SEU/IS/18/ICT/002 - K.DANUSAN

Employee Management System - Final Report

The project focuses on developing an Employee Management Systemthat efficiently manages employee data, daily attendance, salary calculations, welfare services, leave, tasks, and safety. It integrates MongoDB, Node.js, and a user-friendly front end to streamline operations. The system ensures accuracy, enhances productivity, and simplifies management processes, offering a scalable solution for organizational needs.

1. Project Overview

Project Title: Employee Management System (EMS)
Objective:

- Employee Database Management.
- Daily Attendance Tracking.
- Task/Project Management.
- Salary Calculation.
- Welfare Services like health, food, insurance.
- Leave Management.
- Safety Management.

2. Key Features

a) Web Application:

1. User Authentication:

- Sign In/Sign Out validation.
- Session storage for user session tracking.

2. CRUD Operations:

- Add, Update, Delete, View functionalities for employee details, attendance, and other modules via the browser.

3. Modules Developed:
- Add Employee.
- Attendance Tracking.
- Benefits/Welfare Management.
- Leave Management.
- Task/Project Management.
- Safety Management.
- Salary Management.

4. Integrated MongoDB:
- Backend APIs created using Node.js to handle database operations.
- Postman used for testing API functionalities.

b) Android Application:

1. User Interface:
- Attractive and user-friendly UI built using XML in Android Studio.
- Each module corresponds to a separate activity.

2. Modules Implemented:
- Add, View, Update, Delete functionalities replicated from the web app.
- Activities Created:
  - AddEmployeeActivity
  - ViewEmployeesActivity
  - UpdateEmployeeActivity
  - DeleteEmployeeActivity
  - WelfareActivity
  - AttendanceActivity
  - SalaryActivity
  - LeaveManagementActivity
  - SafetyManagementActivity

3. Backend Integration:
- Connected the Android app to MongoDB using Node.js APIs.
- Seamless CRUD operations tested from the mobile application.

4. Testing & Validation:
- Validations added for user input (e.g., required fields, minimum password length).
- Robust error handling for API failures.

3. Technical Architecture

a) Frontend:
- Web Technology: HTML, CSS, JavaScript.
- Mobile Technology: Java and XML in Android Studio.

b) Backend:
- Platform: Node.js.
- Database: MongoDB (using MongoDB Compass and Atlas).

c) Tools Used:
- Design: Figma for UI prototyping.
- Testing: Postman for API testing.
- Deployment: Live server deployment.

4. Database Design

Collections in MongoDB:

1. Employee Collection:
- Fields: EmployeeID, Name, Designation, Department, Salary, etc.

2. Attendance Collection:
- Fields: EmployeeID, Date, Check-In Time, Check-Out Time.

3. Salary Collection:
- Fields: EmployeeID, Basic Pay, Allowances, Deductions, Final Salary.

5. Implementation Process

1. UI Design:
- Designed the web app's layout in Figma.
- Created HTML pages for each module with responsive styles.

2. Backend Setup:
- Configured Node.js server for API development.
- Implemented API endpoints for all CRUD operations, Integrated MongoDB with APIs.

3. Testing:
- Verified APIs using Postman.
- Validated data in MongoDB after CRUD operations.

4. Integration with Android Studio:
- Established a connection between the Android app and MongoDB via APIs.
- Implemented activities for all modules.

5. Final Validation:
- Conducted end-to-end testing on both web and mobile platforms.
- Fixed all bugs and ensured smooth functionality.

6. Challenges & Solutions

Challenges:
- Session management during user login and logout.
- Handling API errors during mobile integration.
- Ensuring data consistency across the web and mobile platforms.

Solutions:
- Used sessionStorage for managing user sessions securely.
- Implemented proper error handling in APIs.
- Conducted multiple test cases to ensure data consistency.

7. Achievements

1. Complete Functionality: Successfully developed a fully functional EMS on both web and mobile platforms.
2. User-Friendly Interface: Designed an attractive and responsive UI.
3. Seamless Integration: Integrated MongoDB with both web and mobile applications.

8. Future Enhancements

1. Role-Based Access Control (RBAC):
- Adding different user roles like Admin, HR, and Employee for accessing specific modules.

2. Notification System:
- SMS or Email notifications for attendance, salary updates, etc.

3. Reporting Module:
- Generating detailed reports for attendance, performance, and salary.

4. Cloud Deployment:
- Host the project on cloud platforms for scalability and reliability.

9. Conclusion

The Employee Management System project is a comprehensive and technically sound solution. It is successfully implemented on both web and mobile platforms, ensuring seamless integration with MongoDB. The project demonstrates real-world applicability and scalability, making it a robust and efficient system.