

Biblioteca Atena

Problemática

En muchas bibliotecas, la gestión de libros físicos y los procesos asociados, como el préstamo y la devolución, se realizan de manera manual o con sistemas obsoletos, lo que genera ineficiencias operativas, falta de control sobre el inventario, acceso limitado a la información y riesgos de seguridad en la protección de datos. Estos problemas se traducen en errores frecuentes, dificultades para localizar libros, pérdida de recursos y una gestión ineficaz del personal y los materiales. Para solucionar estas deficiencias, se propone desarrollar un sistema de gestión bibliotecaria móvil, diseñado exclusivamente para dispositivos Android, que automatice los procesos de registro y control de libros, proporcionando a los trabajadores y administradores una plataforma eficiente y segura.

Solución

Para abordar estos problemas, se propone el desarrollo de un sistema de gestión bibliotecaria móvil, diseñado específicamente para ser utilizado en dispositivos Android. Este sistema proporcionará las siguientes soluciones:

- **Automatización y Eficiencia:** El sistema permitirá a los trabajadores registrar la entrada y salida de libros, así como los préstamos y devoluciones, de manera digital y automatizada, eliminando la necesidad de procesos manuales y reduciendo significativamente los errores.
- **Control Centralizado y Transparencia:** El sistema ofrecerá a los administradores una visión centralizada del inventario de libros.
- **Acceso Móvil y Flexibilidad:** Diseñado para dispositivos móviles, el sistema permitirá a los trabajadores y administradores realizar sus tareas desde cualquier lugar dentro de la biblioteca, proporcionando acceso en tiempo real a la información del inventario y permitiendo una gestión más ágil y flexible.

Objetivo del Proyecto

El objetivo principal es optimizar y automatizar los procesos bibliotecarios, desde la catalogación y gestión de recursos hasta los préstamos y devoluciones, pasando por la gestión de usuarios, al centralizar la información y automatizar tareas repetitivas, estos sistemas liberan al personal bibliotecario para que se concentre en tareas de mayor valor añadido, como la atención al usuario y la promoción de la lectura. Además, ofrecen a los usuarios una experiencia más eficiente y satisfactoria, permitiéndoles buscar y encontrar los recursos que necesitan de forma rápida y sencilla.

Análisis: Definición de requerimientos.

Requisitos Funcionales

RF1: Autenticación de Usuario:

- **Descripción:** El sistema debe permitir a los usuarios autenticarse utilizando sus credenciales.
- **Entrada:** Credenciales de usuario (correo electrónico y contraseña).
- **Procesamiento:** Verificación de las credenciales.

- Salida: Acceso concedido o denegado.

RF2: Creación de usuario Administrador/Trabajador:

- Descripción: El sistema ofrecer a los administradores la capacidad de crear nuevos usuarios administrativos dentro del sistema.
- Entrada: Datos de nuevo usuario (Nombre, documento de identificación correo electrónico, contraseña, rol).
- Procesamiento: Ingreso de Información del Usuario
- Salida: Creación de nuevo usuario

RF3: Registro de Entrada de Libros:

- Descripción: Permitir el registro de nuevos libros en la biblioteca.
- Entrada: Detalles del libro (título, autor, categoría, Cantidad de copias, estado del libro).
- Procesamiento: Almacenamiento de la información en la base de datos.
- Salida: Confirmación del registro exitoso.

RF4: Gestión de Inventario:

- Descripción: Permitir la consulta y actualización de la información de los libros disponibles.
- Entrada: Consulta o actualización de datos.
- Procesamiento: Búsqueda y modificación de registros en la base de datos.
- Salida: Información actualizada sobre los libros.

