



Full Stack Employee Management Project

Created By

Nikhil Suresh Patil

Guide

Junie D Solomon

INDEX

1. Introduction
2. Project Scope
3. Hardware Specification
4. Project Goals
5. Technologies
6. Navigation Element
 1. Header
 2. Add Employee
 3. Employee List
 4. Edit and Update
 5. Delete
 6. Login with Authentication
7. Functionality
8. Conclusion

1. INTRODUCTION:

Welcome to project which has two different versions built with the latest web development technologies. The first version of our project is based on React JS and Tailwind CSS, while the second version is built on Next JS, Tailwind CSS, and Next Auth. Both versions have the same backend technology stack, which includes Spring Boot, SQL, APIs, Hibernate, and JSP.

Project aims to provide a modern, user-friendly, and efficient web application for our users. With the help of React JS and Next JS, we have created two different versions that cater to the different needs of our users.

The React JS version of our project is designed for users who require a lightweight and fast web application with a modern UI/UX. We have utilized Tailwind CSS to ensure that the UI is responsive, easy to use, and aesthetically pleasing. With the power of React JS, we have created a seamless user experience that allows users to navigate through the application quickly and efficiently.

The Next JS version of our project is designed for users who require more advanced features such as authentication and authorization. We have incorporated Next Auth to ensure that our users' data is secure and only accessible to authorized individuals. Additionally, we have used Tailwind CSS to provide a modern and responsive UI/UX.

Both versions of project have the same backend technology stack, which ensures that our users receive a consistent experience regardless of which version they use. The use of Spring Boot, SQL, APIs, Hibernate, and JSP ensures that our application is fast, reliable, and scalable.

2. PROJECT SCOPE

The Employee Management System is a software application that is designed to manage the employee data of an organization. The system is aimed at providing a comprehensive solution for adding, updating, deleting, and showing the list of employees in an organization. The scope of the project is defined below:

1. **Employee Data Management:** The system will allow the user to add, update, and delete the employee data, including personal information, contact details, and employment information. The system will ensure that the data is accurate, consistent, and up-to-date.
2. **Employee List:** The system will allow the user to view a list of employees, including their personal and employment information.
3. **User Authentication and Authorization:** The system will require users to authenticate themselves before they can access the employee data. The system will also provide different levels of access based on the user's role, ensuring that the employee data is secure and confidential.
4. **User-Friendly Interface:** The system will provide a user-friendly interface that is easy to use and navigate. The system will be responsive, ensuring that it can be accessed from different devices such as desktops, laptops, or mobile devices.
5. **Maintenance and Support:** The system will require regular maintenance and support to ensure that it remains up-to-date and meets the evolving needs of the organization. The system will be supported by a team of developers and technical experts who will provide ongoing support and updates.

In conclusion, the Employee Management System will provide a comprehensive solution for managing the employee data of an organization. The system will allow the user to add, update, delete, and view the employee data, and provide a user-friendly interface

3. HARDWARE SPECIFICATION

Hardware used:

Processor: Intel i7 (Intel i3 and above can be used)

RAM: 8 GB

Storage: 2TB (250GB minimum required)

Graphics Card: 4 GB

Display: 1400 PX

Software used:

IDE: Visual Studio

Extensions: 1. React JS code snippets

2. ESLint

3. Prettier

4. Java Extension Pack

5. Debugger for Java

6. Tailwind CSS IntelliSense

7. Next.js Extension Pack

Dependencies: Spring Boot, Spring Data JPA, Hibernate ORM, JDBC, Lombok, Spring Web.

4. PROJECT GOALS

1. Develop a system that allows the user to add, update, and delete the employee data, including personal information, contact details, and employment information.
2. Develop a system that allows the user to view a list of employees
3. Implement user authentication and authorization, ensuring that the employee data is secure and confidential.
4. Develop a user-friendly interface that is easy to use and navigate
5. Provide ongoing maintenance and support, ensuring that the system remains up-to-date and meets the evolving needs of the organization.

