

# UNIVERSIDAD LUTERANA SALVADOREÑA

FACULTAD DE CIENCIAS DEL HOMBRE Y LA NATURALEZA

LICENCIATURA EN CIENCIAS DE LA COMPUTACION



## PROYECTO:

### SISTEMA DE SOPORTE TÉCNICO

#### EQUIPO DESARROLLADOR:

PARADA LÓPEZ CRISTIAN ALEXANDER

AMAYA MEJÍA LUIS FERNANDO

LÓPEZ MARTÍNEZ BERTA EUNICE

GARAY GOCHEZ TANIA LISBETH

DE LEÓN HERNÁNDEZ KATHERINE ALEXANDRA

MAGAÑA RODRÍGUEZ JONATHAN ESAÚ

CORTEZ RIVERA EDWIN SARBELIO

VIGIL MARTÍNEZ JOSÉ RODOLFO

#### CARNET

PL24273

MA24353

LM24157

GG24326

DH24411

MR24207

CR24275

VM01136974

**CÁTEDRA:** PROGRAMACIÓN ORIENTADA A OBJETOS

**HORARIO:** SÁBADOS 10 AM A 1:20PM

**FACILITADOR:** LIC. RIGOBERTO ISRAEL ORELLANA

**SAN SALVADOR, 30 de septiembre del 2024**

# ÍNDICE

1. INTRODUCCIÓN:	1
2. ANÁLISIS DEL PROBLEMA	2
2.1 TEMA	2
2.2 PROBLEMÁTICA A RESOLVER	2
3. OBJETIVOS	3
3.1 OBJETIVO GENERAL	3
3.2 OBJETIVOS ESPECÍFICOS	3
4. BENEFICIOS DEL PROYECTO	4
5. JUSTIFICACIÓN	5
6. METODOLOGÍA Y CICLO DE VIDA UTILIZADOS	6
6.1 Metodología de recopilación de información	6
6.2 MODELO DE CICLO DE VIDA	7
6.3 REQUERIMIENTOS	8
Requerimientos Funcionales	8
Requisitos de Datos	9
Requisitos de Interfaz	9
Requisitos de Navegación	9
6.3.1 TABLAS DE REQUERIMIENTOS	10
7. ANÁLISIS	20
MODELO DE ANÁLISIS SELECCIONADO:	20
Diagramas UML:	20
8. DISEÑO	23
9. DESARROLLO	25
9.1 TÉCNICAS DE PROGRAMACIÓN APLICADAS	25
9.2 HERRAMIENTAS DE DESARROLLO UTILIZADAS	26
9.3 LENGUAJE DE PROGRAMACIÓN Y GESTOR DE BASES DE DATOS	26
9. DOCUMENTACIÓN	27
9.1 TÉCNICAS DE PRUEBA DEL SISTEMA	27
10. CONCLUSIONES	28
11. RECOMENDACIONES	29
12. BIBLIOGRAFÍAS	30
13. ANEXOS	31
13.1. DIAGRAMA DE LA BASE DE DATOS	31
13.2 FOTOGRAFÍAS DEL DESARROLLO	31
13.3 MANUAL DE PROGRAMADOR Y MANUAL DE USUARIO:	32

## TABLA DE ILUSTRACIONES

Ilustración 1. Modelo de Cascada .....	7
Ilustración 2. Diagrama casos de uso.....	20
Ilustración 3. Diagrama de clases.....	21
Ilustración 4. Diagrama de secuencia.....	22
Ilustración 5. Diagrama de clases (UML).....	23
Ilustración 6. Carpeta de modelo.....	25
Ilustración 7. Carpeta de Vista .....	25
Ilustración 8. Carpeta de controlador.....	25
Ilustración 9. VScode .....	26
Ilustración 10. XAMPP .....	26
Ilustración 11. Lenguaje de programación.....	26
Ilustración 12. Gestor base de datos .....	26
Ilustración 13. PHPUnit .....	27
Ilustración 14. Prueba de sistema .....	27
Ilustración 15. Prueba con usuario .....	27
Ilustración 16. Diagrama de base de datos .....	31
Ilustración 17. Reunión 3.....	31
Ilustración 18. Reunión 2.....	31
Ilustración 19. Reunión 1.....	31

## TABLAS

Tabla 1. REQ1 .....	10
Tabla 2. REQ2 .....	10
Tabla 3. REQ3 .....	11
Tabla 4. REQ4 .....	11
Tabla 5. REQ5 .....	12
Tabla 6. REQ 6 .....	12
Tabla 7. REQ 7 .....	13
Tabla 8. REQ 8 .....	13
Tabla 9. REQ 9 .....	14
Tabla 10. REQ 10 .....	14
Tabla 11. REQ 11 .....	15
Tabla 12. REQ 12 .....	15
Tabla 13. REQ 13 .....	16
Tabla 14. REQ 14 .....	16
Tabla 15. REQ 15 .....	17
Tabla 16. REQ 16 .....	17
Tabla 17. REQ 17 .....	18
Tabla 18. REQ 18 .....	18
Tabla 19. REQ 19 .....	19

# 1. INTRODUCCIÓN:

El presente documento describe el desarrollo de un Sistema de Soporte Técnico (SST), diseñado para gestionar y optimizar los procesos relacionados con la administración de incidencias técnicas en diversas ubicaciones. Este sistema está dirigido tanto a administradores como a operadores, quienes desempeñan roles específicos en el manejo de las funciones del sistema, asegurando así una mejor coordinación y resolución de problemas.

La finalidad de este documento es proporcionar una visión integral sobre el análisis, diseño, implementación y beneficios de dicho sistema, detallando cómo resuelve problemas técnicos mediante una estructura eficiente basada en el paradigma de programación orientada a objetos (POO).

Está destinado principalmente a desarrolladores, usuarios finales, y responsables de la gestión técnica en empresas que requieran soluciones para la gestión de problemas técnicos en ubicaciones específicas.

## **2. ANÁLISIS DEL PROBLEMA**

### **2.1 TEMA**

**Sistema de Soporte Técnico para la Gestión de Incidencias.**

### **2.2 PROBLEMÁTICA A RESOLVER**

En la gestión de soporte técnico, es común enfrentar problemas relacionados con la falta de organización y seguimiento de incidencias técnicas. Esto genera retrasos en la resolución de problemas, dificultad para ubicar reportes anteriores y pérdida de tiempo para los técnicos y operadores.

El Sistema de Soporte Técnico (SST) busca resolver estas dificultades mediante una herramienta centralizada, donde los usuarios puedan crear y gestionar tickets de soporte, especificando el detalle del problema y su ubicación. Además, permite administrar usuarios y ubicaciones de manera eficiente, asegurando que los roles y accesos estén claramente delimitados según el perfil del usuario.

El sistema cubre las siguientes necesidades:

- Registro eficiente de tickets de soporte.
- Gestión de ubicaciones y usuarios con opciones de búsqueda y filtrado.
- Diferenciación de roles entre administradores y operadores para mayor control.
- Seguimiento del estado de los tickets en proceso o finalizado.

## **3. OBJETIVOS**

### **3.1 OBJETIVO GENERAL**

Desarrollar un Sistema de Soporte Técnico (SST) que facilite la gestión de incidencias, ubicaciones y usuarios mediante una estructura modular y eficiente basada en PHP y MySQL.

### **3.2 OBJETIVOS ESPECÍFICOS**

1. Implementar un módulo de soporte técnico para registrar, editar, eliminar y visualizar tickets de problemas.
2. Crear un módulo para la gestión de usuarios, permitiendo la administración de accesos y roles (administrador y operador).
3. Incorporar un módulo para la gestión de ubicaciones que facilite el registro, edición y búsqueda de ubicaciones técnicas.
4. Proveer un sistema de búsqueda y filtrado en los módulos de usuarios y ubicaciones para optimizar las consultas.

