****

**UNIVERSIDAD PRIVADA DE TACNA**

**FACULTAD DE INGENIERÍA**

**Escuela Profesional de Ingeniería de Sistemas**

**Proyecto de Mejoramiento de la Aplicación Biblioteca Digital para Gestión Inteligente de Libros**

Curso: *Diseño y Arquitectura de Software*

Docente:  *Mag. Patrick Cuadros Quiroga*

Integrantes:

***AYMA CHOQUE, ERICK YOEL (2021072616)***

***POMA MACHICADO, FABIOLA ESTEFANI (2021070030)***

***TAPIA VARGAS, DYLAN YARIET(2021072630)***

**Tacna – Perú**

***2024***

| CONTROL DE VERSIONES | | | | | |
| --- | --- | --- | --- | --- | --- |
| Versión | Hecha por | Revisada por | Aprobada por | Fecha | Motivo |
| 1.0 | MPV | ELV | ARV | 10/10/2024 | Versión Original |

Mejoramiento de la Aplicación Biblioteca Digital para Gestión Inteligente de Libros

Documento de Arquitectura de Software

Versión *{1.0}*

ÍNDICE GENERAL

Contenido

[1. Antecedentes o introducción 4](#_ur0tc11a9719)

[2. Título 4](#_bbqyofkdxktv)

[3. Autores 4](#_4pnqye5ta1br)

[4. Planteamiento del problema 4](#_kbb02q7cj7yu)

[4.1. Problema 4](#_blv6j8cnfxd9)

[4.2. Justificación 4](#_71t9htop4ad4)

[4.3. Alcance 4](#_xmcuftodxxzg)

[5. Objetivos 4](#_roci47bxxy0j)

[5.1. General 4](#_tlrsygc9xdpi)

[5.2. Específico 4](#_tal16duis5c0)

[6. Referentes teóricos 4](#_vwkn1llvoy7i)

[7. Desarrollo de la propuesta 4](#_zb5h5rcak0jg)

[7.1. Tecnología de información 4](#_qkad593r6nn5)

[7.2. Metodología, técnicas usadas 4](#_k15ftrhqma6l)

[8. Cronograma 5](#_c5diwsc50ram)

**Tema: Mejoramiento de la Aplicación Biblioteca Digital para Gestión Inteligente de Libros**

# Antecedentes o introducción En un mundo donde la información y el conocimiento son cada vez más accesibles, las bibliotecas desempeñan un papel fundamental en la promoción de la lectura y el aprendizaje. Sin embargo, muchos sistemas de gestión de bibliotecas son obsoletos y no se adaptan a las necesidades actuales de los usuarios. La digitalización y la automatización se han convertido en herramientas esenciales para mejorar la eficiencia de estos servicios y ofrecer una experiencia más enriquecedora a los usuarios.

# 

# Este proyecto surge como respuesta a la necesidad de modernizar la gestión de bibliotecas, creando una plataforma que permita administrar de manera efectiva los recursos bibliográficos, optimizar el proceso de préstamos y devoluciones, y facilitar el acceso a la información a través de una interfaz intuitiva y fácil de usar.

# 

# Con el objetivo de fomentar la lectura y el acceso al conocimiento, este sistema está diseñado para adaptarse a las diversas necesidades de los usuarios, permitiendo la búsqueda y consulta de libros de forma rápida y eficiente. Además, se busca crear un espacio virtual donde se pueda promover la cultura y la educación, contribuyendo así al desarrollo de una sociedad más informada y culta.

# Título Biblioteca Digital para Gestión Inteligente de Libros

# Autores

# Erick Ayma Choque

# Dylan Tapia Vargas

# Fabiola Poma Machicado

# Planteamiento del problema

## Problema Las bibliotecas enfrentan desafíos significativos en la gestión de sus recursos, debido a la falta de sistemas automatizados que faciliten el préstamo, la devolución y la consulta de libros. Muchos sistemas existentes son obsoletos, lo que genera ineficiencias, demoras en el servicio y una experiencia insatisfactoria para los usuarios. Además, la dificultad en la búsqueda de información y en el acceso a recursos digitales limita el uso de las bibliotecas, afectando su relevancia en la comunidad.

## Justificación La implementación de un sistema de gestión bibliotecaria moderno es esencial para mejorar la eficiencia operativa y la satisfacción del usuario. Este proyecto no solo permitirá a los bibliotecarios gestionar los libros y los préstamos de manera más efectiva, sino que también facilitará el acceso a la información para los usuarios. La digitalización de los recursos bibliográficos y la creación de una plataforma intuitiva fomentarán el uso de la biblioteca, aumentando la participación de la comunidad en actividades culturales y educativas. Además, al automatizar los procesos, se liberará tiempo al personal para que se dedique a tareas más relevantes, como la promoción de la lectura y la organización de eventos.

