**UNIVERSIDAD NACIONAL MAYOR DE SAN MARCOS**

UNIVERSIDAD DEL PERÚ, DECANA DE AMÉRICA



**SGI - Documento de Arquitectura de Software Actualización**

**Curso:** Gestión de la Configuración

**Integrantes (Grupo N° 1):**

* Balarezo Ramos Luis Jesus
* Balceda Delgado Mariana Alejandra
* Canecillas Contreras Juan Mariano
* Del Aguila Febres Brayan Tadeo
* Durand Caracuzma Marlon Milko
* Huarhua Piñas Edson Sebastián
* Justiniano Quispe Diego André
* Soller Barrenechea Carlos Javier

**SISTEMA DE GESTIÓN DE INCIDENCIAS**

**Documento de Arquitectura de Software**

Versión 1.0

**Lima, Junio del 2023**

**TABLA DE CONTENIDOS**

[**I. Introducción 4**](#_yowyc1xmz3wn)

[**II. Visión general de la arquitectura 4**](#_e46ijt3tljcf)

[**III. Arquitectura de alto nivel 6**](#_mg9fvb4k5upu)

[**IV. Detalles de la arquitectura 8**](#_djg4u4h5a3w6)

[**V. Patrones de diseño y principios arquitectónicos 17**](#_aaeom07qkzj7)

[**VI. Consideraciones de rendimiento y escalabilidad 18**](#_c361k2skl6q2)

[**VII. Consideraciones de seguridad 19**](#_ewawu1d8s9b2)

[**VIII. Consideraciones de despliegue y operación 21**](#_sndxo22unip3)

[**IX. Conclusiones y recomendaciones 22**](#_wj0v3s67y8tu)

# **I. Introducción**

**A. Propósito del documento**

El propósito de este documento es establecer la arquitectura de software para el proyecto del Sistema de Gestión de Incidencias. Este documento proporciona una visión general de la estructura y organización del sistema, así como las decisiones de diseño y las tecnologías utilizadas.

**B. Alcance del proyecto**

El alcance de este proyecto se centra en el desarrollo de un sistema de gestión de incidencias que permita a los usuarios reportar, gestionar y dar seguimiento a las incidencias o problemas que surjan en un entorno empresarial. El sistema proporcionará funcionalidades como la creación de incidencias, asignación de responsables, seguimiento del estado de resolución y generación de informes.

**C. Definición de términos y acrónimos**

* Sistema de Gestión de Incidencias (SGI): Se refiere al sistema de software que se desarrollará para la gestión eficiente de las incidencias en un entorno empresarial.
* Incidencia: Hace referencia a cualquier problema, error o situación anómala reportada por los usuarios del sistema.
* Responsable: Persona o equipo encargado de resolver o atender una incidencia específica.
* Estado de resolución: Estado actual de una incidencia, que puede ser abierto, en progreso, pendiente o cerrado.

# **II. Visión general de la arquitectura**

**A. Objetivos de la arquitectura**

Los objetivos de la arquitectura del Sistema de Gestión de Incidencias son:

* Proporcionar una estructura modular y escalable que permita la adición de nuevas funcionalidades en el futuro.
* Garantizar la eficiencia y rendimiento del sistema, asegurando tiempos de respuesta rápidos y capacidad de manejo de grandes volúmenes de datos.
* Asegurar la seguridad y confidencialidad de la información, implementando medidas de autenticación y control de accesos.
* Facilitar la integración con otros sistemas o herramientas utilizados en la organización.
* Mejorar la usabilidad y experiencia del usuario, ofreciendo una interfaz intuitiva y amigable.
* Permitir la generación de informes y métricas para el seguimiento y análisis de las incidencias.

**B. Descripción general del sistema**

El Sistema de Gestión de Incidencias se compone de los siguientes componentes principales:

* Interfaz de Usuario: Es la capa de presentación del sistema, que permite a los usuarios interactuar con las funcionalidades y realizar acciones como registrar incidencias, consultar el estado de las mismas, asignar responsables, etc. Esta interfaz estará basada en un diseño intuitivo y amigable para facilitar su uso.
* Capa de Negocio: Es la capa que contiene la lógica del negocio, donde se procesan y gestionan las incidencias. Aquí se realizan operaciones como la asignación de responsables, seguimiento del estado de resolución, generación de informes, entre otras funcionalidades.
* Capa de Acceso a Datos: Es la capa encargada de interactuar con la base de datos del sistema. Aquí se realizan operaciones de lectura y escritura de los datos relacionados con las incidencias, permitiendo su almacenamiento, recuperación y actualización.
* Base de Datos: Es el repositorio donde se almacenan los datos del sistema, como la información de las incidencias, los usuarios, los roles, etc. Se utilizará un modelo de base de datos relacional para garantizar la integridad y consistencia de los datos.

