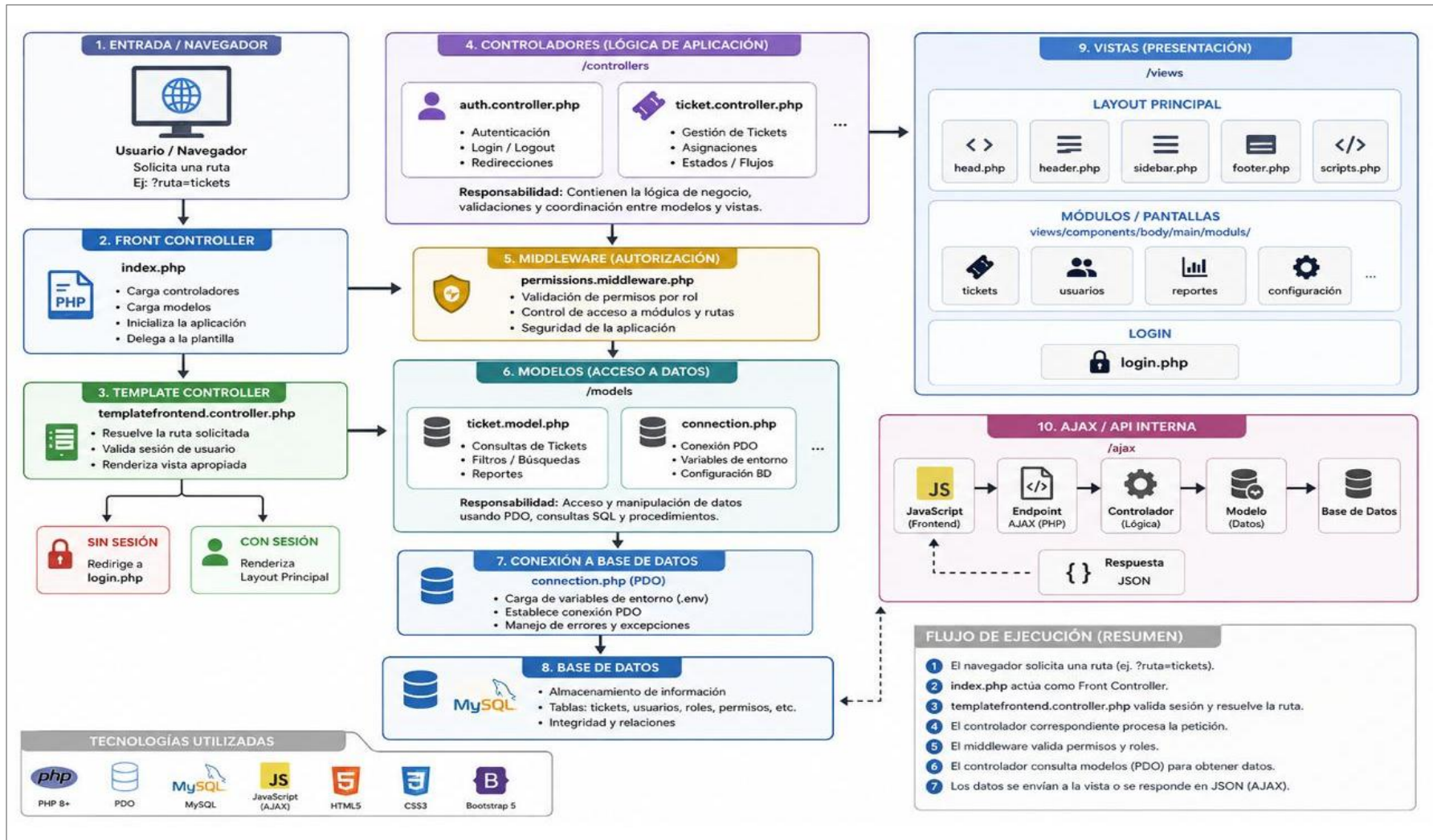
Antes de acceder a las funcionalidades protegidas, el sistema utiliza un mecanismo de middleware de autorización, encargado de verificar la sesión y los permisos asociados al rol del usuario. De esta manera, el acceso a los módulos se encuentra condicionado por los permisos establecidos.

Los controladores gestionan la lógica de aplicación y se comunican con los modelos para realizar las operaciones relacionadas con la base de datos. Los modelos utilizan PDO para establecer la comunicación con MySQL, permitiendo consultar, registrar, actualizar y gestionar la información almacenada.

Finalmente, los resultados son enviados a las vistas para su presentación al usuario. Para determinadas operaciones que requieren actualización de información sin recargar completamente la página, se utiliza AJAX, mediante endpoints internos que procesan la solicitud y retornan respuestas en formato JSON.

Figura 10

MVC en PHP (Clásico) + Endpoints AJAX + SPA Ligera



Nota. Elaboración propia.

3.4.2. Diseño de la base de datos

Para el almacenamiento y gestión de la información del sistema se diseñó una base de datos relacional utilizando MySQL. La estructura fue definida de acuerdo con los requerimientos funcionales identificados, considerando principalmente la gestión de usuarios, roles y permisos, la configuración de las categorías y subcategorías, la administración de títulos y prioridades, así como el registro y seguimiento de los tickets generados por los usuarios.

La entidad central de la solución corresponde a la tabla tickets, debido a que concentra la información principal de cada incidencia o requerimiento registrado. Esta entidad almacena datos como el código del ticket, título, descripción, usuario solicitante, responsable de la atención, estado y fechas asociadas al registro, actualización y cierre. De esta manera, la información almacenada permite realizar el seguimiento del ticket durante las diferentes etapas de su ciclo de atención.

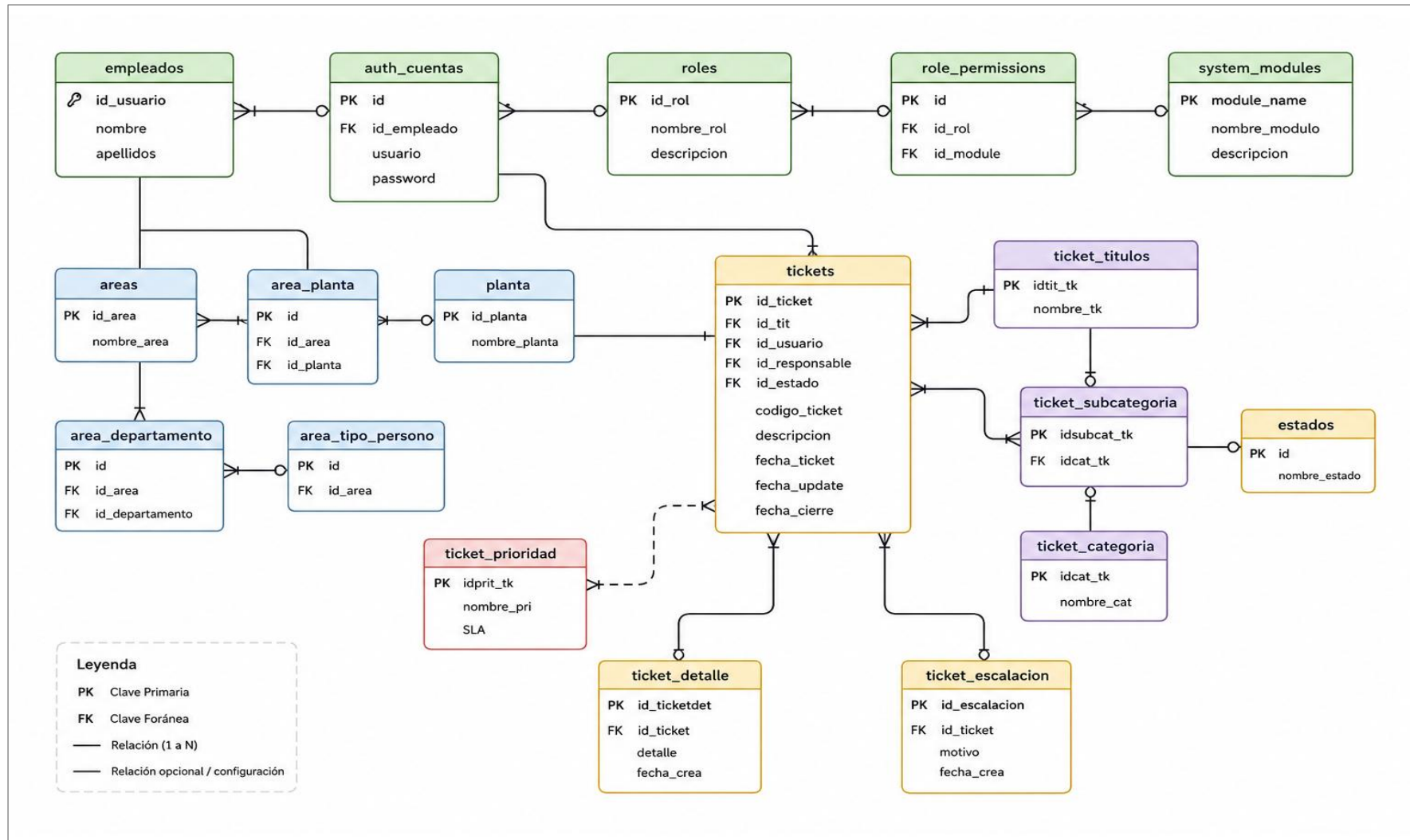
Para la clasificación y configuración de las solicitudes se implementaron las entidades relacionadas con categorías, subcategorías, títulos y prioridades. La tabla ticket_titulos permite definir los títulos disponibles para el registro de solicitudes y asociarlos con la configuración correspondiente. Asimismo, las estructuras relacionadas con ticket_subcategoria y ticket_prioridad permiten organizar la clasificación y los criterios de atención de los tickets. Esta estructura sirve de soporte para la funcionalidad incorporada en la segunda versión del sistema, mediante la cual el usuario selecciona un título predefinido y el sistema utiliza la configuración asociada.

