

**CURSO: CURSO INTEGRADOR II: SOFTWARE**

**DOCENTE:**

**JUAN LUIS MAGALLANES RODRIGUEZ**

**“Diseño de un sistema web de mesa de ayuda para el área de soporte de una entidad financiera”**

INTEGRANTES:

● LEDESMA VARGAS, AXEL LUCAS U19314417

● SÁNCHEZ CORRO, JAVIER SEBASTIAN U19209170

● LIRA POLO, JOSÉ MARCO 1631816

● TUPAHUACCAILLO SANDOVAL, JOSUÉ MARTIN U19213818

***Lima - 2023***

Índice:

[**ASPECTOS GENERALES 3**](#_9n7crbef3r73)

[1.1 Definición del Problema 3](#_aws4r73r8ui3)

[1.1.1 Descripción del Problema 3](#_803bpey5r63p)

[1.2 Definición de Objetivos 4](#_pi69ign1rmuc)

[1.2.1 Objetivo General 4](#_p7xznuft6xs0)

[1.2.2 Objetivos Específicos 4](#_a4iormifvmn2)

[1.2.3 Alcances y Limitaciones 5](#_bwlflgnlue0m)

[1.2.4 Justificación 5](#_vu9wp1vxh8v9)

[1.2.5 Estado del arte 6](#_dkbs3weo6k18)

[**MARCO TEÓRICO 9**](#_l7qafvukxe7)

[2.1 Fundamento teórico 9](#_u0niyf6j40ly)

[2.1.1 Sistema Web 9](#_hze2tvkn3vt8)

[2.1.2 Mesa de ayuda 10](#_mi9d3fc3z9be)

[2.2 Marco Metodológico 10](#_i77ctal73nze)

[2.2.1 Metodología Scrum 10](#_jbbw1bo4dpu7)

[2.2.2 Fases del Scrum 11](#_lrmpeb5om2gt)

[2.2.3 Componentes de Scrum 12](#_6vt4rc9l72lg)

[Las reuniones 12](#_d9f6m1ucseuh)

[Los roles 12](#_esk2hsw4cf8o)

[Los elementos de Scrum 13](#_qvrc44cnriin)

[2.3 Marco Conceptual 13](#_ixu83dhi1vcb)

[JavaScript 13](#_utfl4fdjl3hr)

[PHP 13](#_1174ggwey4jb)

[MySQL 14](#_vkwtetnz2h7q)

[HTML 14](#_fkimdzhwlc93)

[Visual Studio Code 15](#_nhe1hxanwnb4)

[3.1 Fases del Scrum 16](#_1zl92fvpnk0y)

[3.1.1 Fase Inicial 16](#_6rx61zy2ve1u)

[3.1.1.1 Visión del Proyecto 16](#_n9rr0c4ll6sk)

[3.1.1.2 Diagrama de Contexto 16](#_61416wfwb6o3)

[3.1.1.3 Identificación de Actores 17](#_6lxhoh9xr9ah)

[3.1.1.4 Desarrollo de épicas 18](#_55uymv9b0zwo)

[3.1.1.5 Listas de pendientes 30](#_sw1j5jmo46lo)

[3.1.1.6 Plan de lanzamiento 30](#_abd2reyzfqro)

[3.1.1.7 Repositorio del Proyecto 31](#_3a3reyg0hvrp)

[3.1.1.8 Producto Backlog priorizado 31](#_us6ezhtjsx7w)

[3.1.1.9 Planificación de la iteración (Sprint). 33](#_hgutw785wh4m)

[3.1.1.10 Sprint 35](#_6xobdxf6hy16)

[3.1.1.11 Sprint Backlog 37](#_44gaazh5zibw)

[3.1.1.12 Mínimo Producto Viable 38](#_gblhmiullo19)

[**3.2. Arquitectura tecnológica 39**](#_3t7ghcsr6ye8)

[**3.3. Mockups del diseño 40**](#_qwp1ucd92uqk)

[**3.4 Funcionalidades del código 44**](#_8dswyai9evkp)

[**CONCLUSIONES 53**](#_7pauwk96nvh2)

[**RECOMENDACIONES 53**](#_lp7h533za9hd)

[**ANEXOS 53**](#_4h5tuoh22rzp)

[**BIBLIOGRAFÍA 54**](#_8g6c3wqrkbxp)

CAPÍTULO 1

# ASPECTOS GENERALES

### 1.1 Definición del Problema

#### 1.1.1 Descripción del Problema

El área de soporte técnico informático y computacional de la entidad financiera, que es el cliente al cual se le va a diseñar la solución web, realiza los procesos de atención de solicitudes a mano y escritos sobre hojas de papel, sin orientación sobre las situaciones urgentes y generando sobrecarga de trabajo, tanto para el personal del área como para el personal que solicita atención. Esto sin duda causa un retraso en otros sistemas dentro de la estructura organizacional de la entidad financiera. Según indican Sandhu Supreet y Arora Sangeeta (2022), la deficiencia en un sistema conlleva al colapso de los demás sistemas, lo que a su vez y sin remedio, causará una mala atención para los clientes de dicha organización.

Casos anteriores demuestran que, sin un sistema de organización de mesa de ayuda óptimo, los procesos pueden causar cuellos de botella dentro de la empresa o, como en nuestro caso, entidad financiera. Esto se ve apoyado por la investigación de Rodríguez y Pino (2019) en el caso de la empresa HITSS Ecuador. Estos cuellos de botella en los procesos de atención de solicitudes, provocados por la falta de una mesa de ayuda eficiente, conllevan a una pérdida de recursos en la coordinación de esfuerzos sobre las prioridades que deben ser atendidas. Tal como lo enmarcan Faria y Freire (2021) en su investigación.

### 1.2 Definición de Objetivos

#### 1.2.1 Objetivo General

Diseñar un sistema web de mesa de ayuda para el área de soporte de una entidad financiera.

#### 1.2.2 Objetivos Específicos

* Analizar la funcionalidad del sistema a través de la metodología ágil SCRUM.
* Crear la base de datos relacional para el almacenamiento de los usuarios y la solicitud de tickets estableciendo la prioridad de atención a las incidencias de forma automatizada a través de la categoría seleccionada por el usuario que registra la solicitud.
* Generar notificaciones por correo electrónico a los usuarios y al equipo de soporte cada vez que se genere un nuevo ticket o se actualice un ticket existente. (Además, el equipo de soporte también recibirá notificaciones por WhatsApp al momento de crear, asignar y cerrar un ticket.)
* Desarrollar una API basada en inteligencia artificial que habilite al equipo de soporte de la mesa de ayuda para responder a los tickets, utilizando las respuestas generadas por el modelo IA ChatGPT como base preliminar.
* Generar reportes del estado y categoría de los tickets generados.

#### 1.2.3 Alcances y Limitaciones

* Alcances

El software a diseñar permitirá la gestión de pedidos de ayuda al área de soporte técnico computacional de la entidad financiera.

Será diseñado para que se desenvuelva en un entorno de servicio web local o intranet.

Estará enlazado con un servicio de base de datos relacional.

Los clientes o usuarios del software a diseñar serán personal que labora dentro de la misma entidad bancaria.

* Limitaciones
* No habrá una base de datos con amplio almacenamiento debido a su alto costo.
* La automatización de la priorización basada en la categoría seleccionada por el usuario podría no ser siempre precisa.
* La IA puede no proporcionar respuestas precisas.
* En la generación de reportes pueden haber demoras debido a que no son preestablecidos sino personalizados al requerimiento del trabajador.

### 1.2.4 Justificación

El diseño de un sistema web de mesa de ayuda para el área de soporte de esta entidad financiera se presenta como una solución imperativa y estratégica ante los evidentes desafíos que enfrenta la organización en la actualidad. Los procesos obsoletos y manuales para la gestión de pedidos de mantenimiento de tecnología no solo generan ineficiencias operativas y sobrecarga de trabajo, sino que también ponen en riesgo la calidad de atención a los clientes. La experiencia previa en otras empresas, como el caso de HITSS Ecuador y el enfoque de Faria y Freire, ilustra de manera elocuente cómo la falta de recursos tecnológicos y la ausencia de una metodología eficiente pueden desencadenar serias dificultades en la gestión de prioridades y, en última instancia, afectar la satisfacción del cliente.

El diseño de un sistema web de mesa de ayuda es fundamental para modernizar y optimizar los procesos, garantizando así una atención ágil, efectiva y centrada en las necesidades de los clientes, alineando la entidad financiera con las mejores prácticas tecnológicas y asegurando su competitividad en un entorno cada vez más digitalizado. Victor Arroyo (2021) nos indica que los resultados que se obtienen sobre la eficiencia, eficacia y calidad de servicio presentan una mejora sustancial que supera ampliamente las expectativas, pues dichas mejoras superan el 50% respecto a la gestión de incidentes antes de la implementación de un sistema web de mesa de ayuda.

### 1.2.5 Estado del arte

En la Tesis realizada en la Universidad Nacional del Callao, Lima, y titulada 'Sistema web para la gestión de incidencias de la mesa de ayuda de la empresa Nexus Technology' (Quispe, Requena, Soto, 2022), se planteó el objetivo de analizar el impacto de un sistema web desarrollado en los lenguajes JavaScript, TypeScript, Angular, y Node.JS en la optimización de la gestión de incidentes en la mesa de ayuda de la mencionada empresa. El sistema web ofrece un registro de solicitudes de incidencias sencillo de elaborar, donde se puede categorizar las mismas y ver el estado de las incidencias, en caso de solucionarlas se procede a seleccionar el cierre de la incidencia, también cuenta con una sección de informes que registran todas las solicitudes y su estado actual. Los resultados obtenidos demostraron una mejora palpable en el proceso de gestión de incidentes, además de un incremento significativo en el porcentaje de resolución exitosa de problemas mediante la categorización de prioridades para los incidentes reportados en la organización.

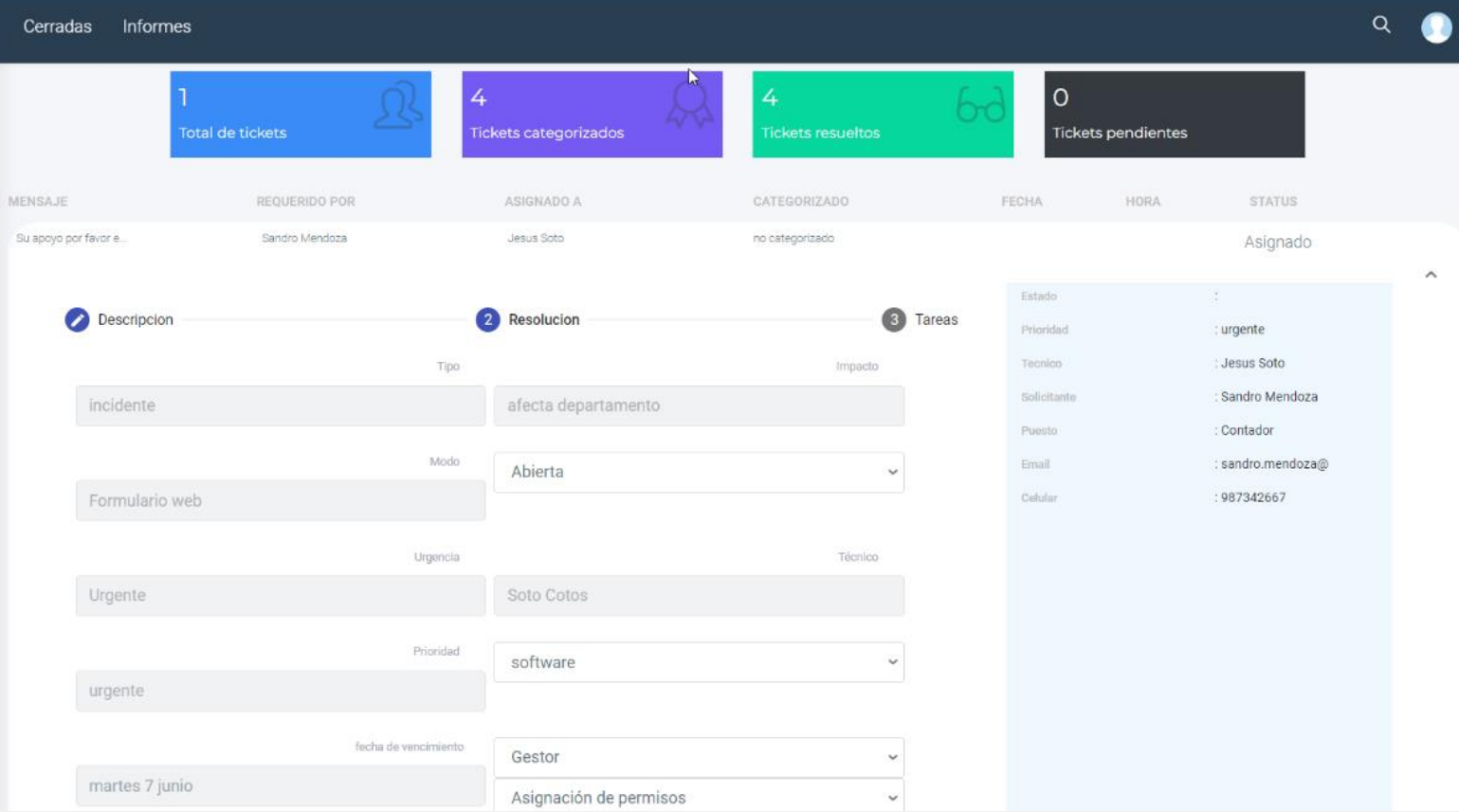


Figura 1: *Detalle de ticket - Sistema web para la gestión de incidencias de la mesa de ayuda de la empresa Nexus Technology*

