Library Management System (A Microservices-based Application)

By- Aman Yadav

Library Management System

This Library Management System is designed using microservices architecture. It manages all essential library operations such as book inventory, member management, and admin functionalities.

Features:

- Users can search, borrow, and return books.
- Members can view their profiles and issued books.
- Admins can manage books, members, and overall system settings.

Tech Stack:

- Backend → Flask (Python)
- Frontend → Angular
- Databases → Separate DB per service
- Deployment → Docker & Docker Compose

Architecture Used – Microservices

- Each service is **independent** and **loosely coupled**.
- Services communicate via **REST APIs** through a central **Gateway Service**.
- Each service has its **own database**, ensuring data isolation.

Advantages:

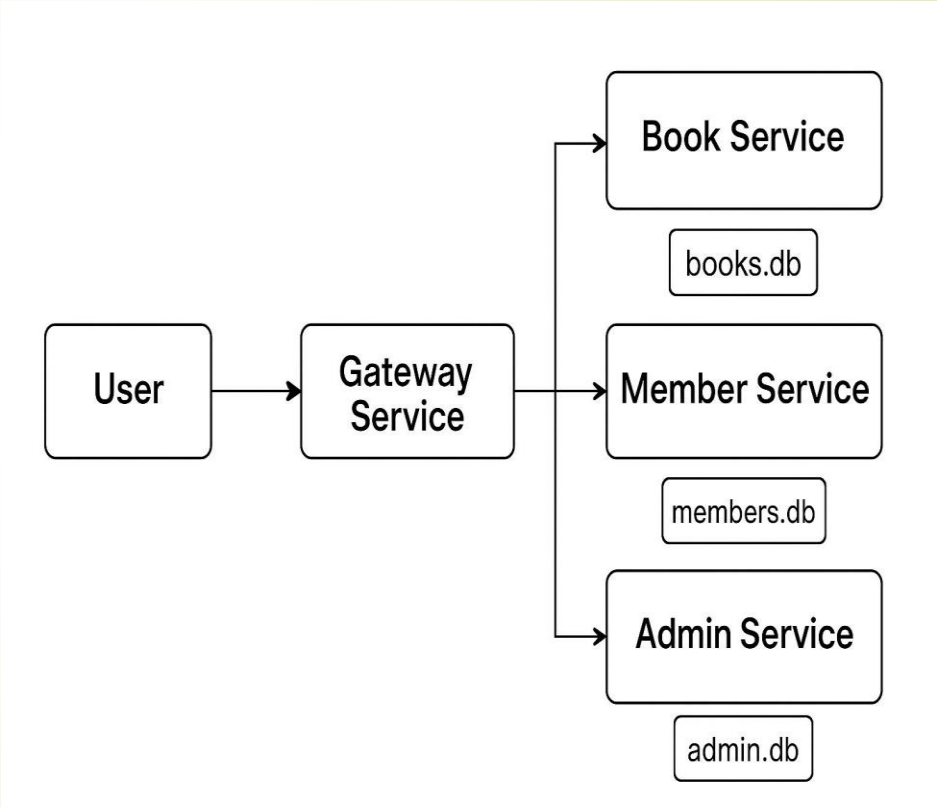
- **Scalability** → Services can be scaled independently.
- **Fault Isolation** → Failure in one service doesn't affect others.
- **Faster Development** → Teams can work on different services in parallel.

Architecture Diagram

User interacts via **Gateway**.
Gateway routes requests to:

- **Book Service** → Handles books CRUD
- **Member Service** → Manages users/members
- **Admin Service** → Admin controls & authentication

Each service connects to its **own database**.



