



INSTITUTE FOR ADVANCED COMPUTING AND
SOFTWARE DEVELOPMENT AKURDI, PUNE

Documentation On

EMPLOYEE ADMINISTRATION SYSTEM

PG-DAC MARCH 2023

Submitted By:

Group No:92

Roll No.

233198

233168

Name:

Sahil Totade (230341220154)

Mayur Kasbe (230341220098)

Mr. Harshal Waghchaure

Mrs. Monika Sindhikar

Project Guide

Mr. Rohit Puranik

Centre Coordinator

Table of Contents

ABSTRACT	4
ACKNOWLEDGEMENT.....	5
SRS DOCUMENT FOR EAS.....	6
1. INTRODUCTION	6
2. PURPOSE	6
3. SCOPE	6
3.1 USER AUTHENTICATION MODULE.....	6
3.2 DEPARTMENT MODULE	6
3.3 MANAGER MODULE	6
3.4 EMPLOYE MODULE	7
3.5 EMPLOYEE SALARY MODULE.....	7
4. FUNCTIONAL REQUIREMENTS	7
4.1 USER AUTHENTICATION MODULE	7
4.2 DEPARTMENT MODULE	8
4.3 MANAGER MODULE	8
4.4 EMPLOYEE MODULE.....	8
4.5 EMPLOYEE SALARY MODULE	8
5. NON FUNCTIONAL REQUIREMENT	9
6. CONSTRAINTS.....	9
7. GLOSSARY	9
8. REFRENCES	9
9. REVISION HISTORY	10
10. TABLE STRUCTURE/SCREEN SHOTS.....	11
10.1 DEPARTMENT TABLE	12
10.2 SALARY	12
10.3 USER.....	13
11. PROJECT CLOSURE REPORT	14
11.1 EAS USING SPRINGBOOT MISCROSERVICES+REACT JS+MYSQL	14
11.2 TECHNOLOGIES USED IN EAS	16
11.3 SOFTWARE USED	16
12. USER MODULES IN EAS	17

13. FUNCTIONAL MODULES	17
14. OVERVIEW	18
15. THE PROJECT STRUCTURE.....	19
16. THE PROJECT FLOW	20
17. ROLES OF USERS	21
19. CONCLUSION.....	43
20. BIBLIOGRAPHY/REFERENCES	44

LIST OF FIGURES

18. DIAGRAMS	33
18.1 ADMIN ACTIVITY DIAGRAM	33
18.2 MANAGER ACTIVITY DIAGRAM	34
18.3 EMPLOYEE ACTIVITY DIAGRAM.....	35
18.4 CLASS DIAGRAM	36
18.5 E-R DIAGRAM	37
18.6 USE CASE DIAGRAM.....	38
18.7 SEQUENCE DIAGRAM	39
18.8 ZERO LEVEL DFD	40
18.9 FIRST LEVEL DFD.....	41
18.10 SECOND LEVEL DFD.....	42

ABSTRACT

An **Employee Administration System Project in Spring Boot and React** empowers administrators to efficiently manage departments, administrators, employees, and salary information. This project delves into the functionalities of an Employee Administration System (EAS) wherein administrators can effortlessly add departments and assign administrators liable for overseeing employee administration. Additionally, the system allows the recording and monitoring of salary information for each employee. Discover how this comprehensive solution enhances organizational efficiency, simplifies administrative tasks, and guarantees accurate control of employee records and compensation. Streamline your employee administration tactics and optimize productivity with an Employee Administration System tailored to satisfy the specific desires of your agency.

An Employee Administration System is developed using Spring Boot Microservices, React JS & MySQL. By using this application, Admins can add Managers in the application based on the departments, and Managers can administer the Employee Data, including their Salary and complete details.

ACKNOWLEDGEMENT

I would like to express my special thanks of gratitude to All my teachers and staff members as well as our co-ordinator who gave me the golden opportunity to do this wonderful project on the topic **Employee Administration System**, which also helped me in doing a lot of Research and I came to know about so many new things I am really thankful to them. Secondly i would also like to thank my parents and friends who helped me a lot in finalizing this project within the limited time frame.

SRS Document for Employee Administration System

1.Introduction

The Employee Administration System is a comprehensive software solution that aims to streamline employee-related operations within an organization. Developed using Spring Boot for the backend and React for the frontend, the system facilitates user authentication, department management, manager registration, employee management, and salary administration. This document outlines the requirements, features, and specifications of the system.

