# Book Management System

## Project Overview

This is a Book Management System built using PHP with a RESTful API architecture. It includes User Management, Book Management, Borrow/Return Books, and Role-Based Authentication (Admin/User).

## Setup and Run Locally

### ****Prerequisites****

Before running the project, ensure you have the following installed:

* Install XAMPP for Apache & MySQL setup
* Install Composer
* Install Postman Desktop setup for testing API requests

### ****Clone the Repository****

cd C:\xampp\htdocs

https://github.com/mohit000011/book-management.git

cd book-management

### ****Install Dependencies****

composer install

### ****Set Up the Database****

* Open **phpMyAdmin** (http://localhost/phpmyadmin/).
* Create a new database: book\_management.
* Execute the following SQL script:

CREATE TABLE users (

id INT AUTO\_INCREMENT PRIMARY KEY,

username VARCHAR(255) NOT NULL UNIQUE,

email VARCHAR(255) NOT NULL UNIQUE,

password VARCHAR(255) NOT NULL,

createdat timestamp NOT NULL,

role ENUM('admin', 'user') NOT NULL DEFAULT 'user'

);

CREATE TABLE books (

id INT AUTO\_INCREMENT PRIMARY KEY,

title VARCHAR(255) NOT NULL,

author VARCHAR(255) NOT NULL,

genre VARCHAR(255) NULL,

published\_year INT NULL

createdat timestamp NOT NULL

);

CREATE TABLE borrowed\_books (

id INT AUTO\_INCREMENT PRIMARY KEY,

user\_id INT NOT NULL,

book\_id INT NOT NULL,

borrowed\_date TIMESTAMP DEFAULT CURRENT\_TIMESTAMP,

returned\_date TIMESTAMP NULL,

FOREIGN KEY (user\_id) REFERENCES users(id) ON DELETE CASCADE,

FOREIGN KEY (book\_id) REFERENCES books(id) ON DELETE CASCADE

);

### ****Configure Environment Variables****

Create a .env file in the root directory and add:

DB\_HOST=localhost

DB\_NAME=book\_management

DB\_USER=root

DB\_PASS=

JWT\_SECRET=secret\_key

### ****Start Apache & MySQL****

* Open **XAMPP Control Panel** → Start **Apache** & **MySQL**.

## API Endpoints & Testing

### ****User Management****

#### **Register a User**

* **Endpoint:** POST http://localhost/book-management/controllers/UserController.php?action=register
* **Body:** (JSON)