RF5: Registro de Clientes:

- Descripción: El sistema ofrecer a los trabajadores la capacidad de crear nuevos usuarios para entrega de libros.
- Entrada: Datos de nuevo usuario (Nombre, documento de identificación, teléfono, correo electrónico).
- Procesamiento: Ingreso de Información de las personas
- Salida: Creación de nuevo usuario

RF6: Registro de Salida de Libros:

- Descripción: Registrar la salida de libros prestados.
- Entrada: Detalles del préstamo (usuario, libro, fecha de devolución, encargado).
- Procesamiento: Almacenamiento de la información en la base de datos.
- Salida: Confirmación del registro de salida.

RF7: Devolución de Libros:

- Descripción: Registrar la devolución de libros.
- Entrada: Detalles de la devolución (usuario, libro, fecha de devolución, encargado).
- Procesamiento: Actualización de la información en la base de datos.
- Salida: Confirmación de la devolución.

Requisitos No Funcionales

RNF1: Rendimiento:

- El sistema debe ser capaz de procesar y responder a las solicitudes de los usuarios en un tiempo máximo de 2 segundos bajo condiciones normales de operación.
- El sistema debe soportar al menos 20 usuarios concurrentes y transacciones sin degradación significativa en el rendimiento.

RNF2: Seguridad:

- Protección de la información almacenada.
- Mecanismos de autenticación y autorización para usuarios del sistema.

RNF3: Compatibilidad:

- La aplicación debe ser compatible con dispositivos Android, soportando versiones del sistema operativo desde Android 8.0 (Oreo) en adelante.
- El sistema debe adaptarse correctamente a diferentes tamaños de pantalla de dispositivos móviles

RNF4: Usabilidad:

- Interfaz intuitiva y fácil de usar.
- Capacitación mínima requerida para los operadores.

RNF5: Escalabilidad:

- El sistema debe estar diseñado para soportar un crecimiento en el número de libros y usuarios sin necesidad de cambios significativos en la infraestructura

Documentación IEEE 830

Diseño Base de Datos

Tabla de Usuarios

Campo	Tipo	Clave	Descripción
idUsuario	INT	PK	Identificador único
nombre	VARCHAR		Nombre del trabajador
cedula	FLOAT		Cedula del trabajador
email	VARCHAR		Correo de inicio
clave	VARCHAR		Contraseña de inicio
rol	ENUM		Rol Trabajador/Administrador
registro	TIMESTAMP DEFAULT		Registro de creación

Tabla de Libros

Campo	Tipo	Clave	Descripción
idLibro	INT	PK	Identificador único
titulo	VARCHAR		Titulo del libro
autor	VARCHAR		Autor del libro
categoria	VARCHAR		Categoría del libro
copias	INT		Cantidad del mismo libro
estado	VARCHAR		Estado del libro
registro	TIMESTAMP DEFAULT		Registro de creación

Tabla de Clientes

Campo	Tipo	Clave	Descripción
idCliente	INT	PK	Identificador único
nombre	VARCHAR		Nombre del cliente
cedula	FLOAT		Cedula del cliente
telefono	FLOAT		Telefono del cliente
email	VARCHAR		Correo del cliente
registro	TIMESTAMP DEFAULT		Registro de creación

Tabla de Prestamos

Campo	Tipo	Clave	Descripción
idPrestamo	INT	PK	Identificador único
idCliente	INT	FK	Llamado cliente
idLibro	INT	FK	Llamado libro
fechaPrestamo	DATE		Fecha prestamo libro
fechaDevolucion	DATE		Fecha devolucion libro
idUsuario	INT	FK	Llamado Trabajador