## 4. BENEFICIOS DEL PROYECTO

El Sistema de Soporte Técnico (SST) beneficia a diferentes actores de la organización al optimizar procesos y proporcionar una herramienta intuitiva para la gestión de problemas técnicos.

### **Beneficiarios:**

- **Usuarios finales (operadores):**
  - Simplificación de la creación y gestión de tickets.
  - Seguimiento centralizado del estado de las incidencias.
- **Administradores:**
  - Control total sobre usuarios, ubicaciones y tickets.
  - Optimización de los procesos internos de soporte técnico.
- **Organización:**
  - Mejora en la calidad del servicio técnico.
  - Reducción del tiempo de respuesta a incidencias.
  - Registro histórico para análisis y toma de decisiones.

## 5. JUSTIFICACIÓN

El Sistema de Soporte Técnico (SST) es una herramienta esencial para mejorar la eficiencia operativa de las empresas que gestionan incidencias técnicas. La justificación de este sistema radica en varios factores clave que responden a la necesidad de optimizar los procesos internos, reducir tiempos de respuesta, y mejorar la calidad del servicio.

La gestión manual de incidencias y la falta de una estructura organizada puede generar confusión y retrasos. El SST ofrece una plataforma centralizada donde tanto administradores como operadores pueden registrar, visualizar y hacer seguimiento de los tickets de manera eficiente. Esto no solo mejora la rapidez en la resolución de problemas, sino que también reduce el margen de error asociado a la gestión manual de incidencias.

Un sistema que permite la administración clara de roles (administradores y operadores) facilita la asignación de responsabilidades y el seguimiento de tareas. Esto asegura que los operadores se enfoquen en sus tareas específicas, mientras que los administradores tienen una visión general del estado de las incidencias. De este modo, se mejora la coordinación entre los distintos miembros del equipo de soporte técnico y se reduce la posibilidad de errores o duplicación de esfuerzos.

El Sistema de Soporte Técnico (SST) representa una solución integral que no solo optimiza la gestión de incidencias técnicas, sino que también mejora la comunicación, reduce costos, y permite tomar decisiones informadas para mejorar la calidad del servicio. La implementación de este sistema contribuye directamente a la mejora del rendimiento operativo y a la satisfacción tanto de los usuarios como de los administradores y operadores del sistema.

## 6. METODOLOGÍA Y CICLO DE VIDA UTILIZADOS

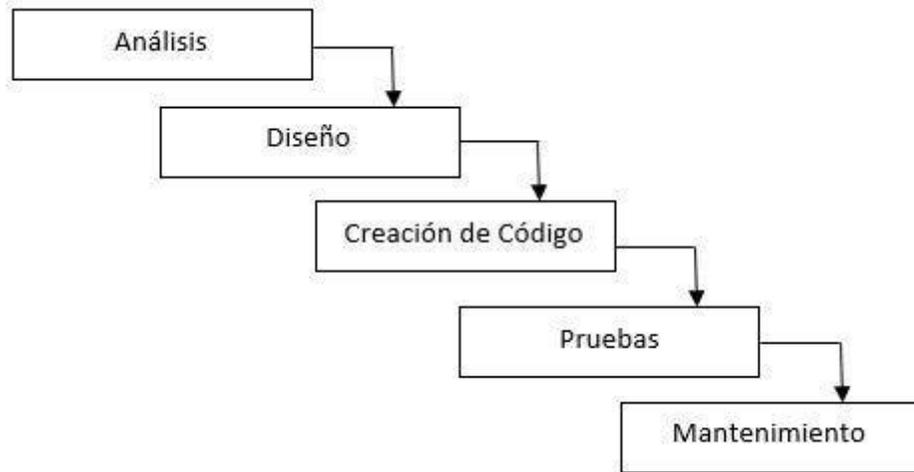
### 6.1 Metodología de recopilación de información

Para la recopilación de información se utilizaron las siguientes técnicas:

1. **Análisis:** Se realizó un análisis para incorporar distintos roles (administrador y operador) e incorporar las ubicaciones de la institución.
2. **Entrevistas:** Realizadas con administradores para definir permisos y flujos de trabajo necesarios.
3. **Revisión documental:** Estudio de sistemas similares y análisis de manuales de gestión de soporte técnico.

## 6.2 MODELO DE CICLO DE VIDA

El ciclo de vida adoptado para el desarrollo de este proyecto es el modelo Cascada, debido a que permite una planificación estructurada y secuencial en cada una de las etapas, desde el análisis de requisitos hasta la implementación y mantenimiento. Este modelo asegura la claridad en los entregables y facilita el seguimiento del proyecto.



*Ilustración 1. Modelo de Cascada*

## 6.3 REQUERIMIENTOS

### Requerimientos Funcionales

**REQ1:** El sistema deberá permitir a los operadores crear, editar, eliminar y consultar tickets de soporte técnico para reportar problemas en ubicaciones específicas.

**REQ2:** El sistema permitirá a los administradores gestionar usuarios (crear, editar, eliminar y buscar usuarios por tipo) y ubicaciones (crear, editar, eliminar y buscar ubicaciones).

**REQ3:** Los operadores solo podrán acceder al módulo de soporte técnico, sin permisos para gestionar usuarios ni ubicaciones.

**REQ4:** Los tickets de soporte deberán incluir los detalles del problema y la ubicación asignada.

**REQ5:** El sistema deberá permitir buscar ubicaciones por nombre y usuarios por tipo (administrador u operador).

### Disponibilidad:

**REQ6:** El sistema estará disponible 24/7 para usuarios con credenciales válidas, tanto administradores como operadores.

### Accesibilidad:

**REQ7:** El sistema deberá ser accesible de manera local desde un navegador compatible.

### Rendimiento:

**REQ8:** El sistema soportará la creación y consulta simultánea de tickets sin que el tiempo de respuesta supere 1 segundo por cada 100 usuarios concurrentes.

### Mantenibilidad:

**REQ9:** El sistema deberá ser diseñado para facilitar la incorporación de nuevas funcionalidades, como reportes avanzados o integración con sistemas de correo.

### Escalabilidad:

**REQ10:** El sistema permitirá agregar más módulos en el futuro, como gestión de inventarios o estadísticas de soporte técnico.

### **Requisitos de Datos**

**REQ11:** El sistema deberá almacenar los datos de usuarios, ubicaciones y tickets en una base de datos MySQL estructurada con relaciones entre tablas.

**REQ12:** Los datos de cada ticket incluirán: ID, ID del usuario asignado, ID de la ubicación, descripción del problema, fecha y estado.

**REQ13:** Los usuarios deberán tener registrados un nombre, contraseña encriptada, tipo (administrador u operador) y estado (activo/inactivo).

**REQ14:** Cada ubicación deberá incluir un nombre, ID único y estado (activa/inactiva).

### **Requisitos de Interfaz**

**REQ15:** La interfaz debe ser clara, intuitiva y en español, permitiendo una navegación sencilla tanto para operadores como administradores.

**REQ16:** El sistema debe mostrar un diseño adaptable a diferentes dispositivos (computadoras, tabletas, smartphones) para facilitar su uso.

**REQ17:** Los operadores y administradores tendrán interfaces específicas según su rol, con botones y opciones claramente diferenciadas.

### **Requisitos de Navegación**

**REQ18:** El sistema debe funcionar correctamente en los principales navegadores web (Google Chrome, Mozilla Firefox, Microsoft Edge, Safari).

**REQ19:** El diseño del sistema debe garantizar que cada módulo sea accesible con un máximo de tres clics desde la página principal.

