

Reporte del Proyecto Final

17/05/25

Diego Martín Domínguez Hernández

Seminario de bases de datos

Prof. José Antonio Aviña Mendez

Índice

Introducción	2
Descripción del sistema	2
Entidades principales	2
Funcionalidades	2
Metodología	3
Diseño de la base de datos	4
Modelo entidad relación	4
Esquema relacional	4
Implementación	5
Creación de la base de datos	5
Configuración del entorno de desarrollo	5
Diagrama de flujo del sistema	9
Mockups	10
Conclusión	21
Referencias	21

Introducción

Las bibliotecas son el pilar fundamental del conocimiento y el aprendizaje; sin embargo, en la era digital, se enfrentan nuevos retos y oportunidades. Este proyecto busca responder a estos desafíos mediante un sistema integral de gestión de bases de datos diseñado específicamente para bibliotecas modernas. El sistema tiene como objetivo optimizar las operaciones bibliotecarias, incluyendo la gestión del inventario de libros, el registro de usuarios, el seguimiento de préstamos y la administración de reservas. Esta implementación las bibliotecas pueden aumentar su eficiencia, mejorar la experiencia de los usuarios y gestionar de manera más efectiva sus recursos en un mundo cada vez más digitalizado.

Descripción del sistema

Este sistema es una solución integral diseñada para optimizar las operaciones diarias de una biblioteca mediante el uso de una bases de datos relacional. Este sistema gestiona de manera eficiente las interacciones entre libros, miembros, préstamos y reservas, proporcionando una estructura organizada y accesible para el manejo de datos.

Entidades principales

En el ámbito de una base de datos, las entidades son los objetos o conceptos del mundo real que el sistema necesita identificar y gestionar mediante la recopilación de datos específicos. En una biblioteca, las entidades principales son los elementos esenciales que permiten organizar y controlar los recursos y las interacciones de la biblioteca de manera eficiente.

Las entidades principales que conforman este sistema incluyen.

- **Libros:** Los recursos que la biblioteca ofrece, ya sean físicos o digitales
- **Miembros:** Los usuarios registrados que utilizan los servicios de la biblioteca
- **Préstamos:** Los registros de las transacciones cuando un miembro toma un libro prestado
- **Reservas:** Las solicitudes para apartar libros que no están disponibles en el momento

Estas entidades están interrelacionadas: un miembro puede realizar varios préstamos o reservas, y un libro puede pasar por multiples préstamos a lo largo del tiempo. Esta red de conexiones es lo que permite al sistema funcionar de manera efectiva, asegurando que los recursos se gestionen adecuadamente y que los usuarios tengan acceso a ellos de forma ordenada.

Funcionalidades

El sistema ofrece un conjunto de herramientas para optimizar la administración de la biblioteca y facilitar el acceso de los usuarios a sus recursos. A continuación, se describen sus principales características:

- Registro y actualización del inventario de libros.
- Gestión de membresías y préstamos.

- Gestión de multas por retrasos.

En resumen, el sistema de gestión combina herramientas prácticas y automatizadas que simplifican las tareas administrativas.

Metodología

El desarrollo se llevó a cabo mediante un enfoque basado en el ciclo de vida de desarrollo de bases de dato, un método que asegura la creación de un sistema robusto y fácil de mantener. El enfoque se compone de cinco etapas principales

Análisis de requisitos Se identificaron las necesidades operativas y funcionales de la biblioteca

Diseño conceptual Con los requisitos, se elaboró un modelo entidad relación para representar de manera abstracta las entidades principales del sistema.

Diseño lógico El modelo ER se transformó en un esquema relacional, definiendo las tablas necesarias junto con sus claves primarias y foráneas, aplicando técnicas para evitar redundancia y garantizar la integridad de los datos.

Diseño físico Se eligió PostgreSQL como el sistema de gestión de bases de datos, debido a su robustez y capacidad para soportar operaciones complejas. En esta fase, se especificaron los tipos de datos de cada campo y se optimizaron las consultas para asegurar un buen rendimiento.

Implementación En la etapa final, se creó la base de datos en PostgreSQL utilizando scripts SQL para definir las tablas e insertar datos de ejemplo. Estos datos permiten probar las funcionalidades del sistema, como la gestión de préstamos y reservas. Además se diseñó un mockup para la interfaz de usuario, aunque su implementación quedó como opcional según los requerimientos del proyecto.