Service Overview & APIs

Book Service

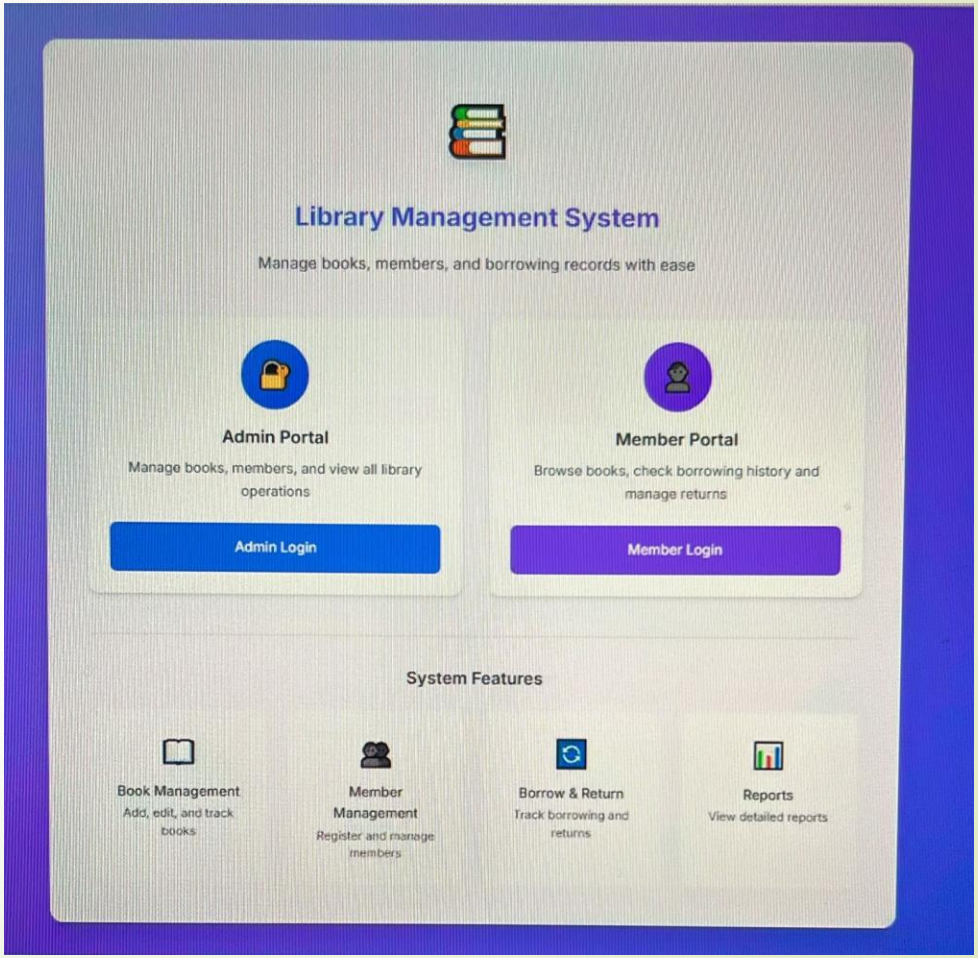
- GET /books → List all books
- POST /books → Add new book
- PUT /books/{id} → Update book
- DELETE /books/{id} → Delete book

Member Service

- GET /members → List all members
- GET /members/{id} → Member by ID
- POST /members → Add new member

Admin Service

- POST /register → Register new admin
- POST /login → Authenticate admin



Containerization

Each microservice is packaged into a Docker container.

Containers created:

- Book Service Container
- Member Service Container
- Admin Service Container
- Gateway/Frontend Container

Benefits:

- Consistent across environments.
- Easy to update/scale services.
- Portable and lightweight.

Deployment - Docker Compose

- Multi-container setup managed via docker-compose.yml.
- One command starts everything:

  docker-compose up
- If you want to force rebuild everything (ignoring cache), use:

