



Nombres y Apellidos

Ruben Yasmiel Ureña Saviñon

Profesor

Francis Ramirez

Matricula

2025-1126

Materia

Programación II

Tema

Documento de Arquitectura del Proyecto

Sistema de Gestión de Bibliotecas (SIGEBI)

Background:

El Sistema de Gestión de Bibliotecas (SIGEBI) surge como respuesta a la necesidad de reemplazar procesos manuales y sistemas aislados utilizados en la biblioteca institucional. Estos procesos generan inconsistencias, errores operativos y falta de trazabilidad en la gestión de usuarios, recursos bibliográficos, préstamos, devoluciones y penalizaciones.

SIGEBI tiene como objetivo centralizar y automatizar la gestión bibliotecaria, asegurando la aplicación uniforme de las reglas del dominio, la confiabilidad de la información y la capacidad de auditoría, alineándose con las normas académicas y administrativas de la institución.

Requerimientos:

Tiene que:

- Registro y control de usuarios autorizados.
- Consulta de recursos bibliográficos físicos.
- Gestión de préstamos y devoluciones.
- Aplicación automática de penalizaciones por retraso.
- Administración del inventario bibliográfico.
- Registro y auditoría de todas las acciones relevantes.
- Generación de notificaciones por eventos clave.

Recomendable tener:

- Configuración de reglas de préstamo (límites, plazos)
- Historial detallado por usuario y recurso.

Podría tener:

- Reportes administrativos básicos.

Lo que no cubre el sistema:

- Gestión financiera.
- Adquisición de nuevos recursos.
- Acceso a recursos digitales externos.
- Gestión de usuarios fuera del contexto bibliotecario.

Componentes del Sistema (SIGEBI)

A continuación, se describen los componentes principales del sistema SIGEBI, definidos a partir del modelo arquitectónico seleccionado (aplicación web monolítica modular).

Componente de Autenticación y Autorización

Descripción:

Este componente se encarga de validar la identidad de los usuarios y controlar el acceso a las funcionalidades del sistema según su rol (usuario de biblioteca, personal bibliotecario o administrador institucional).

Justificación:

Es necesario para garantizar la seguridad del sistema y asegurar que cada actor acceda únicamente a las funciones que le corresponden, cumpliendo con los requisitos no funcionales de seguridad.

Componente de Gestión de Usuarios

- **Descripción:**
Administra la información de los usuarios autorizados de la biblioteca, incluyendo su estado (activo o bloqueado) y su historial de actividades dentro del sistema.
- **Justificación:**
Permite controlar quién puede acceder a los recursos bibliográficos y aplicar restricciones cuando existan penalizaciones activas, asegurando el cumplimiento de las reglas del dominio.

Componente de Gestión de Recursos Bibliográficos

- **Descripción:**
Gestiona el inventario de los recursos bibliográficos físicos de la biblioteca, controlando su identificación, estado y disponibilidad.
- **Justificación:**
Es fundamental para mantener un control confiable del inventario y garantizar que los recursos no sean prestados cuando no están disponibles.

Componente de Gestión de Préstamos

Descripción:

Controla el proceso de préstamos de recursos, incluyendo la validación de disponibilidad, fechas de inicio y vencimiento, y el límite de préstamos por usuario.

Justificación:

Es el núcleo funcional del sistema, ya que automatiza el proceso principal de la biblioteca y asegura la aplicación uniforme de las reglas de préstamo.

Componente de Gestión de Devoluciones y Penalizaciones

Descripción:

Registra las devoluciones de los recursos y aplica penalizaciones en caso de retraso, bloqueando temporalmente al usuario cuando corresponda.

Justificación:

Garantiza el cumplimiento de los plazos establecidos y permite un control justo y consistente del uso de los recursos bibliográficos.

Componente de Notificaciones

Descripción:

Genera notificaciones internas ante eventos relevantes como préstamos realizados, devoluciones, vencimientos y penalizaciones.

Justificación:

Mejora la comunicación con los usuarios y contribuye a reducir retrasos y errores operativos.

Componente de Auditoría y Trazabilidad

Descripción:

Registra todas las acciones relevantes realizadas dentro del sistema, permitiendo la supervisión y el control administrativo.

Justificación:

Es indispensable para garantizar la trazabilidad, facilitar auditorías y asegurar la confiabilidad de la información registrada.

