SDE Internship Assignment (MERN Stack)

Introduction

This report outlines the design, backend structure, and assumptions made during the development of an employee management system. The system allows users to view, add, edit, and delete employee information.

Overall Design

The system utilizes a two-tier architecture with a frontend built using React and a backend powered by Node.js and Express. The frontend handles user interactions and displays data, while the backend manages data persistence and server-side logic.

Frontend Design

The frontend features a responsive design for optimal viewing on various screen sizes.

A navigation bar potentially exists for easy access to different functionalities.

A "container" element likely centers the content within the webpage.

A "btn-add" button triggers the display of an "Add Employee" form.

The "Add Employee" form allows users to input employee details like first name, last name, employee ID, email, and contact number.

A table displays a list of employees with columns for their details.

"Edit" and "Delete" buttons allow users to modify existing employee data.

Backend Design

Node.js and Express provide the server-side framework.

Mongoose serves as an object data modeling (ODM) library for interacting with a MongoDB database.

The backend defines API endpoints for:

Retrieving all employee data (GET /)

Creating a new employee (POST /create)

Updating existing employee data (PUT /update)

Deleting an employee (DELETE /delete/:id)

Assumptions

The user has a basic understanding of web applications.

The server has a running instance of MongoDB.

The frontend and backend communicate through a well-defined API.

Data validation occurs on both the frontend and backend to ensure data integrity.

Error handling is implemented to gracefully handle unexpected situations.

Future Enhancements

User authentication and authorization for secure data access.

Pagination and sorting functionalities for managing large datasets.

Search functionality to easily locate specific employees.

Integration with other systems like payroll or HR management software.

Conclusion

This report provides a high-level overview of the design, backend structure, and assumptions made during the development of this employee management system. The system utilizes a layered architecture and leverages modern web technologies for data management and user interaction. While this initial implementation offers basic CRUD functionalities, it lays the foundation for future enhancements towards a more comprehensive employee management solution.