2. Purpose:

The purpose of the Employee Administration System is to provide an efficient and user-friendly platform for managing employee-related tasks. The system offers a range of functionalities including user authentication, department management, manager registration, employee records management, and salary administration. By automating and centralizing these operations, the system improves organizational efficiency and reduces administrative overhead.

3. Scope:

The system encompasses the following key modules:

3.1 User Authentication Module:

- Secure registration and login functionality for users (Admin, Manager, Employee).
- Role-based authorization to ensure that users can access only their relevant functionalities.
- Password recovery and reset options for users.

3.2 Department Module:

- Ability to add new departments with relevant details.
- Update department details such as name, description, or manager.
- Deletion of departments when needed.
- View a list of all departments within the organization.

3.3 Manager Module:

- Register new managers with personal details and department assignments.
- Update manager information, including personal and contact details.

- Delete manager records as needed.
- View a list of all managers.

3.4 Employee Module:

- Register new employees with personal information, contact details, and department assignments.
- Update employee records with relevant changes.
- Delete employee records when necessary.
- View a list of all employees within the organization.
- Employees can view their own profiles for personal information.

3.5 Employee Salary Module:

- Add salary details for employees, including salary amount, payment mode, and bank details.
- Update salary information when necessary.
- Fetch and display employee salary details.
- Employees can view their own salary details.

4. Functional Requirements:

4.1 User Authentication Module:

1. Secure Registration and Login:
 - Users can register with valid credentials.
 - Secure login process with authentication checks.
2. Role-Based Authorization:
 - Admins can access all modules and functionalities.
 - Managers can manage employees and salary details.
 - Employees have access to their own profiles and salary information.
3. Password Management:
 - Password recovery through email.
 - Password reset functionality.

4.2 Department Module:

1. Department Management:
 - Admins can add new departments with relevant details.
 - Admins can update or delete existing departments.
 - List of departments displayed with their details.

4.3 Manager Module:

1. Manager Registration:
 - Admins can register new managers with personal and department information.
 - Admins can update or delete manager records.
2. Manager Details:
 - Managers can view their own profiles.
 - Admins can view a list of all managers.

4.4 Employee Module:

1. Employee Registration:
 - Managers can register new employees with personal, contact, and department information.
 - Managers can update or delete employee records.
2. Employee Details:
 - Employees can view their own profiles.
 - Managers can view a list of all employees within their departments.

4.5 Employee Salary Module:

1. Salary Administration:
 - Managers can add salary details for employees.
 - Managers can update salary information.
 - Employees can view their own salary details.

5. Non-Functional Requirements:

1. User Interface:
 - Intuitive and user-friendly design.
 - Responsive and consistent across modules.
2. Security:
 - Strong authentication and authorization mechanisms.
 - Passwords stored securely using hashing.
3. Performance:
 - Fast response times for user interactions.
 - Efficient database queries for data retrieval.
4. Scalability:
 - The system should handle a growing number of users and records.
5. Data Backup and Recovery:
 - Regular database backups to prevent data loss.
 - Mechanisms for data recovery in case of system failures.

6. Constraints:

1. The system should be compatible with modern web browsers.
2. Backend: Spring Boot Microservices.
3. Frontend: React JS.
4. Database: MySQL.

7. Glossary:

- **Admin:** User with administrative privileges to manage all system functionalities.
- **Manager:** User responsible for managing employees and their data.
- **Employee:** A user within the organization.
- **Department:** A division within the organization structure.
- **Salary:** Compensation provided to employees for their work.