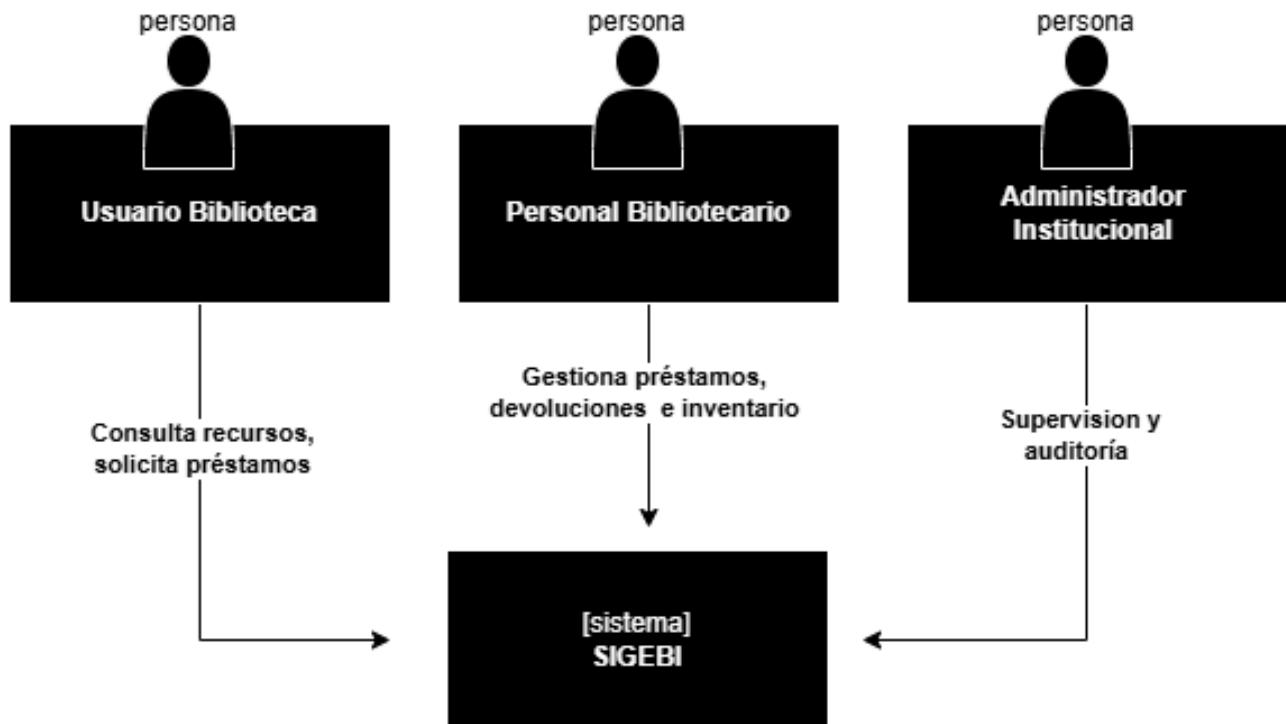
SIGEBI Diagrama de Contexto

Actores:

Usuario de la biblioteca: solicita préstamos y utiliza recursos.

Personal bibliotecario: gestiona préstamos, devoluciones, penalizaciones e inventario.

Administrador institucional: supervisa el sistema y consulta registros.



SIGEBI **Diagrama de Contenedores**

SIGEBI se implementará como una aplicación web con una arquitectura en capas, separando presentación, lógica de negocio y persistencia para garantizar mantenibilidad, seguridad y trazabilidad.