Por otra parte, la gestión de acceso se encuentra estructurada mediante las entidades empleados, auth_cuentas, roles y role_permissions. Esta organización permite separar la información del empleado de sus credenciales de acceso y establecer los permisos correspondientes al rol asignado. De esta forma, el sistema puede controlar las funcionalidades disponibles para los diferentes perfiles de usuario.

La base de datos también contempla entidades relacionadas con la estructura organizacional, como areas, planta, area_planta y area_departamento, además de estructuras destinadas a la administración de los módulos del sistema y al registro de actividades de acceso. Estas entidades permiten complementar la información necesaria para la gestión de usuarios, permisos y funcionalidades de la aplicación.

Figura 11

Modelo simplificado de las principales entidades de la base de datos



Nota. Elaboración propia.

Estructura de las entidades principales

La tabla **tickets** constituye la entidad principal, debido a que concentra la información de las incidencias registradas. Asimismo, las tablas **ticket_detalle** y **ticket_escalacion** permiten complementar la información generada durante la atención de las solicitudes.

Tabla 6

Estructura principal de tickets

Campo	Tipo de dato	Función
id_ticket	INT	Identificador único del ticket.
codigo_ticket	VARCHAR(15)	Código utilizado para identificar la solicitud.
id_tit	INT	Relación con el título seleccionado.
descripcion	MEDIUMTEXT	Descripción proporcionada por el usuario.
adjunto	TEXT	Referencia a la evidencia adjunta, cuando corresponde.
estado	INT	Estado actual del ticket.
id_usuario	INT	Usuario que registra la solicitud.
id_responsable	INT	Responsable asignado para la atención.
fecha_ticket	TIMESTAMP	Fecha de registro del ticket.
fecha_update	TIMESTAMP	Fecha de última actualización.
fecha_cierre	TIMESTAMP	Fecha de cierre de la solicitud.

Nota. Elaboración propia.

La estructura de tickets permite conservar la información necesaria para identificar, clasificar, asignar y realizar el seguimiento de cada solicitud. Las relaciones establecidas con el usuario, responsable, título y estado permiten mantener la trazabilidad del ticket durante su ciclo de atención, desde el registro hasta su cierre.

3.4.3. Diseño de los módulos del sistema

El sistema fue estructurado en módulos funcionales de acuerdo con los requerimientos identificados y con los diferentes perfiles de usuario. Esta organización permite separar las funciones de administración, configuración y gestión de tickets, facilitando el control de acceso y el mantenimiento de la aplicación.

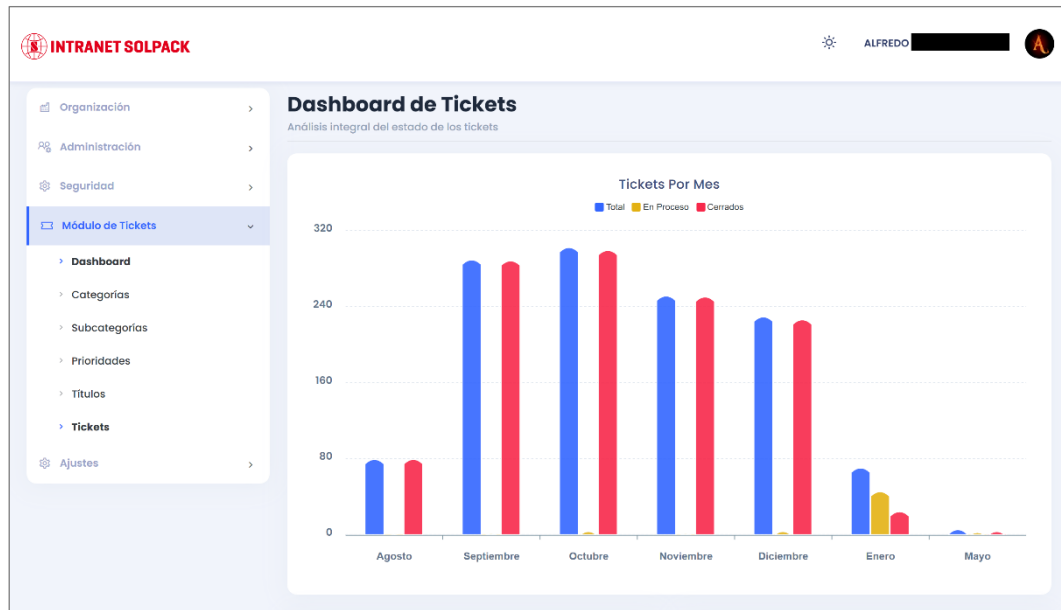
Los principales módulos desarrollados corresponden a usuarios, roles y permisos, categorías y subcategorías, títulos, tickets, detalle y seguimiento de tickets, configuración, reportes y dashboard. Cada módulo cumple una función específica dentro del proceso de gestión de incidencias y su disponibilidad depende de los permisos asignados al usuario.

Tabla 7
Módulos principales del sistema

Módulo	Función principal
Usuarios	Permite gestionar la información de los usuarios y sus cuentas de acceso.
Roles y permisos	Permite definir los perfiles de usuario y controlar el acceso a las funcionalidades del sistema.
Categorías y subcategorías	Permite organizar y clasificar los tipos de solicitudes gestionadas mediante el sistema.
Títulos	Permite establecer títulos predefinidos para las solicitudes y asociarles configuraciones de atención.
Tickets	Permite registrar, asignar, consultar y gestionar las incidencias o requerimientos.
Detalle del ticket	Permite visualizar y registrar información relacionada con la atención, comentarios, evidencias y cambios realizados durante el proceso.
Configuración	Permite administrar parámetros utilizados por los diferentes módulos del sistema.
Reportes	Permite consultar información consolidada sobre los tickets y su atención.
Dashboard	Presenta información resumida del comportamiento de los tickets mediante indicadores y visualizaciones.

Nota. Elaboración propia.

Figura 12
Módulos principales del sistema web



Nota. Elaboración propia.

El módulo de tickets constituye el componente principal de la solución, debido a que concentra el flujo de atención de las solicitudes. Su funcionamiento se encuentra relacionado con los módulos de títulos, categorías, prioridades y usuarios, permitiendo que cada ticket sea registrado, asignado y atendido de acuerdo con la configuración establecida.

