SIMPLELEARN Full Stack Developer - MERN Stack Masters Program

Course – 3 Design a Dynamic Frontend with React

Assessment – Writeup

Student Name : Vedaang Sharma

HR Portal - Project Documentation

1. Project Overview

The **HR Portal Application** is a web-based Human Resource Management system designed to simplify and automate essential HR operations inside an organization. It provides a platform for both employees and HR personnel to interact with core HR services, such as employee management, leave handling, attendance regularization, and user authentication.

Purpose:

- Enable HR departments to digitalize employee management activities.
- Allow employees to register, update details, and interact with HR services.
- Provide a smooth, responsive, and user-friendly interface for HR processes.
- Replace error-prone manual systems with an interactive and automated portal.

This project focuses on the **frontend development** of the portal using React JS, Bootstrap, HTML, CSS, and JavaScript as outlined in the industry-relevant tools list. The documentation format and structure follow the example provided.

---

2. Features

User Authentication

- **Registration:** Employees and HR users can register with name, email, username, password, department, and role.
- **Login:** Validates user credentials and redirects them based on their role.
- **Role-Based Access:** HR users access admin features; employees access personal dashboards.
- **Logout:** Users securely exit their sessions.

Employee Management (HR Only)

- View complete list of employees.
- Add new employees using a structured form.
- Data is stored and updated using JSON and React state.
- Dashboard cards show total employees and pending leave count.

Leave Management

Employees can:

- Apply for leave using a form (from-date, to-date, reason).
- View leave status (Pending, Approved, Rejected).

HR can:

- View all leave requests.
- Approve or reject leave applications.
- Track leave analytics via dashboard counters.

Responsive Dashboards

- Fully responsive HR and Employee dashboards using **Bootstrap Grid System**.
- Navigation bar adjusts to mobile layout.
- Dashboard cards summarize important information.

Data Management Using JSON

- employees.json stores initial employee data.
- React state syncs with JSON for real-time updates.
- Acts as a lightweight mock database for this project.

Custom Theming

- Consistent UI using a global theme (theme.css).

- Smooth animations, enhanced cards, styled tables, focus effects, and modern typography.

---

3. Implementation Details

Frontend Technologies

The project uses the following technologies, similar to how the food ordering website implementation documented its toolset:

- **React.js:** Component-based dynamic UI rendering.
- **Bootstrap 5:** Layout, responsive grid, card components, navbar, forms.
- **JavaScript (ES6):** State updates, event handling, form interactions.
- **HTML5:** Structural layout of components.
- **CSS3 (Custom Theme):** Professional styling and animations.

Key React Components

Authentication Components

- **Login.js:** Validates username/password and redirects by role.
- **Signup.js:** Enhanced employee registration with department and role fields.

Dashboard Components

HRDashboard.js

- Displays statistics cards.
- Renders AddEmployee and EmployeeList components.
- Integrates LeaveApproval for HR-level leave management.

EmployeeDashboard.js

- Shows employee details and department.
- Provides ApplyLeave and MyLeaves components.

Data Handling

Employee Data Management

```
[
  {
    "id": 1,
    "name": "Vedaang Sharma",
    "email": "vedaangsharma2006@gmail.com",
    "department": "Engineering"
  },
  {
    "id": 2,
    "name": "Nidhi Sharma",
    "email": "nidhi.sharma@gmail.com",
    "department": "Science"
  }
]
```

Leave Data Structure

```
const newLeave = {

      id: Date.now(),
      employeeName,
      fromDate,
      toDate,
      reason,
      status: "Pending"
    };
```

Role-Based Navigation Logic

- HR users → HR Dashboard
- Employees → Employee Dashboard
- Current session stored in React state.

User Interface Enhancements

As in the reference project, UI is made professional using:

- Cards
- Buttons

- Styled tables
- Smooth animations
- Navbars

This mirrors the approach in the example documentation.

---

4. Workflow / Working of the Application

1. User Registration/Login

- New users register with detailed information.
- Returning users log in.
- Redirects according to role.

2. HR Dashboard

- HR can view employee list.
- Add new employees.
- Monitor leave requests.
- Approve/Reject requests.

3. Employee Dashboard

- User sees personal info.
- Apply for leave.
- Track leave history/status.

4. Leave Processing

- Employee submits a leave form.
- HR reviews and updates the request.
- Updated status appears instantly for the employee.

5. JSON-Based Storage

- Employees data loaded from employees.json.
- New employees added dynamically.
- Leave requests managed in React state.

6. Logout

- Ends active session and returns user to login page.

---

5. Features Summary

Feature	Description
User Authentication	Register, login, role-based routing
Employee Management (HR)	Add/view employee records, track department data
Leave Application	Employees apply for leave
Leave Approval	HR reviews, approves, or rejects requests
Responsive Dashboards	Fully responsive HR & Employee UI
JSON Data Storage	Lightweight local data management
Professional UI Theme	Cards, navbars, input styling, animations

---

6. Project Deliverables

I am submitting a .zip file of the source code developed by me in the project. The project is also hosted on my GitHub profile and deployed for live viewing through a hosting platform.

Links :

- GitHub Repository : https://github.com/gtathelegend/hr-portal-mern-stack-simplilearn-course-end-project
- Live Project : https://hr-portal-mern-stack-simplilearn-course.onrender.com
- Google Drive Link: https://drive.google.com/drive/folders/1wdeQqOjtkmj1s9ZEb9aQHpKt5ajEuyyh?usp=sharing

---

7. Conclusion

This project demonstrates how a fully functional HR Management Portal can be built using:

- **React.js** for dynamic UI rendering
- **Bootstrap** for responsiveness

- **JSON** for storage
- **JavaScript** for data handling and interactions

The HR Portal simplifies core HR functions such as employee data management, leave tracking, and user role segregation.