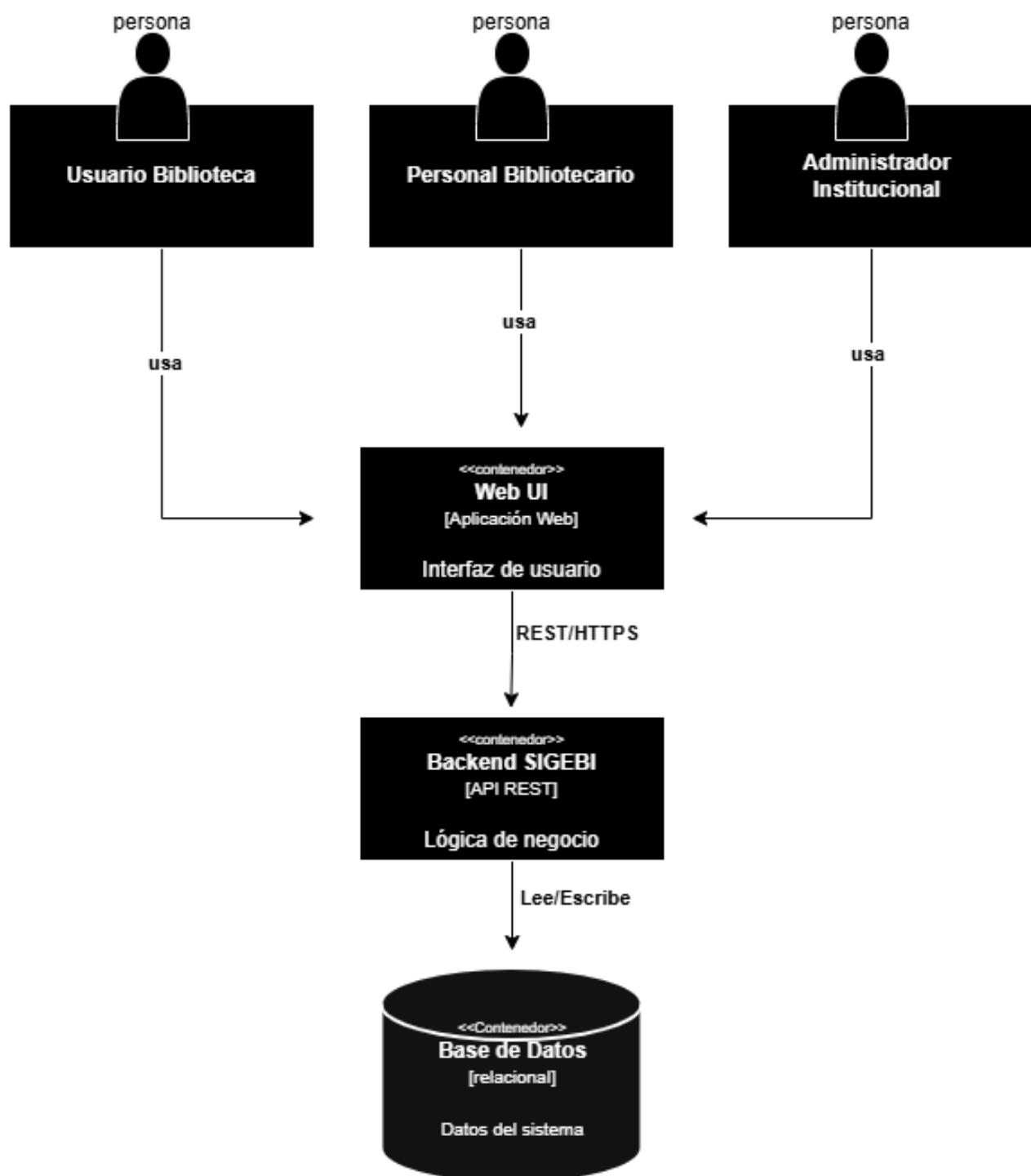
Contenedores principales:

La Web UI es una aplicación web accesible desde un navegador. Presenta diferentes vistas según el rol del usuario, ya sea usuario final, bibliotecario o administrador. Sus responsabilidades incluyen la autenticación de usuarios, la consulta de recursos bibliográficos, la gestión de préstamos y devoluciones, así como la visualización de notificaciones y sanciones asociadas a cada usuario.

El Backend SIGEBI es la aplicación central del sistema. Expone una API REST que es consumida por la Web UI y concentra toda la lógica de negocio junto con las reglas del dominio. Entre sus responsabilidades se encuentran la validación de las reglas de préstamo, la gestión de penalizaciones, el control de los estados de los recursos bibliográficos y el registro de auditoría de las operaciones realizadas en el sistema.

La Base de Datos Relacional es el componente encargado de la persistencia centralizada de la información del sistema. Almacena los datos correspondientes a los usuarios y sus roles, los recursos bibliográficos disponibles, los registros de préstamos y devoluciones, las penalizaciones aplicadas y los registros de auditoría necesarios para el control y trazabilidad del sistema.

El Servicio de Notificaciones es un módulo interno del backend que se encarga de generar notificaciones cuando ocurren eventos relevantes dentro del sistema. Estos eventos incluyen la realización de préstamos, las devoluciones de recursos, los vencimientos de los plazos establecidos y la aplicación de penalizaciones a los usuarios.



SIGEBI Componentes

El Backend SIGEBI se implementa como un monolito modular, donde cada módulo encapsula responsabilidades claras del dominio bibliotecario. Esta estructura facilita mantenibilidad, pruebas y evolución sin la complejidad de microservicios.

Componentes del Backend SIGEBI

1. Módulo de Autenticación y Autorización

- Gestión de inicio de sesión.
- Control de acceso basado en roles (Usuario, Bibliotecario, Administrador).

2. Módulo de Usuarios

- Registro y administración de usuarios autorizados.
- Consulta de estado del usuario (activo, sancionado).
- Historial de actividad del usuario.

3. Módulo de Recursos Bibliográficos

- Alta y mantenimiento del inventario.
- Gestión de estados del recurso (disponible, prestado, reservado).
- Validaciones de integridad (no eliminar con préstamo activo).

4. Módulo de Préstamos

- Creación y control de préstamos.
- Validación de disponibilidad y límites.
- Registro de fechas de inicio y vencimiento.

5. Módulo de Devoluciones y Penalizaciones

- Registro de devoluciones.
- Cálculo automático de penalizaciones por retraso.
- Gestión del estado de sanciones.