Script de la Base de Datos

```
DROP DATABASE IF EXISTS Biblioteca;

CREATE DATABASE Biblioteca;

USE Biblioteca;
```

```
-- Tabla de Usuarios
CREATE TABLE Usuarios (
  id_usuario INT PRIMARY KEY AUTO_INCREMENT,
  nombre VARCHAR(100) NOT NULL,
  cedula FLOAT NOT NULL,
  email VARCHAR(100) NOT NULL,
  clave VARCHAR(100) NOT NULL,
  rol ENUM('trabajador', 'administrador') NOT NULL,
  fecha_registro TIMESTAMP DEFAULT CURRENT_TIMESTAMP
);

-- Tabla de Libros
CREATE TABLE Libros (
  id_libro INT PRIMARY KEY AUTO_INCREMENT,
  titulo VARCHAR(255) NOT NULL,
  autor VARCHAR(255) NOT NULL,
  categoria VARCHAR(100) NOT NULL,
  copias INT NOT NULL,
  estado VARCHAR(255) NOT NULL,
  fecha_registro TIMESTAMP DEFAULT CURRENT_TIMESTAMP
);

-- Tabla de Clientes
CREATE TABLE Clientes (
  id_cliente INT PRIMARY KEY AUTO_INCREMENT,
  nombre VARCHAR(100) NOT NULL,
  cedula FLOAT NOT NULL,
  telefono FLOAT NOT NULL,
  email VARCHAR(100) NOT NULL,
  fecha_registro TIMESTAMP DEFAULT CURRENT_TIMESTAMP
);

-- Tabla de Prestamos
CREATE TABLE Prestamos (
  id_prestamo INT PRIMARY KEY AUTO_INCREMENT,
  id_cliente INT NOT NULL,
  id_libro INT NOT NULL,
  fecha_prestamo DATE NOT NULL,
  fecha_devolucion DATE,
  id_usuario INT NOT NULL,
  estado ENUM('prestado', 'devuelto') NOT NULL DEFAULT 'prestado',
  FOREIGN KEY (id_cliente) REFERENCES Usuarios(id_cliente),
  FOREIGN KEY (id_usuario) REFERENCES Usuarios(id_usuario),
  FOREIGN KEY (id_libro) REFERENCES Libros(id_libro)
);
```

Diagramas UML

Estos diagramas proporcionarán una representación visual y comprensible de los diferentes componentes y flujos del sistema.

Diagrama de Clases

El diagrama de clases modela la estructura estática del sistema, mostrando las clases, sus atributos, métodos y las relaciones entre ellas.