### 6.3.1 TABLAS DE REQUERIMIENTOS

Identificación del Requisito:	Tipo de Requisito	Caso de Uso/Evento
<b>REQ1</b>	Funcional	Creación y gestión de tickets
<b>Objetivos asociados:</b>	1, 3	
<b>Justificación:</b>	El sistema debe permitir a los operadores crear, editar, eliminar y consultar tickets para reportar problemas en ubicaciones específicas, asegurando una correcta gestión de incidencias.	
<b>Condiciones/Datos específicos</b>	Los tickets deben tener detalles sobre el problema, ubicación y estado.	
<b>Criterios de Validación:</b>	Verificar que los tickets se pueden crear, editar, eliminar y consultar correctamente.	
<b>Dependencia</b>	REQ2	
<b>Analistas:</b> <b>Katherine Alexandra De León Hernández</b> <b>Tania Lisbeth Garay Góchez</b>	Proyecto: Diseño e implementación del Sistema de Soporte Técnico para la Universidad Luterana Salvadoreña 2024	

Tabla 1. REQ1

Identificación del Requisito:	Tipo de Requisito	Caso de Uso/Evento
<b>REQ2</b>	Funcional	Gestión de usuarios y ubicaciones
<b>Objetivos asociados:</b>	1, 2, 3	
<b>Justificación:</b>	El sistema permitirá a los administradores gestionar usuarios y ubicaciones, permitiendo la creación, edición, eliminación y búsqueda de ambos.	
<b>Condiciones/Datos específicos</b>	El administrador podrá crear y gestionar usuarios y ubicaciones, mientras que los operadores no tendrán acceso a esta funcionalidad.	
<b>Criterios de Validación:</b>	Verificar que los administradores puedan gestionar usuarios y ubicaciones.	
<b>Dependencia</b>	REQ3	
<b>Analistas:</b> <b>Luis Fernando Mejía Amaya</b> <b>Cristian Alexander Parada López</b>	Proyecto: Diseño e implementación del Sistema de Soporte Técnico para la Universidad Luterana Salvadoreña 2024	

Tabla 2. REQ2

Identificación del Requisito:	Tipo de Requisito	Caso de Uso/Evento
<b>REQ3</b>	Funcional	Acceso restringido a operadores
<b>Objetivos asociados:</b>	2 y 3	
<b>Justificación:</b>	Los operadores solo podrán acceder al módulo de soporte técnico, sin permisos para gestionar usuarios ni ubicaciones, garantizando un control de acceso adecuado.	
<b>Condiciones/Datos específicos</b>	Los operadores no deben poder ver ni editar usuarios ni ubicaciones.	
<b>Criterios de Validación:</b>	Verificar que los operadores solo puedan acceder al módulo de soporte.	
<b>Dependencia</b>	REQ2	
<b>Analistas:</b> <b>Edwin Sarbelio Cortez Rivera</b> <b>Berta Eunice López Martínez</b>	Proyecto: Diseño e implementación del Sistema de Soporte Técnico para la Universidad Luterana Salvadoreña 2024	

Tabla 3. REQ3

Identificación del Requisito:	Tipo de Requisito	Caso de Uso/Evento
<b>REQ4</b>	Funcional	Detalles de tickets
<b>Objetivos asociados:</b>	1	
<b>Justificación:</b>	Los tickets deben incluir detalles del problema y la ubicación para facilitar su seguimiento.	
<b>Condiciones/Datos específicos</b>	Los tickets deben mostrar todos estos datos al ser visualizados o editados.	
<b>Criterios de Validación:</b>	Verificar que los tickets muestren correctamente los detalles del problema, ubicación y estado.	
<b>Dependencia</b>	REQ1	
<b>Analistas:</b> <b>Jonathan Esaú Magaña Rodríguez</b> <b>Tania Lisbeth Garay Góchez</b>	Proyecto: Diseño e implementación del Sistema de Soporte Técnico para la Universidad Luterana Salvadoreña 2024	

Tabla 4. REQ4

Identificación del Requisito:	Tipo de Requisito	Caso de Uso/Evento
<b>REQ5</b>	Funcional	Búsqueda de usuarios y ubicaciones
<b>Objetivos asociados:</b>	4	
<b>Justificación:</b>	El sistema debe permitir buscar ubicaciones por nombre y usuarios por tipo (administrador u operador) para facilitar la gestión y consulta.	
<b>Condiciones/Datos específicos</b>	Los resultados deben ser rápidos y precisos.	
<b>Criterios de Validación:</b>	Verificar que la búsqueda funcione correctamente.	
<b>Dependencia</b>	REQ6	
<b>Analistas:</b> <b>Luis Fernando Mejía Amaya</b> <b>Cristian Alexander Parada López</b>	Proyecto: Diseño e implementación del Sistema de Soporte Técnico para la Universidad Luterana Salvadoreña 2024	

Tabla 5. REQ5

Identificación del Requisito:	Tipo de Requisito	Caso de Uso/Evento
<b>REQ6</b>	Funcional	Disponibilidad
<b>Objetivos asociados:</b>	1 y 2	
<b>Justificación:</b>	El sistema debe estar disponible 24/7 para usuarios con credenciales válidas.	
<b>Condiciones/Datos específicos</b>	Los usuarios solo podrán acceder al sistema con credenciales válidas.	
<b>Criterios de Validación:</b>	Verificar que el sistema esté accesible en cualquier momento.	
<b>Dependencia</b>	REQ7	
<b>Analistas:</b> <b>Luis Fernando Mejía Amaya</b>	Proyecto: Diseño e implementación del Sistema de Soporte Técnico para la Universidad Luterana Salvadoreña 2024	

Tabla 6. REQ 6

<b>Identificación del Requisito:</b>	<b>Tipo de Requisito</b>	<b>Caso de Uso/Evento</b>
<b>REQ7</b>	Funcional	Accesibilidad
<b>Objetivos asociados:</b>	2	
<b>Justificación:</b>	El sistema debe ser accesible desde cualquier navegador compatible de manera local.	
<b>Condiciones/Datos específicos</b>	El sistema debe funcionar correctamente en los principales navegadores.	
<b>Criterios de Validación:</b>	Verificar que el sistema sea accesible desde los navegadores indicados.	
<b>Dependencia</b>	REQ8	
<b>Analistas: Jonathan Esaú Magaña Rodríguez</b>	Proyecto: Diseño e implementación del Sistema de Soporte Técnico para la Universidad Luterana Salvadoreña 2024	

*Tabla 7. REQ 7*

<b>Identificación del Requisito:</b>	<b>Tipo de Requisito</b>	<b>Caso de Uso/Evento</b>
<b>REQ8</b>	Funcional	Rendimiento
<b>Objetivos asociados:</b>	1, 2 y 3	
<b>Justificación:</b>	El sistema debe soportar la creación y consulta simultánea de tickets sin que el tiempo de respuesta sea mayor a 1 segundo por cada 100 usuarios concurrentes.	
<b>Condiciones/Datos específicos</b>	El sistema debe ser capaz de manejar múltiples solicitudes simultáneas.	
<b>Criterios de Validación:</b>	Realizar pruebas de carga y medir el tiempo de respuesta.	
<b>Dependencia</b>	REQ9	
<b>Analistas: Tania Lisbeth Garay Góchez Rodolfo</b>	Proyecto: Diseño e implementación del Sistema de Soporte Técnico para la Universidad Luterana Salvadoreña 2024	

*Tabla 8. REQ 8*

<b>Identificación del Requisito:</b>	<b>Tipo de Requisito</b>	<b>Caso de Uso/Evento</b>
<b>REQ9</b>	Funcional	Mantenibilidad
<b>Objetivos asociados:</b>	1 y 2	
<b>Justificación:</b>	El sistema debe ser fácil de mantener y permitir la incorporación de nuevas funcionalidades en el futuro, como reportes avanzados o integración con otros sistemas.	
<b>Condiciones/Datos específicos</b>	El código debe ser modular y flexible.	
<b>Criterios de Validación:</b>	Evaluar la facilidad para agregar nuevas funcionalidades.	
<b>Dependencia</b>	REQ10	
<b>Analistas: Katherine Alexandra De León Hernández</b>	Proyecto: Diseño e implementación del Sistema de Soporte Técnico para la Universidad Luterana Salvadoreña 2024	