Figura 13
Módulo de Tickets

N° Ticket	Estado	Título	Usuario	Fecha	Hora	Acciones
S260512092651	En Proceso	Ventas - Error al crear oferta de ventas	ANA [redacted]	12/05/26	09:33 am	[icon]
S260510040236	En Proceso	Acceso - Desbloquear usuario de SAP	ANA [redacted]	11/05/26	01:51 am	[icon]
S260108024650	En Proceso	Ventas - Cerrar orden de venta	CARLA [redacted]	10/05/26	05:08 pm	[icon]
S260108105629	En Proceso	Registro - actualización de información en los módulos de intranet	DIANA [redacted]	08/01/26	02:33 pm	[icon]
S260108012207	En Proceso	Ventas - Cancelar entrega (Guía)	ANA [redacted]	08/01/26	02:31 pm	[icon]
S260108022744	En Proceso	Query Manager - Crear - Modificar reporte	LUIS [redacted]	08/01/26	02:29 pm	[icon]
S260108125204	En Proceso	Ventas - Cerrar orden de venta	LUIS [redacted]	08/01/26	12:53 pm	[icon]

Nota. Elaboración propia.

Asimismo, la separación de los módulos permite que las funcionalidades sean habilitadas de acuerdo con los permisos del usuario. De esta manera, la estructura modular implementada facilita la administración del sistema y permite incorporar modificaciones o nuevas funcionalidades sin afectar directamente el funcionamiento de los demás componentes.

3.4.4. Diseño del flujo de atención de tickets

El flujo de atención de tickets fue diseñado considerando las diferentes etapas que intervienen en la gestión de una incidencia o requerimiento. El proceso inicia con el registro de la solicitud por parte del usuario y continúa con su asignación, atención, resolución, validación y cierre.

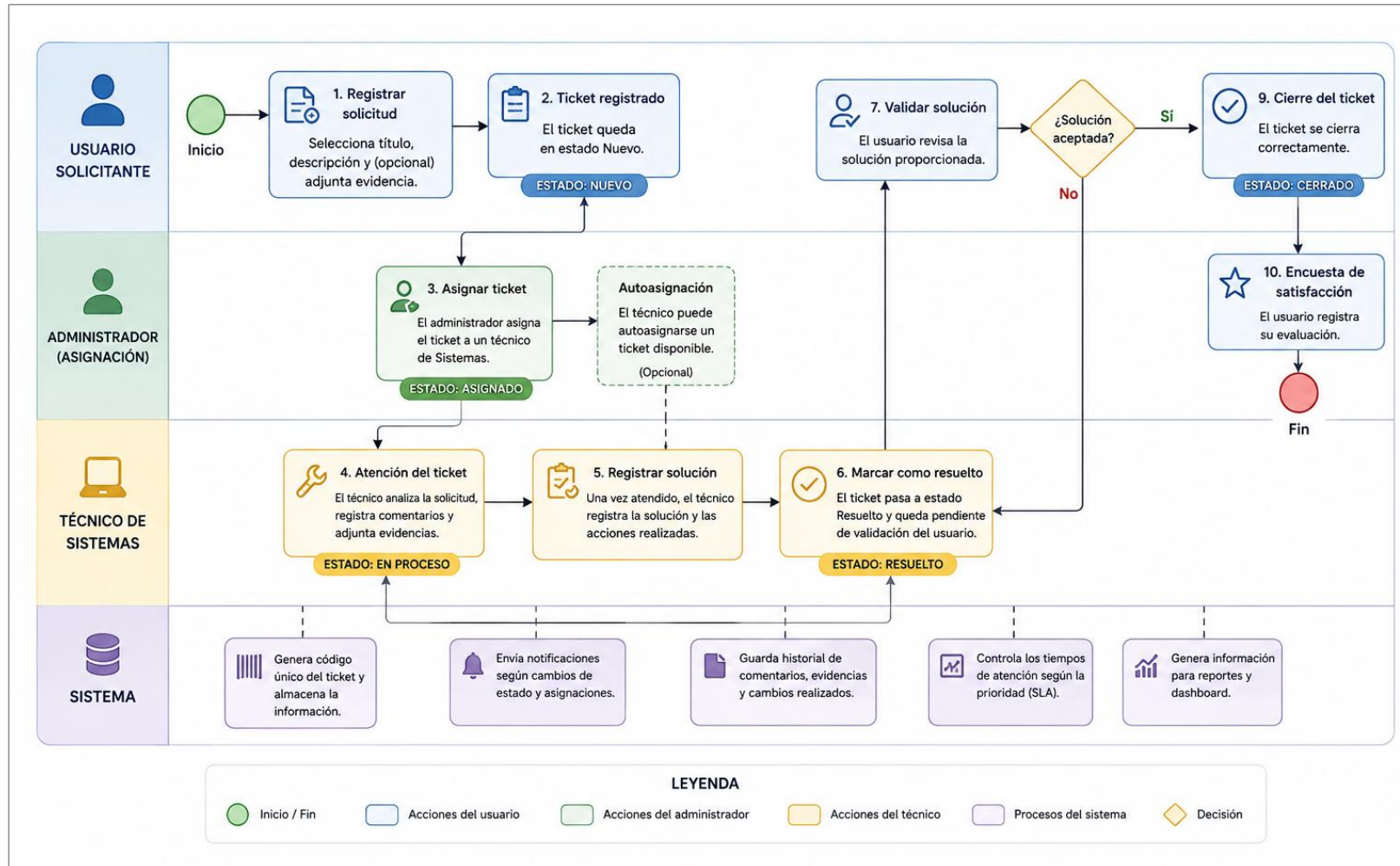
El flujo comienza cuando el usuario solicitante selecciona el título correspondiente, completa la descripción de la incidencia y, de ser necesario, adjunta una evidencia. Una vez registrado el ticket, este adquiere el estado Nuevo y queda disponible para su atención.

Posteriormente, el administrador puede asignar el ticket a un técnico de Sistemas, o el propio técnico puede realizar la autoasignación de un ticket disponible. Una vez asignado, el ticket pasa al estado Asignado y posteriormente a En proceso, durante el cual el técnico realiza las actividades necesarias para atender la solicitud, pudiendo registrar comentarios y adjuntar evidencias relacionadas con la atención.

Una vez finalizadas las actividades de atención, el técnico registra la solución y el ticket pasa al estado Resuelto. El usuario solicitante valida la solución proporcionada y, cuando la atención ha sido aceptada, se procede con el cierre del ticket. Finalmente, se puede registrar la encuesta de satisfacción correspondiente a la atención recibida.

Figura 14

Flujo de atención y gestión de tickets del sistema web.



Nota. Elaboración propia.

3.5. Desarrollo e implementación

3.5.1. Desarrollo del sistema

El desarrollo del sistema se realizó de manera individual, bajo la supervisión de la jefatura del área de Sistemas, tomando como referencia los requerimientos identificados durante el análisis del proceso de atención de incidencias. La implementación se llevó a cabo inicialmente en un entorno local, donde se desarrollaron y probaron progresivamente las funcionalidades de la solución.

Como resultado de este proceso se generaron dos versiones de la solución. La primera versión permitió implementar el registro y gestión básica de tickets, mientras que la segunda incorporó mejoras orientadas a simplificar el registro de solicitudes mediante el uso de títulos predefinidos asociados a una configuración de atención.

En la segunda versión, el formulario de registro fue simplificado para que el usuario únicamente seleccione el título correspondiente, ingrese la descripción de la solicitud y adjunte una evidencia cuando sea necesario. La información asociada al título permite utilizar la configuración previamente establecida para la clasificación, prioridad y SLA del ticket, reduciendo la cantidad de datos que el usuario debe ingresar manualmente.