8. References:

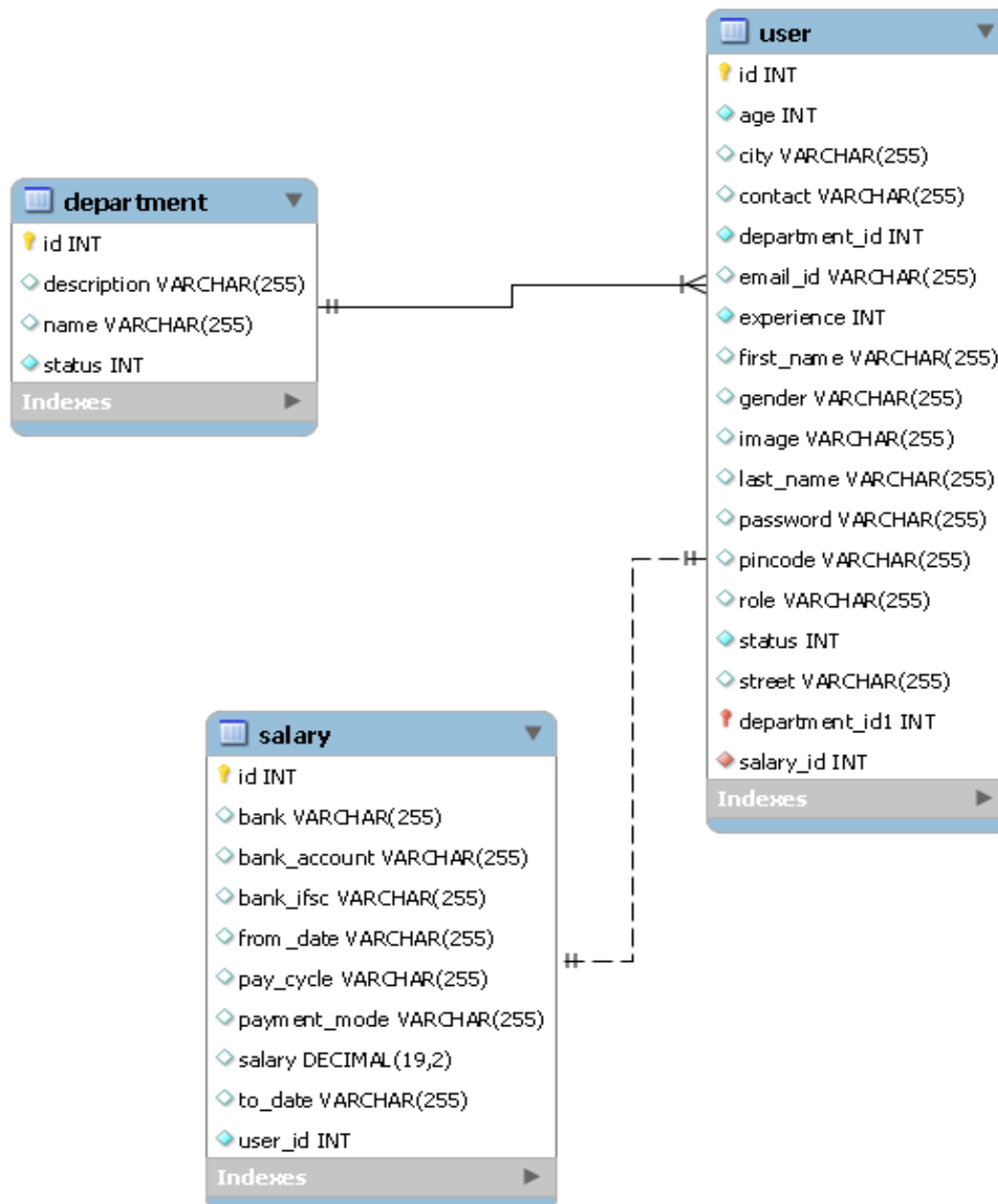
- Spring Boot Documentation
- React JS Documentation

- MySQL Documentation

9. Revision History:

- This Software Requirements Specification provides an in-depth overview of the features, functionality, and requirements of the Employee Administration System. It serves as a blueprint for development, ensuring a comprehensive understanding of the system's purpose, scope, and key components

10.TABLE STRUCTURE / SCREEN-SHOTS:



10.1. department table:

```
mysql> use employee_department_detail;
Database changed
mysql> show tables;
+-----+
| Tables_in_employee_department_detail |
+-----+
| department                           |
+-----+
1 row in set (0.01 sec)

mysql> desc department;
+-----+-----+-----+-----+-----+-----+
| Field          | Type          | Null | Key | Default | Extra          |
+-----+-----+-----+-----+-----+-----+
| id             | int           | NO   | PRI | NULL    | auto_increment |
| description    | varchar(255)  | YES  |     | NULL    |                |
| name           | varchar(255)  | YES  |     | NULL    |                |
| status         | int           | NO   |     | NULL    |                |
+-----+-----+-----+-----+-----+-----+
4 rows in set (0.02 sec)
```