Tabla 9. REQ 9

<b>Identificación del Requisito:</b>	<b>Tipo de Requisito</b>	<b>Caso de Uso/Evento</b>
<b>REQ10</b>	Funcional	Escalabilidad
<b>Objetivos asociados:</b>	4	
<b>Justificación:</b>	El sistema debe ser escalable para permitir la integración de más módulos en el futuro, como la gestión de inventarios.	
<b>Condiciones/Datos específicos</b>	El sistema debe estar diseñado para agregar módulos adicionales sin problemas.	
<b>Criterios de Validación:</b>	Verificar que el sistema pueda integrar nuevos módulos sin afectar el rendimiento.	
<b>Dependencia</b>	REQ11	
<b>Analistas: Cristian Alexander Parada López</b>	Proyecto: Diseño e implementación del Sistema de Soporte Técnico para la Universidad Luterana Salvadoreña 2024	

Tabla 10. REQ 10

Identificación del Requisito:	Tipo de Requisito	Caso de Uso/Evento
<b>REQ11</b>	Funcional	Almacenamiento de Datos
<b>Objetivos asociados:</b>	1 y 3	
<b>Justificación:</b>	El sistema debe almacenar los datos de usuarios, ubicaciones y tickets en una base de datos MySQL, estructurada con relaciones entre tablas, para garantizar la integridad y la facilidad de acceso a la información.	
<b>Condiciones/Datos específicos</b>	La base de datos debe tener tablas interrelacionadas para usuarios, ubicaciones y tickets, con claves foráneas donde sea necesario.	
<b>Criterios de Validación:</b>	Verificar que los datos se almacenen correctamente y que las relaciones entre las tablas estén implementadas adecuadamente.	
<b>Dependencia</b>	REQ12	
<b>Analistas:</b> <b>Luis Fernando Mejía Amaya</b> <b>Edwin Sarbelio Cortez Rivera</b>	Proyecto: Diseño e implementación del Sistema de Soporte Técnico para la Universidad Luterana Salvadoreña 2024	

Tabla 11. REQ 11

Identificación del Requisito:	Tipo de Requisito	Caso de Uso/Evento
<b>REQ12</b>	Funcional	Almacenamiento de Datos de Tickets
<b>Objetivos asociados:</b>	2 y 3	
<b>Justificación:</b>	Los datos de cada ticket deberán incluir información clave como: ID, ID del usuario asignado, ID de la ubicación, descripción del problema, fecha y estado, para asegurar la trazabilidad y seguimiento adecuado de los tickets.	
<b>Condiciones/Datos específicos</b>	Cada ticket debe registrar estos campos de forma precisa, y su estado debe ser actualizado según el progreso del soporte.	
<b>Criterios de Validación:</b>	Verificar que todos los campos mencionados se almacenen correctamente en la base de datos para cada ticket.	
<b>Dependencia</b>	REQ11	
<b>Analistas:</b> <b>Luis Fernando Mejía Amaya</b> <b>Edwin Sarbelio Cortez Rivera</b>	Proyecto: Diseño e implementación del Sistema de Soporte Técnico para la Universidad Luterana Salvadoreña 2024	

Tabla 12. REQ 12

Identificación del Requisito:	Tipo de Requisito	Caso de Uso/Evento
<b>REQ13</b>	Funcional	Almacenamiento de Datos de Usuarios
<b>Objetivos asociados:</b>	1	
<b>Justificación:</b>	Los usuarios deberán tener registrados un nombre, una contraseña encriptada, el tipo de usuario (administrador u operador) y su estado (activo/inactivo), para garantizar la seguridad y control de accesos en el sistema.	
<b>Condiciones/Datos específicos</b>	La contraseña debe ser encriptada utilizando un algoritmo seguro, y el estado de los usuarios debe gestionarse dinámicamente.	
<b>Criterios de Validación:</b>	Verificar que la contraseña esté correctamente encriptada y que los campos de tipo de usuario y estado estén configurados adecuadamente.	
<b>Dependencia</b>	REQ12	
<b>Analistas: Katherine Alexandra De León Hernández</b>	Proyecto: Diseño e implementación del Sistema de Soporte Técnico para la Universidad Luterana Salvadoreña 2024	

Tabla 13. REQ 13

Identificación del Requisito:	Tipo de Requisito	Caso de Uso/Evento
<b>REQ14</b>	Funcional	Almacenamiento de Datos de Ubicaciones
<b>Objetivos asociados:</b>	3	
<b>Justificación:</b>	Cada ubicación deberá tener un nombre único, un ID único y un estado (activa/inactiva) para su correcta gestión y utilización dentro del sistema.	
<b>Condiciones/Datos específicos</b>	Las ubicaciones deben estar registradas con un nombre único y su estado debe poder modificarse dependiendo de su disponibilidad.	
<b>Criterios de Validación:</b>	Verificar que las ubicaciones sean únicas por nombre e ID, y que su estado se mantenga actualizado.	
<b>Dependencia</b>	REQ12	
<b>Analistas: Jonathan Esaú Magaña Rodríguez</b>	Proyecto: Diseño e implementación del Sistema de Soporte Técnico para la Universidad Luterana Salvadoreña 2024	

Tabla 14. REQ 14

<b>Identificación del Requisito:</b>	<b>Tipo de Requisito</b>	<b>Caso de Uso/Evento</b>
<b>REQ15</b>	No Funcional	Interfaz de Usuario
<b>Objetivos asociados:</b>	1 y 3	
<b>Justificación:</b>	La interfaz debe ser clara, intuitiva y en español, para facilitar la navegación tanto para operadores como para administradores, mejorando la experiencia de usuario.	
<b>Condiciones/Datos específicos</b>	La interfaz debe ser sencilla, con botones y menús bien diferenciados para facilitar la navegación.	
<b>Criterios de Validación:</b>	Verificar que los elementos de la interfaz sean fáciles de identificar y que la navegación sea intuitiva.	
<b>Dependencia</b>	REQ15	
<b>Analistas: Jonathan Esaú Magaña Rodríguez</b>	Proyecto: Diseño e implementación del Sistema de Soporte Técnico para la Universidad Luterana Salvadoreña 2024	

Tabla 15. REQ 15

<b>Identificación del Requisito:</b>	<b>Tipo de Requisito</b>	<b>Caso de Uso/Evento</b>
<b>REQ16</b>	No Funcional	Diseño Adaptable
<b>Objetivos asociados:</b>	2 y 3	
<b>Justificación:</b>	El sistema debe tener un diseño adaptable a diferentes dispositivos (computadoras, tabletas, smartphones) para facilitar su uso desde cualquier plataforma.	
<b>Condiciones/Datos específicos</b>	El diseño debe ajustarse a distintos tamaños de pantalla y resoluciones.	
<b>Criterios de Validación:</b>	Verificar que la interfaz funcione correctamente en dispositivos con distintos tamaños de pantalla.	
<b>Dependencia</b>	REQ16	
<b>Analistas: Rodolfo</b>	Proyecto: Diseño e implementación del Sistema de Soporte Técnico para la Universidad Luterana Salvadoreña 2024	

Tabla 16. REQ 16

Identificación del Requisito:	Tipo de Requisito	Caso de Uso/Evento
<b>REQ17</b>	Funcional	Interfaces por Rol de Usuario
<b>Objetivos asociados:</b>	1	
<b>Justificación:</b>	Los operadores y administradores tendrán interfaces específicas según su rol, con botones y opciones claramente diferenciadas, para que cada uno tenga acceso solo a las funciones correspondientes.	
<b>Condiciones/Datos específicos</b>	Las interfaces deben diferir en los módulos y botones disponibles, asegurando que cada usuario vea solo lo que le corresponde según su rol.	
<b>Criterios de Validación:</b>	Verificar que la interfaz se adapta a los distintos roles de usuario y que las opciones disponibles son las correctas.	
<b>Dependencia</b>	REQ15	
<b>Analistas: Rodolfo</b>	Proyecto: Diseño e implementación del Sistema de Soporte Técnico para la Universidad Luterana Salvadoreña 2024	