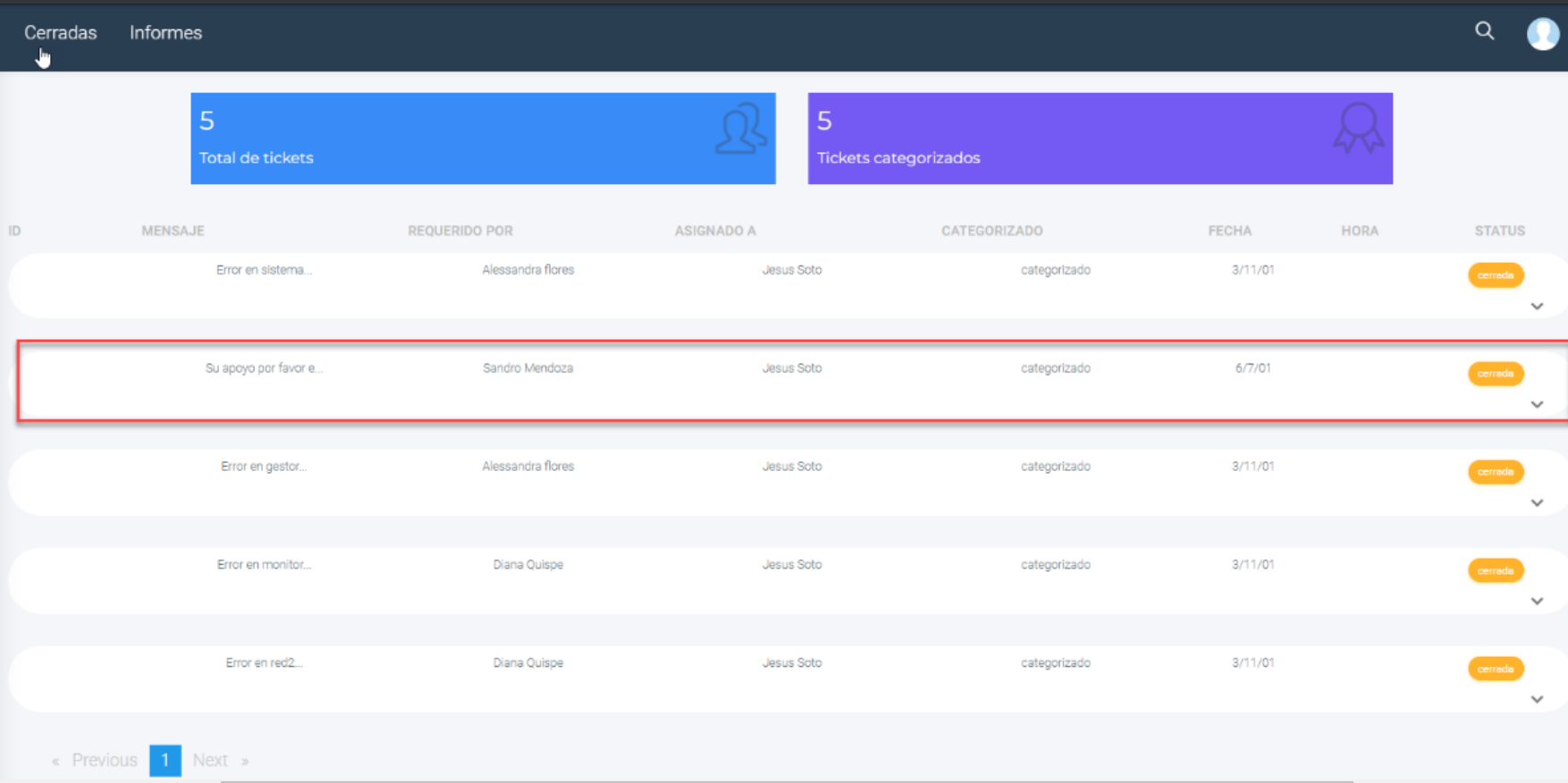
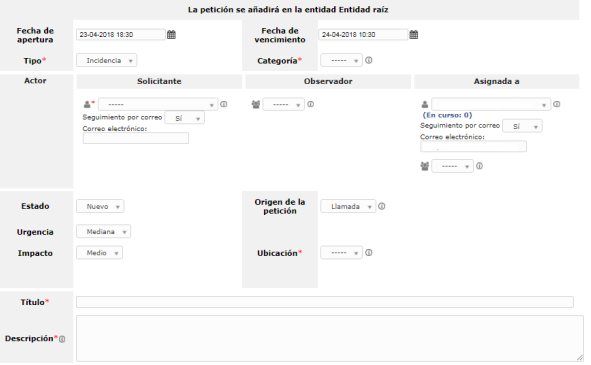
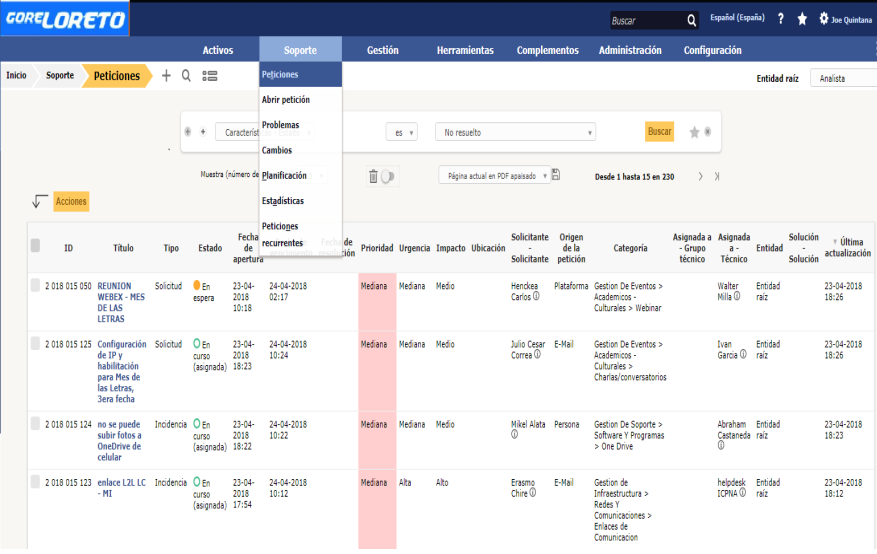


Figura 2: *Tickets generados - Sistema web para la gestión de incidencias de la mesa de ayuda de la empresa Nexus Technology*

En su Tesis llamada "Implantación de Aplicativo web para la mejora de la Gestión de mesa de ayuda en el Gore Loreto 2017" (Arbildo , Quiroz, 2017), se propusieron implementar un aplicativo web basado en ITIL, desarrollado en lenguajes como Java, HTML, CSS, SQL, para optimizar la gestión de la mesa de ayuda en el entorno del Gore Loreto Utilizando un enfoque de correlación no experimental, se analizaron conexiones entre variables en un grupo de 25 trabajadores de Energía y Minas. El aplicativo web cuenta con una interfaz más compleja donde podemos seleccionar el tipo y categoría de la incidencia a solicitar, además se selecciona el solicitante y como queremos que sea la atención una vez enviada la petición se muestra el estado en curso de las peticiones realizadas. Se concluyó que la implementación del aplicativo web está vinculada a mejoras en la gestión de la mesa de ayuda, demostrando efectos significativos en aspectos como el nivel de servicio y la administración de requerimientos. Las conclusiones enfatizan la influencia positiva del aplicativo web en la gestión de la mesa de ayuda, subrayando factores específicos que tuvieron impacto en el proceso.



*Figura 3: Formulario de petición - Implantación del Aplicativo web para la mejora de la Gestión de mesa de ayuda en el Core Loreto 2017" (Arbildo , Quiroz, 2017)*



*Figura 4: Registro de peticiones - Implantación del Aplicativo web para la mejora de la Gestión de mesa de ayuda en el Gore Loreto 2017" (Arbildo , Quiroz, 2017)*

CAPÍTULO 2

# MARCO TEÓRICO

### 2.1 Fundamento teórico

#### 2.1.1 Sistema Web

Un sistema web o aplicación web es un software que se encuentra alojado e implementado en un servidor, del cual pueden servirse los usuarios de manera local o remota, a través de una red interna o intranet, o el internet. Según Zurita (2020), esta tecnología puede ser usada desde cualquier navegador existente, independientemente del sistema operativo que utilice el usuario.

#### 2.1.2 Mesa de ayuda

La mesa de ayuda es una herramienta de uso institucional u organizacional por sobre el cual, la empresa realiza actividades que corresponden a intereses, generalmente ligados, a satisfacer las necesidades de los clientes y mejorar la comunicación entre la organización y estos. Una mesa de ayuda debe estar conformada por especialistas en el área de atención, maximizando la efectividad y evitando soluciones confusas o erróneas. Valverde (2021) dice textualmente: “la mesa de ayuda nos permitirá optimizar nuestros recursos y horas hombre de la empresa, así como también permitirá brindar una solución óptica a todas las incidencias reportadas por los clientes”. Lo cual se entiende que, una mesa de ayuda puede servir para mejorar el desempeño de la organización, produciendo efectividad y eficacia en las partes interesadas.

### 2.2 Marco Metodológico

#### 2.2.1 Metodología Scrum

Scrum es una metodología de desarrollo de software clasificado dentro de las metodologías ágiles. Ésta fue desarrollada por Jeff Sutherland y Ken Schwaber en 1996, donde dieron por primera vez las bases formales para el desarrollo ágil. Se basa principalmente en inspeccionar y adaptar el trabajo de manera continua conforme el proyecto va avanzando.

El equipo que se desenvuelve usando esta metodología intenta siempre ir dando valor al cliente conforme el proyecto avanza, manteniendo una comunicación directa y continua con todas las partes interesadas, lo cual propicia que el desarrollo sea altamente adaptable a los cambios y nuevos requerimientos.

Trigas(2012) nos dice en su libro “Scrum al ser una metodología ágil, tiene como base la idea de creación de ciclos breves para el desarrollo, que comúnmente se les llama iteraciones y que en Scrum se les llamarán sprints”

#### 2.2.2 Fases del Scrum

Según Trigas(2012), también nos habla que existen 5 fases que definen el ciclo del desarrollo ágil.

1. Concepto.- Definición genérica del proyecto y los equipos
2. Especulación.- Con la información anterior se delimitan momentáneamente las características del proyecto.
3. Exploración.- Se insertan las funcionalidades de la fase de especulación.
4. Revisión.- El equipo revisa lo avanzado y si coincide con los objetivos planteados.
5. Cierre.- Se entrega el producto más no se acaba ahí, pues se seguirán haciendo cambios llamados mantenimientos.



Trigas, M. (2012), Ciclo del desarrollo ágil [Figura]. Recuperado de: <http://hdl.handle.net/10609/17885>

#### 

#### 2.2.3 Componentes de Scrum

En el libro de Trigas (2012) este nos indica que existen 3 fases, las cuales se les llaman reuniones, roles y elementos.

##### Las reuniones

Está comprendida también de 3 fases:

* **Planificación del Backlog.-** Definición de un documento inicial donde se colocarán los requisitos del sistema.
* **Seguimiento del Sprint.**- Son las reuniones diarias.
* **Revisión del sprint.-** Se presentan los resultados finales o una demo de lo logrado.

##### Los roles

Se divide en 2 grupos:

* **Los cerdos.-** Personas comprometidas con el proceso de Scrum, entre los cuales podemos tener al Product Owner, ScrumMaster y al Equipo de desarrollo.
* **Las gallinas.-** No son parte del proceso de Scrum pero son necesarios para la retroalimentación. Estos actores pueden ser los Usuarios, Stakeholders y los Managers.