10.2. salary:

```
mysql> use employee_salary_detail;
Database changed
mysql> show tables;
+-----+
| Tables_in_employee_salary_detail |
+-----+
| salary                           |
+-----+
1 row in set (0.00 sec)

mysql> desc salary;
+-----+-----+-----+-----+-----+-----+
| Field          | Type          | Null | Key | Default | Extra          |
+-----+-----+-----+-----+-----+-----+
| id             | int           | NO   | PRI | NULL    | auto_increment |
| bank           | varchar(255)  | YES  |     | NULL    |                |
| bank_account   | varchar(255)  | YES  |     | NULL    |                |
| bank_ifsc      | varchar(255)  | YES  |     | NULL    |                |
| from_date      | varchar(255)  | YES  |     | NULL    |                |
| pay_cycle      | varchar(255)  | YES  |     | NULL    |                |
| payment_mode   | varchar(255)  | YES  |     | NULL    |                |
| salary         | decimal(19,2) | YES  |     | NULL    |                |
| to_date        | varchar(255)  | YES  |     | NULL    |                |
| user_id        | int           | NO   |     | NULL    |                |
+-----+-----+-----+-----+-----+-----+
10 rows in set (0.01 sec)
```

10.3. user

```
mysql> use employee_user_detail;
```

```
Database changed
```

```
mysql> show tables;
```

```
+-----+
| Tables_in_employee_user_detail |
+-----+
| user                            |
+-----+
```

```
1 row in set (0.00 sec)
```

```
mysql> desc user;
```

Field	Type	Null	Key	Default	Extra
id	int	NO	PRI	NULL	auto_increment
age	int	NO		NULL	
city	varchar(255)	YES		NULL	
contact	varchar(255)	YES		NULL	
department_id	int	NO		NULL	
email_id	varchar(255)	YES		NULL	
experience	int	NO		NULL	
first_name	varchar(255)	YES		NULL	
gender	varchar(255)	YES		NULL	
image	varchar(255)	YES		NULL	
last_name	varchar(255)	YES		NULL	
password	varchar(255)	YES		NULL	
pincode	varchar(255)	YES		NULL	
role	varchar(255)	YES		NULL	
status	int	NO		NULL	
street	varchar(255)	YES		NULL	

```
16 rows in set (0.00 sec)
```

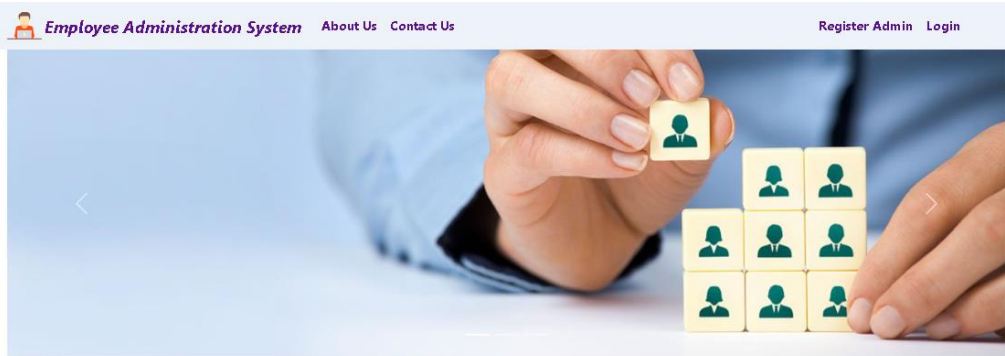
11.Project Closure Report:

11.1.Employee Administration System Project using React JS + Spring Boot Microservices + MySql

An **Employee Administration System Project in Spring Boot and React** empowers administrators to efficiently manage departments, administrators, employees, and salary information. This project delves into the functionalities of an Employee Administration System (EAS) wherein administrators can effortlessly add departments and assign administrators liable for overseeing employee administration. Additionally, the system allows the recording and monitoring of salary information for each employee. Discover how this comprehensive solution enhances organizational efficiency, simplifies administrative tasks, and guarantees accurate control of employee records and compensation. Streamline your employee administration tactics and optimize productivity with an Employee Administration System tailored to satisfy the specific desires of your agency.

An Employee Administration System is developed using Spring Boot Microservices, React JS & MySQL. By using this application, Admins can add Managers in the application based on the departments, and Managers can administer the Employee Data, including their Salary and complete details.

HOME PAGE



Welcome to Employee Administration System

The Employee Administration System is a comprehensive solution for managing employees and their information in an organization. It helps streamline various HR processes, such as employee onboarding, attendance tracking, performance management, and more.