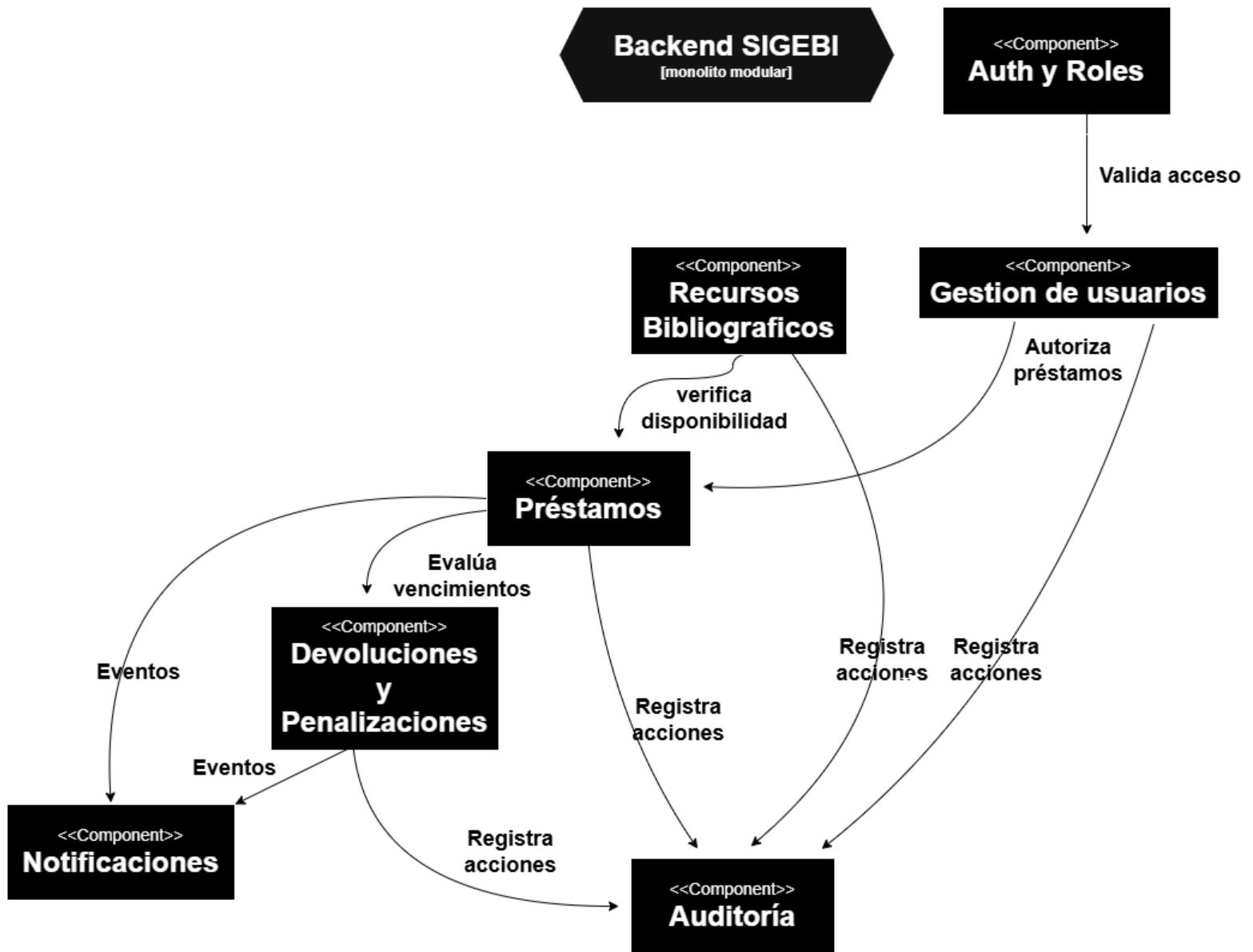
6. Módulo de Notificaciones

- Generación de notificaciones internas por eventos:
 - Préstamo realizado.
 - Devolución.
 - Vencimiento próximo.
 - Penalización aplicada.