{

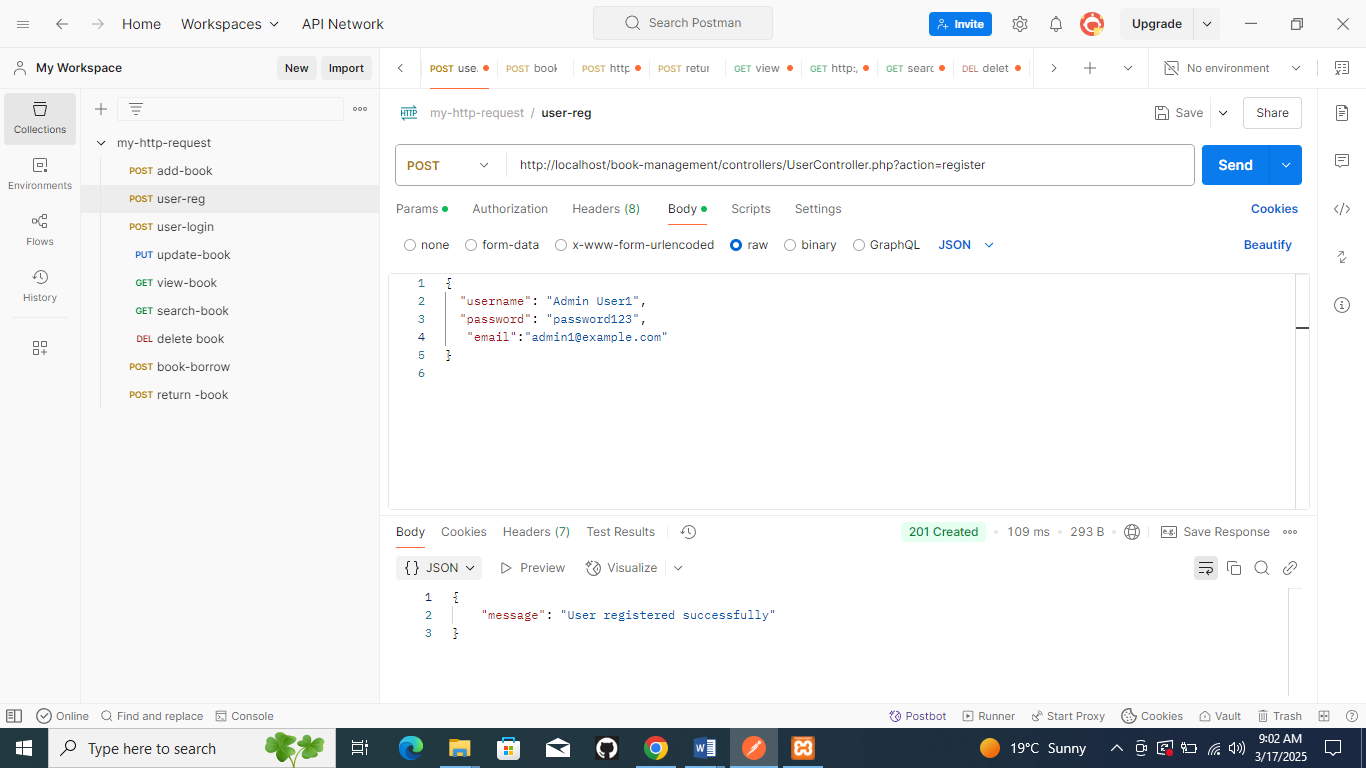
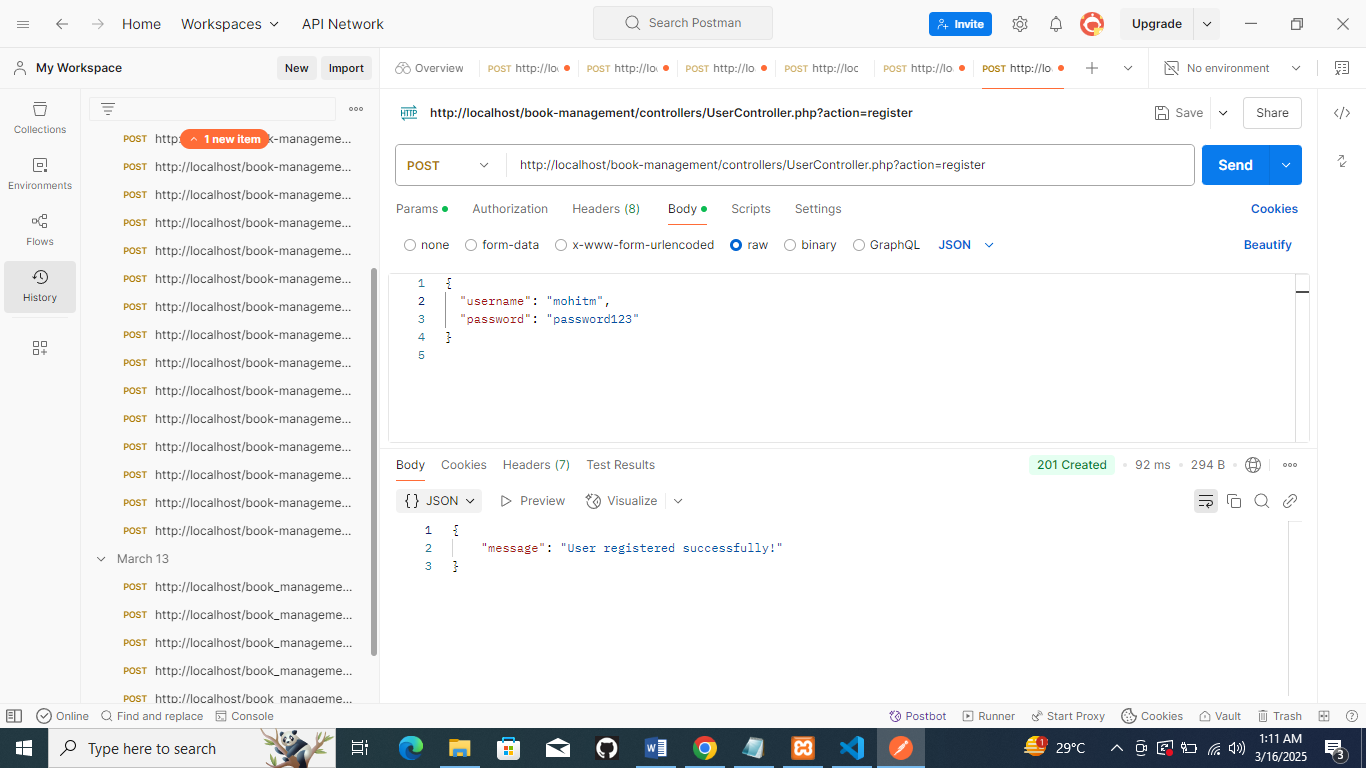
"username": "admin",

"email": "admin@example.com",

"password": "adminpassword",

"role": "admin"

}



#### 

#### **2. Login**

* **Endpoint:** POST http://localhost/book-management/user/login.php
* **Body:**

{

"email": "admin@example.com",

"password": "adminpassword"