La planificación del desarrollo se gestionó mediante un diagrama de Gantt, que facilitó la organización de las tareas y el cumplimiento de los plazos en cada etapa.

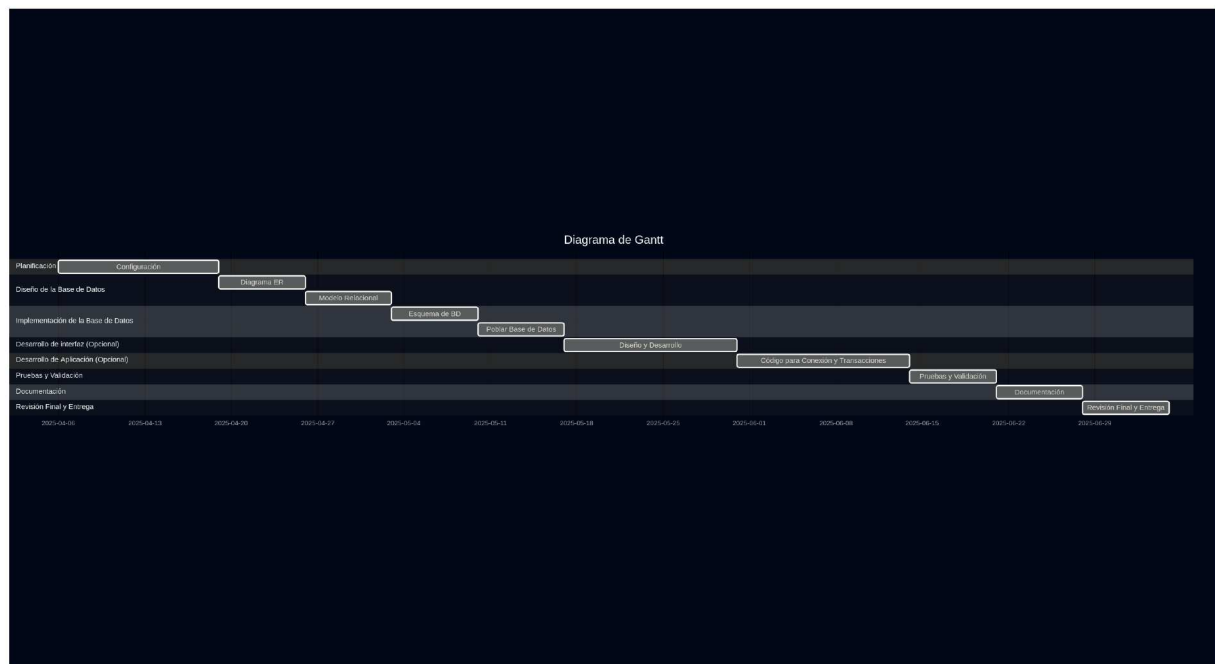


Figura 1: Diagrama de actividades

Diseño de la base de datos

En el diseño se establece cómo se organizan, almacenan y relacionan los datos necesarios para las operaciones de la biblioteca. este diseño se desarrolló utilizando un enfoque estructurado que comenzó con modelo entidad relación y se transformó en esquema relacional, implementado en PostgreSQL.

Modelo entidad relación

Es la representación más visula para las principales entidades del sistema y sus interrelaciones, con las entidades siendo:

- Libros
- Autores
- Préstamos
- Reservas
- Editoriales y categorías

Esquema relacional

El ERD se convirtió en esquema relacional con tablas como:

BOOKS Almacena datos de los libros, con *book id* como clave primaria

AUTHORS Contiene información de los autores, con *author id*

MEMBERS Registra a los usuarios, con *member id*

LOANS Gestiona los préstamos, vinculando *book id* y *member id* mediante claves foráneas.