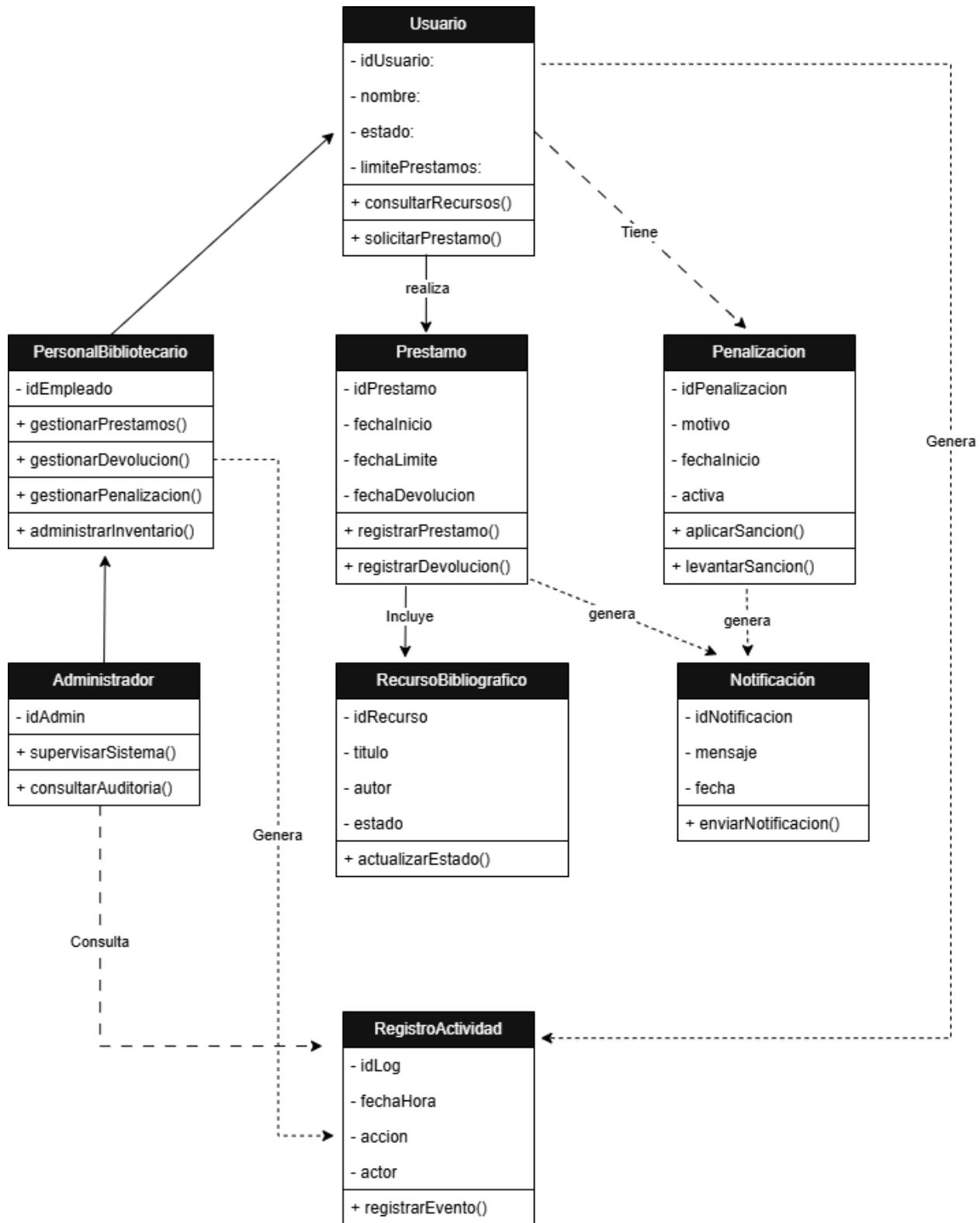
7. Módulo de Auditoría y Trazabilidad

- Registro inmutable de acciones relevantes.
- Soporte para supervisión y auditoría administrativa.

Diagrama componentes:



Vistas Arquitectónicas



Arquitectura de Usuarios y Roles -Herencia-

El diagrama utiliza una estructura de **herencia** para gestionar los permisos:

- **Usuario:** Es la clase base que contiene la información esencial (ID, nombre, estado). Representa a los estudiantes y docentes. Su atributo `limitePrestamos` es clave para validar la regla de negocio que impide préstamos excesivos.
- **PersonalBibliotecario:** Hereda de Usuario y añade las capacidades operativas. Es quien tiene los métodos para ejecutar acciones sobre el inventario y los préstamos.
- **Administrador:** Se sitúa en la cima de la jerarquía, con permisos exclusivos para supervisar el sistema y consultar los registros de auditoría detallados.

2. El Núcleo Operativo, los Préstamos y Recursos

- **RecursoBibliográfico:** Representa los libros o materiales. Su atributo `estado` es fundamental: el sistema consulta si está "Disponible" antes de permitir un préstamo, cumpliendo con el control de inventario.
- **Préstamo:** Es una **clase de asociación** lógica que conecta al Usuario con el Recurso. Almacena la `fechaInicio` y la `fechaLimite`. La lógica de negocio reside aquí para calcular si la

fechaDevoluciónEfectiva superó el límite, lo que dispararía una sanción.