Tabla 17. REQ 17

Identificación del Requisito:	Tipo de Requisito	Caso de Uso/Evento
<b>REQ18</b>	No Funcional	Compatibilidad con Navegadores
<b>Objetivos asociados:</b>	1	
<b>Justificación:</b>	El sistema debe funcionar correctamente en los principales navegadores web (Google Chrome, Mozilla Firefox, Microsoft Edge, Safari) para garantizar la accesibilidad de los usuarios independientemente del navegador que utilicen.	
<b>Condiciones/Datos específicos</b>	El sistema debe ser completamente funcional en estos navegadores populares sin necesidad de ajustes adicionales.	
<b>Criterios de Validación:</b>	Verificar que el sistema se ejecute sin errores en los navegadores mencionados.	
<b>Dependencia</b>	REQ16	
<b>Analistas: Katherine Alexandra De León Hernández</b>	Proyecto: Diseño e implementación del Sistema de Soporte Técnico para la Universidad Luterana Salvadoreña 2024	

Tabla 18. REQ 18

<b>Identificación del Requisito:</b>	<b>Tipo de Requisito</b>	<b>Caso de Uso/Evento</b>
<b>REQ19</b>	No Funcional	Diseño de Navegación
<b>Objetivos asociados:</b>	2 y 3	
<b>Justificación:</b>	El diseño del sistema debe garantizar que cada módulo sea accesible con un máximo de tres clics desde la página principal, mejorando la usabilidad y facilitando la navegación.	
<b>Condiciones/Datos específicos</b>	Las opciones más importantes deben estar accesibles en un máximo de tres clics, sin que los usuarios tengan que buscar demasiado.	
<b>Criterios de Validación:</b>	Verificar que todos los módulos importantes sean accesibles dentro de tres clics desde la página principal.	
<b>Dependencia</b>	REQ17	
<b>Analistas:</b> <b>Tania Lisbeth Garay Góchez</b> <b>Berta Eunice López Martínez</b>	Proyecto: Diseño e implementación del Sistema de Soporte Técnico para la Universidad Luterana Salvadoreña 2024	

*Tabla 19. REQ 19*

## 7. ANÁLISIS

### MODELO DE ANÁLISIS SELECCIONADO:

#### Programación Orientado a Objetos (POO)

Se ha seleccionado el enfoque Orientado a Objetos para realizar el análisis del sistema, aprovechando las herramientas y diagramas UML. Este enfoque facilita la implementación del sistema mediante clases, atributos y métodos que reflejan las entidades reales del proyecto.

#### Diagramas UML:

1. **Diagrama de casos de uso:** Representa las interacciones entre los actores (operador, administrador) y el sistema.

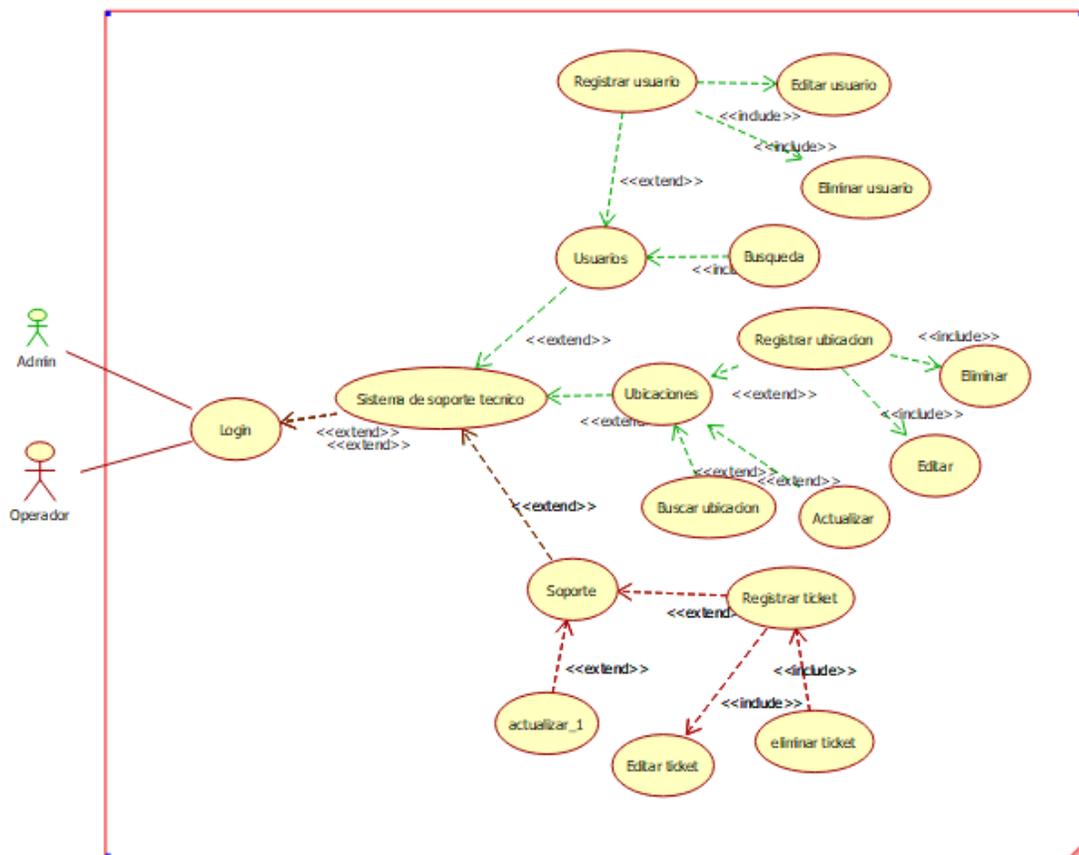


Ilustración 2. Diagrama casos de uso

2. **Diagrama de clases:** Muestra las clases principales (Usuario, Soporte, Ubicación) y sus relaciones.