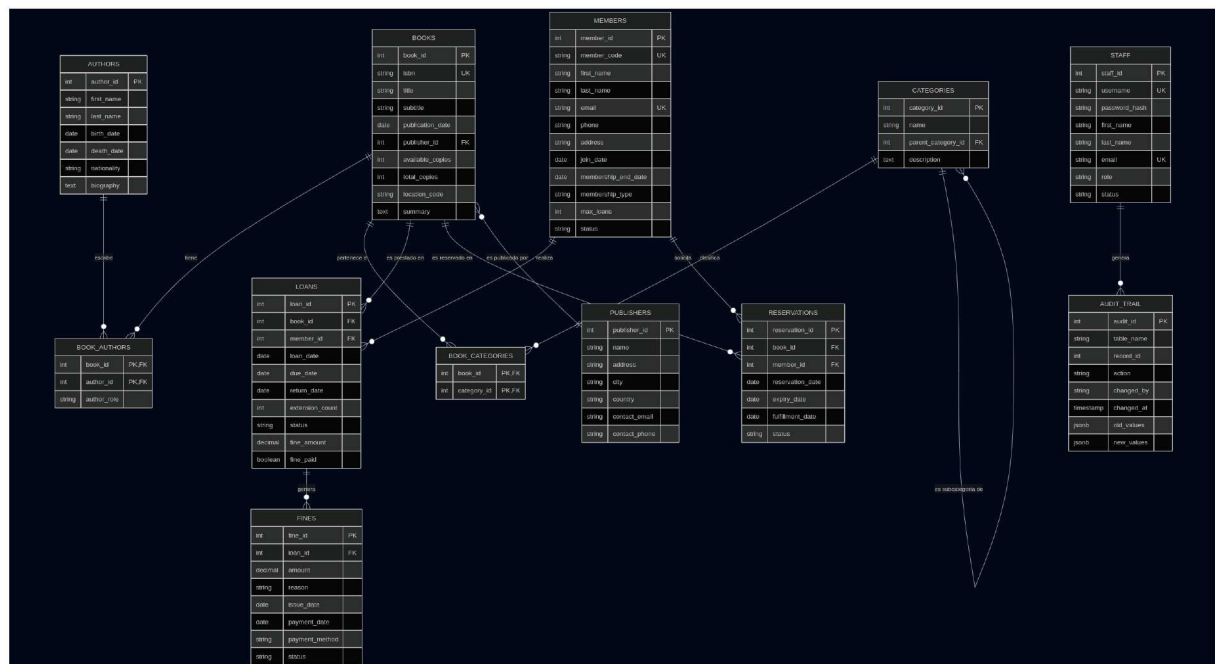


Figura 2: Esquema relacional

Implementación

Se centró en la construcción de una base de datos relacional utilizando PostgreSQL, elegido por su robustez, capacidad de escalabilidad y soporte para operaciones complejas.

Creación de la base de datos

El proceso comenzó con la ejecución de scripts SQL para definir la estructura de la base de datos. Los scripts abarcaron:

Tablas Se crearon tablas para las entidades principales, tales como BOOKS, AUTHORS, MEMBERS, LOANS y RESERVATIONS.

Claves primarias Se asignaron identificadores únicos a cada tabla para garantizar la unicidad de los registros.

Claves foráneas Se establecieron relaciones entre tablas (por ejemplo, vinculando LOANS con BOOKS y MEMBERS) para mantener la integridad referencial.

Restricciones Se aplicaron reglas como NOT NULL y CHECK para asegurar la consistencia de los datos, por ejemplo, evitando fechas de devolución inválidas.

PostgreSQL fue seleccionado como el sistema de gestión de bases de datos por su capacidad de manejar grandes volúmenes de datos y su compatibilidad con consultas avanzadas lo que lo hace ideal para el sistema.

Configuración del entorno de desarrollo

- Se utilizaron contenedores podman para aislar los entornos de desarrollo, facilitando la portabilidad y la reaplicación del sistema en diferentes máquinas.

- Se instaló postgresQL en el sistema local, utilizando la versión más reciente para aprovechar sus mejoras de rendimiento y seguridad.