**C. Stakeholders y sus roles**

En el proyecto del Sistema de Gestión de Incidencias, los stakeholders y sus roles son los siguientes:

* Cliente: Es el usuario o la organización que solicita y financia el desarrollo del sistema. Su rol es asegurarse de que el sistema cumpla con los requisitos y expectativas establecidos, y de que se alcancen los objetivos planteados.
* Usuarios finales: Son las personas que utilizarán el sistema para reportar y gestionar incidencias. Su rol es interactuar con la interfaz de usuario y aprovechar las funcionalidades del sistema para resolver los problemas o incidencias que se presenten.
* Equipo de desarrollo: Está conformado por los ingenieros de software y otros profesionales involucrados en el desarrollo del sistema. Su rol es diseñar, implementar y probar las funcionalidades del sistema, siguiendo las directrices establecidas en la arquitectura y los requisitos del proyecto.
* Administradores del sistema: Son responsables de la configuración, mantenimiento y seguridad del sistema. Su rol es garantizar que el sistema esté disponible, funcione correctamente y cumpla con los estándares de seguridad establecidos.

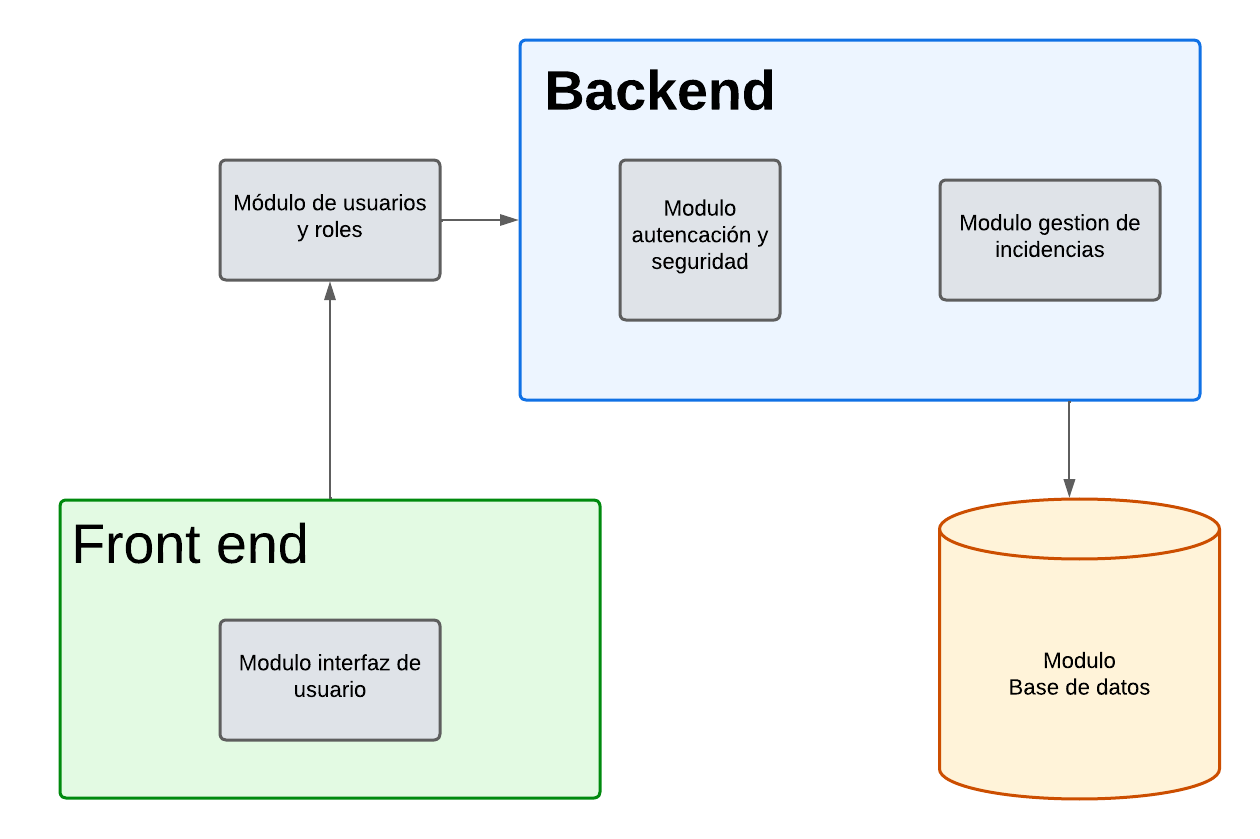
# **III. Arquitectura de alto nivel**

**A. Componentes principales**

Los componentes principales de un sistema de gestión de incidencias en línea son:

* Módulo de gestión de incidencias
* Módulo de usuarios y roles
* Módulo de autenticación y seguridad
* Módulo de interfaz de usuario
* Módulo de base de datos

**B. Diagrama de bloques**

****

**C. Descripción de cada componente**

**1. Módulo de gestión de incidencias**Este módulo permite a los usuarios reportar problemas, realizar seguimiento de las incidencias, asignar tareas a los equipos de soporte y dar seguimiento a la resolución de problemas. Proporciona funciones para la clasificación, priorización y asignación de incidencias, así como la generación de informes y métricas relacionadas con las incidencias.

**2. Módulo de usuarios y roles**En este módulo el administrador gestiona los roles y permisos en el sistema. Permite crear, modificar y eliminar cuentas de usuario, así como definir los roles y los permisos asociados según el tipo de usuario. Proporciona funciones para autenticar y autorizar a los usuarios, garantizando que solo tengan acceso a las funcionalidades y los datos adecuados según su rol y nivel de privilegios.

**3. Módulo de autenticación y seguridad**Este módulo proporciona funciones para la autenticación de usuarios, asegurando que solo usuarios autorizados puedan acceder al sistema. Además, implementa medidas de seguridad, como la encriptación de contraseñas, la gestión de sesiones y la protección contra ataques de seguridad, como la inyección de código malicioso o el acceso no autorizado a los datos.

**4. Módulo de interfaz de usuario**

Este módulo permite a los usuarios interactuar con el sistema a través de una interfaz gráfica intuitiva y fácil de usar. Proporciona funcionalidades como la navegación entre pantallas, la visualización de información, la entrada de datos y la interacción con otros componentes del sistema. También puede incluir características de personalización de la interfaz y adaptación a diferentes dispositivos y tamaños de pantalla.

**5. Módulo de base de datos**

Este componente se encarga de gestionar y almacenar los datos del sistema. Proporciona funcionalidades para la creación, lectura, actualización y eliminación de datos en la base de datos. Permite la persistencia de la información utilizada por otros módulos y proporciona mecanismos de consulta y búsqueda eficientes. También puede incluir la implementación de medidas de seguridad y respaldo de datos para garantizar la integridad y disponibilidad de la información almacenada.

# **IV. Detalles de la arquitectura**

**A. Componente 1: Módulo de gestión de incidencias**

**1. Funcionalidades principales**

El Módulo de Gestión de Incidencias es el componente central del sistema y se encarga de manejar todas las operaciones relacionadas con la gestión de incidencias. Sus funcionalidades principales incluyen:

* Registro de incidencias: Permite a los usuarios reportar y registrar nuevas incidencias en el sistema, proporcionando información detallada sobre el problema encontrado.
* Asignación de responsables: Facilita la asignación de un responsable o equipo encargado de resolver la incidencia reportada.
* Seguimiento del estado: Permite monitorear y actualizar el estado de resolución de las incidencias, manteniendo un registro de los avances y cambios realizados.
* Comunicación y notificaciones: Posibilita la comunicación entre los usuarios y los responsables de las incidencias, a través de comentarios, actualizaciones y notificaciones automáticas.
* Generación de informes: Permite generar informes y métricas sobre las incidencias, facilitando el análisis y la toma de decisiones.

**2. Interacción con otros componentes**

El Módulo de Gestión de Incidencias interactúa con otros componentes del sistema para lograr su funcionamiento completo:

* Interfaz de Usuario: El módulo se integra con la interfaz de usuario para permitir a los usuarios interactuar con las funcionalidades del sistema y acceder a las opciones de gestión de incidencias.
* Capa de Negocio: El módulo se comunica con la capa de negocio para procesar las solicitudes y operaciones relacionadas con las incidencias, como asignación de responsables, actualización de estados, generación de informes, etc.
* Capa de Acceso a Datos: El módulo utiliza la capa de acceso a datos para almacenar y recuperar la información relacionada con las incidencias, como su estado, detalles, comentarios, etc.