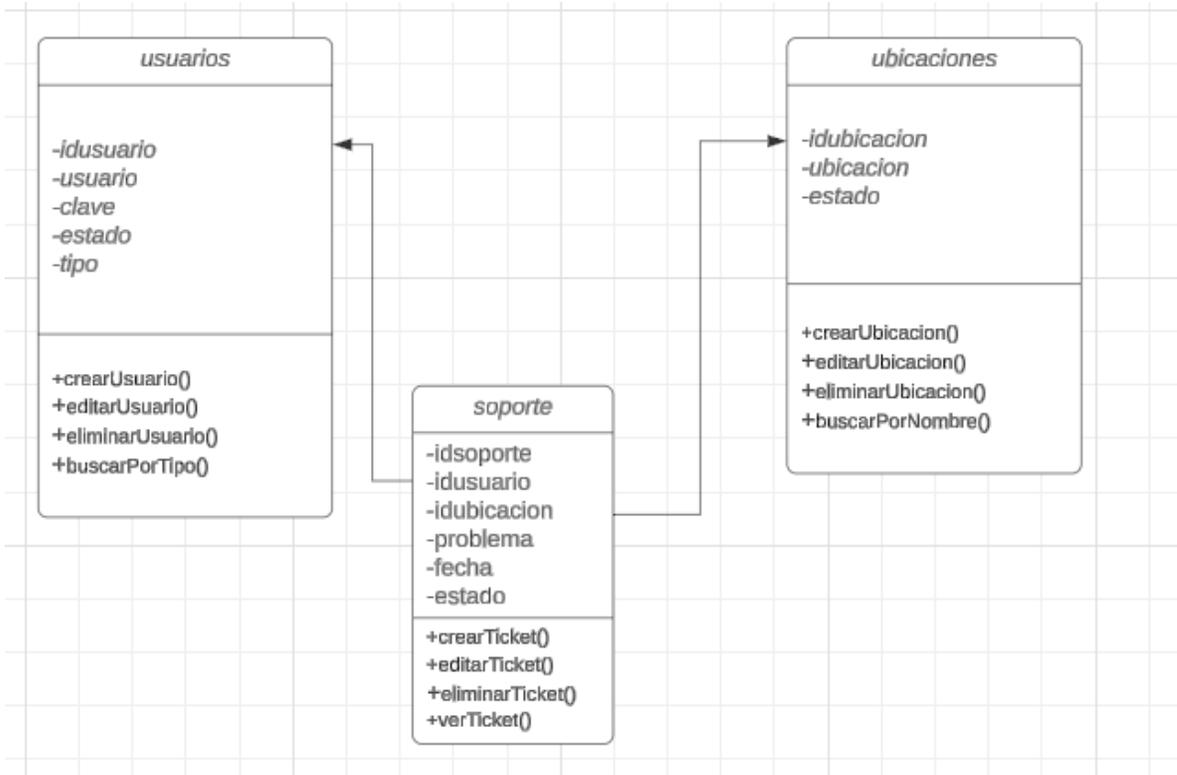


Ilustración 3. Diagrama de clases

3. **Diagrama de secuencia:** Describe el flujo de acciones durante la creación, edición o eliminación de tickets, usuarios o ubicaciones.

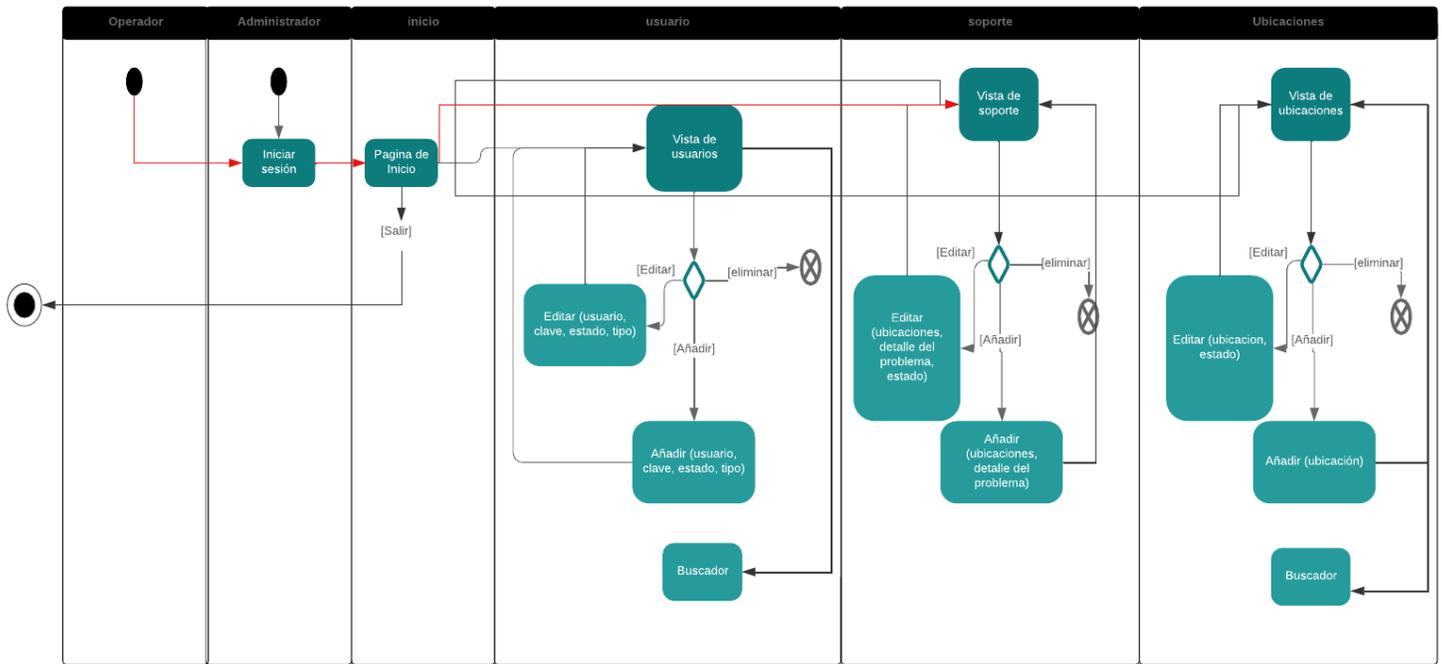


Ilustración 4. Diagrama de secuencia

## 8. DISEÑO

### Técnicas de diseño utilizadas

El diseño del Sistema de Soporte Técnico (SST) se desarrolló bajo el paradigma de la Programación Orientada a Objetos (POO), utilizando herramientas de diseño como diagramas UML para representar la estructura y el comportamiento del sistema.

### Diseño de datos:

- **Diagrama de clases (UML):** Incluye las clases principales (Usuario, Soporte, Ubicación) con atributos y métodos, además de sus relaciones.

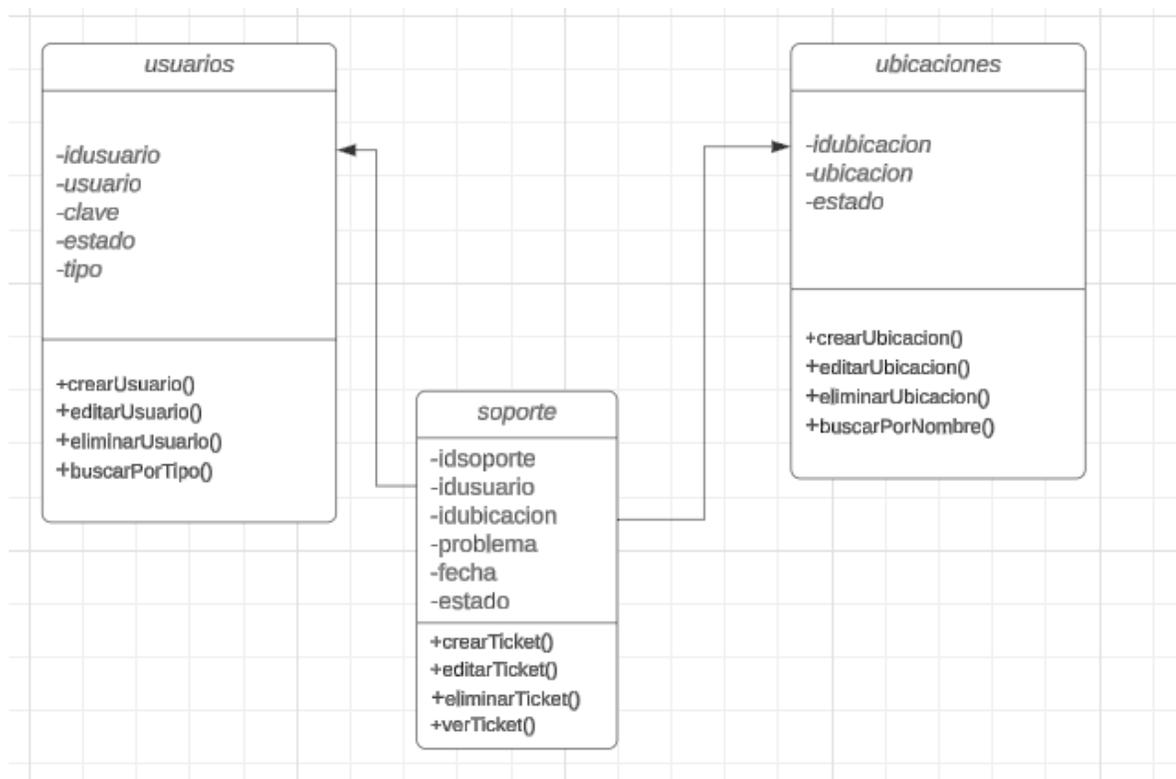


Ilustración 5. Diagrama de clases (UML)

## **Diseño de interfaces:**

### **1. Menú principal:**

- Acceso segmentado según roles (Administrador u Operador).
- Enlaces a módulos: Soporte, Usuarios, Ubicaciones.