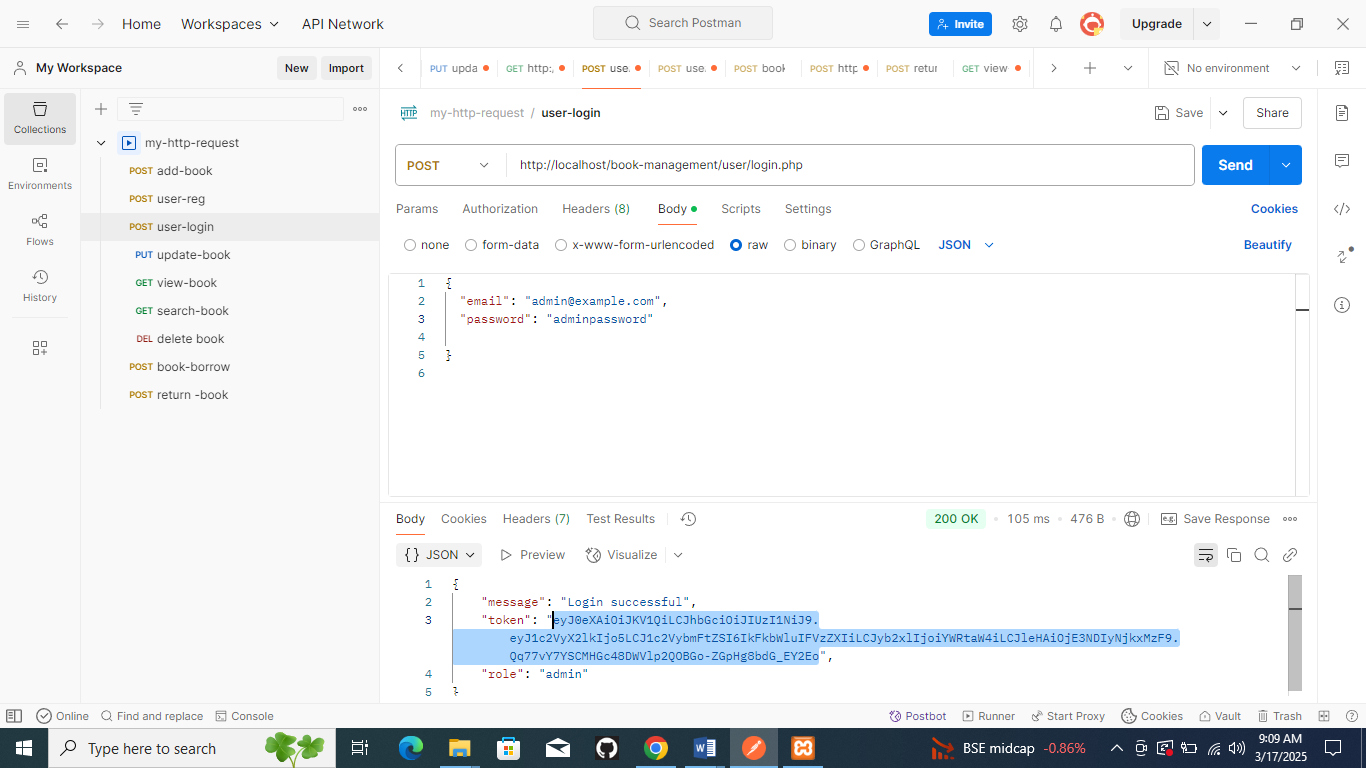
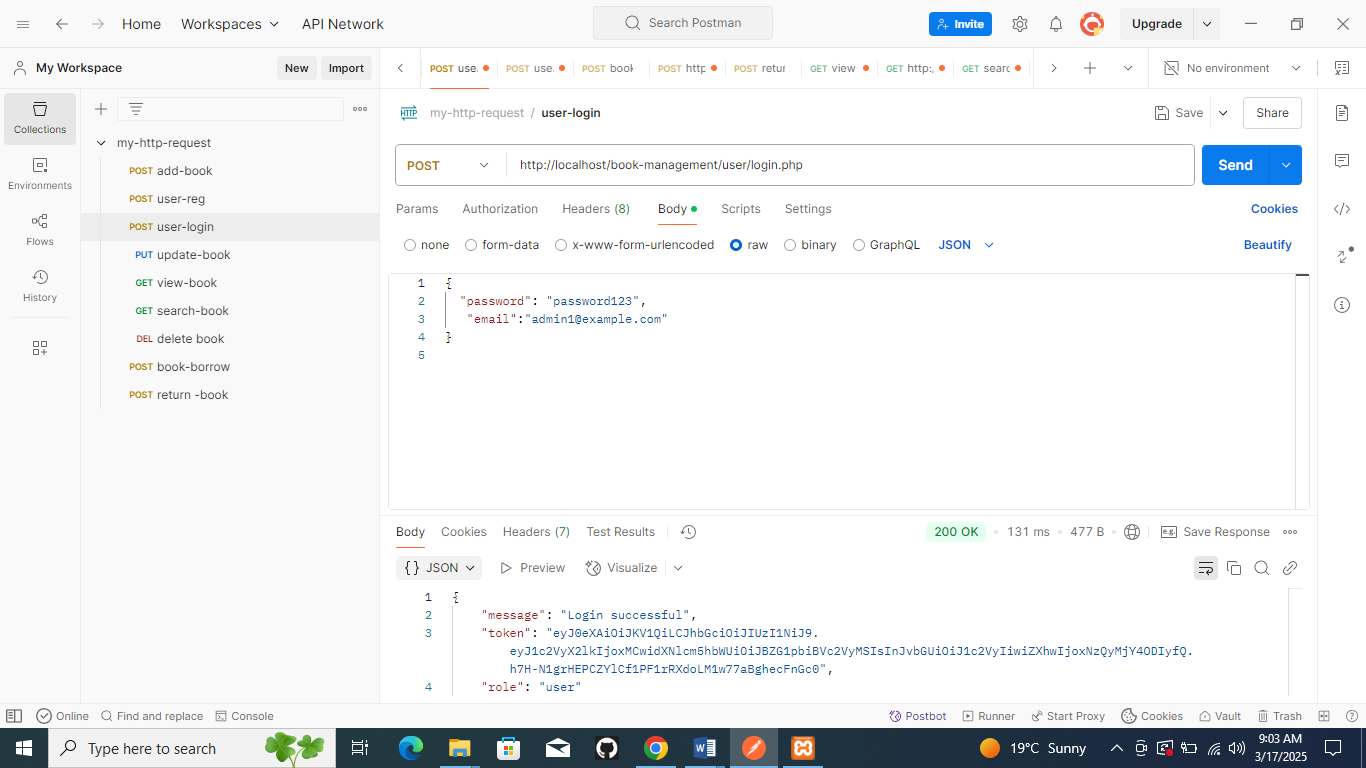
}