****

Trigas, M. (2012), Chiste del cerdo y la gallina [Figura]. Recuperado de: <http://hdl.handle.net/10609/17885>

##### 

##### Los elementos de Scrum

* **Product Backlog.-** Las necesidades del cliente.
* **Sprint Backlog.-** Las tareas que se realizarán en el Sprint.
* **Incremento.-** Lo que se logra en un Sprint, la cual es operativa.

### 2.3 Marco Conceptual

##### **JavaScript**

JavaScript se presenta como un lenguaje de programación de alto nivel con una amplia aplicación en el desarrollo web. Su razón de ser radica en infundir vida a las páginas web, posibilitando la creación de contenido altamente interactivo y dinámico. Distinguiendo de otros lenguajes de programación, JavaScript se ejecuta en el lado del cliente, lo que significa que es interpretado por el navegador web del usuario. Este enfoque brinda a los desarrolladores la capacidad de modificar elementos HTML y CSS en tiempo real, reaccionar ante las acciones del usuario, verificar la validez de formularios, administrar efectos de animación y establecer conexiones con servidores remotos mediante solicitudes HTTP o AJAX. En síntesis, JavaScript se erige como un componente fundamental en la edificación de experiencias en línea atractivas y funcionales (Smith, 2023).

##### **PHP**

PHP, que es una abreviatura de "Preprocesador de Hipertexto", se trata de un lenguaje de programación de alto nivel especialmente diseñado para trabajar en el desarrollo web. Su versatilidad y poder radican en su habilidad para crear aplicaciones web que sean interactivas y sitios que sean dinámicos de manera eficaz. PHP funciona en el servidor, lo que significa que todo su trabajo se realiza antes de que veas una página en tu navegador. Esto le permite a los programadores llevar a cabo una amplia variedad de tareas, desde generar contenido web que se actualiza constantemente hasta manejar bases de datos y verificar la identidad de los usuarios.

Lo que hace que PHP se destaque es su capacidad para conectarse con bases de datos como MySQL, lo que facilita mucho la creación de aplicaciones que almacenan y recuperan datos de manera eficiente. Además, PHP se usa ampliamente en la creación de sistemas de gestión de contenidos (CMS) como WordPress y plataformas de comercio electrónico como WooCommerce. Esto lo convierte en una elección muy popular para construir sitios web que sean personalizables y llenos de vida.

PHP ha sido abrazado por la comunidad de desarrolladores web debido a su facilidad de uso y a la gran cantidad de personas que comparten recursos y bibliotecas. Su flexibilidad y capacidad de adaptarse a proyectos de todos los tamaños hacen que PHP sea una herramienta imprescindible en el kit de herramientas de cualquier desarrollador web (Martínez, 2022).

##### **MySQL**

MySQL es un sistema de gestión de bases de datos relacionales (RDBMS) de código abierto ampliamente utilizado en el desarrollo de aplicaciones y la administración de datos. Proporciona una plataforma eficiente para crear, almacenar y recuperar datos en tablas relacionadas de manera organizada y eficaz. MySQL se ha convertido en un pilar en la gestión de datos, desempeñando un papel crucial en la creación de aplicaciones y sistemas que requieren un manejo robusto y escalable de la información (Jhonson, 2021).

##### **HTML**

HTML (Hypertext Markup Language) es un lenguaje de marcado ampliamente utilizado en el desarrollo web para crear y estructurar el contenido de las páginas en la World Wide Web. Este lenguaje utiliza etiquetas y elementos específicos para describir la información contenida en una página web, como texto, imágenes, hipervínculos y otros medios multimedia. HTML se ha convertido en el lenguaje fundamental para la construcción de sitios web, permitiendo a los desarrolladores definir la estructura y el formato de una página, lo que a su vez proporciona la base para su presentación y funcionamiento.

##### **Visual Studio Code**

Visual Studio Code, también conocido como VS Code y desarrollado por Microsoft, se ha destacado como un entorno de desarrollo altamente versátil y personalizable que ha revolucionado la forma en que los programadores escriben y administran su código. Este entorno de desarrollo integrado (IDE) de código abierto se distingue por su ligereza y un rendimiento excepcional, lo que lo convierte en la elección preferida para quienes buscan una experiencia de desarrollo fluida y sin complicaciones. La verdadera potencia de VS Code radica en su amplia variedad de extensiones, que permiten a los usuarios adaptar su entorno a sus necesidades específicas. Desde el resaltado de sintaxis y la sugerencia de código hasta herramientas avanzadas de depuración y colaboración en tiempo real, esta plataforma ha creado un ecosistema dinámico y diversificado que beneficia a programadores en una amplia gama de campos y proyectos. En última instancia, Visual Studio Code ha establecido un nuevo estándar en el ámbito de los entornos de desarrollo integrado, ofreciendo una experiencia altamente adaptable, rápida y eficiente que se adapta a las cambiantes demandas de la programación moderna (Gómez, 2023).

**ChatGPT**

ChatGPT es un modelo de procesamiento de lenguaje natural desarrollado por OpenAI que se apoya en una estructura de redes neuronales profundas denominada GPT (Generative Pre-trained Transformer). Este sistema se somete a un proceso de entrenamiento utilizando extensas colecciones de datos de texto disponibles en línea, y puede generar respuestas coherentes y contextualmente apropiadas a partir de las instrucciones que recibe en forma de texto. ChatGPT se beneficia de su habilidad para identificar patrones y relaciones presentes en los datos de entrenamiento, lo que le confiere la capacidad de comprender y producir texto en diversas situaciones.

El funcionamiento de ChatGPT se basa en técnicas de aprendizaje automático y estadística, aprovechando una arquitectura de redes neuronales profundas con atención de múltiples cabezas para procesar y generar texto. A través de un proceso de ajuste fino y optimización, este modelo se adapta para cumplir con tareas específicas, como la generación de respuestas en conversaciones escritas. Es fundamental destacar que, a pesar de sus notables habilidades, ChatGPT puede no proporcionar respuestas precisas o adecuadas en todo momento, y su utilización debe ser vigilada y evaluada cuidadosamente, especialmente en aplicaciones críticas.

CAPÍTULO 3

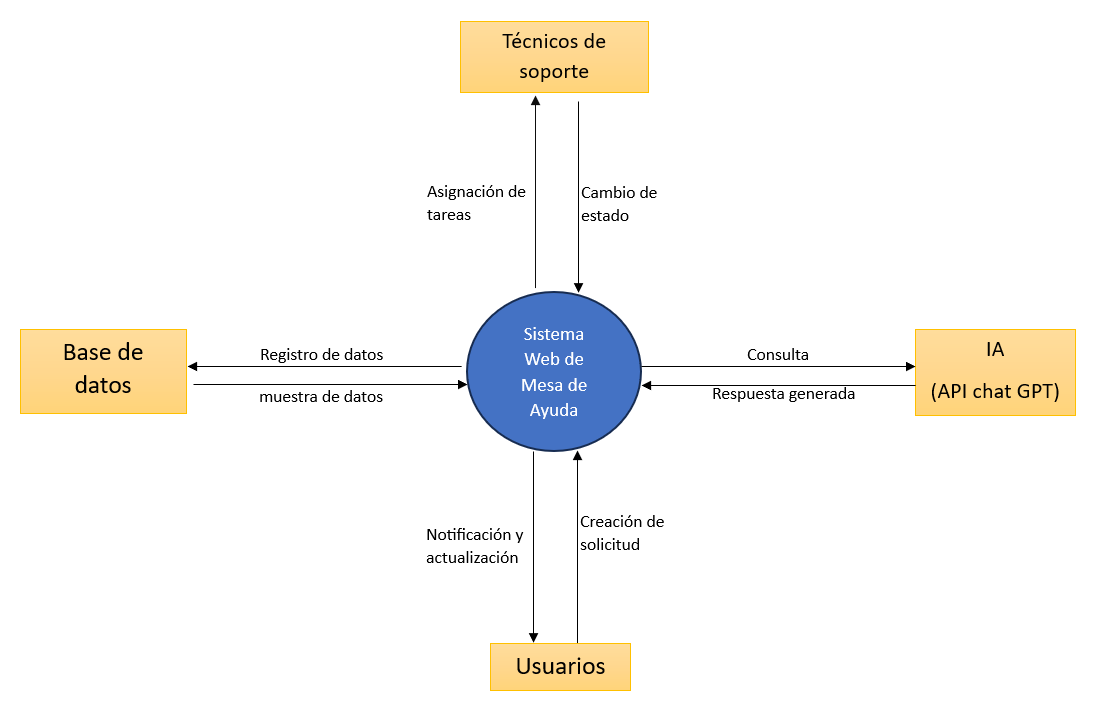
## 3.1 Fases del Scrum

### 3.1.1 Fase Inicial

#### **3.1.1.1 Visión del Proyecto**

El diseño de un sistema web de mesa de ayuda permitirá modernizar y optimizar los procesos de gestión de pedidos de mantenimiento de tecnología garantizando así una atención ágil, efectiva y centrada en las necesidades de los clientes, alineando la entidad financiera con las mejores prácticas tecnológicas y asegurando su competitividad en un entorno cada vez más digitalizado.

#### **3.1.1.2 Diagrama de Contexto**



#### **3.1.1.3 Identificación de Actores**

**Scrum Master**: Axel Lucas Ledesma Vargas

Responsabilidad: Coordinar y liderar el equipo SCRUM, asegurando que se sigan las prácticas y procesos adecuados.

**Stakeholder**: Javier Barrientos

Responsabilidad: Proporcionar la visión y los requisitos del proyecto, así como tomar decisiones estratégicas.

**Equipo de Trabajo**

| **Rol** | **Nombre** | **Área de Responsabilidad** |
| --- | --- | --- |
| Product Owner | Javier Sebastián Sánchez Corro | Definición y priorización de funcionalidades desde la perspectiva del cliente |
| Desarrollador Principal | José Marco Lira Polo | Liderazgo en el desarrollo del sistema web, garantizar la calidad del código |
| Tester | Josué Martín Tupahuaccaillo Sandoval | Realización de pruebas de calidad, asegurar el funcionamiento correcto del sistema |