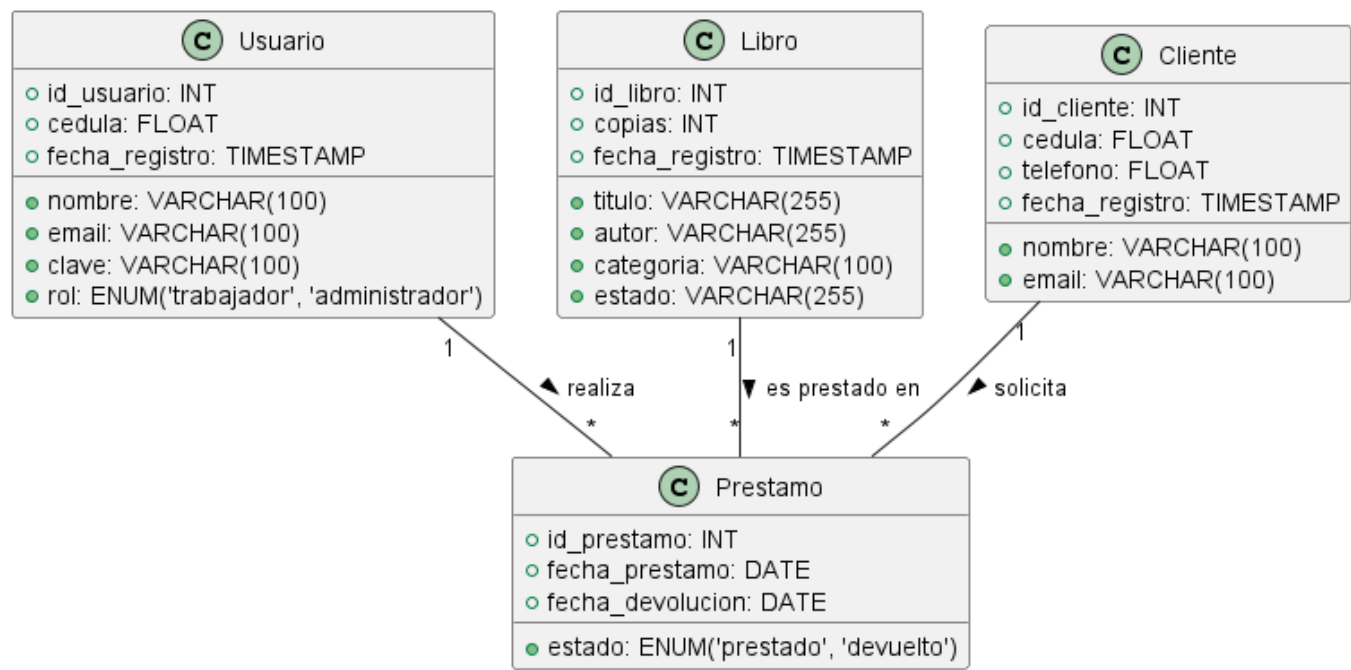


Diagrama de Secuencias

El diagrama de secuencia ilustra cómo interactúan los diferentes objetos del sistema a lo largo del tiempo para llevar a cabo un proceso específico.

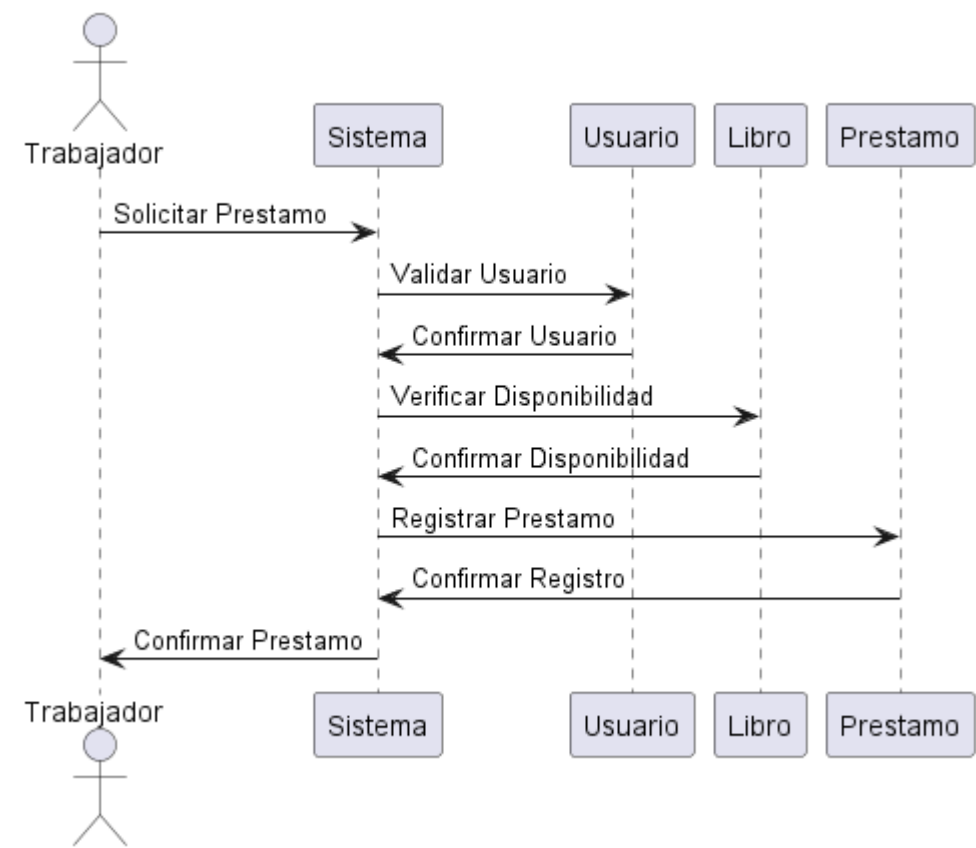
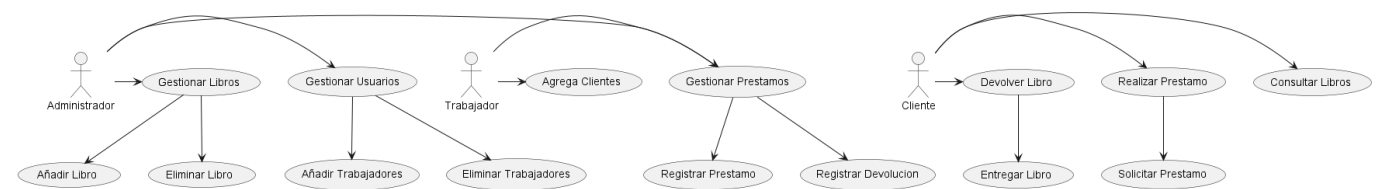


