

FINTECH LAB 1 AND 2

KRISH KUMAR

240958254

71

Introduction

FinTech Lab I and II are designed to provide practical exposure to modern web based and financial application development. The laboratory focuses on hands on training in backend systems, REST based services, version control practices, and cloud based deployment techniques. Students actively use tools and frameworks that are widely adopted in the software and fintech industry.

Course Outcomes

The course helps students understand the core concepts of REST based web services. Learners develop backend applications using the Spring Boot framework. The lab offers practical experience with Git and GitHub for managing source code. Students learn how to deploy applications on Microsoft Azure. The course also strengthens debugging abilities, coding skills, and logical problem solving.

Software and Tools Required

The lab requires Node.js and Angular for development support. Visual Studio Code and Eclipse IDE are used as primary coding environments. MySQL is used for database management. Postman is used for API testing. Git and NVM are used for version control and environment management. Installation is confirmed using version commands such as node -v, npm -v, and git --version.

REST API Overview

REST API refers to Representational State Transfer Application Programming Interface. It allows interaction between client and server through HTTP based communication. Commonly used HTTP methods include GET for fetching data, POST for creating records, PUT for modifying existing data, and DELETE for removing data. REST APIs are stateless in nature, scalable in design, and reliable in performance. Data is usually exchanged in JSON format.

Maven Setup in Eclipse

Maven is installed through the Eclipse Marketplace. Once installed, Maven options become available within Eclipse preferences. Maven is used to manage project dependencies and automate the build process. It also helps in simplifying project configuration and long term maintenance.

Spring Boot Project Creation

The Spring Boot project is generated using Spring Initializr. Important configuration options include Maven as the project type, Java as the programming language, Java version 11 or 17, Jar packaging format, and Spring Web as the dependency. The generated project is then imported into Eclipse as an existing Maven project.

Running and Verifying the Application

The application is executed using the Run option in Eclipse. The embedded server starts on the default port number 8080. Accessing the URL <http://localhost:8080/> confirms that the application is running successfully. The white label error page indicates that the server has started correctly even though no endpoints are defined.

REST Controller Implementation

A basic REST controller is implemented using the `RestController` annotation. The `GetMapping` annotation is used to map HTTP GET requests. When the endpoint is accessed through a browser, a greeting message is returned. This confirms the successful implementation of a REST API.

Git and GitHub Usage

A Git repository is created using the `git init` command. Project files are added and committed with appropriate commit messages. The local repository is connected to GitHub using a remote origin. The code is pushed to the main branch to enable version tracking and backup.

Azure Cloud Deployment

Azure Cloud Shell is used to deploy the application to the cloud. The GitHub repository is cloned into the Azure environment. The Azure Web App Maven plugin is configured for deployment. Application settings include selecting Windows or Linux as the operating system, Java version 17, and Central India as the deployment region. The application is deployed using Maven based commands.

Important Spring Boot Annotations

`SpringBootApplication` is used as the main entry point of the application and enables automatic configuration. `RestController` is used to define REST based web services. `GetMapping` is used to map HTTP GET requests to controller methods.

Conclusion

The FinTech Lab offers strong practical training in backend application development and cloud technologies. Students gain valuable hands on experience with REST APIs, Spring Boot, Git, GitHub, and Azure based deployment.

HTTP My Collection / **Get data** Save Share

GET http://localhost:8080 Send

Docs **Params** Auth Headers (7) Body Scripts • Settings Cookies

Query Params

	Key	Value	Desc...	...	Bulk Edit
	Key	Value	Description		

Body Raw Preview Visualize 200 OK • 369 ms • 187 B • e.g. ...

```
1 Greetings from MIT-FIS!
```