### **2. Pantallas de entrada:**

- Formularios para crear y editar tickets de soporte, usuarios y ubicaciones.
- Validaciones básicas para evitar datos inconsistentes.

### **3. Pantallas de salida:**

- Listado de tickets con filtros por estado.
- Listado de usuarios con búsqueda por rol.
- Listado ubicaciones inactivas.

## 9. DESARROLLO

### 9.1 TÉCNICAS DE PROGRAMACIÓN APLICADAS

El sistema se implementó utilizando el patrón Modelo-Vista-Controlador (MVC) para organizar el código en capas:

- **Modelo:** Manejo de datos y conexión con MySQL, gestionando las consultas necesarias.

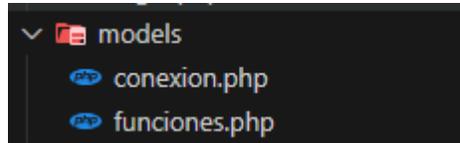


Ilustración 6. Carpeta de modelo

- **Vista:** Representación de interfaces de usuario con HTML y CSS, proporcionando formularios y listados.

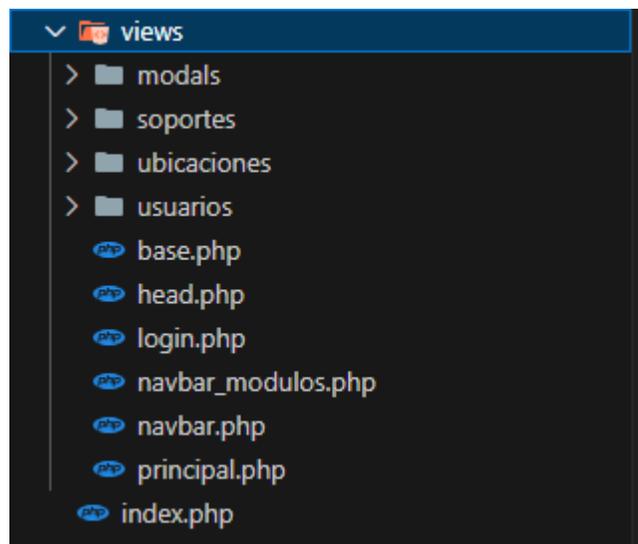


Ilustración 7. Carpeta de Vista

- **Controlador:** Gestión de la lógica del sistema, controlando la interacción entre vistas y modelos.

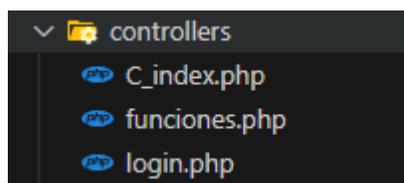


Ilustración 8. Carpeta de controlador

## 9.2 HERRAMIENTAS DE DESARROLLO UTILIZADAS

### 1. Editor de código: Visual Studio Code (versión 1.81).

- Ventajas: Autocompletado, integración con Git, extensiones útiles para PHP y MySQL.



Ilustración 9.  
VScode

### 2. Servidor local: XAMPP con PHP 8.1.

- Incluye Apache y MySQL, facilitando pruebas locales.



Ilustración 10. XAMPP

## 9.3 LENGUAJE DE PROGRAMACIÓN Y GESTOR DE BASES DE DATOS

### 1. Lenguaje de programación: PHP 8.2.12, utilizado para implementar la lógica y funcionalidades.



Ilustración 11. Lenguaje de programación

### 2. Gestor de bases de datos: MySQL 8.2.12, utilizado para almacenar y gestionar datos.

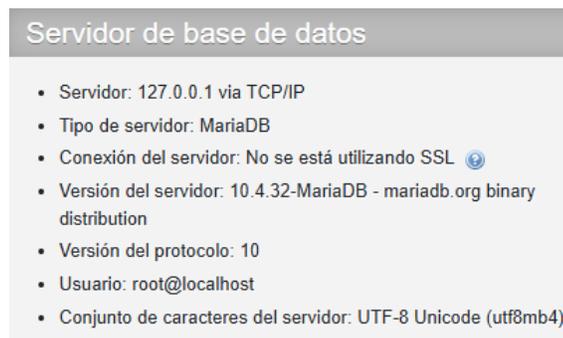


Ilustración 12. Gestor base de datos

## 9. DOCUMENTACIÓN

### 9.1 TÉCNICAS DE PRUEBA DEL SISTEMA

Se aplicaron las siguientes técnicas para validar la funcionalidad:

1. **Pruebas unitarias:** Validación de funciones individuales en el código.

```
PS C:\xampp\htdocs\SST\tests> phpunit ProcesosTest.php
PHPUnit 10.5.38 by Sebastian Bergmann and contributors.

Runtime:       PHP 8.1.10
E                                                     1 / 1 (100%)

Time: 00:00.105, Memory: 8.00 MB

There was 1 error:

1) ProcesosTest::testGetDataUserReturnsCorrectData
Error: Class "Procesos" not found

C:\xampp\htdocs\SST\tests\ProcesosTest.php:21

ERRORS!
Tests: 1, Assertions: 0, Errors: 1.
PS C:\xampp\htdocs\SST\tests>
```

Ilustración 13. PHPUnit

2. **Pruebas funcionales:** Verificación del correcto funcionamiento de los formularios y flujos.

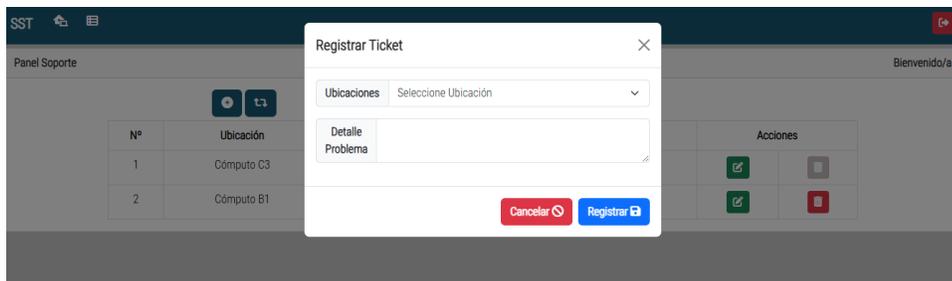


Ilustración 14. Prueba de sistema

3. **Pruebas de usuario:** Realizadas con usuarios finales para evaluar la usabilidad del sistema.

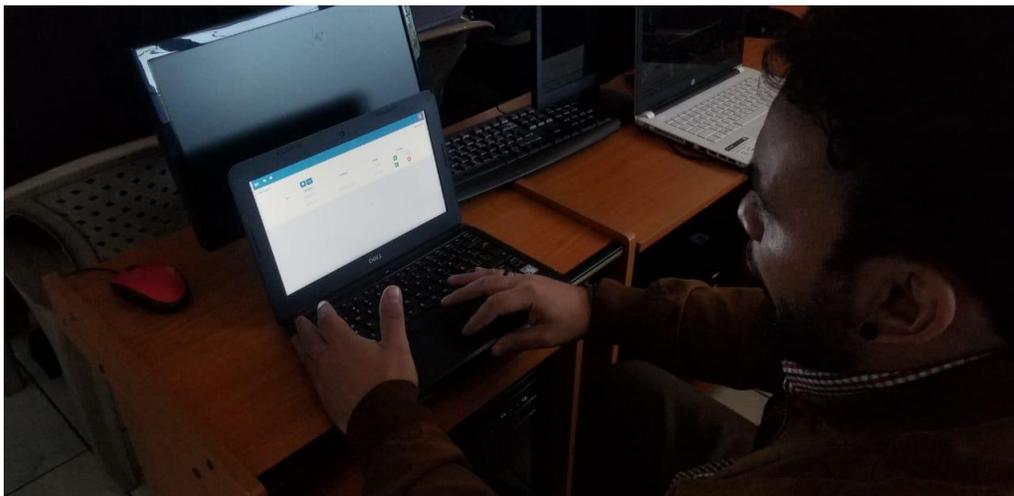


Ilustración 15. Prueba con usuario

## 10. CONCLUSIONES

El desarrollo del Sistema de Soporte Técnico (SST) ha cumplido con el objetivo general de proporcionar una herramienta eficiente para la gestión de incidencias, usuarios y ubicaciones, logrando optimizar procesos y mejorar la coordinación entre administradores y operadores. Mediante una estructura modular basada en PHP y MySQL, el sistema ha demostrado ser una solución eficaz para centralizar las operaciones relacionadas con el soporte técnico, aportando claridad y orden en la administración de incidencias.