# 

# 

#### **3.1.1.4 Desarrollo de épicas**

#### 

|  | | | Enunciado de la historia | | | Criterios de aceptación | | | | |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Épica | Identificador (ID) de la historia | Título | Rol | Característica/ Funcionalidad | Razón / Resultado | Número (#) de escenario | Criterio de aceptación (Titulo) | Contexto | Evento | Resultado/ Comportamiento esperado |
| Gestión de clientes | HU1 | Inicio de sesión | Como cliente | Quiero iniciar Sesion | Para acceder al sistema web | 1 | Inicio de sesión en la mesa de ayuda | En caso el usuario ingrese bien las credenciales | Cuando hace clic en el botón iniciar sesión | El usuario puede ingresar a la mesa de ayuda tras introducir correctamente los datos |
| 2 | mensaje de campos vacíos al iniciar sesión | En caso el usuario no ingrese datos en los campos de inicio de sesión | Cuando hace clic en el botón iniciar sesión | El sistema mostrará un mensaje que diga “Los campos estan vacios” |
| 3 | mensaje de datos incorrectos | En caso el usuario ingrese datos incorrectos campos de inicio de sesión | Cuando hace clic en el botón iniciar sesión | El sistema mostrará un mensaje que diga “El usuario y/o contraseña son incorrectos” |
|  |  |  |  |  |  |  |  |  |  |  |
| Gestión de Tickets | HU2 | Nuevo Ticket | Como cliente | Quiero crear un ticket de solicitud | Para mandar solicitud a la mesa de ayuda | 1 | Formulario creación de tickets | El usuario ha iniciado sesión y desea crear un nuevo ticket | El usuario accede a la sección de creación de tickets. | Se muestra un formulario con campos etiquetados para ingresar los datos del ticket |
| 2 | Validar datos | el usuario llenó los datos del formulario | El usuario intenta enviar el formulario | El sistema valida los datos ingresados y muestra mensajes de error claros si algún campo obligatorio se omite o si se ingresan datos incorrectos. |
| 3 | Selección de Categorías | El usuario desea categorizar el ticket | El usuario intenta seleccionar una categoría en el formulario. | El formulario muestra correctamente las categorías disponibles y selecciona una de ellas para el ticket. |
| 4 | Descripción detallada de ticket | El usuario está proporciona detalles sobre el problema | El usuario utiliza el campo de descripción | El campo de descripción permite al usuario ingresar información detallada sobre el problema |
| 5 | Envío de ticket | El usuario ha completado el formulario y está listo para enviar el ticket | El usuario hace clic en el botón de envío en el formulario | El sistema registra el ticket en la base de datos y asigna un número de seguimiento único. El ticket se crea en el sistema |
| 6 | Confirmación de creación | El usuario ha enviado con éxito el formulario de creación de ticket | El sistema procesa el formulario y crea el ticket | El usuario recibe una confirmación visual inmediata de que el ticket se ha creado correctamente, y se muestra el número de seguimiento del ticket para futuras referencias |
|  |  |  |  |  |  |  |  |  |  |  |
| Gestión de Tickets | HU3 | Detalle del ticket | Como cliente | Quiero editar el ticket | para agregar detalles | 1 | ingresar una descripción adicional | El usuario se encuentra en la página de detalles de un ticket | El usuario agrega detalles adicionales al ticket utilizando el campo de texto para descripciones | El usuario puede escribir y editar libremente en el campo de texto, lo que le permite agregar detalles adicionales o comentarios al ticket. |
| 2 | se registra la descripción adicional como un nuevo detalle | El usuario hace clic en el botón "Enviar." | se envía la información ingresada por el usuario al sistema para su procesamiento | La descripción adicional se registra en la base de datos como un nuevo detalle y se registra con una marca de tiempo adecuada. |
| 3 | la lista de detalles de actividad se actualiza automáticamente | El usuario ha registrado una descripción adicional | La lista de detalles se actualiza automáticamente en la interfaz de usuario. | El sistema debe reflejar la descripción adicional. La información debe mostrarse en orden cronológico, con la última actualización en la parte superior de la lista. |
| 4 | Si el campo de descripción está vacío al hacer clic en "Enviar," se muestra una advertencia. | El usuario intenta registrar una actualización del ticket sin ingresar ninguna descripción | El usuario realiza la acción de enviar la actualización sin proporcionar contenido. | El sistema debe mostrar un mensaje de error que indique al usuario que el campo de descripción está vacío. La operación de registro no se lleva a cabo hasta que se proporcione una descripción válida. |
|  |  |  |  |  |  |  |  |  |  |  |
| Gestión de Tickets | HU4 | Consulta de tickets | Como cliente | Quiero ver mis tickets | para ver el estado de los tickets | 1 | Los usuarios pueden acceder a una sección de consulta donde pueden ver una lista de todos los tickets disponibles. | El usuario desea consultar información sobre los tickets de soporte técnico existentes en el sistema. | El usuario hace clic en la opción "Consultar Tickets" o una funcionalidad similar desde el menú o página principal. | El sistema redirige al usuario a una página que muestra una lista de todos los tickets de soporte técnico disponibles. |
| 2 | La lista de tickets debe incluir detalles como número de ticket, categoría, título, estado y fecha de creación. | El sistema ha cargado la lista de tickets disponibles | El usuario observa la lista de tickets en la página | El usuario puede ver una lista de tickets que incluye información como el número de ticket, la categoría, el título del ticket, su estado actual y la fecha de creación. |
| 3 | Los usuarios pueden utilizar una funcionalidad de búsqueda para encontrar tickets específicos por título, categoría o estado. | La lista de tickets es extensa, y el usuario desea encontrar un ticket específico | El usuario utiliza una barra de búsqueda o filtros para ingresar criterios de búsqueda, como título, categoría o estado. | El sistema filtra automáticamente la lista de tickets en función de los criterios de búsqueda proporcionados por el usuario |
|  |  |  |  |  |  |  |  |  |  |  |
| Gestión de Usuarios | HU5 | Consulta de usuarios | Como soporte | Quiero consultar los usuarios activos | para poder visualizar la lista de accesos al HelpDesk | 1 | Los usuarios de soporte pueden acceder a una sección de consulta donde pueden ver una lista de todos los usuarios activos. | El usuario de soporte desea consultar información sobre los usuarios activos en el sistema. | El usuario de soporte hace clic en la opción "Mantenimiento usuario" o una funcionalidad similar desde el menú o página principal. | El sistema redirige al usuario de soporte a una página que muestra una lista de todos los usuario de soporte activos. |
| 2 | La lista de usuarios debe incluir detalles como nombre, apellido, correo, contraseña y rol. | El sistema ha cargado la lista de usuarios activos. | El usuario de soporte observa la lista de usuarios en la página | El usuario de soporte puede ver una lista de usuarios que incluye información como el nombre, apellido, correo, contraseña y rol. |
| 3 | Los usuarios pueden utilizar una funcionalidad de búsqueda para encontrar usuarios específicos por nombre. | La lista de usuarios es extensa y el usuario desea encontrar un usuario específico | El usuario utiliza una barra de búsqueda para ingresar el nombre como criterio de búsqueda. | El sistema filtra automáticamente la lista de tickets en función de los criterios de búsqueda proporcionados por el usuario. |
| Gestión de Usuarios | HU6 | Nuevo usuario | Como soporte | Quiero crear un nuevo usuario | para poder agregar accesos al HelpDesk | 1 | Formulario creación de usuario | El usuario de soporte ha iniciado sesión y desea crear un nuevo usuario | El usuario accede a la sección de mantenimiento de usuarios. | Se muestra un formulario con campos etiquetados para ingresar los datos del usuario |
| 2 | Validar datos | el usuario llenó los datos del formulario | El usuario intenta enviar el formulario | El sistema valida los datos ingresados y muestra mensajes de error claros si algún campo obligatorio se omite o si se ingresan datos incorrectos. |
| 3 | Confirmación de creación | El usuario de soporte ha enviado con éxito el formulario de creación del usuario | El sistema procesa el formulario y crea el usuario | El usuario de soporte recibe una confirmación visual inmediata de que el usuario se ha creado correctamente. |
| Gestión de Usuarios | HU7 | Modificar usuario | Como soporte | Quiero modificar un usuario | para poder modificar las credenciales de un usuario | 1 | Formulario edicion de usuario | El usuario de soporte ha iniciado sesión y desea crear un nuevo usuario | El usuario accede a la sección de mantenimiento de usuarios. | Se muestra un formulario con campos etiquetados para ingresar los datos del usuario |
| 2 | Validar datos | el usuario llenó los datos del formulario | El usuario intenta enviar el formulario | El sistema valida los datos ingresados y muestra mensajes de error claros si algún campo obligatorio se omite o si se ingresan datos incorrectos. |
| 3 | Confirmación de edición | El usuario de soporte ha enviado con éxito el formulario de edición del usuario | El sistema procesa el formulario y edita el usuario | El usuario de soporte recibe una confirmación visual inmediata de que el usuario se ha editado correctamente. |
| Gestión de Usuarios | HU8 | Eliminar usuario | Como soporte | Quiero eliminar un usuario | para poder modificar desactivar las credenciales de un usuario | 1 | Aviso de Eliminación | El usuario de soporte eligió un usuario y apretó el botón eliminar. | El sistema envía un mensaje emergente al usuario. | El sistema solicita la confirmación del usuario de soporte para la eliminación del usuario seleccionado. |
| 2 | Confirmación de eliminación | El usuario de soporte ha eliminado con éxito el usuario | El sistema elimina el usuario | El usuario de soporte recibe una confirmación visual inmediata de que el usuario se ha eliminado correctamente. |
| Gestion de Usuarios | HU9 | Encriptación de clave desde cuenta de soporte | Como soporte | Cifrado de clave | para poder ocultar la clave y aumentar de la seguridad de la entidad bancaria | 1 | Implementación Exitosa de MD5 | El equipo de desarrollo de seguridad está trabajando en la implementación de la encriptación de claves utilizando el algoritmo MD5. | El equipo completa la implementación de la encriptación de claves con el algoritmo MD5 | Todas las claves almacenadas en la aplicación ahora están encriptadas de manera segura mediante el algoritmo MD5, lo que protege los datos sensibles. Los intentos de acceso no autorizado son ineficaces. |
| Gestion de Usuarios | HU9 | Encriptación de clave desde cuenta de soporte | Como soporte | Cifrado de clave | para poder ocultar la clave y aumentar de la seguridad de la entidad bancaria | 2 | Prueba Exitosa con MD5 | Después de la implementación, se realizan pruebas exhaustivas en la aplicación para verificar la efectividad de la encriptación con MD5 | Las pruebas de seguridad demuestran que las claves encriptadas con MD5 son resistentes a los intentos de acceso no autorizado. | La encriptación de claves con el algoritmo MD5 es exitosa, y la aplicación pasa con éxito todas las pruebas de seguridad relacionadas con la encriptación. |
| Gestion de Usuarios | HU9 | Encriptación de clave desde cuenta de soporte | Como soporte | Cifrado de clave | para poder ocultar la clave y aumentar de la seguridad de la entidad bancaria | 3 | Falla en la Implementación con MD5 | Durante la implementación de la encriptación de claves con MD5, el equipo de desarrollo encuentra desafíos técnicos. | La implementación con MD5 no se completa con éxito debido a problemas técnicos o errores. | Las claves siguen sin estar encriptadas de manera segura con MD5, lo que representa un riesgo de seguridad. El equipo debe abordar y solucionar los problemas técnicos antes de proceder. |
| Gestion de Usuarios | HU9 | Encriptación de clave desde cuenta de soporte | Como soporte | Cifrado de clave | para poder ocultar la clave y aumentar de la seguridad de la entidad bancaria | 4 | Brecha de Seguridad Detectada con MD5 | Después de la implementación de la encriptación con MD5, la aplicación experimenta un intento de acceso no autorizado. | Se detecta un intento de acceso no autorizado a la aplicación. | La encriptación de claves con el algoritmo MD5 debería ser lo suficientemente fuerte como para prevenir con éxito el acceso no autorizado. La aplicación debe registrar el intento y bloquear al atacante. |

#### **3.1.1.5 Listas de pendientes**

| **Épica** | **Identificador (ID) de la historia** | **Título** |
| --- | --- | --- |
| Gestión de clientes | HU1 | Inicio de sesión |
| Gestión de Tickets | HU2 | Nuevo Ticket |
| HU3 | Detalle del ticket |
| HU4 | Consulta de tickets |
| Gestión de Usuarios | HU5 | Consulta de usuarios |
| HU6 | Nuevo usuario |
| HU7 | Modificar usuario |
| HU8 | Eliminar usuario |
| Gestión de Usuarios | HU9 | Encriptación de clave desde cuenta de soporte |

#### **3.1.1.6 Plan de lanzamiento**

| **Sprint** | **Duración** | **Épicas** | **Historias de Usuario** |
| --- | --- | --- | --- |
| Sprint 1 | 3 semanas | Gestión de clientes, Gestión de Tickets | HU1: Inicio de sesión, HU2: Nuevo Ticket, HU3: Detalle del ticket |
| Sprint 2 | 3 semanas | Gestión de Tickets, Gestión de Usuario | HU4: Consulta de tickets, HU5: Consulta de usuarios, HU6: Nuevo usuario |
| Sprint 3 | 2 semanas | Gestión de Usuario | HU7: Modificar usuario, HU8: Eliminar usuario, HU9 Encriptación de clave desde cuenta de soporte |

#### **3.1.1.7 Repositorio del Proyecto**

El repositorio del proyecto es el centro de colaboración y desarrollo para el software en cuestión. Todo el código fuente, y recursos relacionados se encuentran disponibles en el enlace de GitHub:

<https://github.com/JoseLiraP123/ProyectoIntegrador2>

El repositorio se organiza en varias carpetas clave:

**Código Fuente:** En la carpeta principal, se encuentra el código fuente del software, estructurado y comentado para facilitar su comprensión y desarrollo continuo.

**Issues y Problemas:** Utilizamos la función de "Issues" en GitHub para rastrear problemas, errores y tareas pendientes.

**Branches y Versiones:** Mantenemos diferentes ramas (branches) para el desarrollo y versiones estables del software, lo que permite un seguimiento ordenado del progreso y la gestión de versiones anteriores.