* **Response:**

{

"token": "jwt\_token"

}



### ****Book Management****

#### **Add a Book (Admin Only)**

* **Endpoint:** POST http://localhost/book-management/controllers/BookController.php?action=add
* **Headers:** Authorization: Bearer jwt\_token
* **Body:**

{

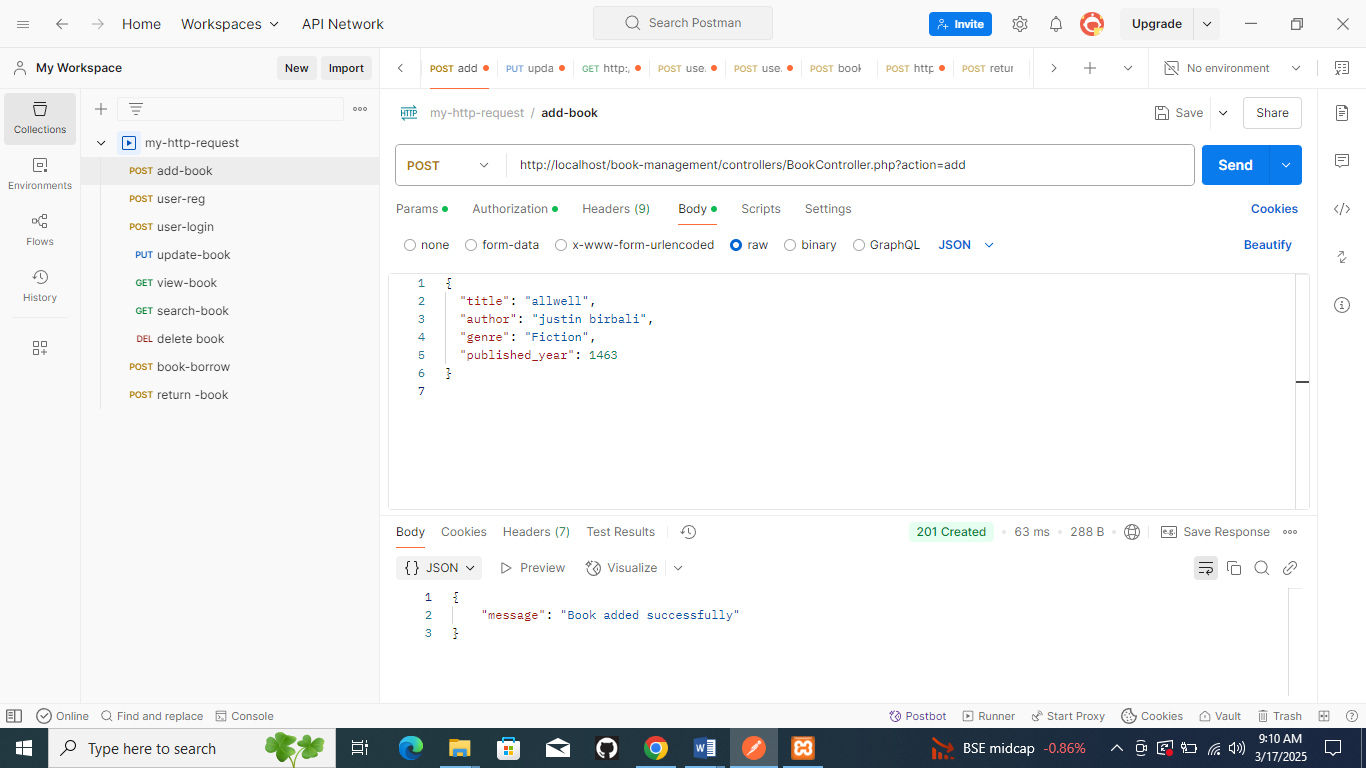
"title": "Book Title",

"author": "Author Name",

"genre": "Fiction",

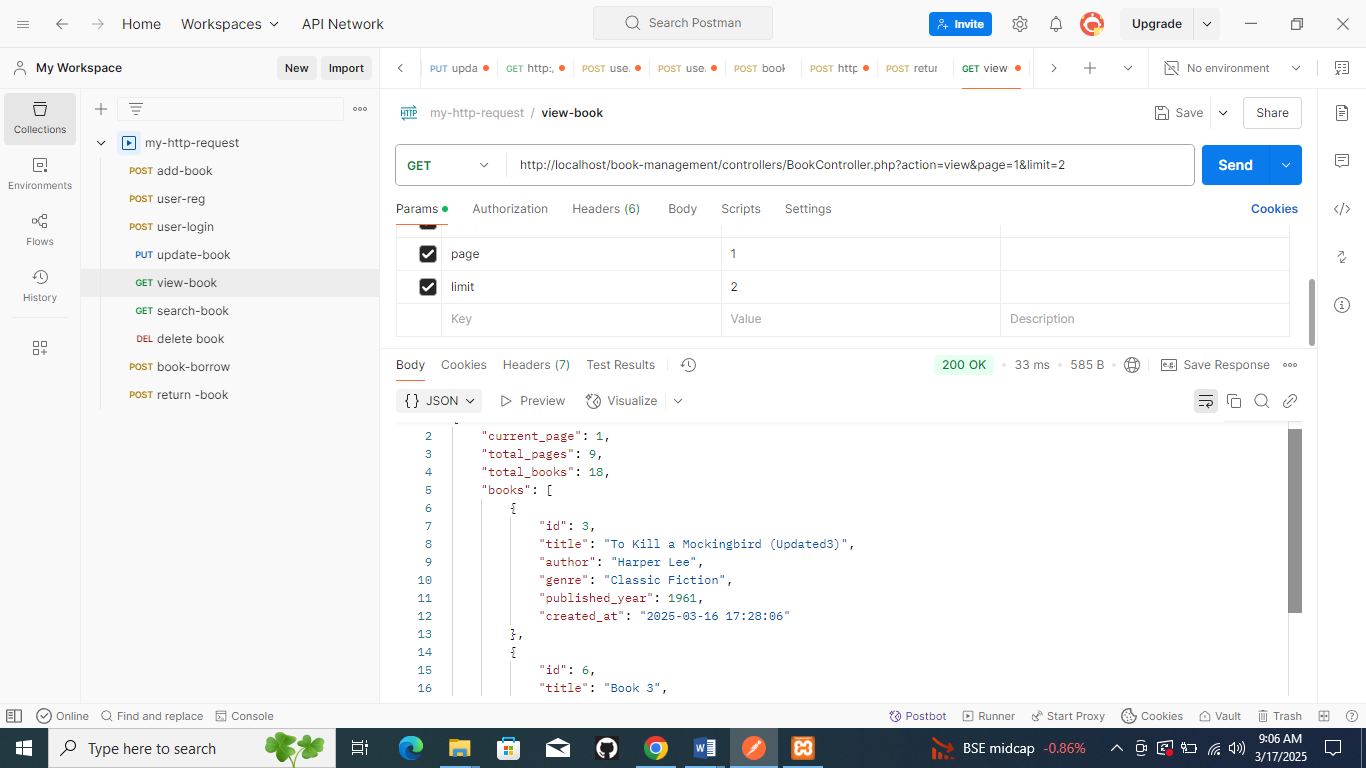
"published\_year": 2024

}



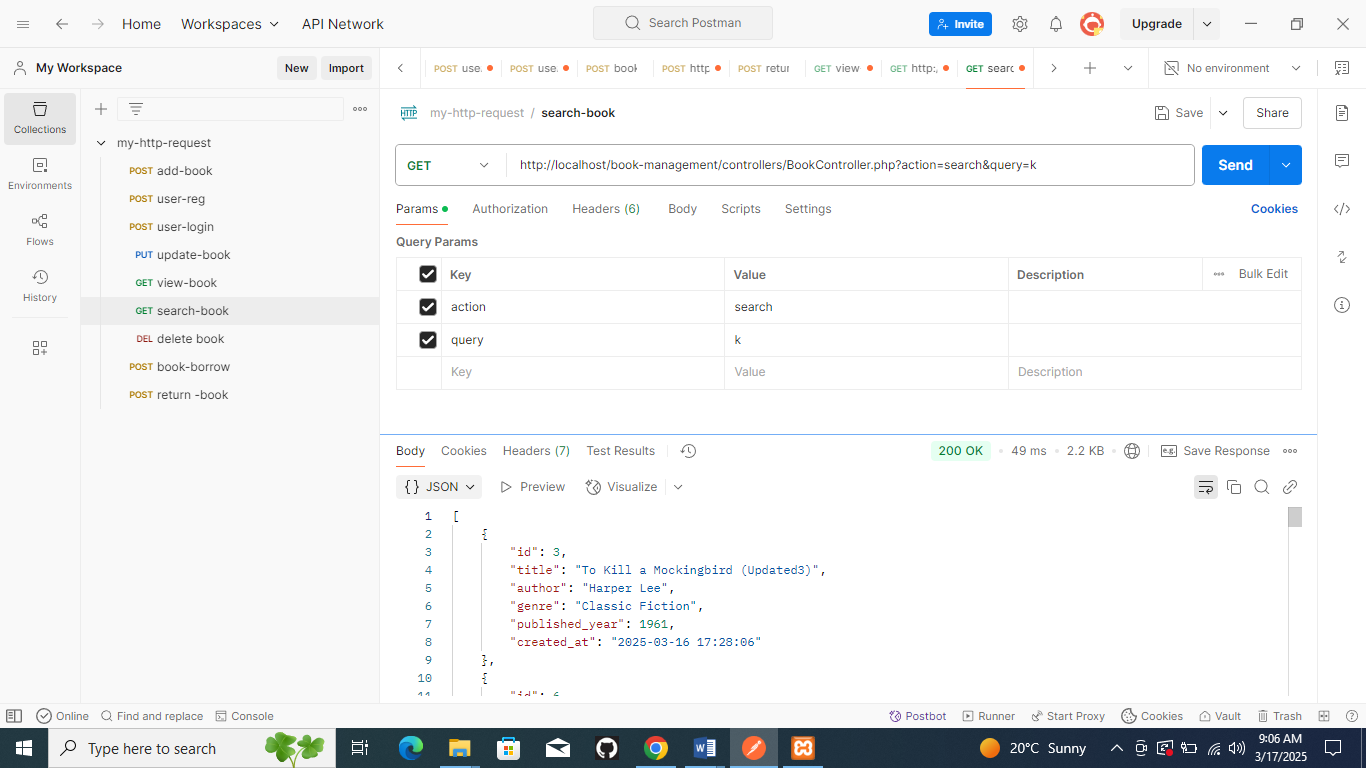
#### **View All Books with Pagination**

* **Endpoint:** GET http://localhost/book-management/controllers/BookController.php?action=view&page=1&limit=2



#### **Search Books**

* **Endpoint:** GET http://localhost/book-management/controllers/BookController.php?action=search&query=K



#### **Update Book Details (Admin Only)**

* **Endpoint:** PUT http://localhost/book-management/controllers/BookController.php?action=update
* **Headers:** Authorization: Bearer jwt\_token
* **Body:**