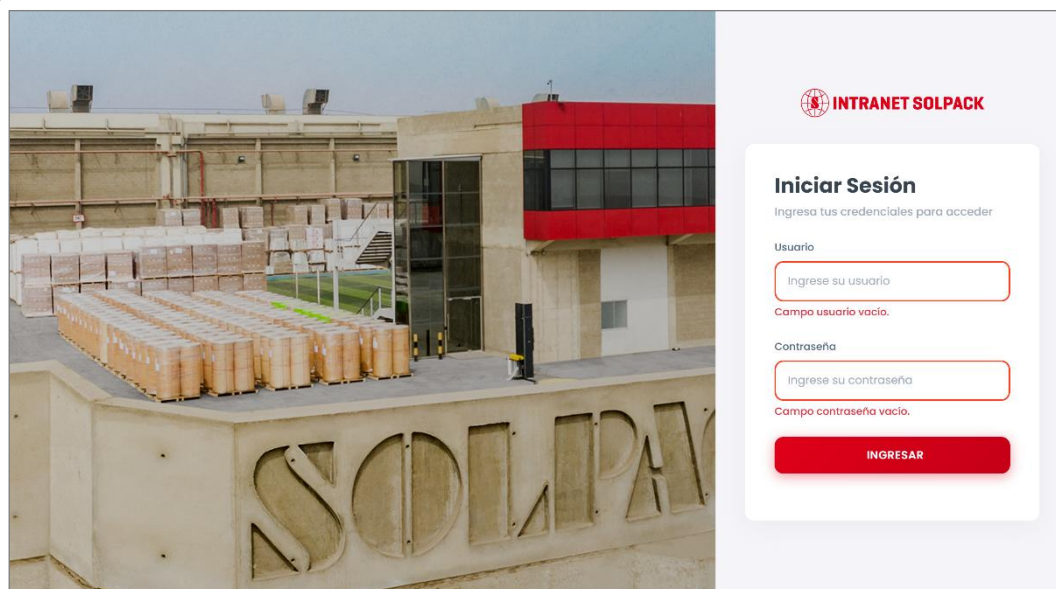
3.5.2. Implementación del módulo de usuarios y accesos

El módulo de usuarios y accesos fue implementado para administrar las cuentas que utilizan el sistema y controlar las funcionalidades disponibles de acuerdo con el rol asignado. Para ello, se estableció una estructura de autenticación y autorización que permite identificar al usuario que accede a la aplicación y restringir el acceso a los módulos según los permisos definidos.

El acceso al sistema se realiza mediante un formulario de autenticación, en el cual el usuario proporciona sus credenciales. Una vez validadas, el sistema inicia la sesión correspondiente y determina las funcionalidades disponibles de acuerdo con el perfil asignado.

Figura 15

Interfaz de autenticación del sistema web

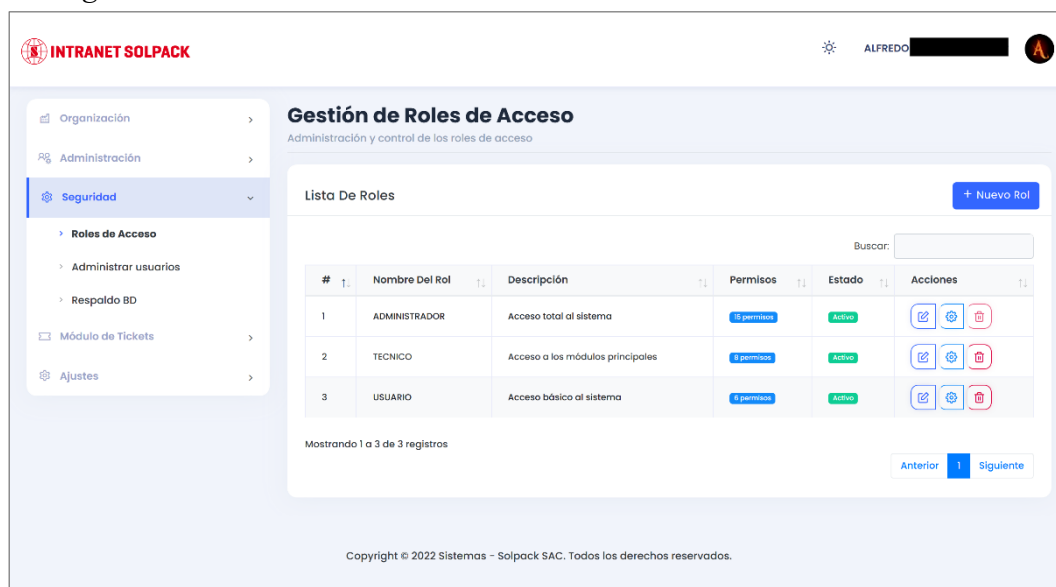


Nota. Elaboración propia.

Posteriormente, el administrador puede gestionar los usuarios y sus respectivos roles y permisos. Esta configuración permite establecer diferentes niveles de acceso dentro de la aplicación, de manera que cada usuario pueda utilizar únicamente las funcionalidades correspondientes a su perfil.

Figura 16

Modulo de gestión de roles de acceso



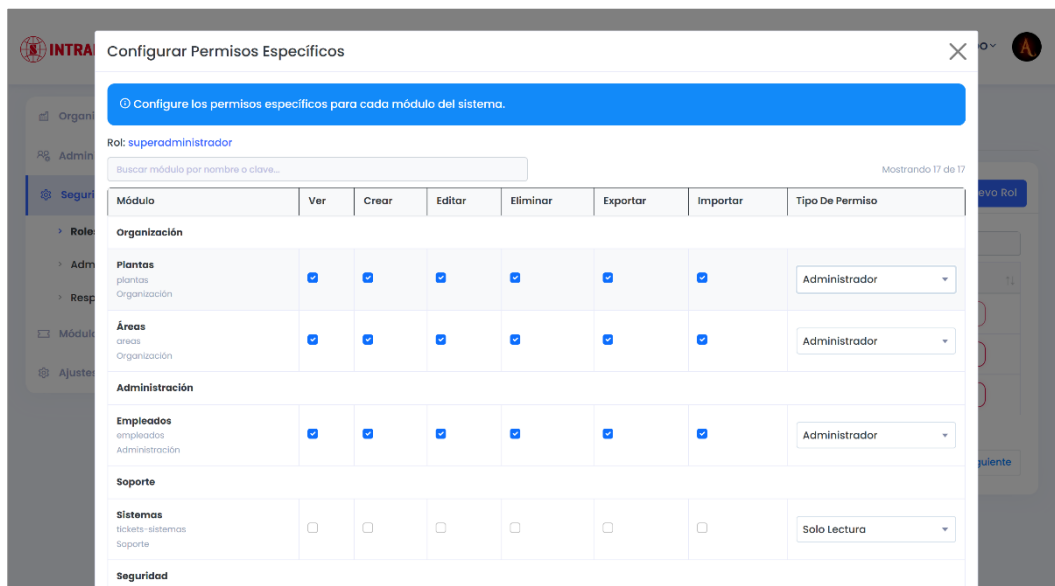
Nota. Elaboración propia.

En relación con la solución desarrollada, el control de acceso se encuentra vinculado con los módulos disponibles en el sistema. De esta manera, la autenticación permite identificar

al usuario y la autorización determina las operaciones que puede realizar, contribuyendo al control de acceso a la información y a las funcionalidades de la aplicación.

Figura 17

Configuración de acceso a los módulos



Nota. Elaboración propia.

3.5.3. Implementación del módulo de gestión de tickets

El módulo de gestión de tickets constituye el componente principal del sistema, debido a que permite registrar y realizar el seguimiento de las incidencias y requerimientos reportados por los usuarios. Su implementación contempla el registro de la solicitud, asignación del responsable, atención, actualización del estado, incorporación de comentarios y evidencias, resolución y cierre del ticket.

Para el registro de una solicitud, el usuario accede al formulario correspondiente y selecciona un título predefinido, ingresa la descripción de la incidencia y, cuando es necesario, adjunta una evidencia. La información asociada al título permite utilizar la configuración previamente establecida para la clasificación, prioridad y SLA, reduciendo la cantidad de datos que el usuario debe ingresar manualmente.

Figura 18
Formulario de registro de tickets

Nota. Elaboración propia.

Una vez registrado, el ticket queda disponible para su atención con el estado correspondiente. El administrador puede asignarlo a un técnico, mientras que los técnicos también pueden autoasignarse los tickets disponibles. Durante la atención, el responsable puede actualizar el estado, registrar comentarios y adjuntar evidencias relacionadas con las actividades realizadas.

Figura 19
Detalle y seguimiento de un ticket

Nota. Elaboración propia.

El ciclo de atención contempla los estados definidos para el sistema, permitiendo identificar el avance de cada solicitud desde su registro hasta su resolución. Una vez que el técnico finaliza la atención, registra la solución y el ticket pasa al estado Resuelto. Posteriormente, el usuario solicitante valida la solución proporcionada y, cuando corresponde, se realiza el cierre del ticket.