#### **3.1.1.8 Producto Backlog priorizado**

| **Prioridad** | **Épica** | **Identificador (ID) de la historia** | **Título** | **Descripción detallada de la historia** | **Puntos de historia** | **Dependencias** |
| --- | --- | --- | --- | --- | --- | --- |
| Alta | Gestión de clientes | HU1 | Inicio de sesión | Los usuarios deben poder iniciar sesión en la aplicación. | 5 | - |
| Alta | Gestión de Tickets | HU2 | Nuevo Ticket | Los usuarios deben poder crear un nuevo ticket de soporte. | 8 | - |
| Alta | Gestión de Tickets | HU3 | Detalle del ticket | Los usuarios deben poder ver los detalles de un ticket. | 5 | HU2 |
| Alta | Gestión de Tickets | HU4 | Consulta de tickets | Los usuarios deben poder consultar los tickets existentes. | 8 | - |
| Alta | Gestión de Usuario | HU5 | Consulta de usuarios | Los usuarios deben poder consultar la lista de usuarios. | 5 | - |
| Alta | Gestión de Usuario | HU6 | Nuevo usuario | Los usuarios deben poder agregar un nuevo usuario. | 8 | - |
| Alta | Gestión de Usuario | HU7 | Modificar usuario | Los usuarios deben poder modificar la información del usuario. | 5 | HU5, HU6 |
| Alta | Gestión de Usuario | HU8 | Eliminar usuario | Los usuarios deben poder eliminar un usuario existente. | 8 | HU5, HU6 |
| Alta | Gestion de Usuario | HU9 | Encriptación de clave desde cuenta soporte | Se aplica el cifrado MD5 para el aumento de la seguridad de la pagina | 7 | HU6,HU7 |

#### **3.1.1.9 Planificación de la iteración (Sprint).**

| **Fecha** | **Historia de Usuario** | **Tareas de Desarrollo** | **Responsable** | **Estado** |
| --- | --- | --- | --- | --- |
| Semana 4 | HU2: Nuevo Ticket | Diseño de interfaz de usuario | Jose Lira | Terminado |
|  |  | Desarrollo del formulario | Jose Lira | Terminado |
| Semana 4 | HU2: Nuevo Ticket | Implementación de la lógica de negocio | Javier Sanchez | Terminado |
| Semana 4 | HU2: Nuevo Ticket | Pruebas de unidad | Axel Ledesma | Terminado |
| Semana 4 | HU4: Consulta de tickets | Diseño de la página de consulta | Jose Lira | Terminado |
|  |  | Desarrollo de la funcionalidad | Jose Lira | Terminado |
| Semana 5 | HU4: Consulta de tickets | Implementación de la lógica de consulta | Javier Sanchez | Terminado |
| Semana 5 | HU2: Nuevo Ticket | Pruebas de integración y correcciones | Axel Ledesma | Terminado |
| Semana 6 | HU4: Consulta de tickets | Pruebas de integración y correcciones | Josue Tupahuacaillo | Terminado |
| Semana 6 | Revisión | Preparación para la revisión del sprint | Equipo | Terminado |
|  | Retrospectiva | Realización de la retrospectiva | Equipo | Terminado |
| Semana 7 | HU5: Consulta de usuarios | Diseño de la interfaz de usuario | Javier Sanchez | Terminado |
|  |  | Desarrollo de la funcionalidad | Javier Sanchez | Terminado |
| Semana 7 | HU5: Consulta de usuarios | Implementación de la lógica de consulta | Jose Lira | Terminado |
| Semana 7 | HU5: Consulta de usuarios | Pruebas de unidad | Axel Ledesma | Terminado |
| Semana 7 | HU6: Nuevo usuario | Diseño de la interfaz de usuario | Javier Sanchez | Terminado |
|  |  | Desarrollo del formulario | Javier Sanchez | Terminado |
| Semana 8 | HU6: Nuevo usuario | Implementación de la lógica de negocio | Jose Lira | Terminado |
| Semana 8 | HU5: Consulta de usuarios | Pruebas de integración y correcciones | Axel Ledesma | Terminado |
| Semana 8 | HU6: Nuevo usuario | Pruebas de integración y correcciones | Josue Tupahuacaillo | Terminado |
| Semana 8 | Revisión | Preparación para la revisión del sprint | Equipo | Terminado |
|  | Retrospectiva | Realización de la retrospectiva | Equipo | Terminado |
| Semana 8 | HU7: Modificar usuario | Diseño de la interfaz de usuario | Jose Lira | Terminado |
|  |  | Desarrollo de la funcionalidad | Jose Lira | Terminado |
| Semana 8 | HU7: Modificar usuario | Implementación de la lógica de modificación | Javier Sanchez | Terminado |
| Semana 8 | HU7: Modificar usuario | Pruebas de unidad | Axel Ledesma | Terminado |
| Semana 8 | HU8: Eliminar usuario | Diseño de la interfaz de usuario | Jose Lira | Terminado |
|  |  | Desarrollo de la funcionalidad | Jose Lira | Terminado |
| Semana 8 | HU8: Eliminar usuario | Implementación de la lógica de eliminación | Javier Sanchez | Terminado |
| Semana 8 | HU7: Modificar usuario | Pruebas de integración y correcciones | Axel Ledesma | Terminado |
| Semana 8 | HU8: Eliminar usuario | Pruebas de integración y correcciones | Josue Tupahuacaillo | Terminado |
| Semana 8 | Revisión | Preparación para la revisión del sprint | Equipo | Terminado |
|  | Retrospectiva | Realización de la retrospectiva | Equipo | Terminado |
| Semana 10 | HU9 Encriptación desde cuenta de soporte | Implementacion del cifrado MD5 | Javier Sanchez | Terminado |
| Semana 11 |  | Implementación del dashboard para las cuentas de usuario y soporte | Jose Lira | Terminado |
| Semana 11 – ?? | Búsqueda de Nuevas Tareas |  |  | Pendiente |

#### **3.1.1.10 Sprint**

**Sprint 1**

**El Sprint 1 está conformado por la HU1, HU2 y HU3.**

HU1: Inicio de sesión

Descripción: Yo, como desarrollador, quiero implementar una función de inicio de sesión en la aplicación para permitir que los usuarios se autentiquen de manera segura. Esto incluirá la capacidad de ingresar un nombre de usuario y una contraseña válidos para acceder al sistema.

HU2: Nuevo Ticket

Descripción: Yo, como desarrollador, quiero desarrollar la funcionalidad "Nuevo Ticket" para permitir que los usuarios creen un registro de ticket en el sistema. Esto implica proporcionar un formulario donde los usuarios puedan ingresar información relevante, como el título, la descripción y la prioridad del ticket.

HU3: Detalle del ticket

Descripción: Yo, como desarrollador, quiero crear la funcionalidad "Detalle del ticket" para que los usuarios puedan ver información detallada sobre un ticket existente en el sistema. Esto incluirá la visualización de todos los datos asociados a un ticket, como su estado, la descripción completa, la fecha de creación y cualquier comentario relacionado.

**Sprint 2**

**El Sprint 2 está conformado por la HU4, HU5 y HU6.**

HU4: Consulta de tickets

Descripción: Yo, como desarrollador, quiero implementar la funcionalidad de "Consulta de tickets" que permita a los usuarios buscar y ver información detallada sobre los tickets existentes en el sistema. Esto incluirá opciones de búsqueda y filtros para facilitar la consulta.

HU5: Consulta de usuarios

Descripción: Yo, como desarrollador, quiero desarrollar la funcionalidad de "Consulta de usuarios" para permitir a los administradores y usuarios autorizados buscar y acceder a la información de otros usuarios registrados en el sistema. Esto incluirá la visualización de detalles de usuario y opciones de búsqueda.

HU6: Nuevo usuario

Descripción: Yo, como desarrollador, quiero crear la funcionalidad "Nuevo usuario" para permitir a los administradores y usuarios registrados agregar nuevos usuarios al sistema. Esto incluirá la capacidad de ingresar información de usuario, como nombre, correo electrónico y contraseña, para la creación de cuentas.

**Sprint 3**

**El Sprint 3 está conformado por la HU7 y HU8.**

HU7: Modificar usuario

Descripción: Yo, como desarrollador, quiero implementar la funcionalidad de "Modificar usuario" para permitir a los administradores y usuarios autorizados realizar cambios en la información de un usuario existente. Esto incluirá la capacidad de actualizar datos como el nombre, el correo electrónico y la contraseña.

HU8: Eliminar usuario

Descripción: Yo, como desarrollador, quiero desarrollar la funcionalidad de "Eliminar usuario" para permitir a los administradores y usuarios autorizados eliminar cuentas de usuario del sistema de manera segura. Esto incluirá un proceso de confirmación antes de la eliminación.

**Sprint 4**

El Sprint 4 está conformado por la HU9.

Descripción:

Yo, como desarrollador, quiero implementar la funcionalidad de "Encriptación de clave desde la cuenta de soporte" para permitir a los administradores de soporte técnico mejorar la seguridad de las claves almacenadas en la aplicación. Esto implica la capacidad de encriptar claves de usuario directamente desde una cuenta de soporte técnico. Esta funcionalidad debe ser utilizada con precaución y solo por personal autorizado.

Criterios de Aceptación:

Los administradores de soporte técnico pueden acceder a la funcionalidad de "Encriptación de clave" desde sus cuentas.

La funcionalidad de encriptación de clave permite a los administradores de soporte técnico seleccionar una clave de usuario y encriptarla de manera segura.

Se registra una auditoría de todas las encriptaciones de claves realizadas desde cuentas de soporte técnico.

La encriptación de clave es exitosa y protege de manera segura las claves de usuario.

Solo el perso**nal autorizado tiene acceso a esta funcionalidad desde cuentas de soporte técnico.**

#### **3.1.1.11 Sprint Backlog**

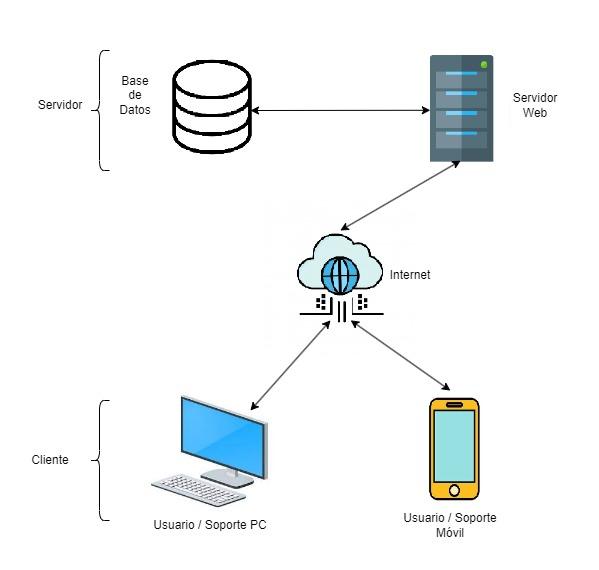
| **Historia de Usuario** | **Tareas de Desarrollo** | **Responsable** | **Estado** |
| --- | --- | --- | --- |
| HU2: Nuevo Ticket | Diseño de interfaz de usuario | Jose Lira | Terminado |
| Desarrollo del formulario | Jose Lira | Terminado |
| Implementación de la lógica de negocio | Javier Sanchez | Terminado |
| Pruebas de unidad | Axel Ledesma | Terminado |
| HU4: Consulta de tickets | Diseño de la página de consulta | Jose Lira | Terminado |
| Desarrollo de la funcionalidad | Jose Lira | Terminado |
| Implementación de la lógica de consulta | Javier Sanchez | Terminado |
| Pruebas de integración y correcciones | Josue Tupahuacaillo | Terminado |
| HU5: Consulta de usuarios | Diseño de la interfaz de usuario | Javier Sanchez | Terminado |
| Desarrollo de la funcionalidad | Javier Sanchez | Terminado |
| Implementación de la lógica de consulta | Jose Lira | Terminado |
| Pruebas de unidad | Axel Ledesma | Terminado |
| HU6: Nuevo usuario | Diseño de la interfaz de usuario | Javier Sanchez | Terminado |
| Desarrollo del formulario | Javier Sanchez | Terminado |
| Implementación de la lógica de negocio | Jose Lira | Terminado |
| Pruebas de integración y correcciones | Josue Tupahuacaillo | Terminado |
| HU7: Modificar usuario | Diseño de la interfaz de usuario | Jose Lira | Terminado |
| Desarrollo de la funcionalidad | Jose Lira | Terminado |
| Implementación de la lógica de modificación | Javier Sanchez | Terminado |
| Pruebas de unidad | Axel Ledesma | Terminado |
| HU8: Eliminar usuario | Diseño de la interfaz de usuario | Jose Lira | Terminado |
| Desarrollo de la funcionalidad | Jose Lira | Terminado |
| Implementación de la lógica de eliminación | Javier Sanchez | Terminado |

#### **3.1.1.12 Mínimo Producto Viable**

| **Historia de Usuario** | **Descripción** |
| --- | --- |
| Iniciar Sesión | Como usuario, quiero poder iniciar sesión en el sistema para acceder a las funcionalidades. |
| Crear Nuevo Ticket | Como usuario, quiero poder crear un nuevo ticket para solicitar asistencia o informar un problema. |
| Detalles del Ticket | Como usuario, quiero poder consultar los detalles de un ticket existente para obtener información sobre el estado y la resolución. |
| Consulta de Tickets | Como usuario, quiero poder buscar y consultar la lista de tickets existentes para obtener una visión general. |
| Consulta de Usuarios | Como equipo de soporte, quiero poder consultar la lista de usuarios para ver quiénes están registrados en el sistema. |
| Agregar Usuario | Como equipo de soporte, quiero poder agregar nuevos usuarios al sistema. |
| Modificar Usuario | Como equipo de soporte, quiero poder modificar la información de un usuario existente. |
| Eliminar Usuario | Como equipo de soporte, quiero poder eliminar un usuario |