A continuación, se mostrará el schema utilizado para la creación de la base de datos.

```
-- Crear la tabla PUBLISHERS primero, ya que es referenciada por BOOKS
CREATE TABLE PUBLISHERS (
    publisher_id SERIAL PRIMARY KEY,
    name VARCHAR(100),
    address TEXT,
    city VARCHAR(50),
    country VARCHAR(50),
    contact_email VARCHAR(100),
    contact_phone VARCHAR(20)
);

-- Crear la tabla AUTHORS
CREATE TABLE AUTHORS (
    author_id SERIAL PRIMARY KEY,
    first_name VARCHAR(50),
    last_name VARCHAR(50),
    birth_date DATE,
    death_date DATE,
    nationality VARCHAR(50),
    biography TEXT
);

-- Crear la tabla CATEGORIES (auto-referenciada para categorías jerárquicas)
CREATE TABLE CATEGORIES (
    category_id SERIAL PRIMARY KEY,
    name VARCHAR(50),
    parent_category_id INTEGER,
    description TEXT,
    FOREIGN KEY (parent_category_id) REFERENCES CATEGORIES(category_id)
);

-- Crear la tabla BOOKS
CREATE TABLE BOOKS (
    book_id SERIAL PRIMARY KEY,
    title VARCHAR(255),
    genre VARCHAR(50),
    publication_date DATE,
    publisher_id INTEGER,
    available_copies INTEGER,
    location_code VARCHAR(50),
    summary TEXT,
    FOREIGN KEY (publisher_id) REFERENCES PUBLISHERS(publisher_id)
);

-- Crear la tabla BOOK_AUTHORS para la relación muchos-a-muchos entre BOOKS y
  AUTHORS
CREATE TABLE BOOK_AUTHORS (
```

```

    book_id INTEGER,
    author_id INTEGER,
    PRIMARY KEY (book_id, author_id),
    FOREIGN KEY (book_id) REFERENCES BOOKS(book_id),
    FOREIGN KEY (author_id) REFERENCES AUTHORS(author_id)
);

-- Crear la tabla BOOK_CATEGORIES para la relacion muchos-a-muchos entre BOOKS
y CATEGORIES
CREATE TABLE BOOK_CATEGORIES (
    book_id INTEGER,
    category_id INTEGER,
    extension_count INTEGER,
    PRIMARY KEY (book_id, category_id),
    FOREIGN KEY (book_id) REFERENCES BOOKS(book_id),
    FOREIGN KEY (category_id) REFERENCES CATEGORIES(category_id)
);

-- Crear la tabla MEMBERS
CREATE TABLE MEMBERS (
    member_id SERIAL PRIMARY KEY,
    first_name VARCHAR(50),
    last_name VARCHAR(50),
    email VARCHAR(100),
    phone VARCHAR(20),
    address TEXT,
    join_date DATE,
    membership_end_date DATE,
    membership_type VARCHAR(20),
    max_loans INTEGER,
    status VARCHAR(20)
);

-- Crear la tabla LOANS
CREATE TABLE LOANS (
    loan_id SERIAL PRIMARY KEY,
    book_id INTEGER,
    member_id INTEGER,
    loan_date DATE,
    due_date DATE,
    return_date DATE,
    FOREIGN KEY (book_id) REFERENCES BOOKS(book_id),
    FOREIGN KEY (member_id) REFERENCES MEMBERS(member_id)
);

-- Crear la tabla FINES
CREATE TABLE FINES (
    fine_id SERIAL PRIMARY KEY,
    loan_id INTEGER,
    amount DECIMAL(10,2),
    reason VARCHAR(255),

```



```

    issue_date DATE,
    payment_date DATE,
    status VARCHAR(20),
    FOREIGN KEY (loan_id) REFERENCES LOANS(loan_id)
);

-- Crear la tabla RESERVATIONS
CREATE TABLE RESERVATIONS (
    reservation_id SERIAL PRIMARY KEY,
    book_id INTEGER,
    member_id INTEGER,
    reservation_date DATE,
    expiry_date DATE,
    status VARCHAR(20),
    FOREIGN KEY (book_id) REFERENCES BOOKS(book_id),
    FOREIGN KEY (member_id) REFERENCES MEMBERS(member_id)
);

-- Crear la tabla STAFF
CREATE TABLE STAFF (
    staff_id SERIAL PRIMARY KEY,
    username VARCHAR(50),
    password_hash VARCHAR(255),
    first_name VARCHAR(50),
    last_name VARCHAR(50),
    email VARCHAR(100),
    role VARCHAR(20),
    status VARCHAR(20)
);

-- Crear la tabla AUDIT_TRAIL
CREATE TABLE AUDIT_TRAIL (
    audit_id SERIAL PRIMARY KEY,
    table_name VARCHAR(50),
    record_id INTEGER,
    action VARCHAR(20),
    changed_by VARCHAR(50),
    changed_at TIMESTAMP,
    old_values JSONB,
    new_values JSONB
);