With our system, administrators can efficiently manage employee records, assign roles and permissions, generate reports, and oversee the overall employee management workflow. Employees can access their profiles, view their schedules, request leave, and communicate with their managers. Managers can easily track employee performance, approve requests, and ensure smooth operations within their teams.

[Get Started](#)



Manage Employees and Managers

Simplify the process of managing employee data along with their salary information. The Employee Management System allows you to easily store and update employee details such as personal information, contact details, job positions, and salary records. Keep track of employee compensation, benefits, and payroll management effectively.

Efficiently manage managers and employee departments with our Employee Management System. Assign managers to specific departments, delegate responsibilities, and oversee the performance of both managers and their respective teams. Streamline communication, collaboration, and decision-making within departments for improved productivity and coordination.

[Get Started](#)



EMPLOYEE ADMINISTRATION SYSTEM

The Employee Administration System is a comprehensive solution for managing employees and their information in an organization.

ABOUT US

[Link 1](#)
[Link 2](#)
[Link 3](#)
[Link 4](#)

CONTACT US

[Link 1](#)
[Link 2](#)
[Link 3](#)
[Link 4](#)

CAREERS

[Link 1](#)
[Link 2](#)
[Link 3](#)
[Link 4](#)

LINKS

[Link 1](#)
[Link 2](#)
[Link 3](#)
[Link 4](#)

Login from here

[Log in](#)

© 2023 Copyright iacsd.com

11.2. Technologies Used in Employee Administration Systems

The key technologies for developing the Employee Administration System include Spring Boot Microservices, React JS, MySQL, Maven, Bootstrap, and Java.

1. **Java - Spring Boot Microservices:** For the Backend Development of the Employee Administration System project, Spring Boot Microservices have been used.
2. **React JS:** React JS is a popular Javascript Library used for the development of the project's frontend.
3. **MySQL:** MySQL is a widely used relational database management system used for storing employee data.
4. **Maven:** Maven has been used for simplifying the project and dependency management.
5. **BootStrap:** Bootstrap, a popular CSS framework, has been used for styling the project's frontend to create a responsive user interface.

11.3. Software Used

1. **STS (Spring Tool Suite):** For developing the backend of the Employee Administration System project using Spring Boot Microservices.
2. **VS Code (Visual Studio Code):** For developing the frontend of the Employee Administration System project using React JS.
3. **MySQL Workbench:** For efficient querying and manipulation of employee information.

12.User Modules in Employee Administration System

The project has three user modules:

12.1.ADMINISTRATOR MODULE

12.2.MANAGER MODULE

12.3.EMPLOYEE MODULE

13.Functional Modules

13.1.User Authentication Module:

- User Authentication and Authorization with Spring Boot and React.
- Registration and Login systems have been implemented so that only authenticated users (Admin, Manager, or Employee) can perform their functionalities.

13.2.Department Module:

- Add Department, Update Department & Delete Department, View Departments.

13.3.Administrator Module:

- Register Administrator, Update Administrator, Delete Administrator, View Administrators.

13.4.Employee Module:

- Register Employee, Update Employee, Delete Employee, View Employees, View My Profile.

13.5.Employee Salary Module:

- Add Employee Salary, Update Salary, Fetch Employee Salary Details, View My Salary Detail.