## Alcance El alcance de este proyecto incluye el desarrollo de una plataforma de gestión bibliotecaria que abarque las siguientes funcionalidades:

## 

## Registro de libros: Permitir la adición, modificación y eliminación de libros en el sistema.

## Gestión de préstamos: Facilitar el préstamo y la devolución de libros, registrando automáticamente la información pertinente.

## Búsqueda y consulta de recursos: Proporcionar una interfaz de búsqueda avanzada para que los usuarios encuentren fácilmente los libros que desean.

## Informes y estadísticas: Generar reportes sobre el uso de los recursos, préstamos realizados y estadísticas de usuarios, que permitirán a la biblioteca tomar decisiones informadas.

## 

# Objetivos

## General

## Desarrollar un sistema de gestión bibliotecaria que facilite la administración de recursos, mejore la eficiencia en el préstamo y la devolución de libros, y ofrezca a los usuarios una plataforma intuitiva para la búsqueda y consulta de información.

## Específico

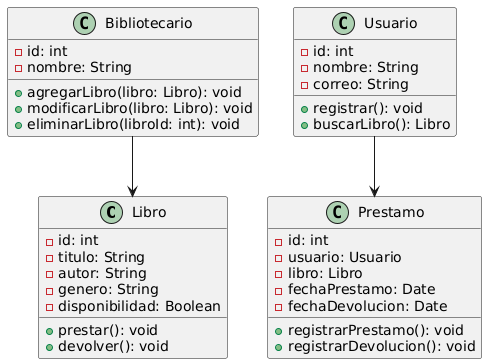
* Implementar un módulo de registro de libros que permita a los bibliotecarios añadir, modificar y eliminar libros de manera eficiente, asegurando que la información esté siempre actualizada.
* Desarrollar un sistema de gestión de préstamos que automatice el proceso de préstamo y devolución, incluyendo recordatorios de fechas de vencimiento y notificaciones de libros no devueltos.
* Crear una interfaz de usuario amigable que permita a los usuarios realizar búsquedas avanzadas por título, autor, género y otros criterios, facilitando el acceso a la información.
* Integrar un módulo de informes y estadísticas que proporcione datos sobre el uso de los recursos, patrones de préstamo y actividad de los usuarios, ayudando a la administración a tomar decisiones informadas.
* Fomentar la participación comunitaria mediante la inclusión de funciones que permitan a los usuarios dejar reseñas y calificaciones de los libros, así como participar en eventos y actividades organizadas por la biblioteca.

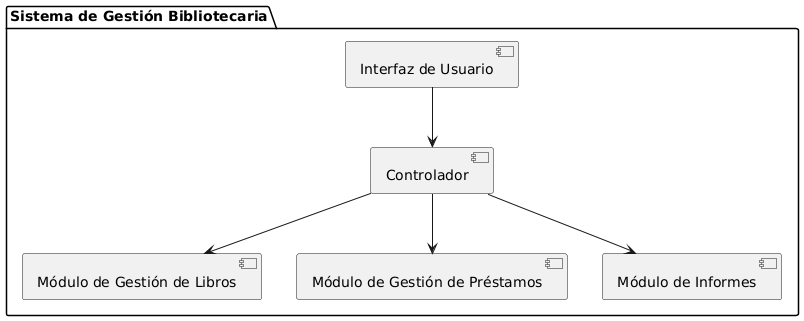
# Referentes teóricos

Diagramas de Casos de Uso, Diagrama de Clases, Diagrama de Componentes y Arquitectura.  
6.1. Diagrama de Casos de Uso