## **3.2. Arquitectura tecnológica**

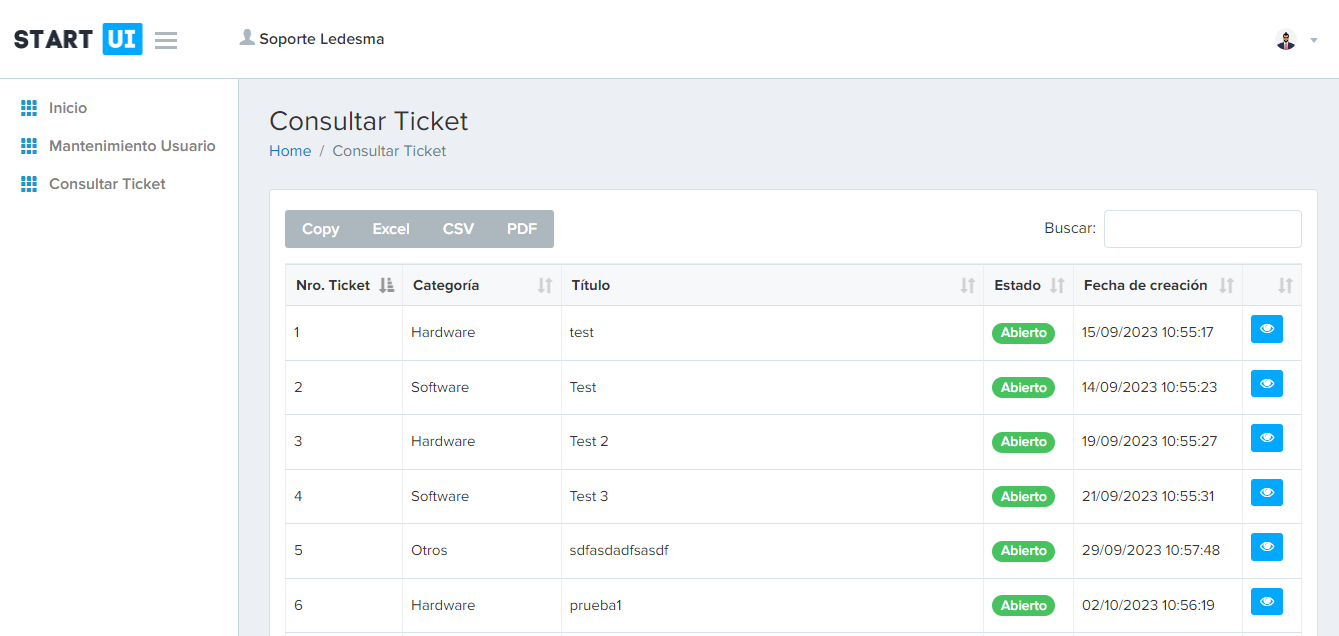
Una arquitectura tecnológica en un proyecto de desarrollo de software es el diseño estructural que define cómo los componentes del sistema interactúan entre sí y con el entorno. Esta arquitectura establece las pautas para la organización, comunicación y funcionamiento de los elementos tecnológicos, garantizando la eficiencia, escalabilidad y robustez del software, así como la adaptación a los requisitos del negocio y a futuras expansiones o cambios. Considerando esta definición, el sistema estará desarrollado con una arquitectura Cliente –Servidor, ya que al tener múltiples usuarios (Usuarios y Soportes) entramos en la necesidad de utilizar una arquitectura una arquitectura que posibilite la separación de la interfaz gráfica y los datos.. A continuación,se muestra la arquitectura expresada en un gráfico:



## 

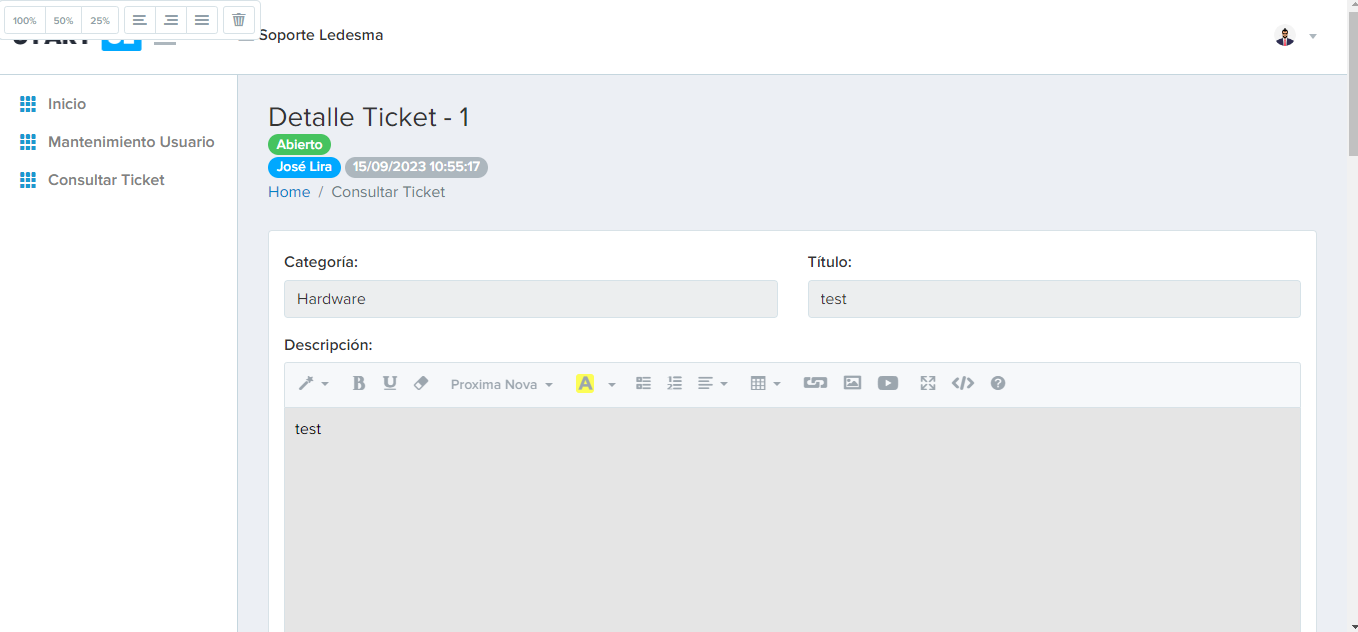
## **3.3. Mockups del diseño**

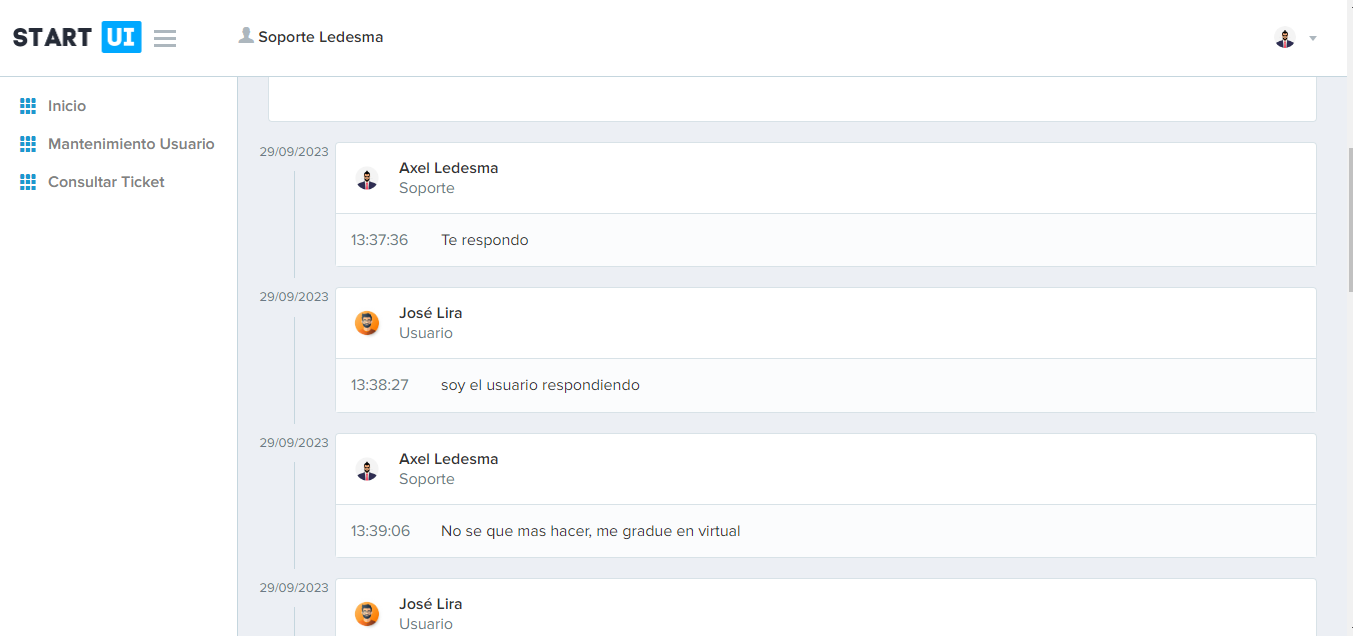
**Diseño consulta de ticket**



La siguiente pantalla muestra la consulta de tickets desde una perspectiva de rol Soporte donde se visualizan los filtros, ordenamientos y la opción de búsqueda.