{

"id": 3,

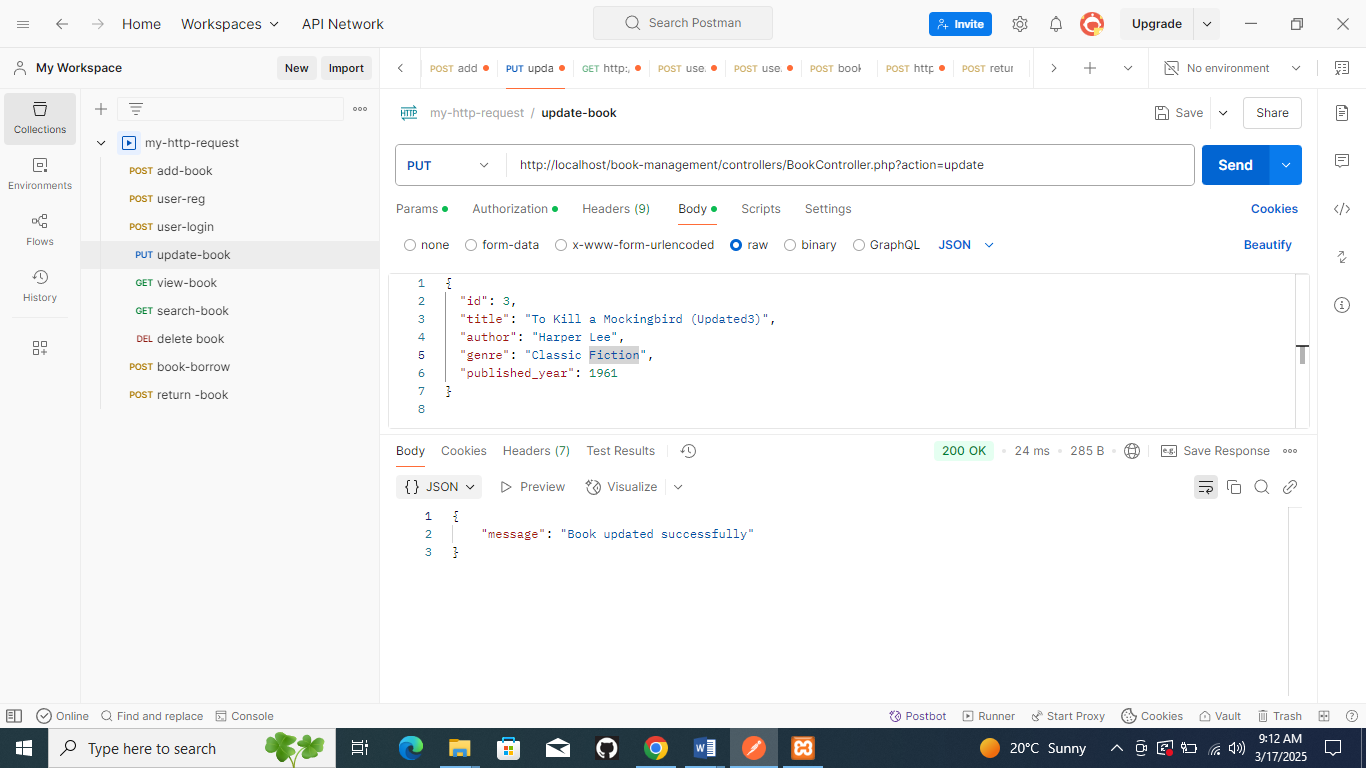
"title": "Updated Title",

"author": "Updated Author",

"genre": "Classic fiction",

"published\_year": 1961

}



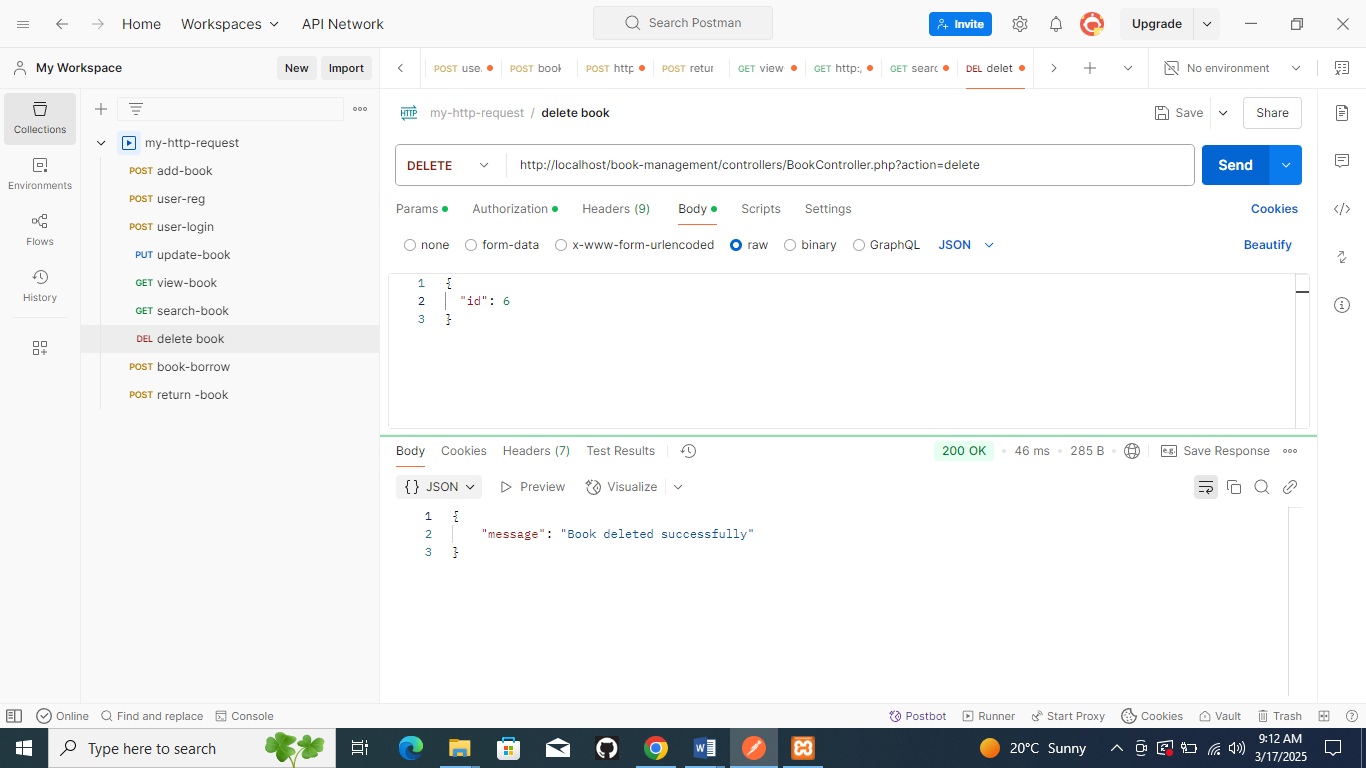
#### **Delete a Book (Admin Only)**

* **Endpoint:** DELETE http://localhost/book-management/controllers/BookController.php?action=delete
* **Headers:** Authorization: Bearer jwt\_token
* **Body:**