```

Diagrama de flujo del sistema

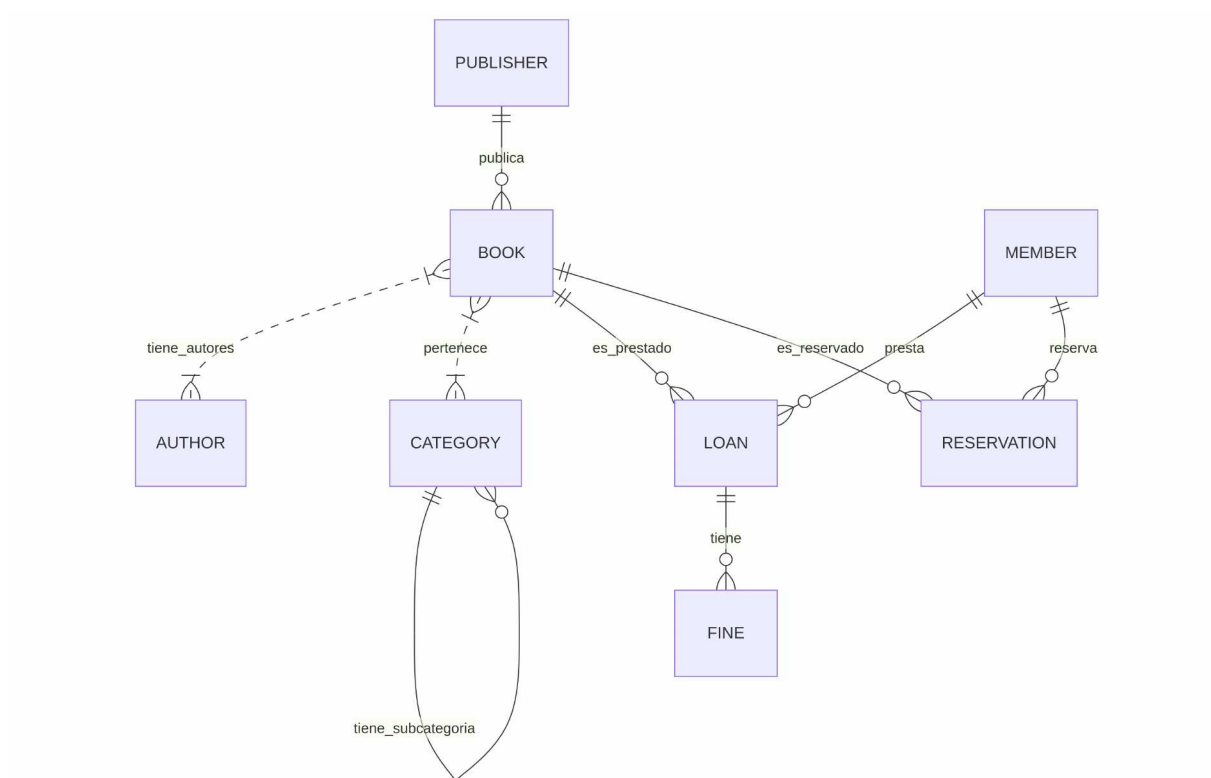


Figura 3: Diagrama de flujo del sistema

Mockups

A continuación, una demostración general de cómo podría verse una aplicación de dicho sistema de gestión de una biblioteca.

¡A leer!

¿Quieres leer algo nuevo pero te da tirria visitarnos? ¡No hay problema! Estrenamos nueva sección en la cual puedes pedir préstamos de libros virtuales.


 ¡Veamos!

Nuevos libros

<div>Título</div> <div>AutorAño</div>	<div>Título</div> <div>AutorAño</div>	<div>Título</div> <div>AutorAño</div>	<div>Título</div> <div>AutorAño</div>
<div>Título</div> <div>AutorAño</div>	<div>Título</div> <div>AutorAño</div>	<div>Título</div> <div>AutorAño</div>	<div>Título</div> <div>AutorAño</div>

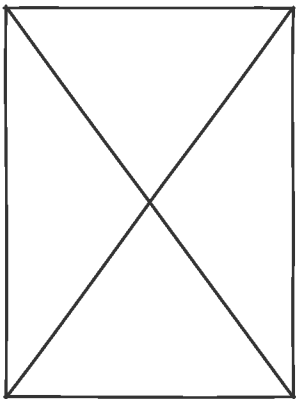
Filtros [Borrar](#)