**3. Decisiones de diseño y tecnologías utilizadas**

En el diseño del Módulo de Gestión de Incidencias, se han tomado las siguientes decisiones y se han utilizado las siguientes tecnologías:

* Patrón de diseño MVC (Modelo-Vista-Controlador): Se ha adoptado este patrón para separar la lógica de negocio, la presentación y el control de las acciones, facilitando la modularidad y mantenibilidad del código.
* Lenguaje de programación: Se ha utilizado un lenguaje de programación orientado a objetos como Java o C#, que permita el desarrollo de una aplicación robusta y escalable.
* Framework de desarrollo: Se ha empleado un framework como Spring (para Java) o ASP.NET (para C#) para agilizar el desarrollo y proporcionar funcionalidades adicionales, como inyección de dependencias y manejo de peticiones HTTP.
* Base de datos: Se ha elegido una base de datos relacional, como MySQL o SQL Server, para almacenar y gestionar la información de las incidencias.
* Protocolos de comunicación: Se han utilizado protocolos estándar, como HTTP o HTTPS, para la comunicación entre el módulo y otros componentes del sistema, así como para la interacción con la interfaz de usuario.

**B. Componente 2: Módulo de usuarios y roles**

1. **Funcionalidades principales:**

El Módulo de Usuarios y Roles se encarga de gestionar la administración de usuarios y sus respectivos roles dentro del sistema de gestión de incidencias. Sus funcionalidades principales incluyen:

* Registro de usuarios: Permite la creación de nuevos usuarios en el sistema, solicitando información como nombre, correo electrónico, contraseña, etc.
* Gestión de roles: Permite asignar roles y permisos a los usuarios, definiendo qué acciones y funcionalidades pueden realizar en el sistema.
* Actualización de perfiles: Permite a los usuarios actualizar su información de perfil, como nombre, correo electrónico, contraseña, etc.
* Búsqueda y filtrado de usuarios: Facilita la búsqueda y filtrado de usuarios en función de criterios específicos, como nombre, rol, etc.

1. **Interacción con otros componentes:**

El Módulo de Usuarios y Roles interactúa con otros componentes del sistema para asegurar una adecuada administración de usuarios y roles:

* Módulo de Gestión de Incidencias: El módulo se integra con el Módulo de Gestión de Incidencias para permitir la asignación de responsables a las incidencias, basándose en los roles y permisos asignados a los usuarios.
* Módulo de Autenticación y Seguridad: El módulo interactúa con el Módulo de Autenticación y Seguridad para validar las credenciales de los usuarios al momento de acceder al sistema y asegurar que solo los usuarios autorizados puedan administrar usuarios y roles.

1. **Decisiones de diseño y tecnologías utilizadas:**

En el diseño del Módulo de Usuarios y Roles se han tomado las siguientes decisiones y se han utilizado las siguientes tecnologías:

* Base de datos: Se utiliza una base de datos relacional para almacenar la información de los usuarios y sus roles. Se pueden emplear tecnologías como MySQL
* Encriptación de contraseñas: Para garantizar la seguridad de las contraseñas, se utiliza un algoritmo de encriptación robusto, como bcrypt, para almacenar las contraseñas en la base de datos.
* Framework de autenticación y autorización: Se puede utilizar un framework como Spring Security o ASP.NET Identity para manejar la autenticación y autorización de los usuarios, permitiendo la configuración de roles y permisos.
* Interfaz de usuario: Se diseña una interfaz intuitiva y fácil de usar que permita la gestión eficiente de usuarios y roles, brindando opciones de creación, actualización y búsqueda de usuarios, así como asignación de roles.

**C. Componente 3:Módulo de interfaz de usuario**

**1. Funcionalidades principales**

El Módulo de Interfaz de Usuario proporciona la interfaz gráfica que permite a los usuarios interactuar con el sistema de gestión de incidencias. Sus funcionalidades principales incluyen:

* Registro e inicio de sesión: Permite a los usuarios registrarse en el sistema y luego iniciar sesión para acceder a las funcionalidades.
* Visualización de incidencias: Muestra una lista de incidencias registradas, permitiendo a los usuarios ver los detalles y el estado de cada incidencia.
* Creación y edición de incidencias: Permite a los usuarios crear nuevas incidencias y editar la información de las incidencias existentes.
* Seguimiento y actualización de incidencias: Permite a los usuarios realizar un seguimiento de las incidencias asignadas, actualizar su estado y agregar comentarios o notas relevantes.
* Generación de informes: Proporciona opciones para generar informes y estadísticas sobre las incidencias, permitiendo un análisis más profundo y una toma de decisiones informada.

**2. Interacción con otros componentes**

El Módulo de Interfaz de Usuario interactúa con otros componentes del sistema para proporcionar una experiencia de usuario completa:

* Módulo de Gestión de Incidencias: El módulo se comunica con el Módulo de Gestión de Incidencias para acceder y modificar la información relacionada con las incidencias, así como para reflejar los cambios realizados en la interfaz de usuario.
* Módulo de Usuarios y Roles: El módulo interactúa con el Módulo de Usuarios y Roles para mostrar la información de los usuarios y sus roles correspondientes en la interfaz de usuario, y permitir acciones relacionadas con la administración de usuarios.