Figura 20
Listado de los tickets

	N° Ticket	Estado	Título	Encargado	Fecha
1	S260512094532	Resuelto	Acceso - Desbloquear usuario de SAP	JHONY ALEX HUAMANCHUMO BANCES	12/05/26
2	S260512094238	Resuelto	Ventas - Cancelar entrega (Guía)	JHONY ALEX HUAMANCHUMO BANCES	12/05/26
3	S260512092651	En proceso	Ventas - Error al crear oferta de ventas	JHONY ALEX HUAMANCHUMO BANCES	12/05/26
4	S260510040236	En proceso	Acceso - Desbloquear usuario de SAP	ALFREDO [redacted]	10/05/26
5	S260510035517	Resuelto	Ventas - Cancelar entrega (Guía)	JHONY ALEX HUAMANCHUMO BANCES	10/05/26
6	S260108012207	En proceso	Ventas - Cancelar entrega (Guía)	MARGARITA [redacted]	08/01/26
7	S260107031452	En proceso	Ventas - Cancelar entrega (Guía)	MARGARITA [redacted]	07/01/26
8	S260107023809	En proceso	Ventas - Cancelar entrega (Guía)	MARGARITA [redacted]	07/01/26

Nota. Elaboración propia.

De esta manera, el módulo permite mantener la información asociada a cada solicitud durante las diferentes etapas de atención, facilitando su consulta y seguimiento por parte de los usuarios involucrados.

3.5.4. Implementación del módulo de configuración de tickets

El módulo de configuración de tickets fue implementado para establecer previamente los criterios utilizados en la clasificación y atención de las solicitudes. Para ello, el sistema permite administrar categorías, subcategorías, títulos, prioridades y SLA, de manera que la información utilizada durante el registro de un ticket pueda ser previamente configurada por el administrador.

Figura 21
Módulo de tickets - Gestión de subcategorías

Id	Subcategoría	Categoría	Estado	Fecha Creación	Acción
1	Computadoras	Equipos	Activa	2025-05-15 12:41:50	[Edit] [Delete]
2	Laptops	Equipos	Activa	2025-05-15 12:41:50	[Edit] [Delete]
3	Periféricos	Equipos	Activa	2025-05-15 12:41:50	[Edit] [Delete]
4	Impresoras	Equipos	Activa	2025-05-15 12:41:50	[Edit] [Delete]
5	Teléfonos IP	Equipos	Activa	2025-05-15 12:41:50	[Edit] [Delete]
6	Lectores QR	Equipos	Activa	2025-05-15 12:41:50	[Edit] [Delete]
7	Monitores	Equipos	Activa	2025-05-15 12:42:26	[Edit] [Delete]
8	Sap Business One	Software	Activa	2025-05-15 12:45:07	[Edit] [Delete]

Nota. Elaboración propia.

La incorporación de los títulos predefinidos permitió simplificar el formulario de registro implementado en la segunda versión del sistema. A diferencia de la primera versión, en la que el usuario debía proporcionar una mayor cantidad de información, la nueva implementación permite seleccionar un título previamente configurado, completar la descripción de la solicitud y adjuntar una evidencia cuando sea necesario.

Figura 22
Módulo de tickets – Gestión de títulos

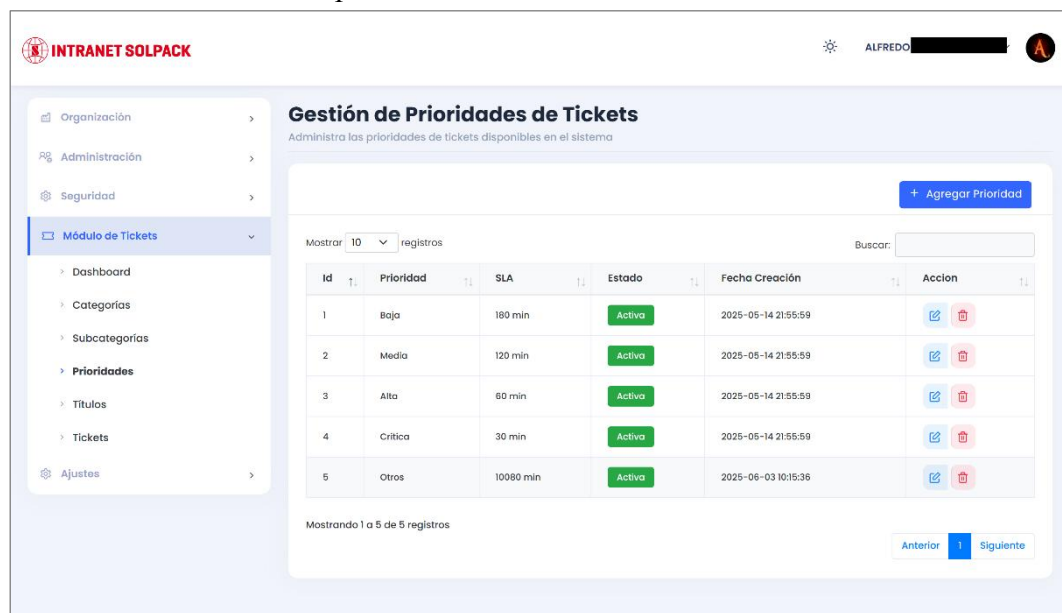
Id	Título	Subcategoría	Categoría
1	Ventas - Cancelar entrega (Guía)	Sap Business One	Software
2	Ventas - Cerrar orden de venta	Sap Business One	Software
3	Operaciones de stock - Cancelar solicitud de traslado	Sap Business One	Software
4	Compras - Cancelar entrada de mercancía	Sap Business One	Software
5	Ventas - Error al crear oferta de ventas	Sap Business One	Software
6	Datos maestros de artículo - Activar - Desactivar SKU para venta	Sap Business One	Software
7	Datos maestros de artículo - Anexar arte en SAP	Sap Business One	Software
8	Datos maestros socio de negocio - Actualizar propietario - Empleado del dpto de ventas	Sap Business One	Software
9	VPN - Configurar acceso remoto	Acceso remoto	Internet
10	APP - Configurar acceso remoto Anydesk - Teamviewer	Acceso remoto	Internet

Nota. Elaboración propia.

A partir del título seleccionado, el sistema utiliza la configuración asociada para establecer los criterios correspondientes de clasificación, prioridad y SLA. De esta manera, se reduce la intervención del usuario en la configuración del ticket y se busca mantener criterios uniformes para la atención de las solicitudes.

Figura 23

Módulo de tickets - Gestión de prioridades



Id	Prioridad	SLA	Estado	Fecha Creación	Accion
1	Baja	180 min	Activa	2025-05-14 21:55:59	[Icono]
2	Media	120 min	Activa	2025-05-14 21:55:59	[Icono]
3	Alta	60 min	Activa	2025-05-14 21:55:59	[Icono]
4	Crítica	30 min	Activa	2025-05-14 21:55:59	[Icono]
5	Otros	10080 min	Activa	2025-06-03 10:15:36	[Icono]

Nota. Elaboración propia.

La configuración implementada también permite realizar modificaciones cuando los criterios de atención requieren ser actualizados, sin necesidad de modificar directamente el formulario utilizado por los usuarios. Esto proporciona mayor flexibilidad para adaptar el sistema a las necesidades operativas del área de Sistemas.

3.5.5. Despliegue del sistema

Para la puesta en funcionamiento del sistema se habilitó un servidor con Windows Server, en el cual se configuró el entorno necesario para ejecutar la aplicación web y gestionar la información almacenada. Como parte de la configuración del servidor se instaló XAMPP, utilizado para proporcionar los servicios requeridos por la aplicación, incluyendo PHP y MySQL.