  docker-compose build --no-cache

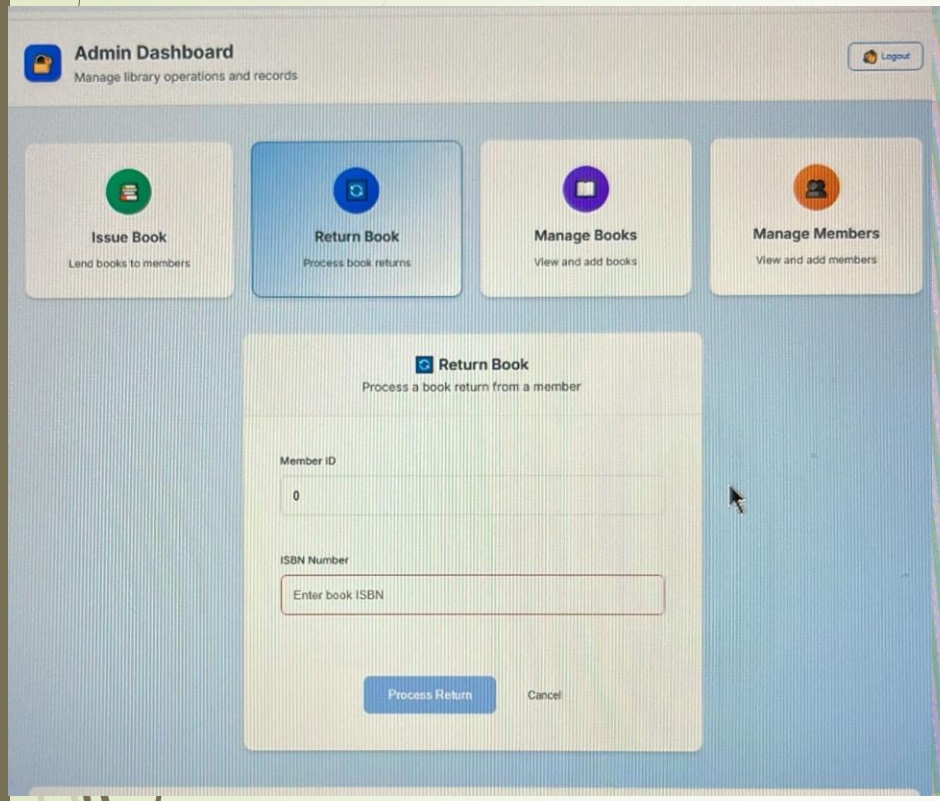
Advantages:

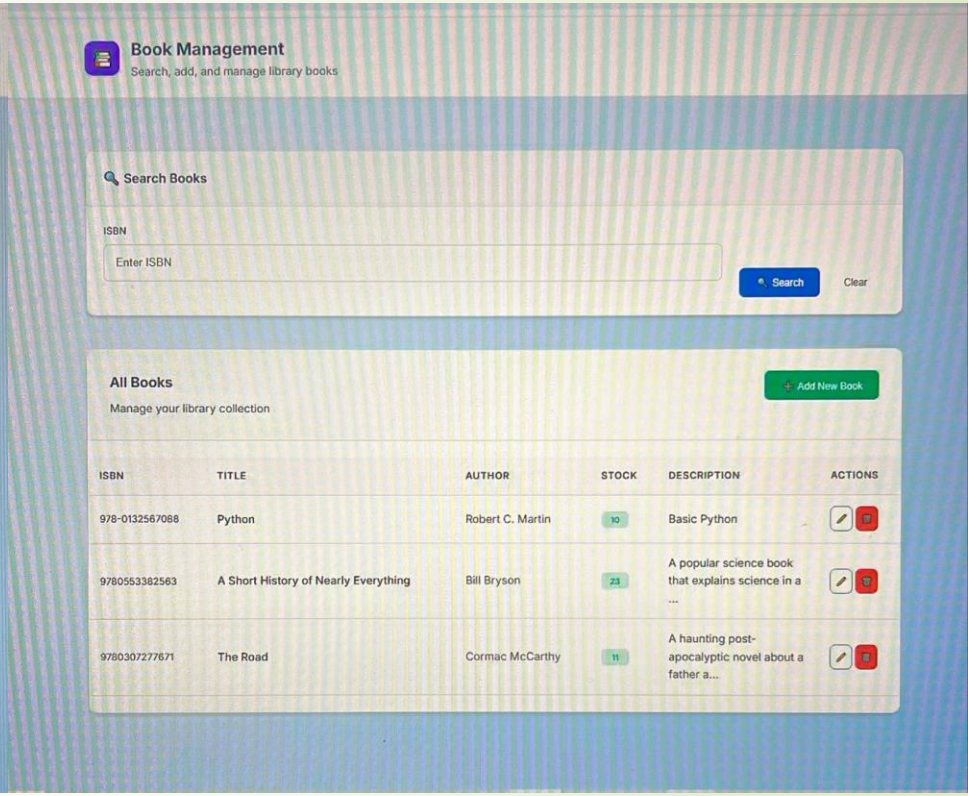
- Simplifies multi-service management.
- Handles service dependencies automatically.
- Allows scaling specific services without affecting others.