5. TECHNOLOGIES


1. **React JS:** React JS is a popular JavaScript library used for building user interfaces. It was developed by Facebook and is now widely used by developers worldwide. React JS is based on a component-based architecture, which allows developers to build complex UIs by breaking them down into smaller, reusable components.
2. **Next JS:** Next JS is a framework built on top of React JS that provides server-side rendering, static site generation, and other features to improve the performance and SEO of React applications. It also includes features such as automatic code splitting, route prefetching, and more.
3. **Tailwind CSS:** Tailwind CSS is a utility-first CSS framework that helps developers quickly and easily style their applications. It provides a set of pre-defined CSS classes that can be used to style HTML elements. Tailwind CSS is highly customizable and can be extended to include custom styles and classes.
4. **Next Auth:** Next Auth is a library for adding authentication to Next JS applications. It supports various authentication providers, such as Google, Facebook, GitHub, and more. It also provides features like session management, email verification, and more.
5. **Java:** Java is a widely used programming language, known for its reliability, security, and platform independence. Java applications are typically compiled to byte code, which can be run on any platform that has a Java Virtual Machine (JVM) installed. Java is commonly used for building web applications, mobile apps, and enterprise applications.

6. **SQL:** SQL stands for Structured Query Language, and it is a standard language used for managing relational databases. SQL is used for creating, modifying, and querying databases, as well as performing other operations like backups and restores.
7. **REST API:** REST stands for Representational State Transfer, and it is a style of web architecture used for building web services. RESTful APIs provide a standard way for different applications to communicate with each other over the web. RESTful APIs use HTTP requests to perform operations like retrieving data, creating, modifying, and deleting resources. They are widely used for building web and mobile applications.

6. NAVIGATION ELEMENT


1. Header:

The header is created same in both UI with the help of tailwind CSS which allows to write minimum code without creating css page and to make it more aesthetic.