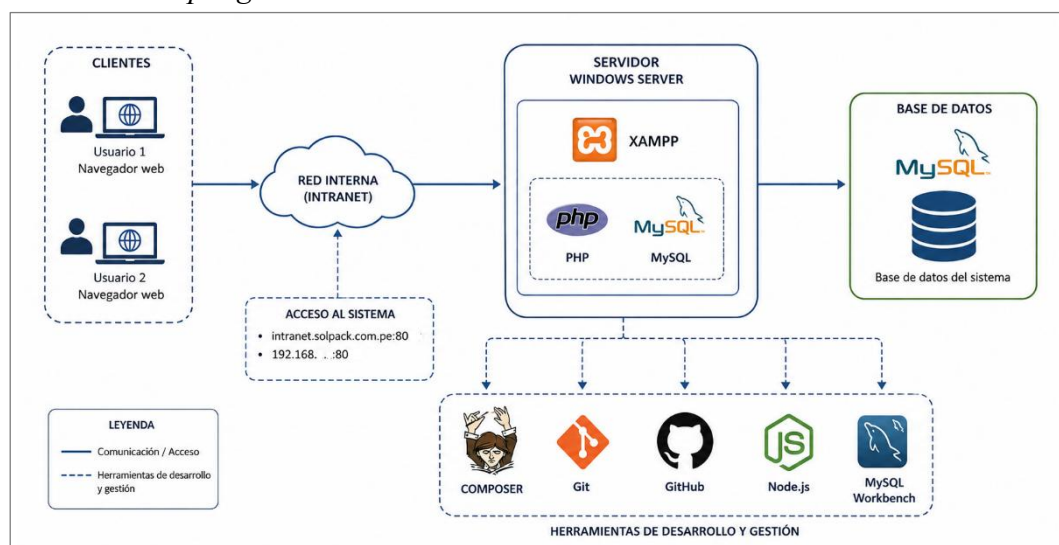
Asimismo, se incorporaron herramientas complementarias para la administración y desarrollo de la solución. Se utilizó MySQL Workbench para la gestión y administración de la base de datos, mientras que Composer permitió gestionar las dependencias del proyecto. Para

el control y gestión del código fuente se emplearon Git y GitHub, y Node.js fue incorporado como parte del entorno utilizado durante el desarrollo de la aplicación.

Una vez configurado el entorno, la aplicación fue integrada en el servidor para permitir el acceso de los usuarios mediante la red interna de la organización. El sistema puede ser accedido desde un navegador web utilizando la dirección configurada para la intranet, mediante el dominio interno correspondiente o directamente a través de la dirección IP del servidor.

Figura 24

Arquitectura de despliegue del sistema web



Nota: Elaboración propia.

CONCLUSIONES

Se implementó el sistema web *Intranet Solpack*, orientado a centralizar la gestión de incidencias y solicitudes de los usuarios mediante un flujo estructurado de registro, asignación, atención, resolución y cierre de tickets. La solución permitió disponer de un canal centralizado para gestionar las solicitudes dirigidas al área de Sistemas.

El sistema permite registrar y realizar el seguimiento de las incidencias durante su ciclo de atención, almacenando información relacionada con el usuario solicitante, título, responsable, estado, descripción, evidencias y fechas de registro, actualización y cierre. Asimismo, la utilización de estados permite identificar la situación de cada ticket durante el proceso de atención.

La configuración de categorías, subcategorías, títulos, prioridades y SLA permitió establecer criterios previamente definidos para la gestión de las solicitudes. En particular, la incorporación de títulos predefinidos en el formulario de registro simplificó la información que debe proporcionar el usuario, quien actualmente selecciona el título correspondiente, registra la descripción y adjunta una evidencia cuando es necesario.

La implementación de roles y permisos permitió establecer diferentes niveles de acceso al sistema, de acuerdo con las funciones de los usuarios involucrados. De esta manera, el usuario solicitante puede consultar sus tickets, mientras que los responsables de atención gestionan los tickets disponibles y asignados, y el administrador dispone de funciones de gestión y configuración.

La solución fue desplegada en un servidor Windows Server dentro de la red interna de la organización, integrando el entorno de ejecución de PHP y MySQL mediante XAMPP y permitiendo el acceso al sistema a través de la intranet institucional. Asimismo, se incorporaron herramientas de apoyo para la gestión de dependencias, control de versiones y administración de la base de datos.

RECOMENDACIONES

Se recomienda establecer un programa periódico de capacitación para los usuarios del sistema web *Intranet Solpack*, especialmente ante la incorporación de nuevos usuarios o la implementación de nuevas funcionalidades. Esto permitirá fortalecer el conocimiento sobre el uso adecuado del sistema y aprovechar las funcionalidades disponibles para el registro y seguimiento de las solicitudes.

Se recomienda realizar actividades periódicas de mantenimiento y supervisión del sistema, considerando la actualización de sus componentes, revisión de los mecanismos de seguridad y generación de copias de respaldo de la base de datos. Estas acciones permitirán preservar la disponibilidad e integridad de la información registrada y reducir los riesgos asociados a posibles fallos del sistema.

Se recomienda utilizar periódicamente la información generada por los reportes y el dashboard para analizar el comportamiento de las solicitudes, identificando los tipos de incidencias más frecuentes, los tiempos de atención y otros aspectos relevantes de la gestión. Esta información puede servir como apoyo para identificar oportunidades de mejora en el servicio brindado por el área de Sistemas.

Se recomienda mantener un mecanismo de retroalimentación con los usuarios del sistema, con el propósito de identificar necesidades, dificultades y nuevas funcionalidades que puedan incorporarse en futuras versiones. La información obtenida permitirá orientar las mejoras del sistema de acuerdo con las necesidades operativas de la organización.

Se recomienda considerar la escalabilidad de la solución para futuras ampliaciones, de manera que el sistema pueda incorporar nuevos módulos, funcionalidades o integraciones conforme evolucionen las necesidades de Solpack S.A.C. Esto permitirá que la solución pueda continuar adaptándose al crecimiento de la organización.

REFERENCIAS

- Anderson. (2010). Kanban Successful Evolutionary Change for Your Technology Business. Blue Hole Press, Washington DC. - References - Scientific Research Publishing. <https://www.scirp.org/reference/referencespapers?referenceid=2444636>
- Beck, Beedle, Bennekum, Cockburn, Cunningham, Fowler, & Grenning. (2021). Manifesto for Agile Software Development. <https://agilemanifesto.org/>
- Bootstrap. (n.d.). Get started with Bootstrap v5.3. Retrieved September 1, 2026, from <https://getbootstrap.com/docs/5.3/getting-started/introduction/>
- Ghazizadeh, E., Jeong, M., Angel, J. P., Que, Y., Ghazizadeh, E., Jeong, M., Angel, J. P., & Que, Y. (2019). IT Service Management and Incident Management: Literature Review and a Case Study. ACIS 2019 Proceedings. <https://aisel.aisnet.org/acis2019/31>
- Greenlaw, R., & Hepp, E. M. (2003). Internet, Overview. Encyclopedia of Information Systems, 667–681. <https://doi.org/10.1016/B0-12-227240-4/00096-4>
- Handoko, D., & Girsang, A. S. (2018). Service Desk Implementation with Information Technology Infrastructure Library Framework (Study Case Financial Company). Journal of Physics: Conference Series, 1090(1), 012059. <https://doi.org/10.1088/1742-6596/1090/1/012059>
- Jäntti, M., Shrestha, A., & Cater-Steel, A. (2012). Towards an Improved IT Service Desk System and Processes: A Case Study. International Journal on Advances in Systems and Measurements, 5. http://www.iariajournals.org/systems_and_measurements/2012,
- Krasner, G. E., & Pope, S. T. (1988). A Cookbook for Using View-Controller User the Model-Interface Paradigm in Smalltalk-80.
- MDN, C. (1998). La web y los estándares web - Aprende desarrollo web | MDN. https://developer.mozilla.org/es/docs/Learn_web_development/Getting_started/Web_standards/The_web_standards_model
- MDN Contributors. (2025). Ajax - Glossary | MDN. <https://developer.mozilla.org/en-US/docs/Glossary/AJAX>
- MDN Contributors. (2025). Trabajando con JSON - Aprende desarrollo web | MDN. https://developer.mozilla.org/es/docs/Learn_web_development/Core/Scripting/JSON
- Molay, Bruce. (2003). Understanding Unix/Linux programming. 530. <https://www.pearson.com/en-us/subject-catalog/p/understanding-unixlinux-programming-a-guide-to-theory-and-practice/P200000003280/9780130083968>

