Project Title	Employee Management System
Technologies	MERN
Domain	Industry
Project Level	Difficult

Table

Table

1. Problem Statement:

The project for an employee management system mostly deals with employee data including departments, salaries, and leaves. The system also displays every piece of available employee data, such as name, contacts, photos, and more. Additionally, you may manage departments by just putting their names into the system. The two components of this project are the Admin Panel and the Employee Panel. The administrator gets rapid access to anything in the overview of this web application. Each employee's compensation may be regulated by the administrator. An administrator must first select a department that is open for selection, then the employee for that department. He or she must fill out this part with information on compensation, including basic pay and benefits.

Additionally, the admin may view each payroll transaction's invoice and generate a PDF copy from the salary history area. The administrator is also in charge of each employee's leave requests. This section lists each employee's name, leave dates, application date, reason with a brief justification, and current status (pending, accepted, or rejected). After managing their leave, the employee may check their status from their own employee panel. After adding a staff member's record, the system does not request any login information; instead, it uses the staff member's email and phone number as their login information. The system also displays employee counts, total pay paid, and other information on the admin's dashboard.

1. Features:

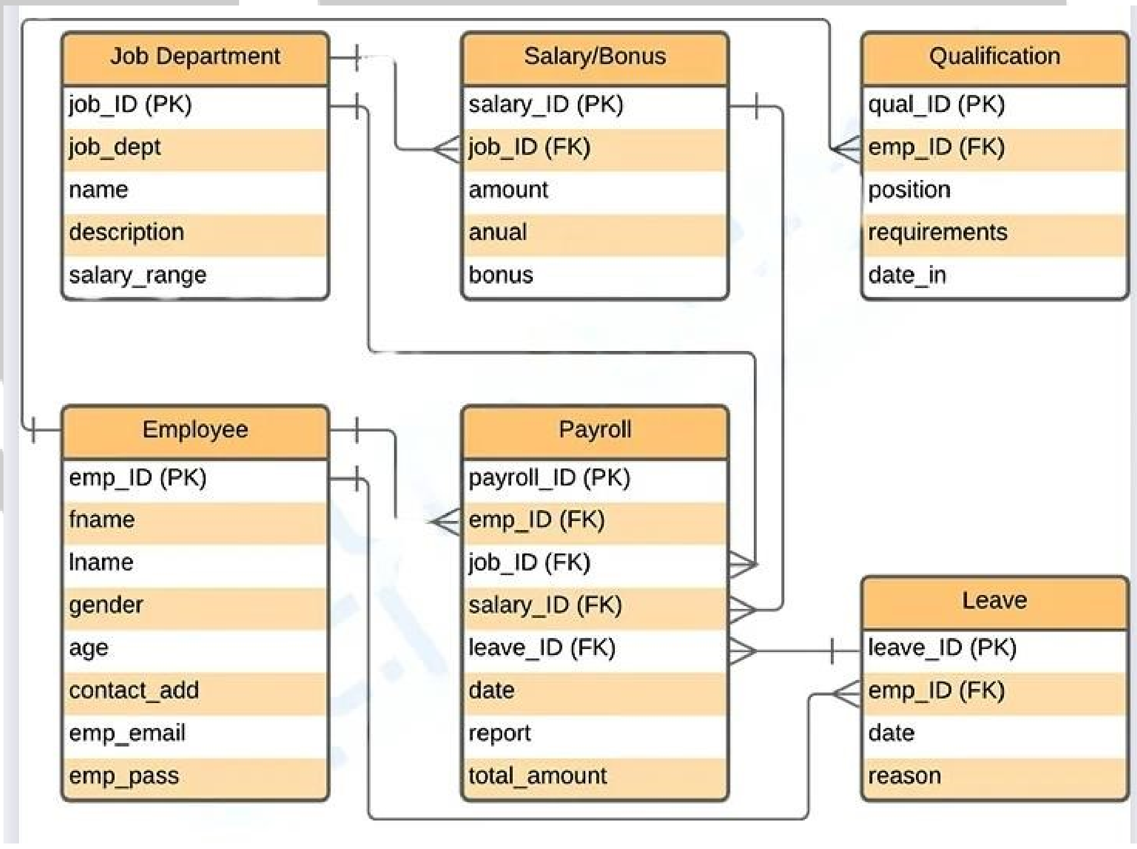
1.1. Admin Side:

- Management Employee
- Manage Leave Application
- Attendance Management System
- Manage Profile