Employee Management System

React Js



Employee Management System

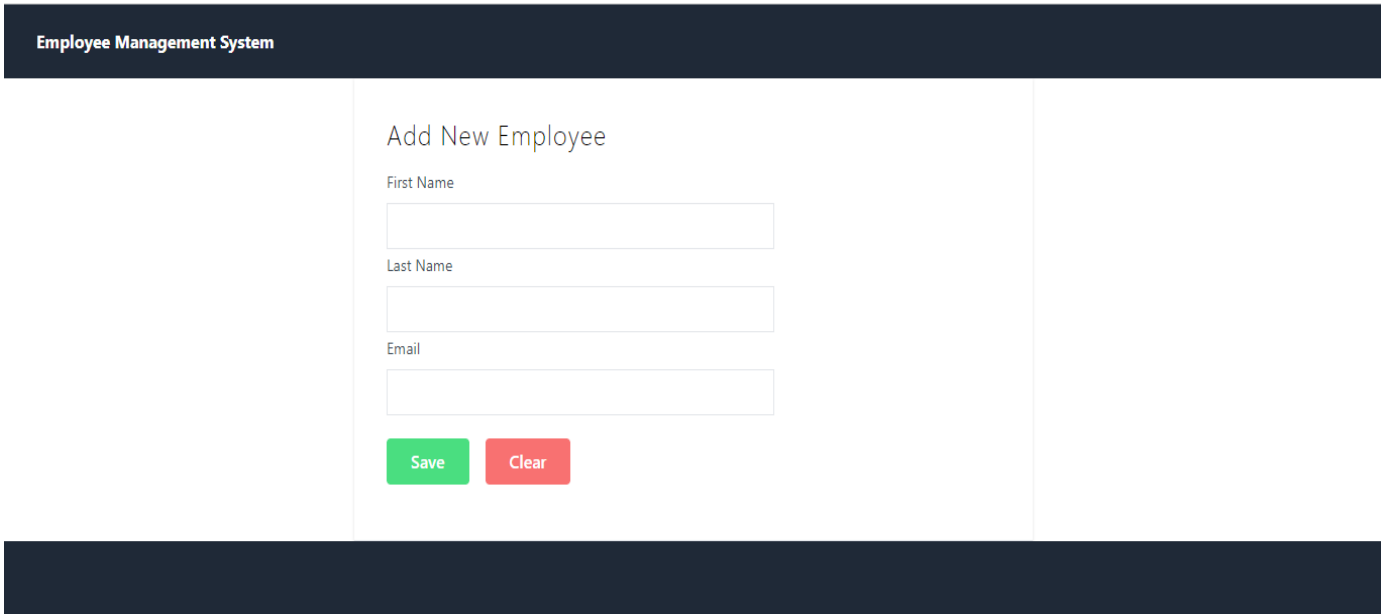


Next Js

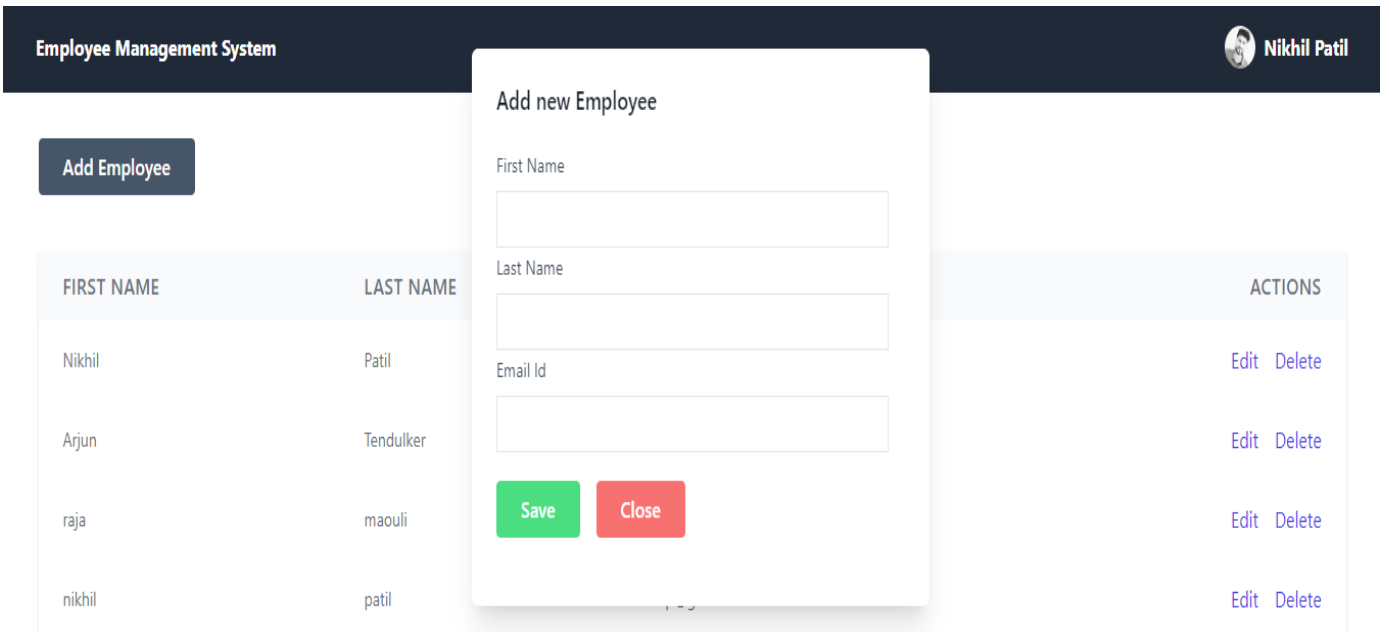
2. Add Employee:

As the page start after header the next Element is Button which used to add the personal data of the employee as such button indicates “Add Employee” on it when it is clicked it will take us to input form where we add employee details with respect to given input boxes as shown in below image.