**3. Decisiones de diseño y tecnologías utilizadas**

En el diseño del Módulo de Interfaz de Usuario, se han tomado las siguientes decisiones y se han utilizado las siguientes tecnologías:

* Diseño responsivo: Se implementa un diseño responsivo que se adapta a diferentes dispositivos y tamaños de pantalla, proporcionando una experiencia de usuario consistente en diferentes plataformas.
* Frameworks y bibliotecas de front-end: Se utilizan frameworks y bibliotecas de front-end, como React.js o Angular, para facilitar el desarrollo de la interfaz de usuario y mejorar la usabilidad y la interactividad.
* API RESTful: Se utiliza una API RESTful para la comunicación entre la interfaz de usuario y los componentes del backend, permitiendo una separación clara de responsabilidades y una fácil integración con otros sistemas o servicios.
* Diseño intuitivo: Se prioriza un diseño intuitivo y fácil de usar, siguiendo los principios de usabilidad y accesibilidad, para que los usuarios puedan navegar y realizar acciones de manera eficiente y sin problemas.

**D. Componente 4: Módulo de autenticación y seguridad**

**1. Funcionalidades principales**

El Módulo de Autenticación y Seguridad es esencial para garantizar la protección y el acceso seguro al sistema de gestión de incidencias. Sus funcionalidades principales incluyen:

* Autenticación de usuarios: Permite a los usuarios ingresar al sistema mediante la verificación de sus credenciales, como nombre de usuario y contraseña.
* Gestión de sesiones: Controla y administra las sesiones de los usuarios, asegurando que solo usuarios autenticados tengan acceso a las funcionalidades del sistema.
* Protección contra ataques: Implementa medidas de seguridad, como protección contra ataques de fuerza bruta, intentos de inicio de sesión repetidos y detección de actividad sospechosa.
* Gestión de roles y permisos: Asigna roles y permisos a los usuarios, permitiendo restringir el acceso y las acciones según los privilegios asignados.
* Registro de actividades: Registra las acciones realizadas por los usuarios, creando un registro de auditoría para fines de seguimiento y seguridad.

**2. Interacción con otros componentes**

El Módulo de Autenticación y Seguridad interactúa con otros componentes del sistema para garantizar la integridad y seguridad de las funcionalidades:

* Módulo de Usuarios y Roles: El módulo se integra con el Módulo de Usuarios y Roles para autenticar y autorizar a los usuarios en función de sus roles y permisos asignados.
* Módulo de Gestión de Incidencias: El módulo colabora con el Módulo de Gestión de Incidencias para restringir el acceso a las incidencias y las acciones relacionadas con ellas, según los roles y permisos del usuario autenticado

**3. Decisiones de diseño y tecnologías utilizadas**

En el diseño del Módulo de Autenticación y Seguridad, se han tomado las siguientes decisiones y se han utilizado las siguientes tecnologías:

* Algoritmos de encriptación: Se emplean algoritmos de encriptación robustos, como bcrypt o SHA-256, para almacenar las contraseñas de los usuarios de manera segura en la base de datos.
* Tokens de autenticación: Se utiliza un sistema de tokens, como JSON Web Tokens (JWT), para gestionar la autenticación de usuarios y permitir el acceso a las funcionalidades del sistema durante una sesión.
* Capas de seguridad: Se aplican medidas de seguridad en todas las capas de la aplicación, incluyendo la capa de presentación, la capa de negocio y la capa de acceso a datos, para proteger la información y prevenir ataques.
* Implementación de políticas de seguridad: Se establecen políticas de seguridad, como contraseñas fuertes, caducidad de sesiones y bloqueo de cuentas después de intentos fallidos de inicio de sesión, para fortalecer la protección del sistema contra amenazas.

**E. Componente 5: Módulo de base de datos**

**1. Estructura de la base de datos**

El Módulo de Base de Datos es responsable de almacenar y gestionar la información del sistema de gestión de incidencias de manera eficiente y segura. La estructura de la base de datos se organiza en tablas y relaciones que representan las entidades y las interacciones entre ellas. Algunas de las principales entidades en la base de datos pueden incluir: usuarios, incidencias, comentarios, roles, etc. La estructura de la base de datos se define de acuerdo con las necesidades del sistema y se diseña para permitir un almacenamiento óptimo y consultas eficientes.