Diagrama de Casos de Uso

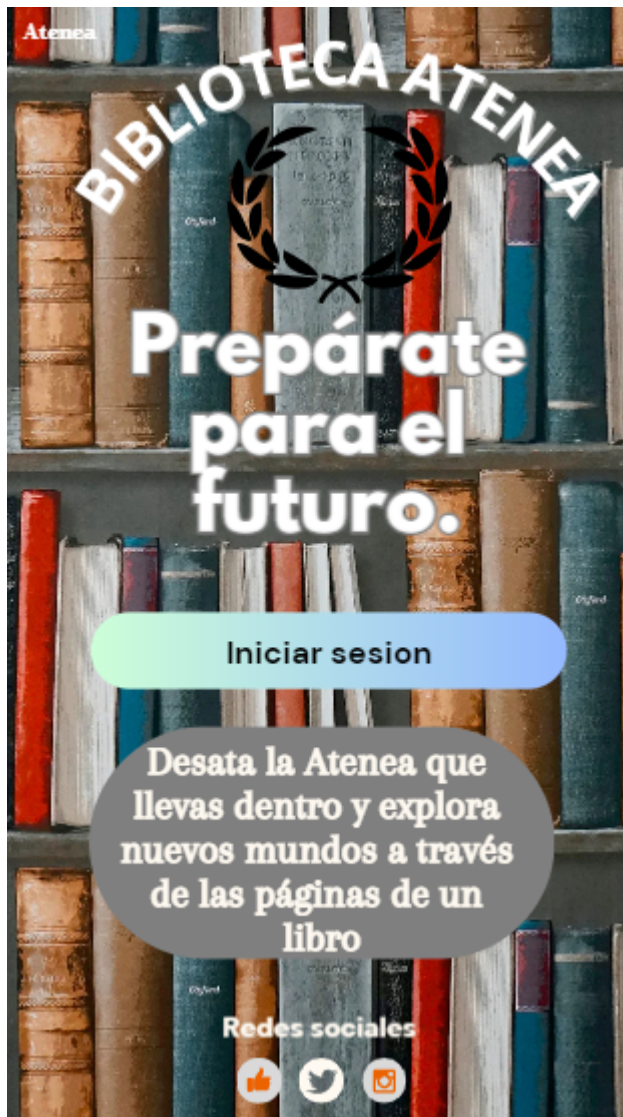
El diagrama de casos de uso describe las interacciones entre los actores (conductor y administrador) y el sistema de parqueadero, destacando los principales casos de uso o funcionalidades.



Mockup

El diseño de modelado se centra en la interfaz gráfica e interactiva desde la perspectiva tanto del usuario como del administrador. En él, se incluyen componentes como inicio de sesión, registros de usuarios y de libros, devoluciones y prestamos a clientes, entre otras cosas. Puedes encontrar la visualización respectiva en el siguiente enlace.

[Link Modelado](#)



Ver Planificacion

La planificación del proyecto en el marco de trabajo para desarrollo ágil de Software Scrum, se puede ver en el siguiente enlace redireccionado a Trello y al Drive:

| [Google Drive](#) | [Trello](#) |