14.Overview

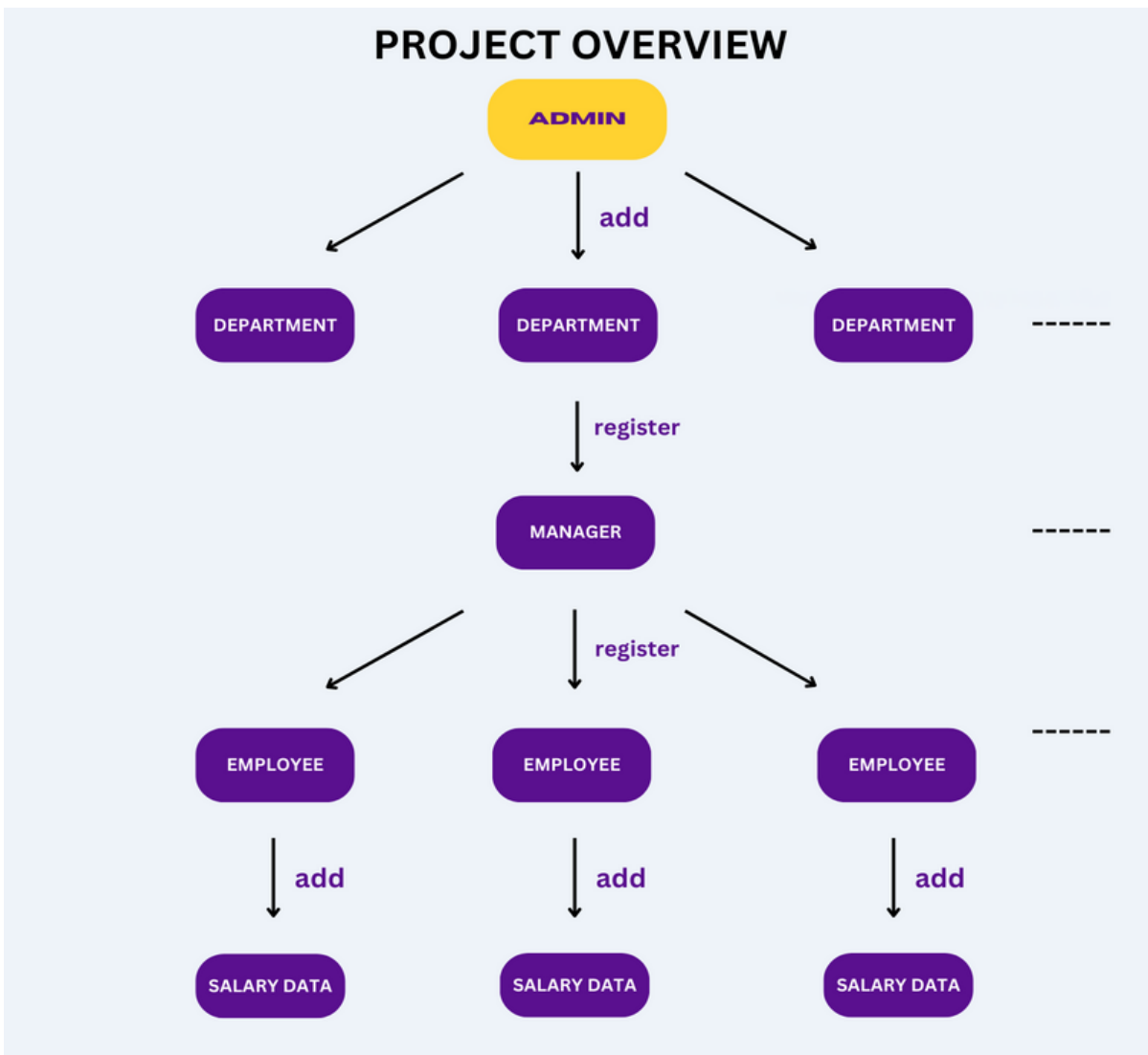
The Employee Administration System Project consists of three modules:

14.1.Department Administration: Admins can add and manage departments.

14.2.Manager Administration: Admins can add and manage managers.

14.3.Employee Administration: Managers can register and manage employees, including salary information.

The Admin is responsible for managing the overall structure, Managers oversee employees within their assigned departments, and Employees can view their profiles and salary details.



15.The Project Structure

15.1.BACKEND: For the development of the backend, Spring Boot Microservices have been used. There are five microservices in the backend.

Employee Administration System Spring Boot Microservices:

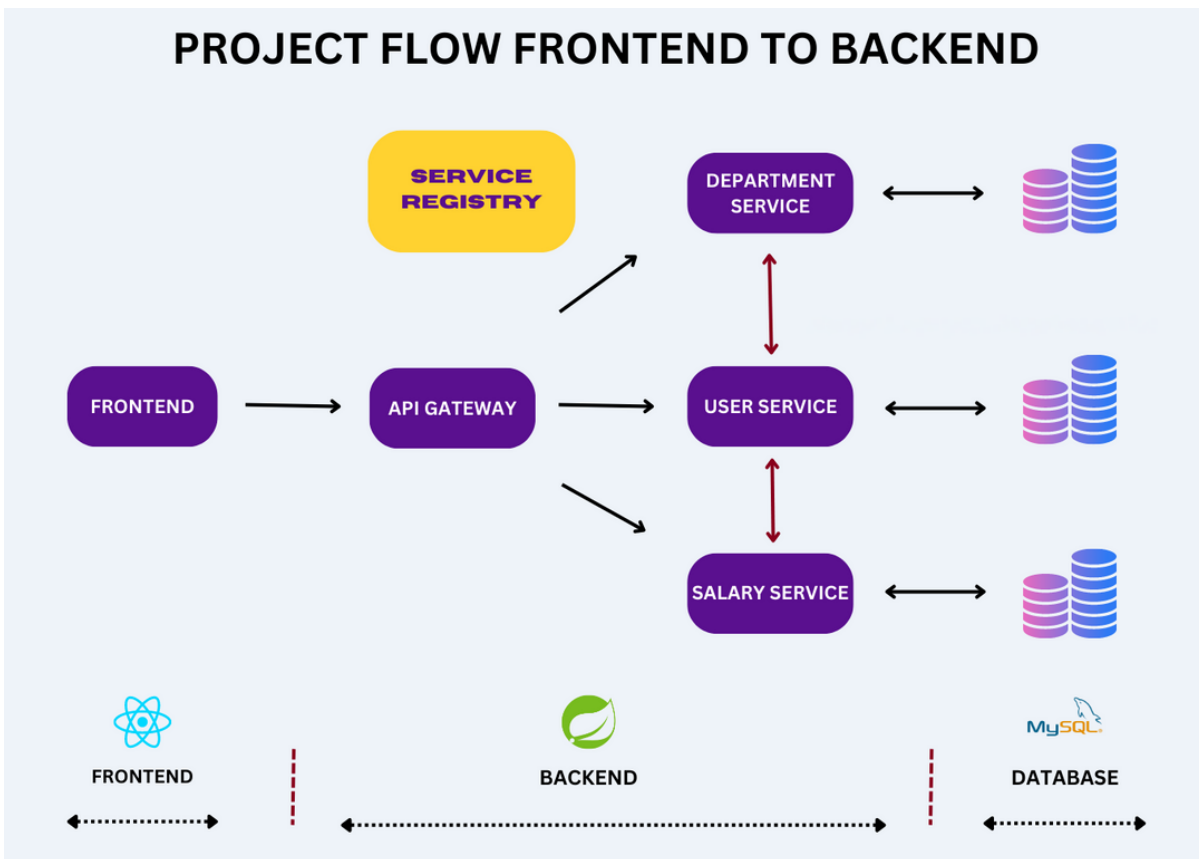
1. **API Gateway Service:** An API Gateway acts as a single entry point for client applications to interact with various microservices within the system.
2. **Service Registry Service:** A Service Registry facilitates service discovery and allows microservices to locate and communicate with each other dynamically.
3. **Employee User Service:** This microservice handles all user-related operations.
4. **Employee Department Service:** This microservice handles all department-related operations.
5. **Employee Salary Service:** This microservice handles all employee salary-related operations.

The request first goes to the API Gateway Service during the request initialization from the frontend. The microservices communicate with each other to form the required response, which is then sent to the frontend. Each microservice connects to a database (e.g., MySQL, Postgres, Oracle) to retrieve and store data.

FRONTEND: The frontend of the Employee Administration System is developed using React Js.

DATABASE: For the Database, MySQL Server has been used for storing the data. Other databases like Postgresql, Oracle, etc., can be configured based on requirements.

16.The Project Flow



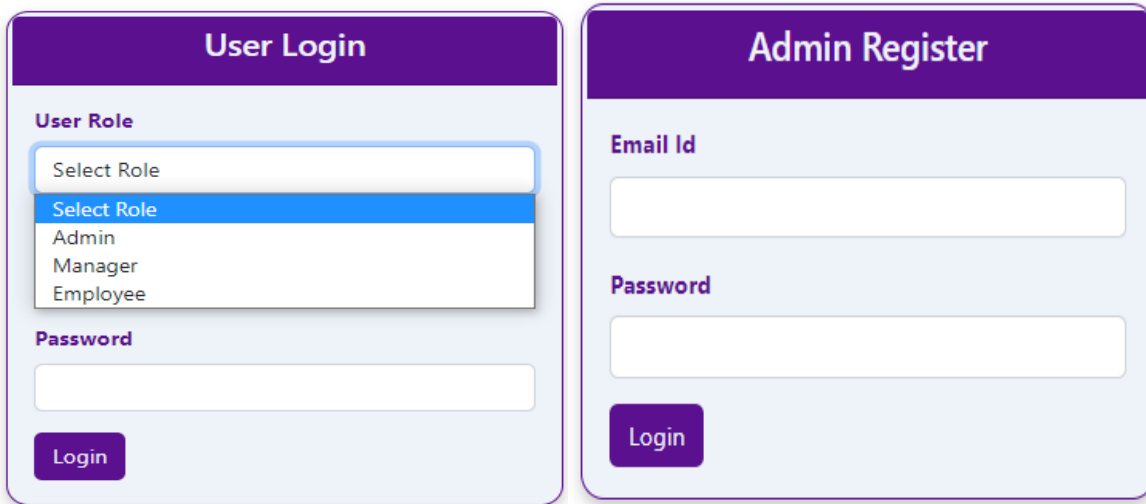
1. All microservices are registered in the SERVICE REGISTRY.
2. Requests flow from the Frontend to the Backend.
3. In the Backend, the request goes to the API GATEWAY.
4. The API GATEWAY routes the request to the required SERVICE (user service, department service, or salary service).
5. Each microservice communicates with one another to retrieve data.
6. Each microservice connects to a database for data retrieval and storage.

