

Maids Server Application Documentation

Library Management System using Spring Boot

Prerequisites

1. install java17 and add it to Environment Variables.
2. install maven and add it to Environment Variables.
3. install mySql and add it to Environment Variables (optional).

Running Application

1. Navigate to your project directory in your terminal.
2. Execute the following command: **mvn clean install**
3. Execute the following command: **mvn spring-boot:run**

Or use this command instead: **java -jar target/Maids.jar**

Authentication

- **Basic Authentication** with username (**maids**) and password (**password**) is implemented and needs to be added in the header of any endpoint request except createBook and getAllBooks endpoints which is a public endpoint.

Interacting with API Endpoints

- You can use tools like Postman to test APIs.
- There is a file (**Maids.postman_collection.json**) in the same folder containing a postman collection for all possible APIs with their authentication and all possible responses (success and failures) as examples.
- Just import the collection into postman and start using/testing predefined requests.

Additional Tips

- Application will be running using in-memory database (h2).
- You can use mySql database only if you add comments in properties file on:
 - **spring.datasource.driverClassName=org.h2.Driver**
 - **spring.datasource.url=jdbc:h2:mem:maids**

- And remove comment on these lines:
 - **spring.datasource.url=jdbc:mysql://localhost:3306/maids?createDatabaseIfNotExist=true&characterEncoding=UTF-8&characterSetResults=UTF-8**
 - **spring.jpa.properties.hibernate.dialect=org.hibernate.dialect.MySQL57Dialect**
- Then create a db user with username (**maids**) and password (**m@iDsP@s\$**) to ensure connectivity of application to database.