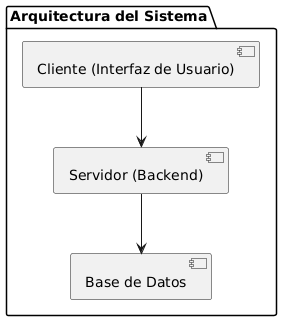


6.2. Diagrama de Clases



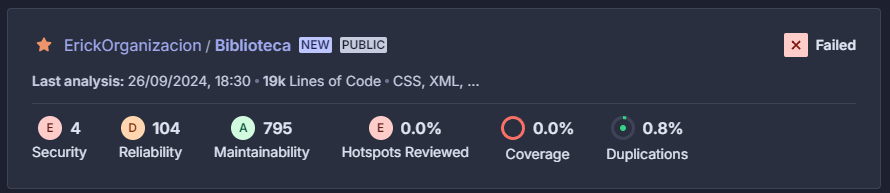
6.3. Diagrama de Componentes  


6.4. Diagrama de Arquitectura

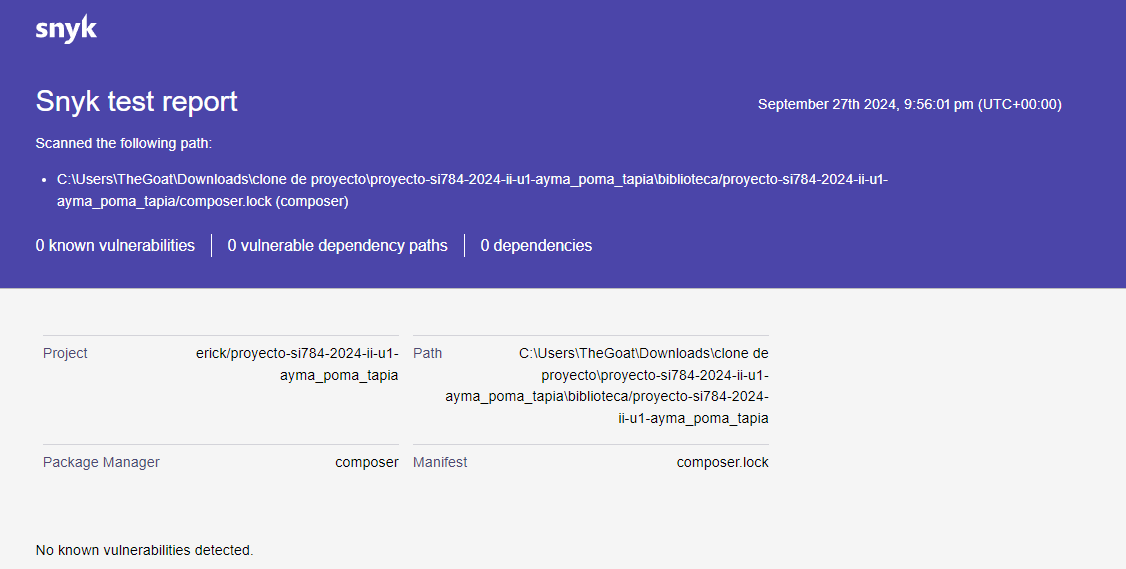


# Desarrollo de la propuesta

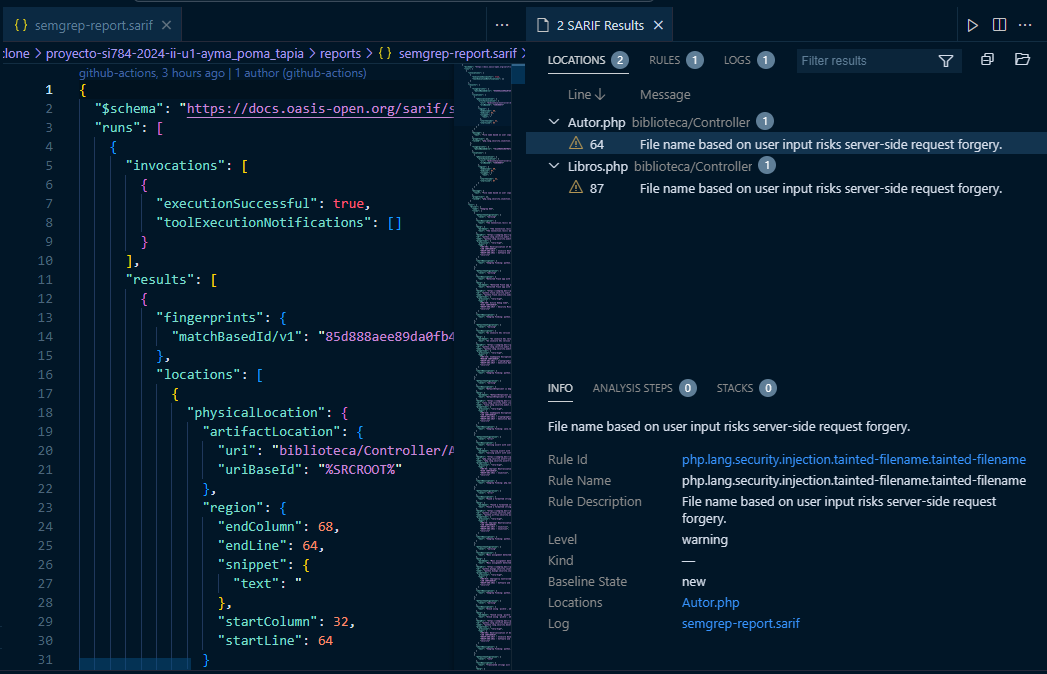
**SonarQube**

****

**Snyk**

****

**Semgrep**

****

## Tecnología de información Para el desarrollo del sistema de gestión bibliotecaria, se han utilizado diversas tecnologías que garantizan la seguridad, eficiencia y escalabilidad de la aplicación:

## **Lenguajes de Programación**:

## **PHP** para el backend, por su facilidad para la construcción de aplicaciones web y su amplia documentación.

## **JavaScript** para la interacción dinámica en el frontend, proporcionando una experiencia de usuario fluida.

## **Lenguajes de Marcado y Estilo**:

## **HTML**: Se utilizó HTML5 para estructurar el contenido del sitio web, asegurando que sea semántico y accesible.

## **CSS**: Se empleó CSS3 para diseñar y dar estilo a la interfaz de usuario, asegurando que la experiencia visual sea atractiva y responsiva. Se utilizó un enfoque modular y posiblemente preprocesadores como SASS para facilitar el mantenimiento de estilos.