After that, your services will be running at:

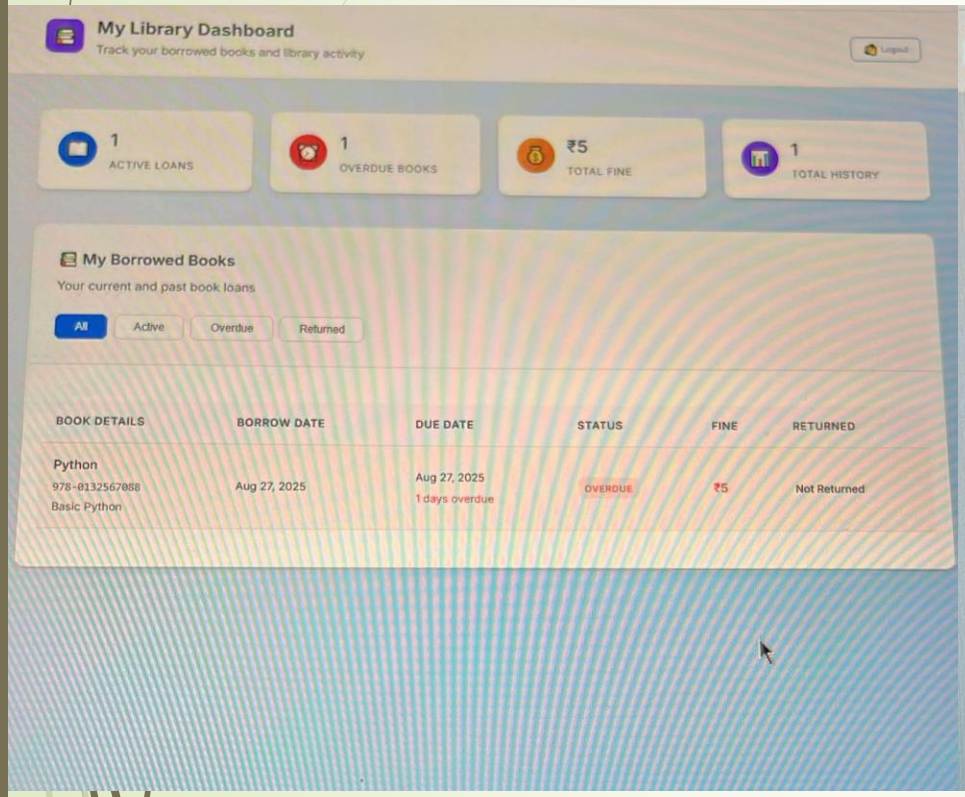
- http://localhost:5001 → Book Service
- http://localhost:5002 → Member Service
- http://localhost:5003 → Admin Service
- http://localhost:5000 → Gateway