{

"id": 6

}



### ****Borrow & Return Books****

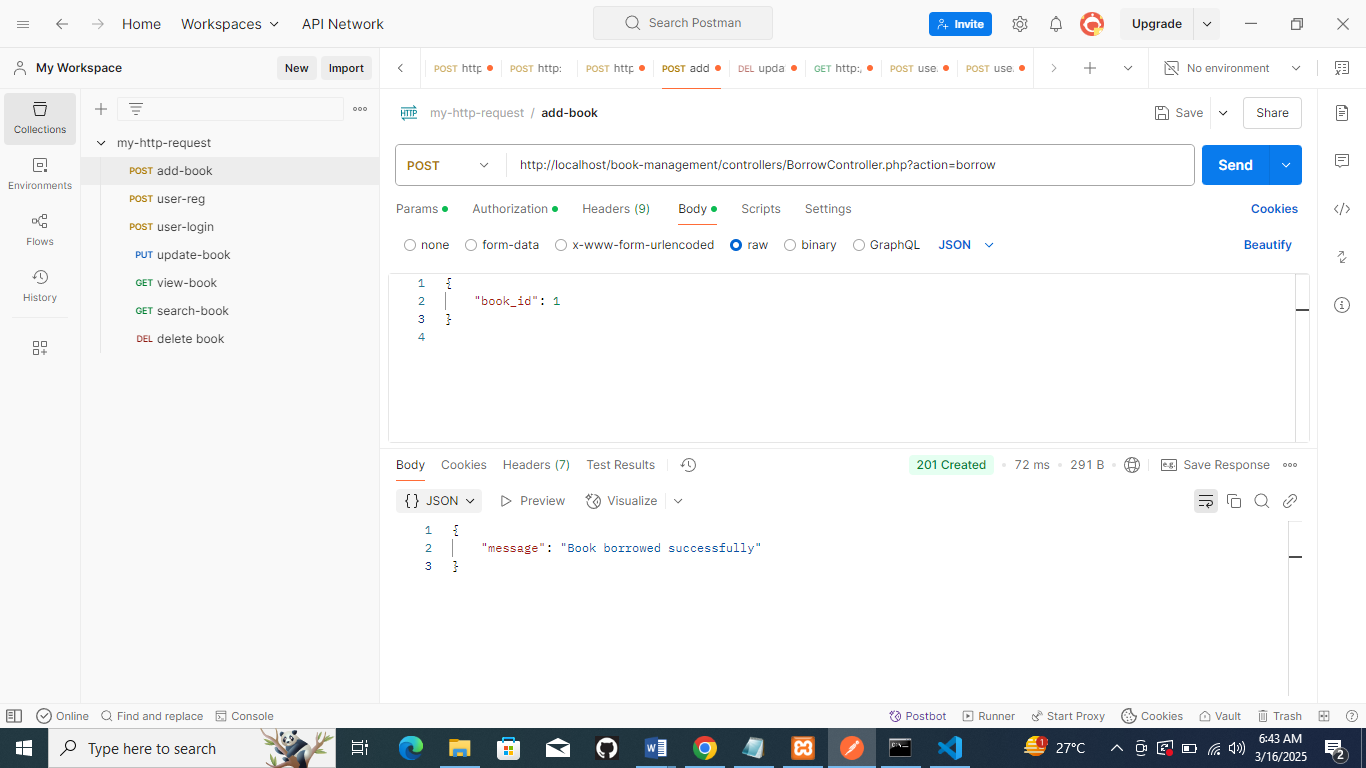
#### **Borrow a Book**

* **Endpoint:** POST http://localhost/book-management/controllers/BorrowController.php?action=borrow
* **Headers:** Authorization: Bearer jwt\_token
* **Body:**

{

"book\_id": 1

}



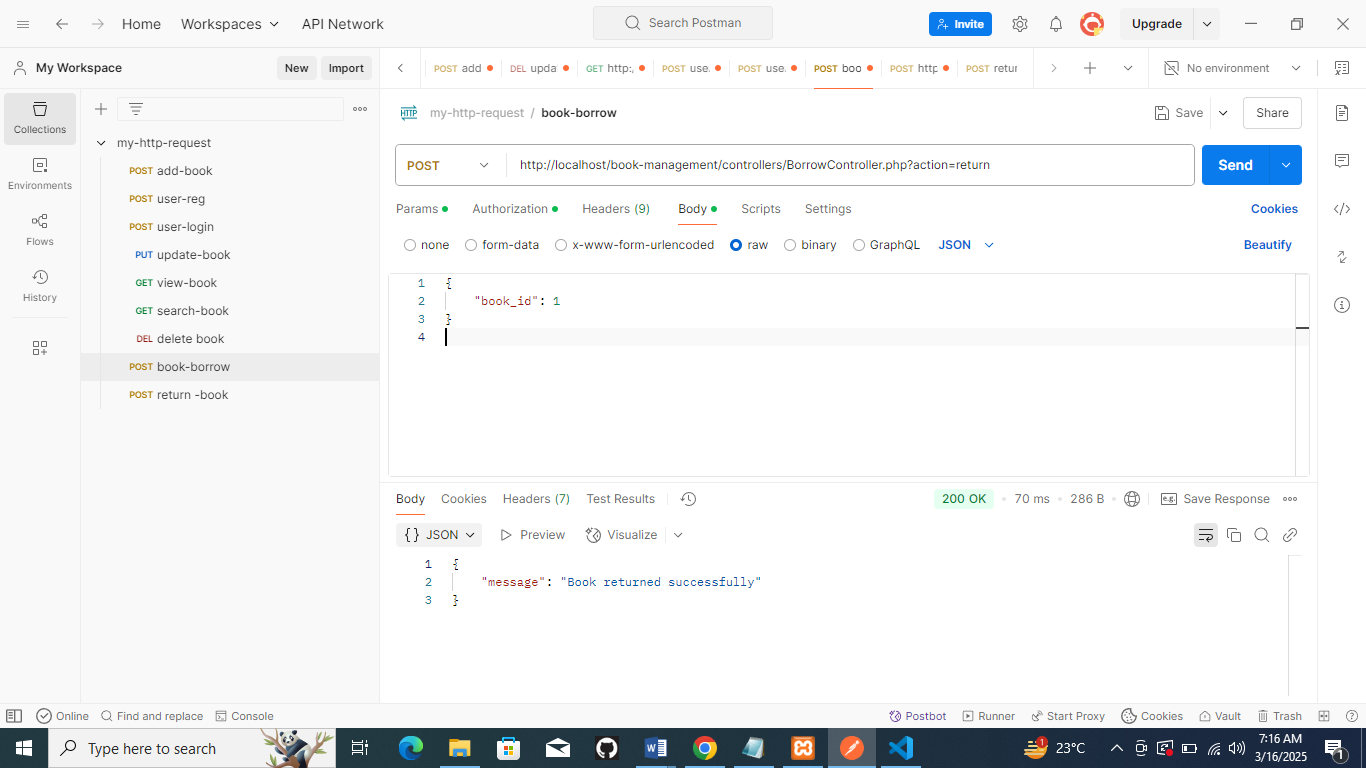
#### **Return a Book**

* **Endpoint:** POST http://localhost/book-management/controllers/BorrowController.php?action=return
* **Headers:** Authorization: Bearer jwt\_token
* **Body:**

{

"book\_id": 1

}



## Additional Features

* **Role-Based Access Control (Admin/User)**
* **Viewing Books with pagination**
* **Deployment on local server using XAMPP**