Más nuevos 

Acción 

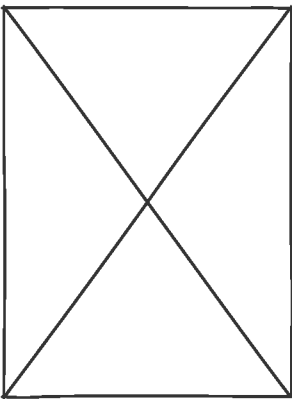
Género

- ☒ Acción
- ☐ Romance
- ☐ Sci-Fi
- ☐ Filosofía
- ☐ Otros



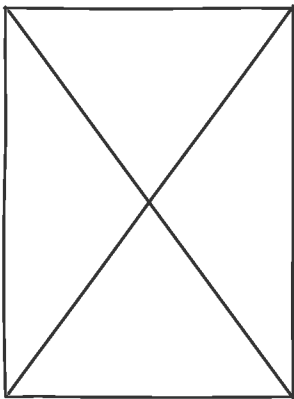
Título
(Año)

Autor



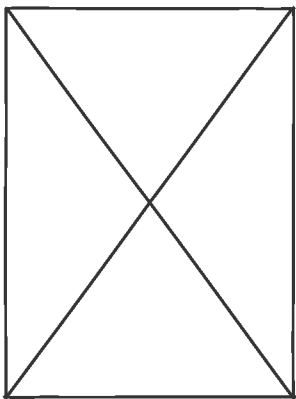
Título
(Año)

Autor



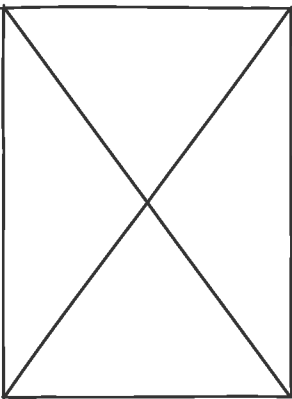
Título
(Año)

Autor



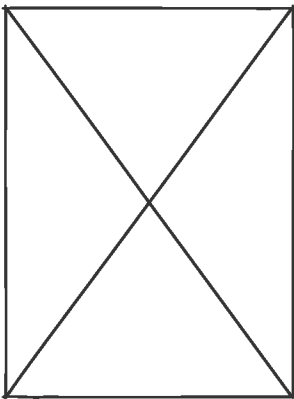
Título
(Año)

Autor



Título
(Año)

Autor






Título
(Año)

Autor



[Home](#) > [Category X](#) > Product Title



Título (Año)

Versión [¿Qué escoger?](#)

 Digital

 Física

Editorial

▼

Días a pedir prestado

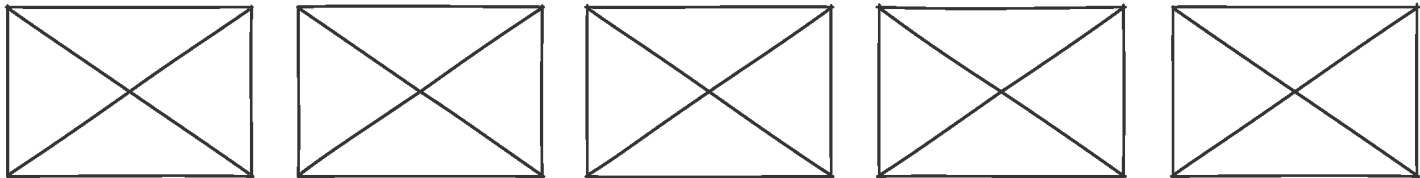
1

▼

Add To Cart

Si no sabes qué libros escoger, ¡puedes ir a "los más leídos" e inspirarte!




Related Products



Email

Suscribirse

[Inicio](#) > [Categoría](#) > Título



Libro añadido

Título de libro (Edición, año)
Autor | x días préstamo

Ver carrito

Préstamo

Días que lo pedirás prestado

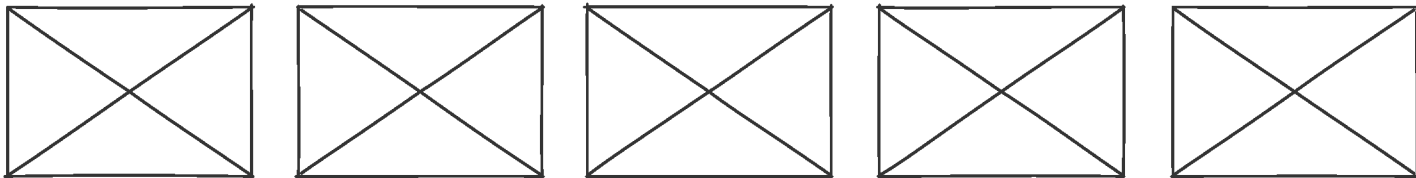
1

▼


Pedir prestado

¡Parece que has tomado una decisión asertiva!