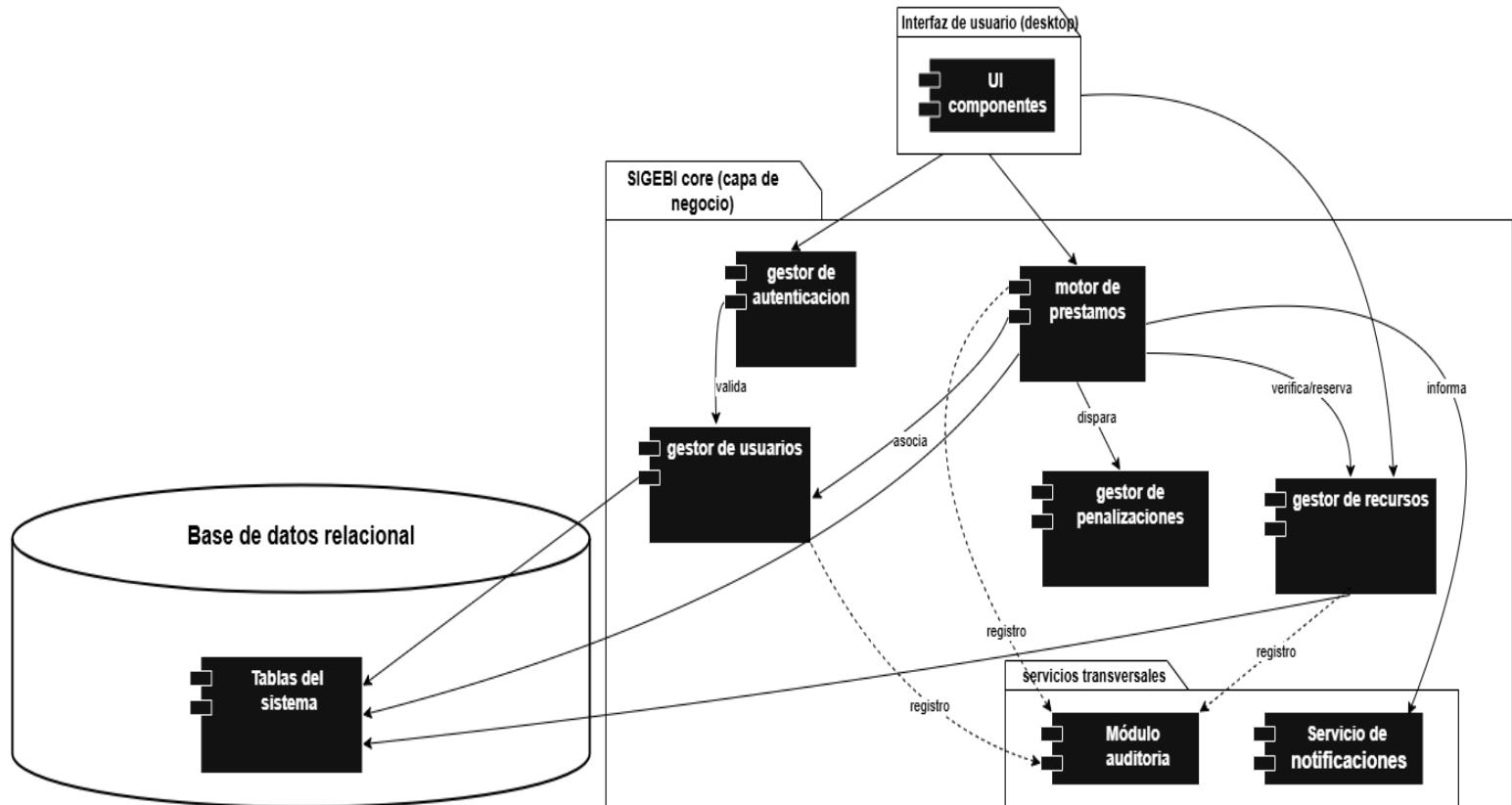
3. Sistema de Control y Sanciones

- **Penalización:** Esta clase está vinculada directamente al **Usuario**. Si un registro de penalización tiene el atributo activa en verdadero, el sistema bloquea las funciones de préstamo del usuario. Esto responde directamente al requerimiento de aplicar reglas uniformes de disciplina bibliotecaria.