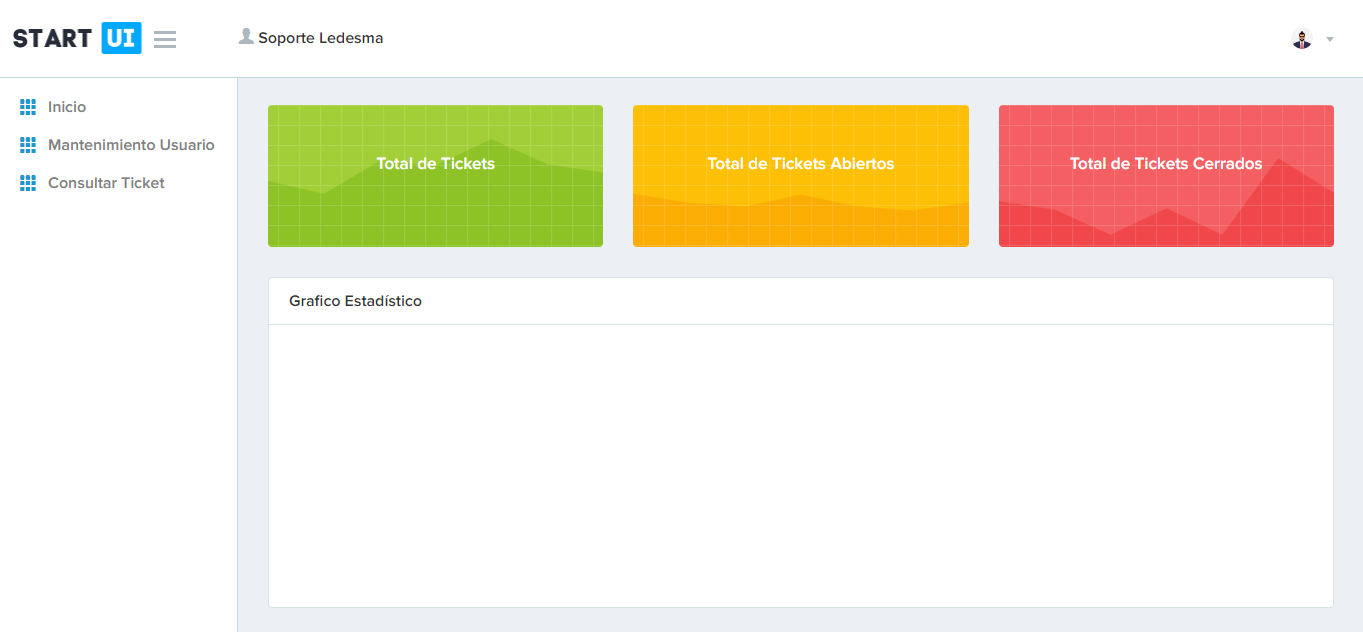
**Diseño detalle de ticket**

****

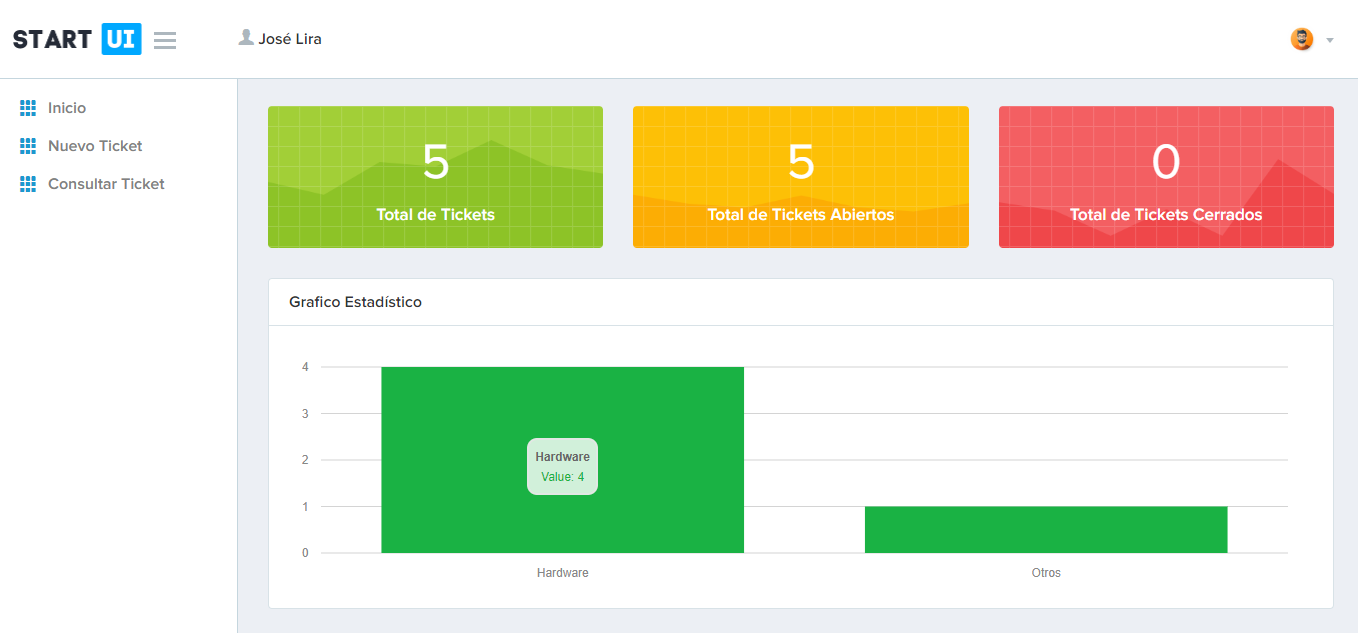
****

La siguiente imagen muestra las pantallas de detalles de ticket donde observamos la información sobre aquel ticket y el seguimiento de las respuestas tanto del usuario como los usuario de soporte.

**Diseño Interfaz Principal**

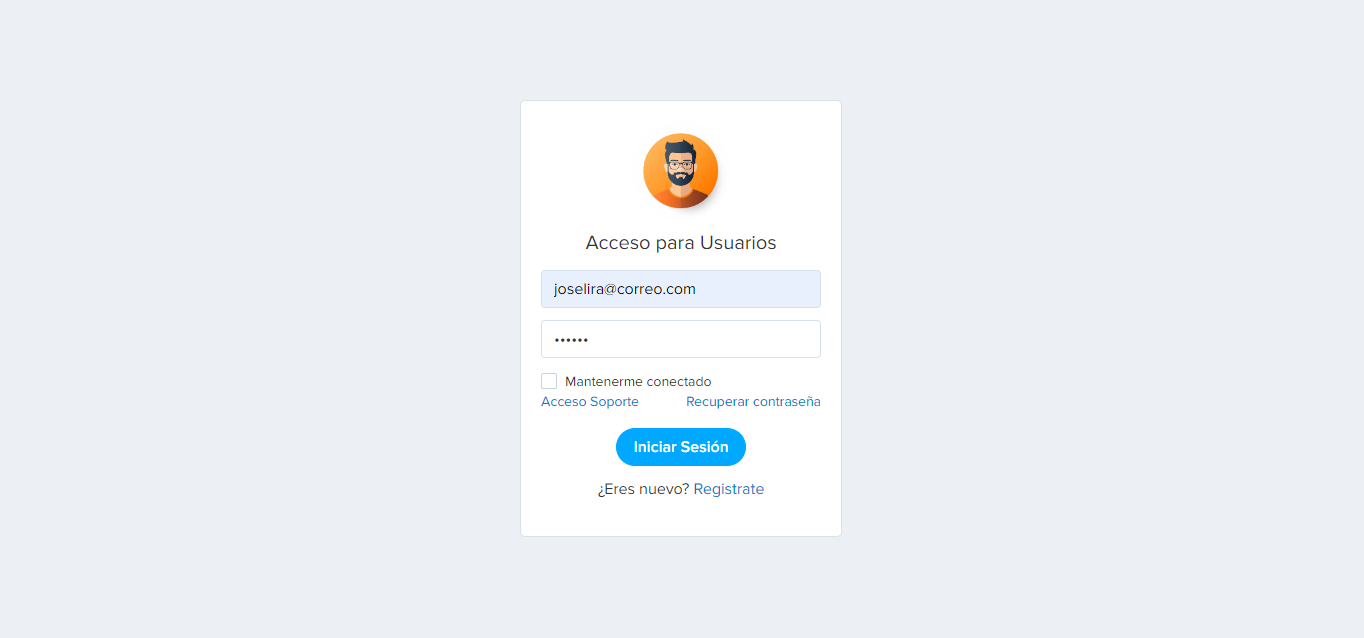
****

Esta pantalla representa la gráfica que tiene el usuario de rol de soporte en la interfaz principal evidenciando que está en proceso el desarrollo de esta funcionalidad.

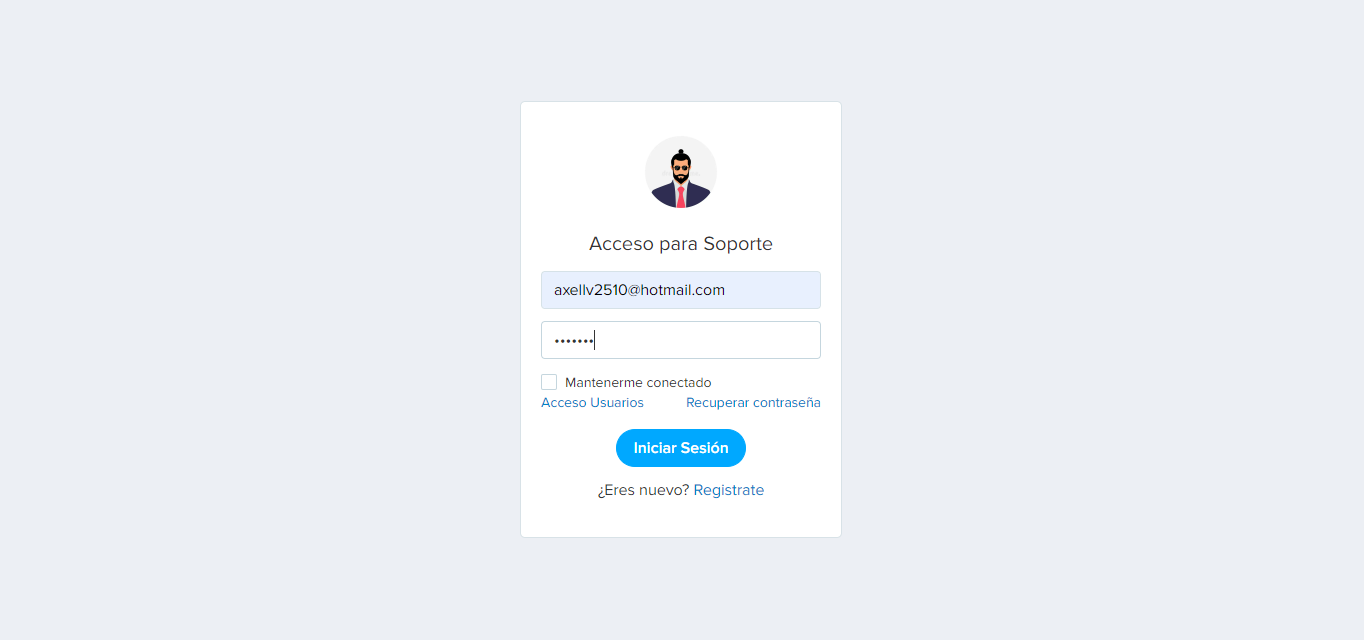
****

Esta pantalla representa la gráfica que tienen los usuarios en la interfaz principal.

**Diseño Login**

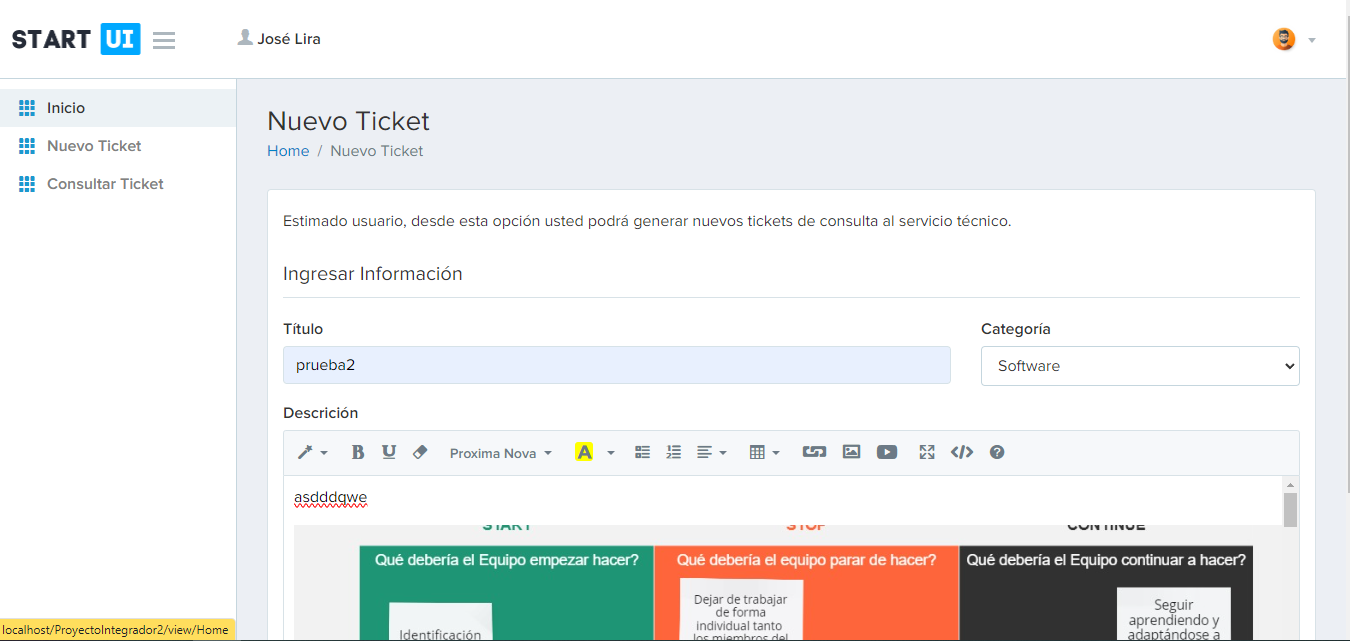
****

La siguiente pantalla muestra el login de acceso para los usuarios.

****

La siguiente pantalla muestra el login de acceso para los usuarios de soporte.

**Diseño Nuevo Ticket**

****

La pantalla muestra el formulario de creación de un nuevo ticket donde evidenciamos funcionalidades como la posibilidad de agregar imágenes en la descripción del ticket..

## **3.4 Funcionalidades del código**

1. Crear la base de datos relacional para el almacenamiento de los usuarios y la solicitud de tickets estableciendo la prioridad de atención a las incidencias de forma automatizada a través de la categoría seleccionada por el usuario que registra la solicitud.

Para este cumplir este objetivo, necesitamos de una clase llamado Conectar con la siguiente estructura:



En dicho código definimos los parámetros de conexión a la base de datos mysql y capturamos los posibles errores durante la conexión para mostrárselo al usuario.

1. Generar notificaciones por correo electrónico a los usuarios y al equipo de soporte cada vez que se genere un nuevo ticket o se actualice un ticket existente. Además, el equipo de soporte también recibirá notificaciones por WhatsApp al momento de crear, asignar y cerrar un ticket.

Esta funcionalidad aún no se ha desarrollado.