## **Bases de Datos**:

## **MySQL**, un sistema de gestión de bases de datos relacional, fue seleccionado por su estabilidad, escalabilidad y facilidad de uso en proyectos de mediana envergadura.

## **Control de Versiones**:

## **GitHub** es la plataforma utilizada para el control de versiones, facilitando la colaboración en equipo y la integración de cambios a lo largo del ciclo de vida del proyecto.

## **Herramientas de Seguridad**:

## **SonarQube**: Una plataforma para la revisión continua de código, enfocada en la calidad y seguridad del software, identificando vulnerabilidades y puntos de mejora.

## **Snyk**: Herramienta de análisis de vulnerabilidades en bibliotecas y dependencias de código abierto, usada para proteger el proyecto frente a fallas de seguridad conocidas.

## **Semgrep**: Herramienta de análisis estático de código que permite identificar patrones inseguros o errores de programación, ayudando a mantener la integridad del código.

## Metodología, técnicas usadas Para el desarrollo del sistema, se adoptó una metodología **Ágil**, específicamente **Scrum**, debido a las siguientes características:

## **Iteraciones Cortas**: Cada Sprint tiene una duración de 1 semana, en las cuales se planifican, desarrollan y entregan incrementos del sistema.

## **Revisión Continua**: Se utilizó una retroalimentación constante para ajustar las funcionalidades y mejorar el producto.

## **Trabajo en Equipo**: Roles como el Product Owner, Scrum Master y el equipo de desarrollo están claramente definidos para asegurar la eficiencia en la toma de decisiones.

#### Técnicas Usadas:

## **Desarrollo Frontend**:

## Se implementaron prácticas de diseño responsivo utilizando **CSS** para asegurar que la aplicación sea accesible en diversos dispositivos.

## Se utilizó **HTML5** para estructurar adecuadamente el contenido y facilitar la accesibilidad y SEO.

## **Integración Continua**: A través de GitHub Actions, se asegura que cada cambio en el código se integre correctamente y sea probado antes de ser liberado en el entorno de producción.

## **Pruebas Automatizadas**: Se implementaron pruebas unitarias y de integración para verificar el correcto funcionamiento del sistema.

## **Análisis Estático de Código**: Herramientas como **SonarQube**, **Snyk** y **Semgrep** fueron utilizadas para realizar análisis estático del código, previniendo errores comunes y vulnerabilidades.

## 

# Cronograma

| **Actividad** | **Responsable** | **Tiempo Estimado** | **Recursos Necesarios** |
| --- | --- | --- | --- |
| **1. Configuración Inicial del Proyecto** | Equipo de Desarrollo | 1 semana | - Herramientas de desarrollo (IDE, Git) |
|  |  |  | - Acceso a SonarQube |
| **2. Análisis Inicial con SonarQube** | Equipo de QA | 1 semana | - SonarQube instalado y configurado |
|  |  |  | - Acceso a la base de código |
| **3. Identificación de Vulnerabilidades** | Equipo de Desarrollo | 1 semana | - Informes de SonarQube |
| **4. Priorización de Problemas** | Product Owner, Equipo de Desarrollo | 3 días | - Herramienta de gestión de proyectos |
| **5. Resolución de Vulnerabilidades** | Equipo de Desarrollo | 3 semanas | - Recursos de desarrollo y tiempo de codificación |
|  |  |  | - Sesiones de revisión de código |
| **6. Re-análisis con SonarQube** | Equipo de QA | 1 semana | - Acceso a SonarQube |
| **7. Documentación de Soluciones** | Documentador del Proyecto | 1 semana | - Herramientas de documentación |
| **8. Entrenamiento del Equipo** | Líder del Proyecto | 2 días | - Material de formación |
| **9. Revisión Final y Pruebas** | Equipo de QA | 1 semana | - Herramientas de prueba |
| **10. Implementación y Despliegue** | Todo el equipo | 1 semana | - Servidores de producción |
|  |  |  | - Recursos para la configuración del entorno |
| **11. Monitoreo Post-Despliegue** | Equipo de QA | 2 semanas | - Herramientas de monitoreo (SonarQube, etc.) |