In case of Next Js UI when the button is clicked it shows the small input box which is created using headless UI which allows it to appear on existing page as shown below.



React Js Add New Employee Page



Next Js Add Employee with Headless UI

3. Employee List:

Employee list is similar in both react Js and Next Js it is created using hooks, API's which are essential in displaying the data which is fetched from backend as SQL database to java from java to local host directory.

FIRST NAME	LAST NAME	EMAIL ID	ACTIONS
Nikhil	Patil	nikhilpatil@gmail.com	Edit Delete
Arjun	Tendulkar	ajTen@gmds.com	Edit Delete
raja	maouli	rajma.@sm.com	Edit Delete
nikhil	patil	np@gm.com	Edit Delete
Ragesh	SAnd	RS@gm.com	Edit Delete
lo	ok	look@me.come	Edit Delete
jfjnbvbkdff	fvkfd	sva	Edit Delete

4. Edit and Update:

The edit function is given to in the employee list to edit the data of each employee if necessary after clicking on the edit button it take us to view the input boxes where the data can be changed.

For Next Js it shows the input boxes which is displayed over the employee list.

5. Delete:

Delete is displayed right next to the edit button as name suggest it deletes the data from database

[Edit](#) [Delete](#)

Update Employee

First Name

Last Name

Email

React Js

FIRST NAME		LAST NAME		ACTIONS	
Nikhil		Patil		Edit	Delete
Arjun		Tendulkar		Edit	Delete
raja		maouli		Edit	Delete
nikhil		patil		Edit	Delete
Ragesh		SAnd		Edit	Delete
In		nk	lnnk@me.com	Edit	Delete

Update User

First Name

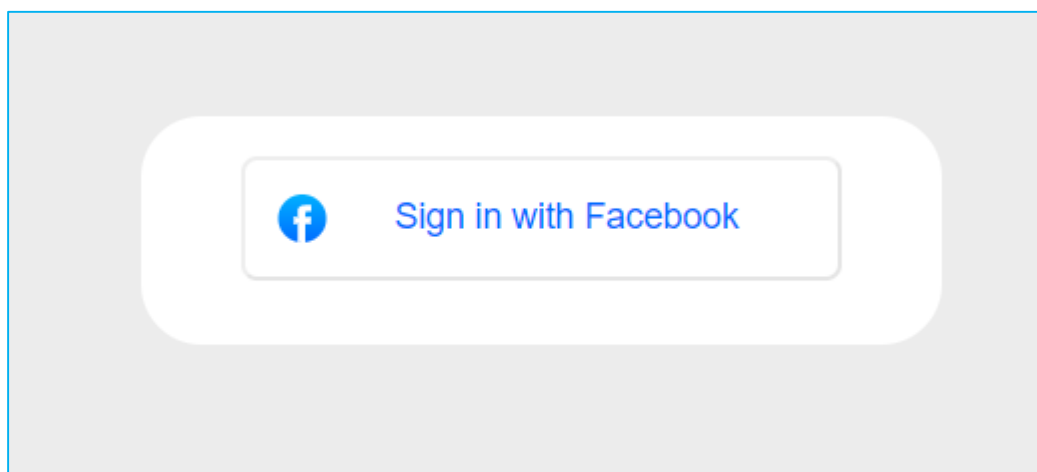
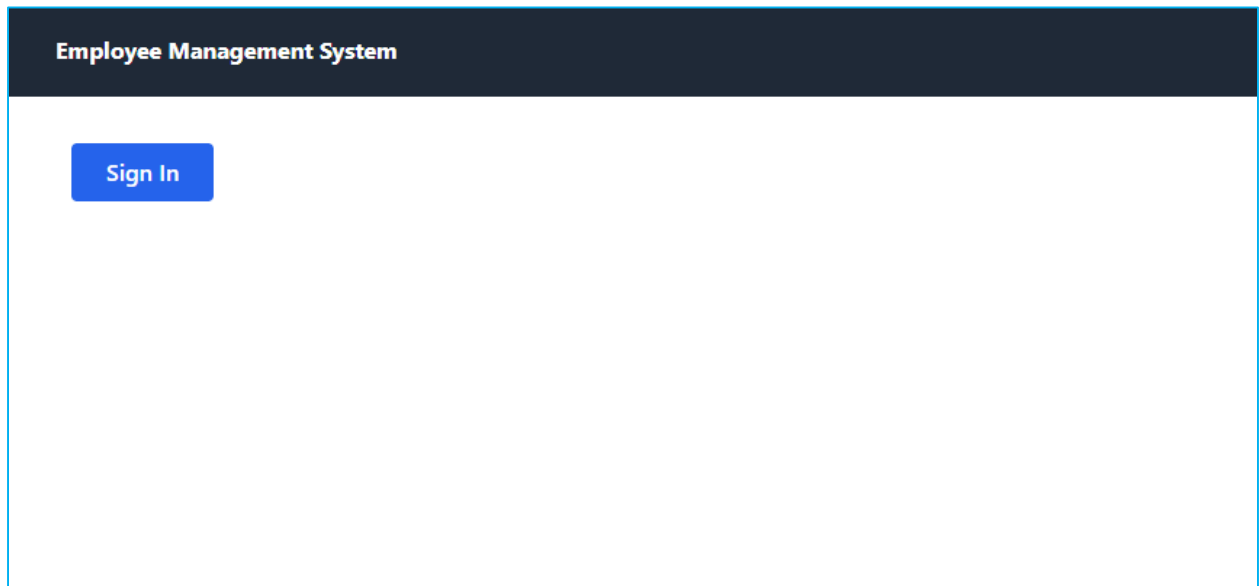
Last Name