**2. Relaciones y tablas utilizadas**

La base de datos del sistema de gestión de incidencias puede incluir varias tablas y relaciones para almacenar la información de manera coherente. Algunas de las relaciones y tablas comunes pueden ser:

* Tabla de usuarios: Almacena la información de los usuarios, como nombre, correo electrónico, contraseña, etc.
* Tabla de incidencias: Contiene los detalles de cada incidencia registrada, como título, descripción, estado, fecha de creación, etc.
* Tabla de comentarios: Guarda los comentarios realizados por los usuarios en relación con las incidencias, manteniendo un registro histórico de las interacciones.
* Tabla de roles: Almacena los roles asignados a los usuarios y sus correspondientes permisos.
* Otras tablas relacionadas: Dependiendo de los requisitos específicos del sistema, pueden existir otras tablas relacionadas para almacenar información adicional, como archivos adjuntos, registros de auditoría, etc.

**3. Decisiones de diseño y tecnologías utilizadas**

En cuanto al diseño y las tecnologías utilizadas en el Módulo de Base de Datos, se pueden tomar las siguientes decisiones:

* Sistema de gestión de base de datos (SGBD): Se elige un SGBD adecuado, como MySQL, según los requisitos del sistema y las preferencias del equipo de desarrollo.
* Normalización de datos: Se aplica la normalización de datos para eliminar la redundancia y garantizar la integridad de los datos en la base de datos.
* Índices y optimización de consultas: Se crean índices en las tablas para mejorar el rendimiento de las consultas y se optimizan las consultas frecuentes para minimizar el tiempo de respuesta.
* Respaldo y recuperación: Se implementa un plan de respaldo y recuperación de datos para garantizar la disponibilidad y la integridad de la información en caso de fallos o pérdidas.
* Seguridad de la base de datos: Se aplican medidas de seguridad, como encriptación de datos sensibles y control de acceso basado en roles, para proteger la información almacenada en la base de datos.

# **V. Patrones de diseño y principios arquitectónicos**

**A. Patrones utilizados en la arquitectura**

En la arquitectura del Sistema de Gestión de Incidencias se pueden utilizar diferentes patrones de diseño para abordar los desafíos específicos del sistema. Algunos ejemplos de patrones que podrían aplicarse son:

* Patrón Modelo-Vista-Controlador (MVC): Se utiliza para separar la lógica de negocio (modelo), la presentación (vista) y la interacción del usuario (controlador), facilitando el mantenimiento y la escalabilidad del sistema.
* Patrón Repositorio: Se utiliza para abstraer el acceso a los datos y proporcionar una capa de abstracción entre el código de la aplicación y la base de datos, mejorando la modularidad y facilitando los cambios en la tecnología de almacenamiento.
* Patrón de Autenticación y Autorización: Se emplea para gestionar la autenticación de los usuarios y controlar el acceso a las funcionalidades y los datos del sistema.

**B. Principios arquitectónicos aplicados**

En el diseño de la arquitectura del Sistema de Gestión de Incidencias, se aplican principios arquitectónicos que guían la toma de decisiones y garantizan la calidad y eficacia del sistema. Algunos principios arquitectónicos comunes que podrían aplicarse son:

* Separación de preocupaciones: Se separan las responsabilidades y funcionalidades del sistema en diferentes componentes para facilitar la comprensión, el mantenimiento y la reutilización del código.
* Escalabilidad: Se diseña la arquitectura para permitir la escalabilidad del sistema, de modo que pueda manejar un mayor número de incidencias, usuarios y cargas de trabajo sin degradar el rendimiento.
* Modularidad: Se divide el sistema en módulos independientes y cohesivos, lo que facilita el desarrollo paralelo, la sustitución de componentes y la incorporación de nuevas funcionalidades en el futuro.
* Flexibilidad: Se diseña la arquitectura de manera que pueda adaptarse a cambios y requisitos futuros sin necesidad de realizar modificaciones extensas o afectar la estabilidad del sistema.

# **VI. Consideraciones de rendimiento y escalabilidad**

**A. Requisitos de rendimiento**

En el contexto del Sistema de Gestión de Incidencias, es fundamental establecer los requisitos de rendimiento que el sistema debe cumplir para garantizar una experiencia fluida y eficiente para los usuarios. Algunos aspectos a considerar son:

* Tiempo de respuesta: Se deben definir los objetivos de tiempo de respuesta para diferentes operaciones del sistema, como la creación de incidencias, la búsqueda de registros o la generación de informes. Por ejemplo, se podría establecer un tiempo de respuesta máximo de 2 segundos para la consulta de incidencias.
* Capacidad de procesamiento: Se deben identificar los volúmenes de datos y la carga de trabajo esperada en el sistema. Esto ayudará a dimensionar adecuadamente los recursos de hardware y software para soportar la cantidad prevista de usuarios y transacciones simultáneas sin degradar el rendimiento.
* Escalabilidad horizontal y vertical: Es importante determinar si el sistema requerirá escalabilidad horizontal (agregar más servidores o nodos) o vertical (aumentar los recursos en un único servidor) para manejar el crecimiento del sistema. Esto implica evaluar los mecanismos de balanceo de carga, la capacidad de distribución de la base de datos y otras consideraciones arquitectónicas.

