

Detailed, step-by-step, build/installation/initialization/launch instructions

- Install the latest version of JDK.
- Node.js and npm:

Install Node.js and npm for managing frontend dependencies.

- MySQL Database:

Set up a MySQL database and note down the connection details.

Steps:

Part 1: Spring Boot Backend

1. Create a Spring Boot Project:

To start a new Spring Boot project, use your favorite IDE or Spring Initializer. Add dependencies such as MySQL, Spring Web, and Spring Data JPA.

2. Configure MySQL:

Enter your MySQL database connection information in the application.properties or application.yml file setup.

3. Create JPA Entity and Repository:

Provide JPA annotations to define the Employee entity. Establish a JPA repository for fundamental CRUD functions.

4. Build Get All Employee REST API:

Implement a REST API endpoint to fetch all employees.

5. Build Create and Update Employee REST APIs:

Implement REST API endpoints for creating and updating employees.

6. Build Get Employee by ID and Delete Employee REST APIs:

Implement REST API endpoints for fetching a single employee by ID and deleting an employee.

7. Test Your Backend:

Run your Spring Boot application and test the APIs using tools like Postman.

Part 2: React Frontend

1. Create React Application:

Use create-react-app or your preferred method to set up a new React application.

2. Add Bootstrap 5:

Integrate Bootstrap 5 into your React app. You can use npm to install Bootstrap.

3. Configure Routing:

Set up routing using react-router-dom to navigate between different components.

4. Create ListEmployeeComponent:

Build a component to display the list of employees using the useState hook.

5. Create AddEmployeeComponent:

Build a form component for adding new employees.

6. Connect React to Backend:

Use fetch or axios to connect your React app with the Spring Boot backend APIs.

7. Testing the Frontend:

Running the React application and ensure that it communicates properly with the backend.

8. Build and Launch

9. Build the Spring Boot JAR:

Use your IDE or Maven to build the Spring Boot project, resulting in a JAR file.

10. Run the Spring Boot Application:

Execute the JAR file or run the application from your IDE.

11. Build and Launch React App:

Use npm to build your React app (npm run build) and configure your Spring Boot application to serve the static files.

- **Access the Application:**

Open the browser and navigate to the specified URL (usually http://localhost:8080) to access the application.