- Oracle. (2021). What Is a Relational Database | Oracle. <https://www.oracle.com/database/what-is-a-relational-database/>
- Oracle. (2026). MySQL:: MySQL 8.4 Reference Manual:: 1 General Information. <https://dev.mysql.com/doc/refman/8.4/en/introduction.html>
- People. (2001). ITIL 4 Practitioner: Incident Management | peoplecert.org. <https://www.peoplecert.org/browse-certifications/it-governance-and-service-management/ITIL-1/itil4-practices-incident-management-3684>
- PHP Documentation Group. (2026). PHP: Introducción - Manual. <https://www.php.net/manual/es/introduction.php>
- Robert, E. (2007). Java SE Application Design With MVC. <https://www.oracle.com/technical-resources/articles/javase/mvc.html>
- Salah, S., Maciá-Fernández, G., Díaz-Verdejo, J. E., & Sánchez-Casado, L. (2015). A Model for Incident Tickets Correlation in Network Management. *Journal of Network and Systems Management* 2015 24:1, 24(1), 57–91. <https://doi.org/10.1007/S10922-014-9340-6>
- Schwaber, & Sutherland. (2020). Scrum Guide | Scrum Guides. <https://scrumguides.org/scrum-guide.html>
- Valente, M. Tulio. (2024). Software Engineering: A Modern Approach. In softengbook: 1.3.2 (Number 1). Self-published. <https://softengbook.org>

ANEXOS

Anexo 01: Código SQL de la base de datos

Código de la base de datos de las tablas principales del sistema

```
1  -- =====
2  -- BASE DE DATOS DEL SISTEMA WEB DE GESTIÓN DE INCIDENCIAS
3  -- SOLPACK S.A.C.
4  -- =====
5
6  • CREATE DATABASE IF NOT EXISTS solpack_intranet;
7  • USE solpack_intranet;
8
9
10 -- =====
11 -- 1. USUARIOS Y CONTROL DE ACCESO
12 -- =====
13
14 • CREATE TABLE roles (
15     id_rol INT AUTO_INCREMENT PRIMARY KEY,
16     nombre VARCHAR(100) NOT NULL,
17     descripcion VARCHAR(255)
18 );
19
20 • CREATE TABLE empleados (
21     id_empleado INT AUTO_INCREMENT PRIMARY KEY,
22     nombres VARCHAR(100) NOT NULL,
23     apellidos VARCHAR(100) NOT NULL,
24     correo VARCHAR(150),
25     id_rol INT NOT NULL,
26     estado TINYINT(1) DEFAULT 1,
27
28     CONSTRAINT fk_empleado_rol
29         FOREIGN KEY (id_rol)
30         REFERENCES roles(id_rol)
31 );
32
33 • CREATE TABLE auth_cuentas (
34     id_cuenta INT AUTO_INCREMENT PRIMARY KEY,
35     id_empleado INT NOT NULL,
36     usuario VARCHAR(100) NOT NULL UNIQUE,
37     password VARCHAR(255) NOT NULL,
38     estado TINYINT(1) DEFAULT 1,
39
40     CONSTRAINT fk_cuenta_empleado
41         FOREIGN KEY (id_empleado)
42         REFERENCES empleados(id_empleado)
43 );
```

```

46      -- =====
47      -- 2. CATEGORÍAS Y SUBCATEGORÍAS
48      -- =====
49
50      ● CREATE TABLE ticket_categoria (
51          id_categoria INT AUTO_INCREMENT PRIMARY KEY,
52          nombre VARCHAR(100) NOT NULL,
53          descripcion VARCHAR(255),
54          estado TINYINT(1) DEFAULT 1
55      );
56
57      ● CREATE TABLE ticket_subcategoria (
58          id_subcategoria INT AUTO_INCREMENT PRIMARY KEY,
59          id_categoria INT NOT NULL,
60          nombre VARCHAR(100) NOT NULL,
61          descripcion VARCHAR(255),
62          estado TINYINT(1) DEFAULT 1,
63
64          CONSTRAINT fk_subcategoria_categoria
65              FOREIGN KEY (id_categoria)
66              REFERENCES ticket_categoria(id_categoria)
67      );
68
69      -- =====
70      -- 3. PRIORIDADES
71      -- =====
72
73
74      ● CREATE TABLE ticket_prioridad (
75          id_prioridad INT AUTO_INCREMENT PRIMARY KEY,
76          nombre VARCHAR(50) NOT NULL,
77          descripcion VARCHAR(255),
78          tiempo_sla INT,
79          estado TINYINT(1) DEFAULT 1
80      );

```

```

83  -- =====
84  -- 4. TÍTULOS PREDEFINIDOS
85  -- =====
86
87  ● ○ CREATE TABLE ticket_titulos (
88      id_tit INT AUTO_INCREMENT PRIMARY KEY,
89      titulo VARCHAR(200) NOT NULL,
90      id_categoria INT,
91      id_subcategoria INT,
92      id_prioridad INT,
93      sla INT,
94      estado TINYINT(1) DEFAULT 1,
95
96      CONSTRAINT fk_titulo_categoria
97          FOREIGN KEY (id_categoria)
98          REFERENCES ticket_categoria(id_categoria),
99
100     CONSTRAINT fk_titulo_subcategoria
101         FOREIGN KEY (id_subcategoria)
102         REFERENCES ticket_subcategoria(id_subcategoria),
103
104     CONSTRAINT fk_titulo_prioridad
105         FOREIGN KEY (id_prioridad)
106         REFERENCES ticket_prioridad(id_prioridad)
107 );
108
109  -- =====
110  -- 5. ESTADOS DE LOS TICKETS
111  -- =====
112
113
114  ● ○ CREATE TABLE ticket_estado (
115      id_estado INT AUTO_INCREMENT PRIMARY KEY,
116      nombre VARCHAR(50) NOT NULL,
117      descripcion VARCHAR(255)
118  );

```

```

121  -- =====
122  -- 6. TICKETS
123  -- =====
124
125  ● CREATE TABLE tickets (
126      id_ticket INT AUTO_INCREMENT PRIMARY KEY,
127      codigo_ticket VARCHAR(30) NOT NULL UNIQUE,
128      id_tit INT NOT NULL,
129      descripcion TEXT NOT NULL,
130      adjunto VARCHAR(255),
131      id_estado INT NOT NULL,
132      id_usuario INT NOT NULL,
133      id_responsable INT,
134      fecha_ticket DATETIME DEFAULT CURRENT_TIMESTAMP,
135      fecha_update DATETIME DEFAULT CURRENT_TIMESTAMP
136          ON UPDATE CURRENT_TIMESTAMP,
137      fecha_cierre DATETIME NULL,
138      CONSTRAINT fk_ticket_titulo
139          FOREIGN KEY (id_tit)
140          REFERENCES ticket_titulos(id_tit),
141      CONSTRAINT fk_ticket_estado
142          FOREIGN KEY (id_estado)
143          REFERENCES ticket_estado(id_estado),
144      CONSTRAINT fk_ticket_usuario
145          FOREIGN KEY (id_usuario)
146          REFERENCES empleados(id_empleado),
147      CONSTRAINT fk_ticket_responsable
148          FOREIGN KEY (id_responsable)
149          REFERENCES empleados(id_empleado)
150  );
151
152  -- =====
153  -- 7. DETALLE Y SEGUIMIENTO DEL TICKET
154  -- =====
155
156
157  ● CREATE TABLE ticket_detalle (
158      id_detalle INT AUTO_INCREMENT PRIMARY KEY,
159      id_ticket INT NOT NULL,
160      id_usuario INT NOT NULL,
161      comentario TEXT,
162      adjunto VARCHAR(255),
163      fecha_registro DATETIME DEFAULT CURRENT_TIMESTAMP,
164
165      CONSTRAINT fk_detalle_ticket
166          FOREIGN KEY (id_ticket)
167          REFERENCES tickets(id_ticket),
168
169      CONSTRAINT fk_detalle_usuario
170          FOREIGN KEY (id_usuario)
171          REFERENCES empleados(id_empleado)
172  );