**B. Estrategias de escalabilidad**

La escalabilidad es un aspecto crítico en el diseño de un sistema de gestión de incidencias, ya que permite adaptarse al crecimiento y a las demandas cambiantes. Algunas estrategias que se pueden emplear para garantizar la escalabilidad del sistema son:

* Distribución de carga: Utilizar un enfoque de balanceo de carga para distribuir la carga de trabajo entre múltiples servidores. Esto puede lograrse mediante el uso de un equilibrador de carga que redirija las solicitudes de los usuarios a diferentes servidores, evitando la sobrecarga de uno en particular.
* Caché de datos: Implementar una capa de caché para almacenar en memoria los datos y consultas frecuentemente utilizados. Esto reduce la carga en la base de datos y mejora el tiempo de respuesta del sistema.
* Particionamiento de la base de datos: Si el tamaño de la base de datos crece considerablemente, se puede considerar el particionamiento de la base de datos en fragmentos más pequeños, distribuyendo los datos en diferentes servidores. Esto permite un mejor rendimiento y acceso eficiente a los datos.
* Uso de servicios en la nube: Considerar la posibilidad de utilizar servicios en la nube, como la infraestructura como servicio (IaaS) o plataforma como servicio (PaaS), que ofrecen la capacidad de escalar recursos de manera flexible y bajo demanda.
* Monitoreo y ajuste continuo: Implementar un sistema de monitoreo para supervisar el rendimiento del sistema en tiempo real y realizar ajustes cuando sea necesario. Esto puede implicar ajustes en la configuración de los servidores, optimización de consultas de base de datos u otras mejoras en la infraestructura.

# **VII. Consideraciones de seguridad**

**A. Amenazas y riesgos de seguridad**

* Acceso no autorizado: Existe el riesgo de que personas no autorizadas obtengan acceso al sistema de gestión de incidencias y manipulen o accedan a información confidencial.
* Fuga de información confidencial: Si la información de las incidencias contiene datos sensibles o confidenciales, existe el riesgo de que se divulgue o se filtre, lo que podría tener implicaciones legales o dañar la reputación de la organización.
* Ataques de denegación de servicio (DDoS): Los sistemas de gestión de incidencias pueden ser blanco de ataques DDoS, donde se sobrecarga el sistema con un alto volumen de tráfico, lo que resulta en la interrupción del servicio y la incapacidad de gestionar adecuadamente las incidencias.
* Inyección de código malicioso: Los atacantes podrían intentar inyectar código malicioso en el sistema de gestión de incidencias para obtener acceso no autorizado, robar información o causar daño en general.
* Fallos en la autenticación y autorización: Si existen debilidades en los mecanismos de autenticación y autorización, los atacantes podrían obtener credenciales de usuario válidas o acceder a funcionalidades para las cuales no tienen permiso.

**B. Estrategias de mitigación**

* Autenticación segura: Utilizar métodos sólidos de autenticación, como contraseñas seguras, autenticación de dos factores (2FA) para garantizar que solo los usuarios autorizados puedan acceder al sistema.
* Control de acceso: Se establecen políticas y mecanismos de control de acceso para limitar los privilegios de los usuarios y garantizar que solo tengan acceso a la información y las funcionalidades necesarias para llevar a cabo sus tareas.
* Encriptación de datos: Se aplican técnicas de encriptación para proteger la confidencialidad de la información almacenada y transmitida, como el uso de protocolos seguros (por ejemplo, HTTPS) y el cifrado de bases de datos.
* Auditoría y registro de actividades: Registrar y auditar todas las actividades realizadas en el sistema de gestión de incidencias, incluyendo el seguimiento de accesos, modificaciones y acciones realizadas por los usuarios. Esto puede ayudar a detectar comportamientos sospechosos o no autorizados.
* Actualizaciones y parches: Mantener el sistema de gestión de incidencias actualizado con las últimas actualizaciones de seguridad y parches para mitigar posibles vulnerabilidades conocidas.
* Pruebas de seguridad: Realizar pruebas de penetración y auditorías de seguridad periódicas para identificar posibles brechas o debilidades en el sistema y tomar medidas para corregirlas.
* Capacitación y concientización en seguridad: Proporcionar capacitación regular a los usuarios y miembros del equipo sobre las mejores prácticas de seguridad, como el uso de contraseñas seguras, la identificación de correos electrónicos de phishing y la protección de información confidencial.