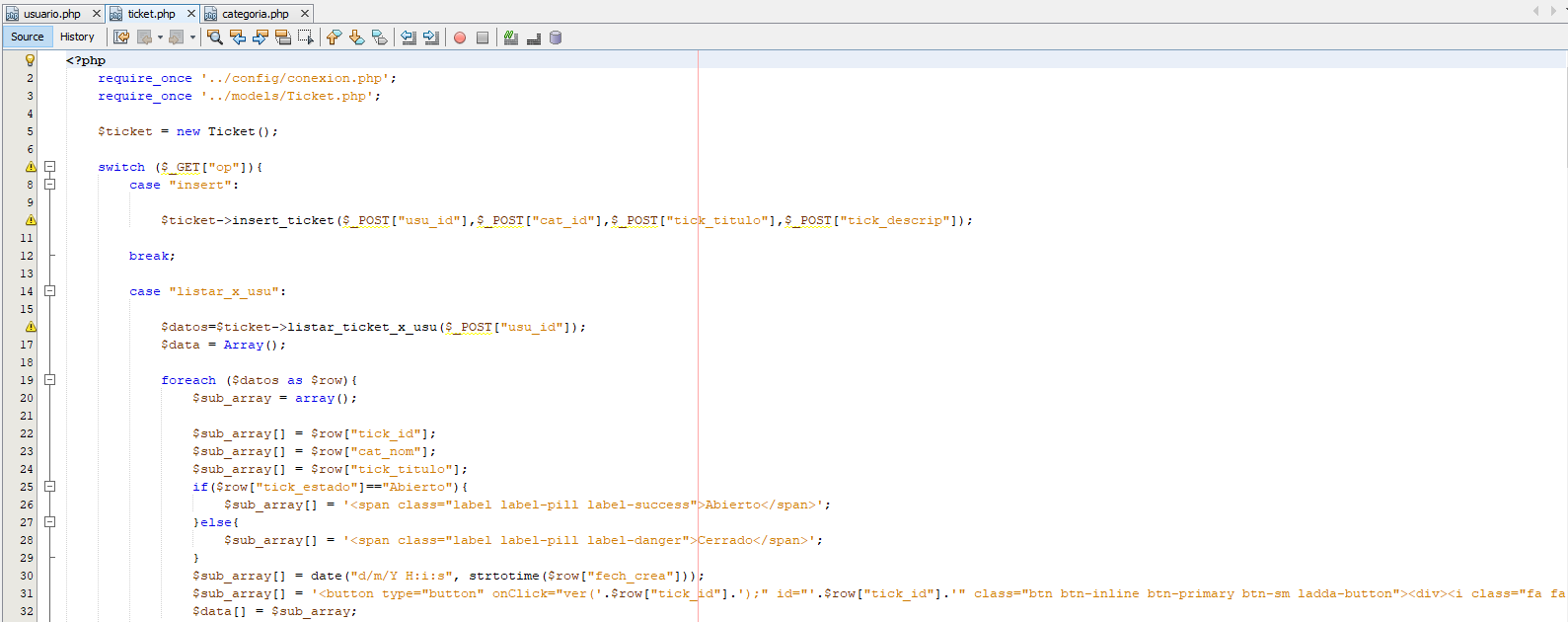
1. Desarrollar una API basada en inteligencia artificial que habilite al equipo de soporte de la mesa de ayuda para responder a los tickets, utilizando las respuestas generadas por el modelo IA ChatGPT como base preliminar.

Esta funcionalidad aún no se ha desarrollado.

1. Generar reportes del estado y categoría de los tickets generados.

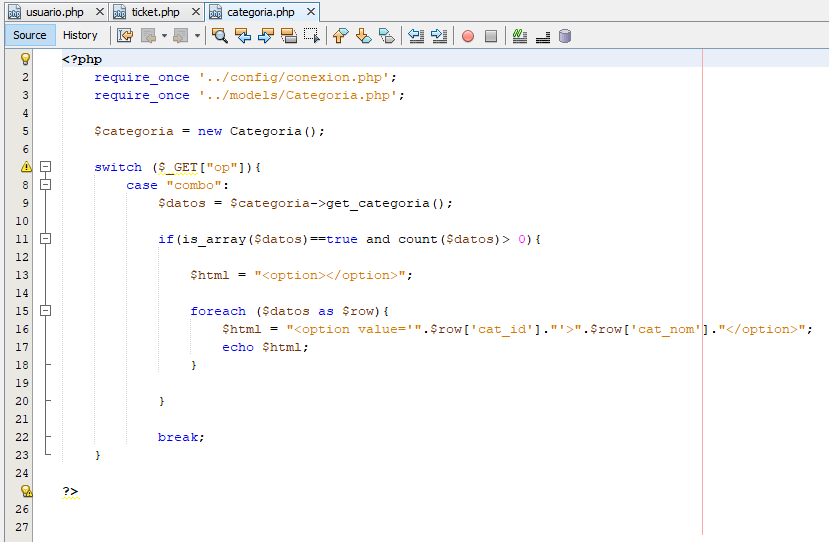
Para generar reportes del estado de los tickets y para lograr listar las categorías utilizamos los siguientes controladores y modelos:

* Controlador Ticket



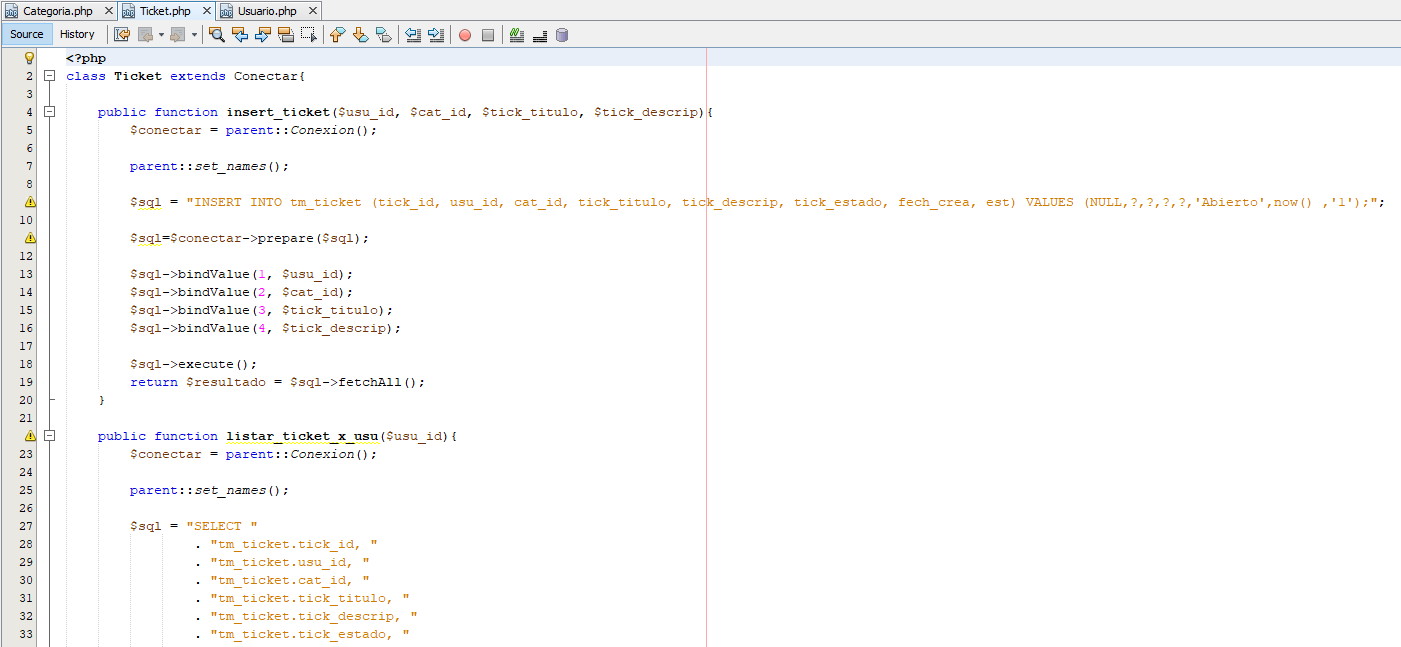
Esta funcionalidad permite que los tickets se puedan generar, consultar o eliminar además de obtener un listado filtrado por usuario.

* Controlador Categoría



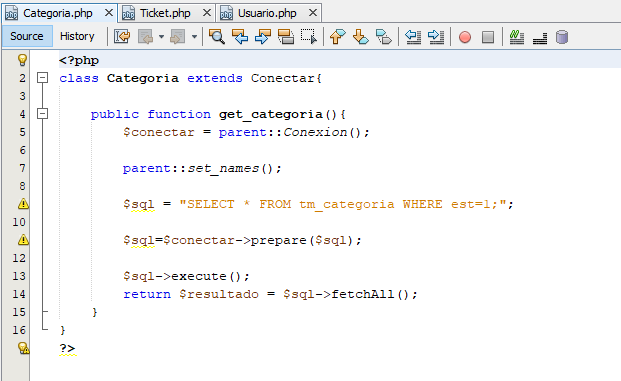
Esta funcionalidad permite cargar la lista de categorías que se pueden elegir para crear un nuevo ticket.

-Modelo Ticket:



La siguiente imagen forma parte del Modelo Ticket donde se muestra la estructura de la función para insertar ticket y listar ticket por usuario

-Modelo Categoría:



Esta imagen muestra la funciona para obtener las categorías que se listaran cuando se busque crear un nuevo ticket

Por otro lado, para la funcionalidad que permite listar las categorías en la creación de un nuevo ticket está conformado por las siguientes líneas de código:



En el php de la página de nuevo ticket, el código sombreado presenta la funcionalidad de cargar la lista de categorias en un combobox del formulario de nuevo ticket



En el js, el código sombreado permite acceder al controlador categoría y eligiendo la opción “combo” logramos obtener la data que necesitamos para listarlo en el combobox.

Por otro lado, para la funcionalidad de obtener los gráficos de los total de ticket, los tickets abiertos y cerrados tenemos las siguientes partes de código:



En el php, la parte del código que da forma a los gráficos se muestra en la imagen anterior.



En el js, podemos observar como hace uso del controlador usuario para obtener la data necesaria para realizar el gráfico desde el rol de Usuario (rol id = 1)



Por otro lado, para el rol de soporte, la funcionalidad aún no está terminada para graficar los reportes de total de ticket, total de ticket abiertos y cerrados.

# CONCLUSIONES

# RECOMENDACIONES

# ANEXOS

# BIBLIOGRAFÍA

* Narayana Reddy, T., & Pushpa Latha, K. (2019). Impact of Banc Assurance on Customers in Purchasing Insurance Policies with Special Reference to Kurnool District of Andhra Pradesh. International Journal of Innovative Technology and Exploring Engineering (IJITEE), DOI: 10.35940/ijitee.F1144.0486S419.
* Sandhu, S., & Arora, S. (2022). Customers' usage behavior of e-banking services: Interplay of electronic banking and traditional banking. International Journal of Finance and Economics, DOI: 10.1002/ijfe.2266.
* Quispe Casafranca, C., Requena Yañez, E., & Soto Costos, J. M. (2022). Sistema web para la gestión de incidencias de la mesa de ayuda de la empresa nexus technology.
* Arbildo Flores, J. P., Quiroz Arista, J. D. (2018). IMPLANTACIÓN DE APLICATIVO WEB BASADA EN ITIL Y MEJORAMIENTO DE LA GESTIÓN DE MESA DE AYUDA EN EL GORE LORETO 2017.
* Rodriguez, C., & Pino, R. (2019). Chatbot con habilidades sociales para interactuar con Usuarios del Servicio de mesa de ayuda que brinda la empresa "Hitss Ecuador." Universidad de Guayaquil. Facultad de Ciencias Matemáticas y Físicas.
* Faria, D., & Freire, L. (2021). Implementación de un sistema de mesa de ayuda informático Help Desk para la gestión de requerimientos que se presentan en un SOC. ESPOL. FIEC.
* Arroyo, V., & Guizado, J. (2021). Sistema Web para el Proceso de Mesa de Ayuda en la Sub Dirección de Tecnología de la Información de la Superintendencia Nacional de Educación Superior Universitaria.
* Zurita, B. (2020). SISTEMA WEB PARA LA GESTIÓN ACADÉMICA Y ADMINISTRATIVA DE EMPRESA DE CAPACITACIÓN PROFESIONAL DIENAV. Universidad Tecnológica Israel. Recuperado de <https://repositorio.uisrael.edu.ec/handle/47000/2489>
* Pedraza, H. (2021). Implementación de una mesa de ayuda para optimizar la gestión de incidencias para el área de informática en la empresa CIRTEXTILES. Universidad Tecnológica del Perú. Recuperado de <https://hdl.handle.net/20.500.12867/4523>
* Trigas, M. (2012). Metodología Scrum. Universitat Oberta de Catalunya. Recuperado de <http://hdl.handle.net/10609/17885>