En relación con el primer objetivo específico, se implementó exitosamente un módulo de soporte técnico que permite registrar, editar, eliminar y visualizar tickets de problemas. Este módulo ha simplificado la gestión de incidencias técnicas, facilitando el acceso a la información relevante y garantizando un seguimiento detallado del estado de cada incidencia.

Asimismo, se logró la creación de un módulo para la gestión de usuarios que permite administrar accesos y roles. Esta funcionalidad ha sido clave para diferenciar responsabilidades entre administradores y operadores, lo que asegura un control efectivo sobre las actividades realizadas en el sistema y protege la integridad de los datos.

Por otra parte, el sistema incorpora un módulo para la gestión de ubicaciones, que facilita el registro, edición y búsqueda de ubicaciones técnicas. Esto ha permitido a los usuarios localizar con rapidez las áreas afectadas y asociarlas a los tickets de soporte, optimizando los tiempos de respuesta y aumentando la eficiencia operativa.

Finalmente, se implementó un sistema de búsqueda y filtrado tanto en el módulo de usuarios como en el de ubicaciones, lo que mejora significativamente la experiencia del usuario al agilizar las consultas y garantizar el acceso inmediato a la información requerida. Este logro refuerza la funcionalidad del sistema, haciéndolo más práctico y accesible para los distintos actores involucrados en el proceso de soporte técnico.

## 11. RECOMENDACIONES

- Evolución hacia Metodologías Ágiles: Aunque el modelo Cascada fue adecuado para la planificación estructurada, la incorporación de metodologías ágiles como Scrum podría mejorar la adaptabilidad a cambios durante el desarrollo, permitiendo iteraciones más rápidas y entregas incrementales.
- Mejora Continua: Es recomendable establecer un proceso de mejora continua basado en el feedback de los usuarios finales. Esto incluye actualizaciones regulares del sistema para adaptarse a nuevas necesidades o tecnologías emergentes.
- Fortalecer las medidas de seguridad, como la implementación de validaciones más estrictas para prevenir inyecciones SQL y accesos no autorizados.
- Considerar la migración a una arquitectura escalable si la demanda de usuarios y datos crece significativamente.
- Automatización de Tareas Repetitivas: Incorporar mecanismos de automatización para tareas comunes, como la asignación de tickets a técnicos disponibles o el envío de notificaciones por correo, podría optimizar aún más los procesos.

## 12. BIBLIOGRAFÍAS

METODOLOGÍAS DE GESTIÓN DE PROYECTOS TEMA 1 MODELO WATERFALL O EN CASCADA. (n.d.).

[https://www.dtagency.tech/cursos/metodologias\\_gestion\\_proyectos/tema\\_1-ModeloWaterfall.pdf](https://www.dtagency.tech/cursos/metodologias_gestion_proyectos/tema_1-ModeloWaterfall.pdf)

Durand, S. (n.d.). DISCAPACIDAD E INTEGRIDAD Manual Autoformativo Interactivo Análisis y requerimientos de software.

[https://repositorio.continental.edu.pe/bitstream/20.500.12394/4281/1/DO\\_FIN\\_103\\_MAI\\_UC0939\\_2018.pdf](https://repositorio.continental.edu.pe/bitstream/20.500.12394/4281/1/DO_FIN_103_MAI_UC0939_2018.pdf)

elhacker.INFO - Descargas Cursos, Manuales, Tutoriales y Libros. (n.d.). Elhacker.info.

<https://elhacker.info/manuales/Lenguajes%20de%20Programacion/0185-programacion-orientada-a-objetos.pdf>

Grupo EIDOS Ejemplar gratuito. (1999). Introduccion a la Programacion Orientada a Objetos.

<https://kataix.umag.cl/~ruribe/Utilidades/Introduccion%20a%20la%20Programacion%20Orientada%20a%20Objetos.pdf>

## 13. ANEXOS

### 13.1. DIAGRAMA DE LA BASE DE DATOS



Ilustración 16. Diagrama de base de datos

### 13.2 FOTOGRAFÍAS DEL DESARROLLO

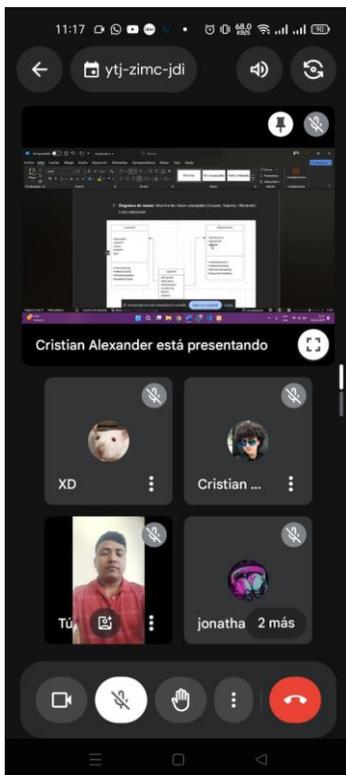


Ilustración 19. Reunión 1

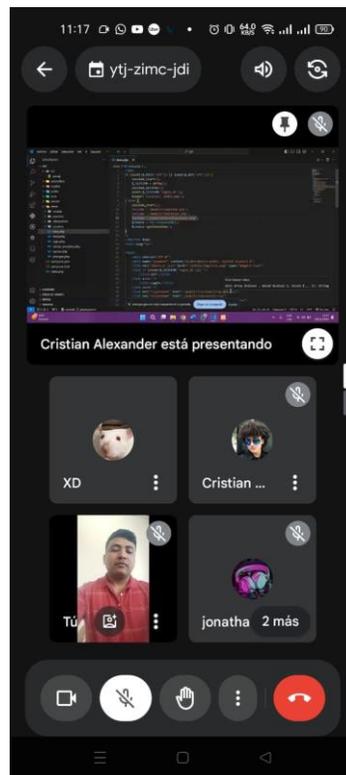


Ilustración 18. Reunión 2

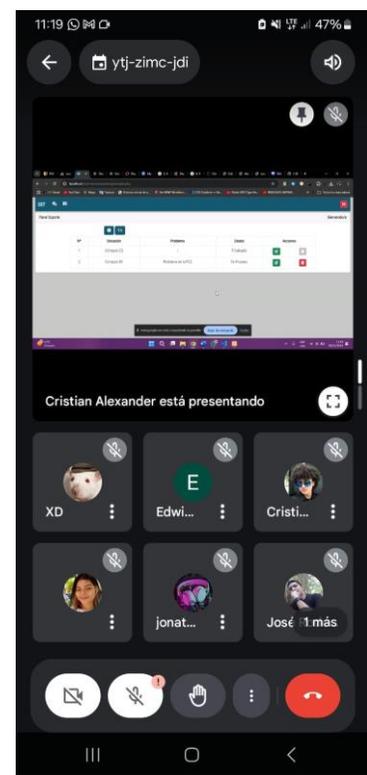


Ilustración 17. Reunión 3

### 13.3 MANUAL DE PROGRAMADOR Y MANUAL DE USUARIO:

<https://drive.google.com/drive/folders/1215orDw-cGrUHK5MmLfOfurx5lShlaNy?usp=sharing>