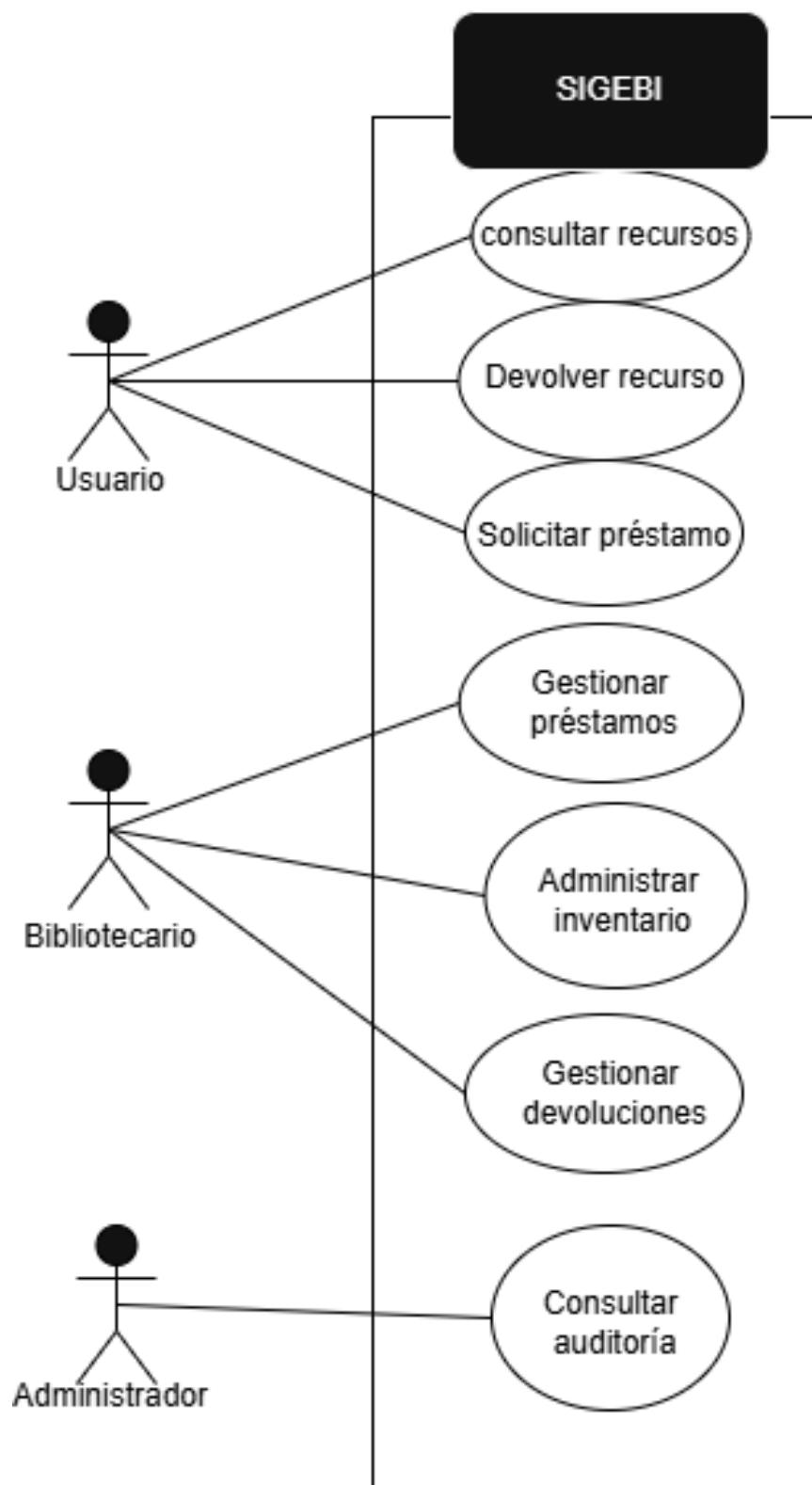
4. Soporte y Trazabilidad, los Requerimientos No Funcionales

- **RegistroActividad -Log-:** No es una entidad que el usuario manipule, sino un receptor de eventos. Cada vez que se crea un Préstamo o se modifica un Recurso, se **genera** un registro aquí con la fecha y la acción realizada.
- **Notificación:** Actúa como el sistema de alertas. Está relacionada con Préstamo y Penalización para informar al usuario sobre vencimientos de plazos o nuevas sanciones aplicadas.

Vista de Desarrollo – Componentes



Casos de Uso:

