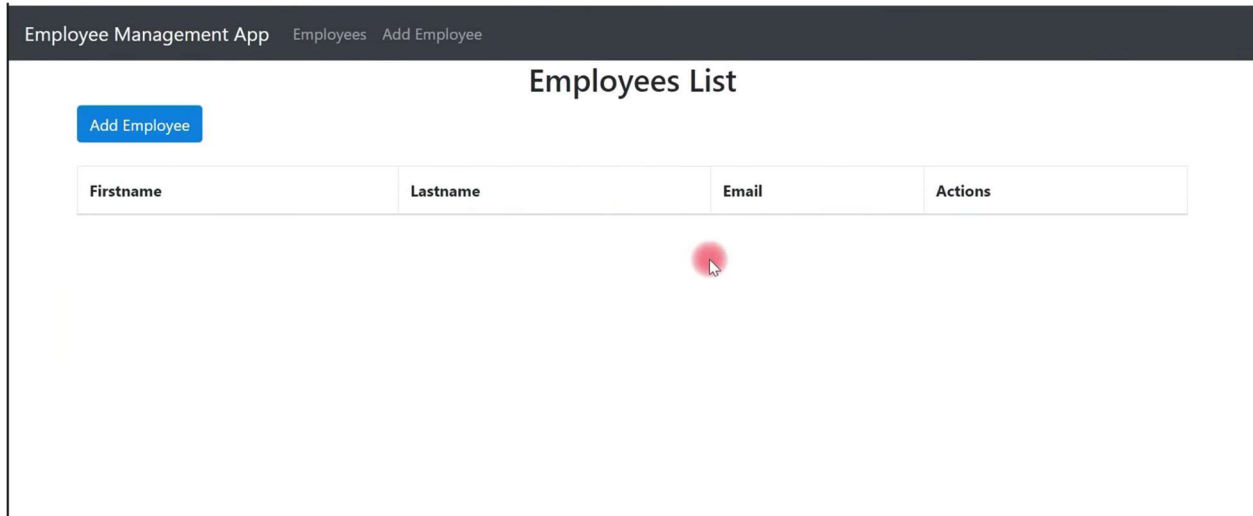


Employee Management System (Full Stack Application)

Project Description: The Employee Management System is a full-stack web application built with Spring Boot, React, Bootstrap, Axios, and MySQL. This application allows users to efficiently manage employee records. Users can add new employees, update their information, and delete employees as needed. The project integrates multiple technologies to create a seamless user experience for employee management.

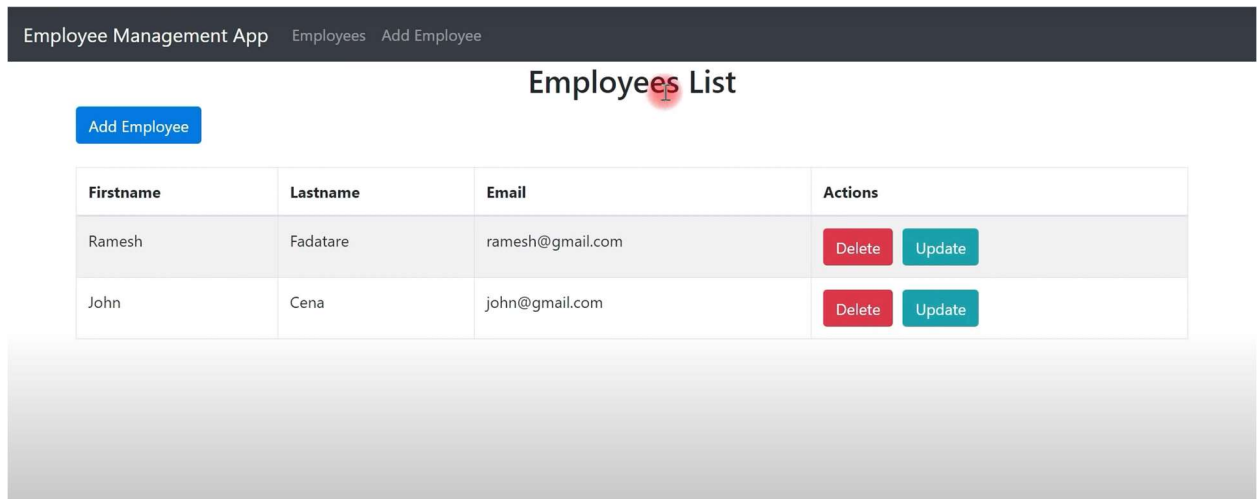
Key Features:



1. **Employee Addition:** Upon launching the application, users are presented with the option to add a new employee. They need to enter the first name, last name, and email of the employee. The entered data is stored in the MySQL database.
2. **Employee Listing:** The front page of the application displays the list of employees, including the employee details (first name, last name, and email) that were previously added. The list is dynamically rendered using React.

A screenshot of the 'Add Employee' form in a web application. The form has a dark header with the text 'Employee Management App', 'Employees', and 'Add Employee'. Below the header, the title 'Add Employee' is centered. The form contains three input fields: 'First Name', 'Last Name', and 'Email Id :'. The 'First Name' field is currently active, with a cursor visible. Below the input fields, there are two buttons: 'Save' (green) and 'Cancel' (red).

3. **Employee Update:** Users can choose to update the details of any employee from the list. They can modify the first name, last name, or email, and the changes are updated in the database using Spring Data JPA.



4. **Employee Deletion:** The application provides the option to delete any employee from the list. Upon deletion, the employee record is removed from the database.

Technologies Used:

1. **Spring Boot:** Used to create the backend of the application, handle HTTP requests, and manage the application's business logic. Spring Boot's rapid development capabilities, Spring Data JPA, and IoC container are utilized.
2. **Spring Data JPA:** Interacts with the MySQL database, performs CRUD operations, and manages entity relationships.
3. **React:** Developed the frontend user interface using React, allowing for a dynamic and responsive UI that enhances the user experience.
4. **Bootstrap:** Used to enhance the frontend design, providing responsive and visually appealing components.
5. **Axios:** An HTTP client library for making asynchronous HTTP requests from the frontend to the backend, enabling smooth communication between React and Spring Boot.
6. **MySQL Database:** Used to store employee data. MySQL is a robust relational database management system.
7. **RESTful APIs:** Spring Boot provides RESTful APIs to handle communication between the frontend (React) and backend (Spring Boot).

Application Setup:

1. Configure MySQL Database:

- Make sure you have MySQL installed and running on your machine.
- Create a new MySQL database for the project.

2. Configure Spring Boot Backend:

- Navigate to the backend directory
- Open **src/main/resources/application.properties** and configure the MySQL database connection settings by setting the **spring.datasource.url**, **spring.datasource.username**, and **spring.datasource.password**.

3. Run Spring Boot Backend:

- Start the Spring Boot application from the backend directory

4. Configure React Frontend:

- Navigate to the frontend directory:
- Open **src/api/api.js** and set the **BASE_URL** to the URL where your Spring Boot backend is running (by default, it should be **http://localhost:8080**).

5. Install Dependencies and Start React Frontend:

- Install the frontend dependencies
- Start the React development server

6. Access the Application:

- Open your web browser and visit **http://localhost:3000**. You should see the Employee Management System in action.
- Use the provided user interface to add, update, and delete employees.

Project Benefits:

- Efficient employee record management.
- Integration of backend (Spring Boot) and frontend (React) technologies.
- Practical demonstration of full stack development skills.
- Use of Bootstrap for responsive and visually appealing design.
- Axios library for seamless communication between frontend and backend.

By following these steps, users can deploy the Employee Management System on their local environment. It showcases proficiency in Java, Spring Boot, React, Bootstrap, Axios, and MySQL.