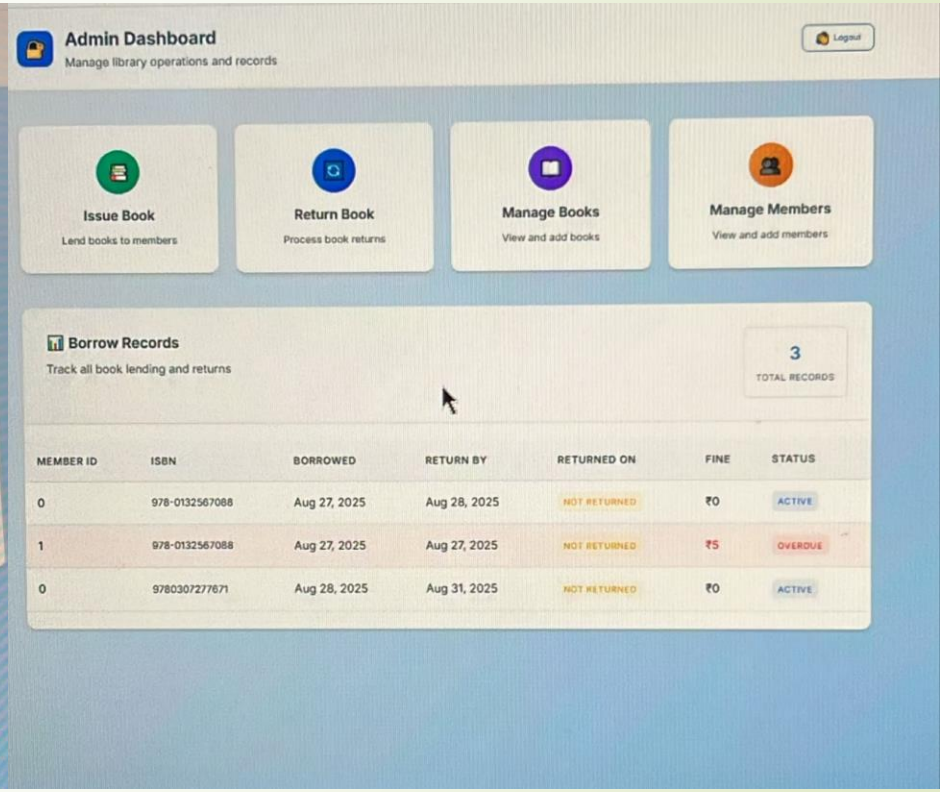
RESULTS


Admin Dashboard
Manage library operations and records
Logout
Issue Book
Lend books to members
Return Book
Process book returns
Manage Books
View and add books
Manage Members
View and add members
Return Book
Process a book return from a member
Member ID
0
ISBN Number
Enter book ISBN
Process Return
Cancel


Book Management
Search, add, and manage library books
Search Books
ISBN
Enter ISBN
Search
Clear
All Books
Manage your library collection
Add New Book
ISBN
TITLE
AUTHOR
STOCK
DESCRIPTION
ACTIONS
978-0132567088
Python
Robert C. Martin
10
Basic Python
9780553382563
A Short History of Nearly Everything
Bill Bryson
23
A popular science book that explains science in a ...
9780307277671
The Road
Cormac McCarthy
11
A haunting post-apocalyptic novel about a father a...

RESULTS


My Library Dashboard
Track your borrowed books and library activity
Logout
1
ACTIVE LOANS
1
OVERDUE BOOKS
₹5
TOTAL FINE
1
TOTAL HISTORY
My Borrowed Books
Your current and past book loans
All
Active
Overdue
Returned
BOOK DETAILS
BORROW DATE
DUE DATE
STATUS
FINE
RETURNED
Python
978-0132567088
Basic Python
Aug 27, 2025
Aug 27, 2025
1 days overdue
OVERDUE
₹5
Not Returned


Admin Dashboard
Manage library operations and records
Logout
Issue Book
Lend books to members
Return Book
Process book returns
Manage Books
View and add books
Manage Members
View and add members
Borrow Records
Track all book lending and returns
3
TOTAL RECORDS
MEMBER ID
ISBN
BORROWED
RETURN BY
RETURNED ON
FINE
STATUS
0
978-0132567088
Aug 27, 2025
Aug 28, 2025
NOT RETURNED
₹0
ACTIVE
1
978-0132567088
Aug 27, 2025
Aug 27, 2025
NOT RETURNED
₹5
OVERDUE
0
9780307277671
Aug 28, 2025
Aug 31, 2025
NOT RETURNED
₹0
ACTIVE

THANK YOU