```

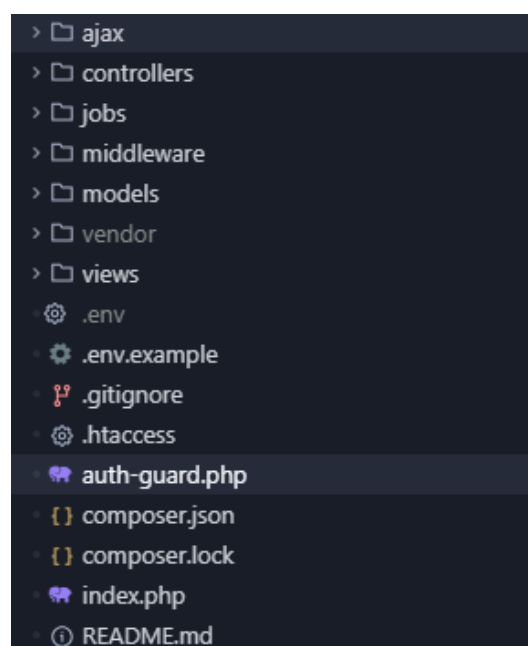
```

175      -- =====
176      -- 8. ESCALAMIENTO
177      -- =====
178
179      CREATE TABLE ticket_escalacion (
180          id_escalacion INT AUTO_INCREMENT PRIMARY KEY,
181          id_ticket INT NOT NULL,
182          id_usuario_origen INT,
183          id_usuario_destino INT,
184          motivo TEXT,
185          fecha_escalacion DATETIME DEFAULT CURRENT_TIMESTAMP,
186
187          CONSTRAINT fk_escalacion_ticket
188              FOREIGN KEY (id_ticket)
189              REFERENCES tickets(id_ticket),
190
191          CONSTRAINT fk_escalacion_origen
192              FOREIGN KEY (id_usuario_origen)
193              REFERENCES empleados(id_empleado),
194
195          CONSTRAINT fk_escalacion_destino
196              FOREIGN KEY (id_usuario_destino)
197              REFERENCES empleados(id_empleado)
198      );

```

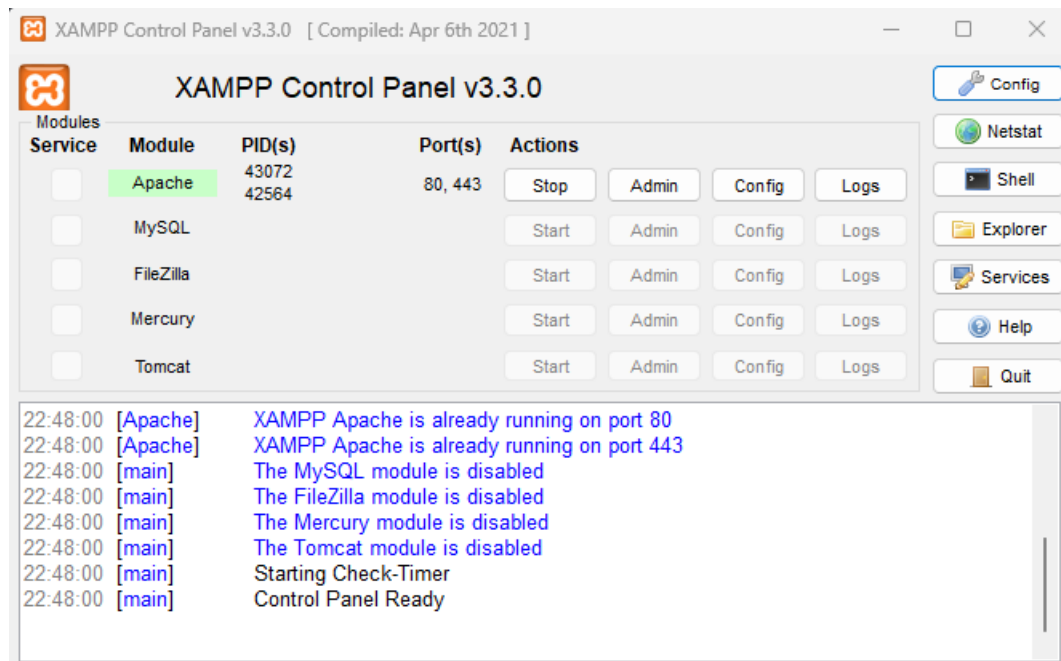
Anexo 02: Estructura del proyecto

En el desarrollo del sistema web se utilizó el patrón de arquitectura Modelo-Vista-Controlador (MVC) para organizar los componentes de la aplicación.

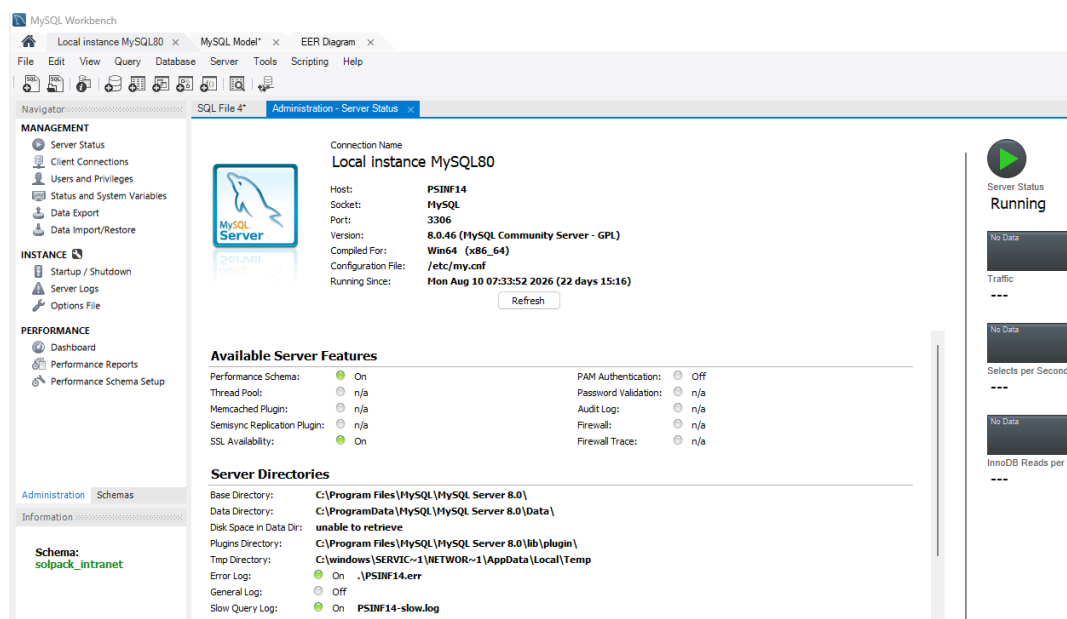


Anexo 03: Configuración del entorno de desarrollo

Configuración de XAMPP para el acceso web



Ejecución de MySQL para el acceso a los datos

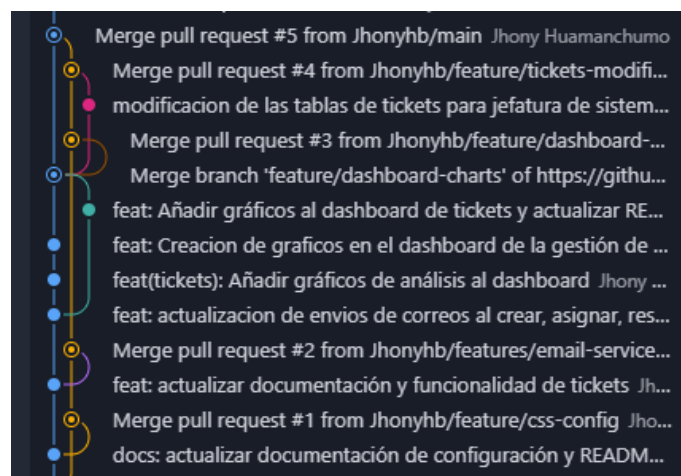


Configuración de la conexión de la base de datos con el sistema web.

```
.env.example x
.env.example

1 # =====
2 # Configuración de la Aplicación
3 # =====
4 APP_NAME="Intranet V2"
5 APP_ENV=local
6 APP_DEBUG=true
7 APP_URL=http://localhost
8 APP_KEY=base64:your_app_key_here
9
10 # =====
11 # Configuración de Base de Datos
12 # =====
13 DB_CONNECTION=mysql
14 DB_HOST=127.0.0.1
15 DB_PORT=3306
16 DB_DATABASE=mibase
17 DB_USERNAME=root
18 DB_PASSWORD=
19 DB_CHARSET=utf8mb4
20 DB_COLLATION=utf8mb4_unicode_ci
21
```

Parte del control de versiones del código del proyecto con Git y GitHub.



Anexo 04: Dashboard Administrativo.

Dashboard del administrador para el monitoreo y toma de decisión respecto a la atención de las incidencias.