¡Puede te gusten estos libros!



Carrito

Libro	Días
<div><div></div><div><div>Título (Año)</div><div>Autor, Editorial</div></div></div> <div><div>1</div><div>▼</div></div>	
Total: \$00.00	

Total (1 libro)

Descuento

\$00.00

Envío

\$00.00

Pago por día excedido

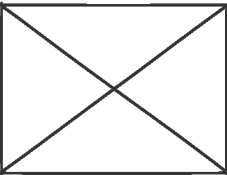
\$00.00

Total

\$0

¡Presta pa'la orquesta!

Carrito

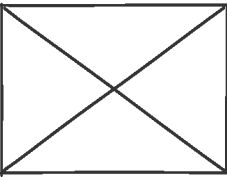


Título (año)
Autor, editorial

\$1

Días

Borrar

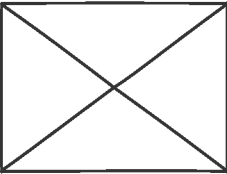


Título (año)
Autor, editorial

\$2

Días

Borrar



Título (año)
Autor, editorial

\$3

Días

Borrar

Subtotal: \$70.00

Envío: \$0.00

Multa \$5.00

Estimate

\$11.00

Código de estudiante = descuento

Apply

Pedir prestado

 Pagar

Pagar con  Pay

Carrito > Información de cliente > Información de envío > Método de pago

Información de cliente

Ya tienes cuenta? [Iniciar sesión](#)

Email

Información de envío

Nombre(s)

Apellido(s)

Institución (opcional)

Dirección

Número (opcional)

Calle

Colonia

C.P.

< [Volver al carrito](#)

[Continuar a métodos de envío](#)

Total (1 libro)

Subtotal

\$20.00

Envío

-

Multa

-

Código de estudiante = descuento

Apply

Total

\$20.00



¡Suscríbete para más sorpresas!

Email

Suscribirse

Carrito > Información de cliente > **Información de envío** > Método de pago

Dirección de envío calle, número, colonia, C.P.

[Edit](#)

Método de envío

- ☒ Recoger en tienda
- \$0
- ☐ Recoger en punto medio
- \$5
- ☐ Servicio de paquetería
- \$10
- ☐ Descargar versión digital
- \$0

[< Volver a información de cliente](#)

Ir a método de pago

Total (1 libro)

Subtotal	\$20.00
Envío	\$2.20
Multa	\$1.40

Código de estudiante = descuento

Apply

Total **\$23.60**



¡Suscríbete para más sorpresas!

Email

Suscribirse

Carrito > Información de cliente > **Información de envío** > Método de pago

Dirección lugar, lugar, lugar

[Editor](#)

Método de envío recoger/envío paquetería

[Editor](#)

Método de pago

Número de tarjeta



Nombre

MM/YY

CVV



Dirección de envío

- ☒ La misma que la tarjeta
- ☐ Una dirección distinta

Recuérdame

☐ Guardar mi información para usarla después

[< Volver a información de cliente](#)

Completar préstamo

Total (1 libro)

Subtotal	\$20.00
Envío	\$2.20
Multa	\$1.40

Código de estudiante = descuento

Apply

Total **\$23.60**



¡Suscríbete para más sorpresas!

Email

Suscribirse



Préstamo 131217312

Gracias, fulanitx!

Tu préstamo está confirmado.

Hemos aceptado tu orden y estamos preparándola.

Información

Dirección

Calle

Método de envío

Recoger en biblioteca

Método de pago

 Termina en 1234 — \$23.60

total (1 libro)

Subtotal	\$20.00
Envío	\$2.20
Multa	\$1.40

Código de estudiante = descuento

Apply

Total \$23.60



Conclusión

El desarrollo del sistema ha culminado en una solución integral y funcional que aborda de manera eficiente los retos de gestión de biblioteca moderna. A través de un enfoque estructurado, se logró construir un sistema capaz de administrar libros, miembros, préstamos y reservas.

Referencias

- Elmasri, R., Navathe, S. B. (2015). *Fundamentals of database systems* (7th ed.). Pearson.
- Connolly, T., Begg, C. (2014). *Database systems: A practical approach to design, implementation, and management*. Pearson Education.