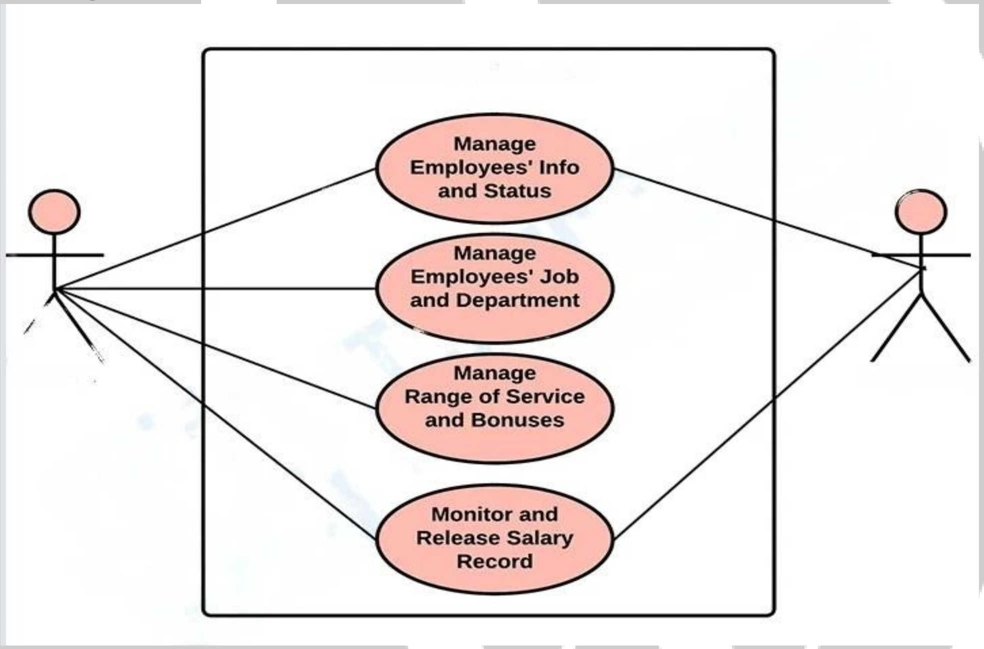
1.2. Employee Side

- Apply for leave
- View leave
- View Attendance
- Mark attendance

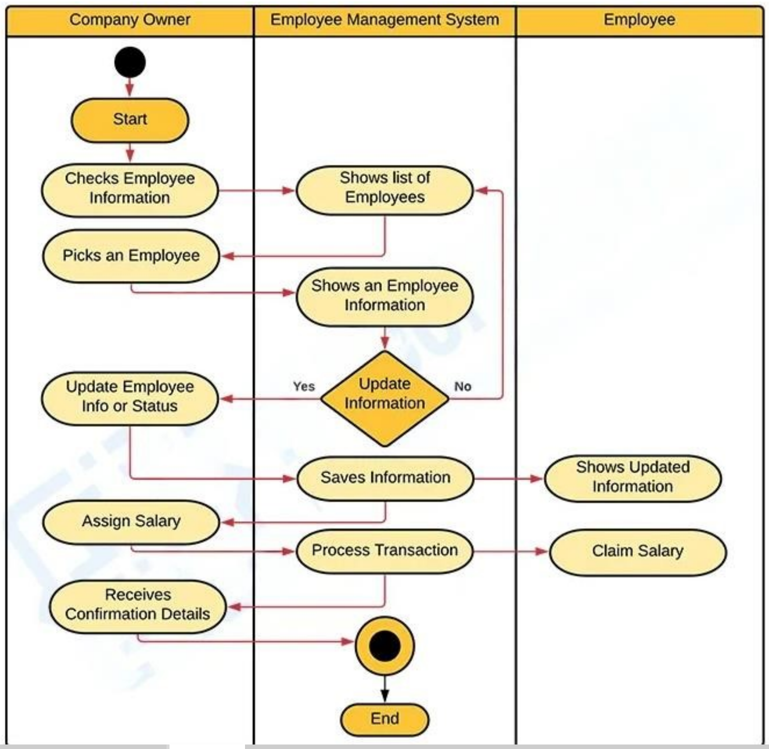
2. ER DIAGRAM:



3.Use case diagram:


Manage Employees' Info and Status
Manage Employees' Job and Department
Manage Range of Service and Bonuses
Monitor and Release Salary Record

4.Activity Diagram:


Company Owner
Employee Management System
Employee
Start
Checks Employee Information
Shows list of Employees
Picks an Employee
Shows an Employee Information
Update Employee Info or Status
Yes
Update Information
No
Saves Information
Shows Updated Information
Assign Salary
Process Transaction
Claim Salary
Receives Confirmation Details
End

5. Project Evaluation metrics:

5.1. Code:

- You are supposed to write code in a modular fashion ● Safe: It can be used without causing harm.
- Testable: It can be tested at the code level.
- Maintainable: It can be maintained, even as your codebase grows.
- Portable: It works similarly in every environment (operating system).
- You have to maintain your code on GitHub.
- You must keep your GitHub repo public so anyone can check your code.
- Proper readme file you have to maintain for any project development.
- You should include the basic workflow and execution of the entire project in the readme file on GitHub.
- Follow the coding standards.

5.2. Database:

MongoDB is a source-available cross-platform document-oriented database program. Classified as a NoSQL database program, MongoDB uses JSON-like documents with optional schemas.

5.3. API Details or User Interface:

You have to expose your complete solution as an API or try to create a user interface for your model testing. Anything will be fine for us.

5.4. Deployment:

Deploy the application on your preferred service.

5.5. Solutions Design:

You have to submit complete solution design strategies in High-level Document (HLD), Lowlevel Document (LLD), and Wireframe documents**.**

5.6. System Architecture:

You have to submit a system architecture design in your wireframe document and architecture document.

5.7. Optimization of solutions:

Try to optimize your solution on the code level, and architecture level, and mention all of these things in your final submission.

6. Submission requirements:

6.1. High-level Document:

You have to create a high-level document design for your project. You can reference the HLD form below the link.

Sample link: HLD Document Link

6.2. Low-level document:

You have to create a Low-level document design for your project; you can refer to the LLD from the link below.

Sample link: LLD Document Link

6.3. Architecture:

You have to create an Architecture document design for your project; you can refer to the Architecture from the link below.

Sample link: Architecture sample link

6.4. Wireframe:

You have to create a Wireframe document design for your project; refer to the Wireframe from the link below.

Demo link: Wireframe Document Link

6.5. Project code:

You have to submit your code to the GitHub repo in your dashboard when the final submission of your project.

Demo link: Project code sample link

6.6. Detail project report:

You have to create a detailed project report and submit that document as per the given sample.

Demo link: DPR sample link

6.7. Project demo video:

You have to record a project demo video for at least 5 Minutes and submit that link.