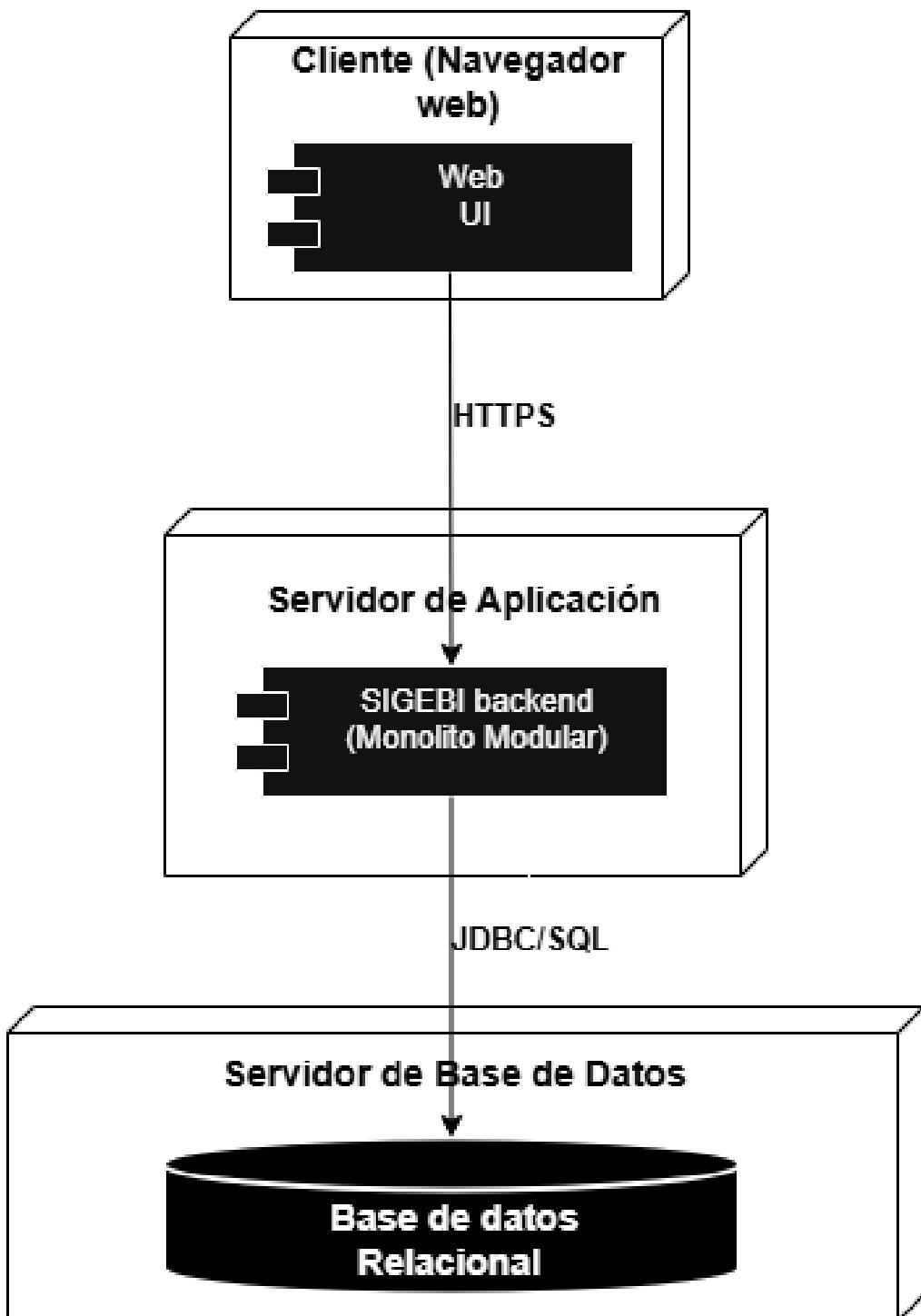


Descripción de Casos de Uso

Caso de Uso: Solicitar Préstamo

- Actor: Usuario de la biblioteca
- Descripción: El usuario solicita el préstamo de un recurso disponible.
- Precondiciones: Usuario activo y sin penalización.
- Flujo principal:
 1. Usuario selecciona recurso.
 2. Sistema valida disponibilidad y reglas.
 3. Sistema registra el préstamo.
 4. Sistema notifica al usuario.
- Flujos alternos:
 - Recurso no disponible → solicitud rechazada.
 - Usuario penalizado → solicitud rechazada.
- Postcondiciones: Préstamo registrado y recurso marcado como prestado.

Vista de Despliegue



Decisiones Arquitectónicas

Las principales decisiones arquitectónicas tomadas para el sistema SIGEBI son las siguientes:

- **Arquitectura web:**
Se eligió una aplicación web por su facilidad de acceso, mantenimiento centralizado y adecuación al entorno institucional.
- **Arquitectura monolítica modular:**
Se optó por un monolito modular debido al alcance académico del proyecto, reduciendo la complejidad y facilitando la comprensión del sistema.
- **Base de datos relacional:**
Se seleccionó una base de datos relacional para garantizar integridad, consistencia y trazabilidad de la información.
- **Separación en capas:**
Se implementa una separación entre presentación, lógica de negocio y persistencia para mejorar mantenibilidad y escalabilidad.

Requisitos No Funcionales

Los principales requisitos no funcionales del sistema son:

- **Seguridad:**
El sistema debe permitir el acceso únicamente a usuarios autorizados.
- **Disponibilidad:**
El sistema debe estar disponible durante el horario de operación de la biblioteca.
- **Usabilidad:**
La interfaz debe ser clara y fácil de usar para todos los actores.
- **Rendimiento:**
El sistema debe responder a las solicitudes en tiempos razonables.
- **Trazabilidad:**
Todas las acciones relevantes deben quedar registradas para fines de control y auditoría.

Riesgos Técnicos

A continuación, se identifican los principales riesgos técnicos del proyecto y sus estrategias de mitigación:

- **Falla del servidor de aplicación:**
Mitigación: respaldos periódicos y monitoreo del sistema.
- **Errores humanos en la operación:**
Mitigación: controles de validación y roles bien definidos.
- **Crecimiento futuro del sistema:**
Mitigación: diseño modular que permita la evolución del sistema.

Conclusiones

El Documento de Arquitectura de Software presentado define una arquitectura clara y coherente para el sistema SIGEBI, alineada con los objetivos del proyecto y el contexto institucional.

La utilización del modelo de vistas arquitectónicas permite comprender el sistema desde diferentes perspectivas, asegurando que los requisitos funcionales y no funcionales sean cubiertos de manera adecuada.

La arquitectura propuesta es simple, mantenible y adecuada para el alcance académico del proyecto.