Email Id

Next Js

6. Login with Authentication:

In next Js we have used the next auth dependency to keep the employee data safe from unauthorized user. To authenticate we have used Facebook app which asks the user to log in using Facebook ID and Password



7. FUNCTIONALITY

The website built using React Js/Next Js, tailwind CSS, Headless UI, and Java, Rest API, JSP, SQL will include several functional features that enhance the user experience. These features include:

Header: the header includes Web Application name and additionally in Next Js the User profile and User Name is shown and on clicking on the user image it sign out the user from the application.

Add Employee: When the button is clicked it opens the add employee form which takes employee's personal details in the input box and additional two buttons are provided one is "save" button by clicking on it the data is stored in the SQL database and other button is "Clear" which clears the data present in the form.

Edit: When the button is clicked it opens the Edit employee form which shows employee's personal details present in the input box and additional two buttons are provided one is "UPDATE" button by clicking on it the data is updated in the SQL database and other button is "Cancel" which takes us back to the employee List page.

Delete: Right next to Edit the delete button is available which erases the data on click of the button.

Login Authentication: The authentication secures the data from unauthorized user with the help of Facebook authentication app.

8. Conclusion

This project demonstrates the power of using React JS hooks and APS to efficiently retrieve data from the back end. The use of Java JSP Hibernate and SQL adds an extra layer of functionality to the application, allowing for continuous updates and smooth data management.

The front end of the application is user-friendly, with easy-to-use buttons such as the "Add Employee" button and "Edit" and "Delete" buttons present in the employee list. The use of Next Auth and Tailwind CSS, Headless UI further adds to the overall user experience by ensuring seamless navigation and a sleek design.

The integration of Hibernate and REST API helps maintain the data in the back end and ensures that updates are reflected in real-time on the front end. The "Edit" button allows users to update employee details, while the "Delete" button completely removes the personal details of an employee from the database.

Overall, this project demonstrates the efficient and effective use of modern web development technologies to create a robust and user-friendly application. The use of React JS hooks, APS, Java JSP Hibernate, and SQL has proven to be a winning combination in creating an application that is both functional and visually appealing.