# **VIII. Consideraciones de despliegue y operación**

**A. Requisitos de despliegue**

* Infraestructura de servidor: Se requiere un entorno de servidor adecuado que cumpla con los requisitos de capacidad, rendimiento y disponibilidad del sistema. Esto puede incluir servidores físicos o virtuales, capacidad de almacenamiento suficiente y recursos de red adecuados.
* Plataforma de software: Es necesario seleccionar la plataforma de software adecuada para el sistema de gestión de incidencias en línea. Esto puede incluir la elección de un sistema operativo, un servidor web, una base de datos y otros componentes necesarios para el funcionamiento del sistema.
* Configuración de red: Se deben configurar adecuadamente las reglas de firewall, enrutamiento y seguridad de red para permitir el acceso seguro al sistema de gestión de incidencias en línea y protegerlo de posibles amenazas.
* Conectividad a internet: El sistema de gestión de incidencias en línea debe estar conectado a internet de manera estable y confiable para permitir que los usuarios accedan al sistema y registren incidencias en tiempo real.

**B. Arquitectura de despliegue**

* Arquitectura cliente-servidor: Se utiliza una arquitectura cliente-servidor, donde el sistema de gestión de incidencias se divide en una interfaz de usuario (cliente) y un servidor centralizado que maneja la lógica de negocio y almacena los datos de las incidencias.

**C. Requisitos de mantenimiento y operación**

* Actualizaciones y parches: El sistema está actualizado con las últimas versiones de software y parches de seguridad para garantizar la corrección de errores y la protección contra vulnerabilidades conocidas.
* Monitorización y gestión de rendimiento: Se emplean herramientas de monitorización y gestión de rendimiento para supervisar la salud del sistema, el rendimiento de los servidores, el uso de recursos y la disponibilidad. Esto permite identificar y solucionar problemas de rendimiento o de capacidad a tiempo.
* Copias de seguridad y recuperación ante desastres: Es esencial realizar copias de seguridad regulares de los datos del sistema de gestión de incidencias y establecer procedimientos de recuperación ante desastres para garantizar la integridad y disponibilidad de los datos en caso de fallos o incidentes.

# **IX. Conclusiones y recomendaciones**

**A. Evaluación de la arquitectura**

* La arquitectura del sistema de gestión de incidencias en línea ha sido evaluada en función de su capacidad para satisfacer los requisitos de la organización, como la escalabilidad, la disponibilidad, la seguridad y el rendimiento.
* Se evaluó si la plataforma gestiona eficientemente las incidencias, desde su registro y asignación hasta su resolución y cierre.
* La arquitectura maneja de manera efectiva el flujo de trabajo de las incidencias, asegurando una asignación adecuada, la comunicación entre los equipos involucrados y la generación de informes y métricas relevantes.

**B. Recomendaciones para futuras mejoras**

* Mejorar la usabilidad de la interfaz de usuario: La interfaz de usuario debe ser intuitiva y fácil de usar, permitiendo a los usuarios registrar incidencias de manera eficiente y acceder a la información relevante de forma rápida. Se deben considerar mejoras en la navegación, la organización de la información y la simplicidad en los procesos.
* Ampliar las integraciones: El sistema de gestión de incidencias puede beneficiarse de integraciones con otras herramientas y sistemas utilizados en la organización, como sistemas de monitoreo, herramientas de seguimiento de errores o sistemas de gestión de proyectos. Esto facilitará la colaboración y la eficiencia en la resolución de incidencias.
* Implementar inteligencia artificial y aprendizaje automático: El uso de tecnologías como la inteligencia artificial y el aprendizaje automático puede mejorar la detección y clasificación de incidencias, así como proporcionar recomendaciones y soluciones automatizadas. Esto puede acelerar la resolución de incidencias y reducir la carga de trabajo manual.
* Reforzar la seguridad: Continuar mejorando la seguridad del sistema de gestión de incidencias en línea, implementando prácticas de autenticación sólidas, encriptación de datos, protección contra ataques de denegación de servicio y auditoría de actividades. Se deben realizar evaluaciones periódicas de seguridad y mantenerse al día con las últimas medidas de protección.
* Ofrecer soporte y capacitación continuos: Proporcionar soporte técnico y capacitación constante a los usuarios del sistema de gestión de incidencias, asegurándose de que comprendan plenamente su funcionalidad y cómo utilizarlo de manera efectiva. Esto ayudará a maximizar la adopción y el beneficio del sistema.