17.ROLE OF USERS

17.1) Admin Role

i) Admin Register & Login

Admin can register into the system by using email and password as shown below.

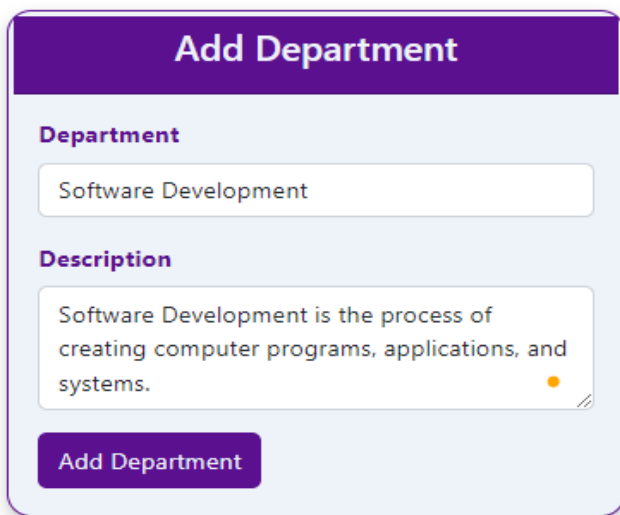


The image displays two side-by-side web forms. The left form, titled 'User Login', features a 'User Role' dropdown menu with options 'Select Role', 'Admin', 'Manager', and 'Employee'. Below this is a 'Password' input field and a 'Login' button. The right form, titled 'Admin Register', includes an 'Email Id' input field, a 'Password' input field, and a 'Login' button.

Admin can log in to the system by using email, password, and his role as shown below. So basically this is role-based login that means, by using the same login page, Admin, Manager & Employee can log in by selecting their role as shown below.

ii) Add Department

Admin can add the departments in the system as shown below.



The image shows a web form titled 'Add Department'. It contains a 'Department' input field with the text 'Software Development' and a 'Description' text area with the text 'Software Development is the process of creating computer programs, applications, and systems.' Below these fields is an 'Add Department' button.

iv) View All Departments

Admin can view all the departments as shown below.

All Departments		
Department	Description	Action
Software Development	Software Development is the process of creating computer programs, applications, and systems.	<button>Remove</button> <button>Update</button>
Marketing	Marketing is the process of creating, communicating, delivering, and exchanging offerings that have value for customers, clients, partners, and society at large.	<button>Remove</button> <button>Update</button>
Deployment	Deployment is the process of making a software system available for use.	<button>Remove</button> <button>Update</button>
Delivery	Delivery is the process of ensuring that a product or service is delivered to the customer in a timely, efficient, and effective manner.	<button>Remove</button> <button>Update</button>

v) Delete or Update Department

Admin can delete the departments by clicking on the remove button as shown above. And also he can update the Departments by clicking on the update button, after clicking on the update button, the admin will be able to see the below page.

Update Department

Department

Description

Marketing is the process of communicating, and exchanging offerings that have value for clients.

Update Department

vi) Manager Register

Admin can register manager for the department in employee management system application

Register manager

First Name

Last Name

Email Id

Password

User Gender

Select Gender

Department

Select Department

Contact No

Age

Experience

Street

City

Pincode

Select Profile Image




Choose File

No file chosen

Register User

vii) View All Managers

Admin can view all the managers present in the EMS application as shown below.

All Manager													
Manager	Manager Id	First Name	Last Name	Email Id	Contact	Department	Experience	Age	Gender	Street	City	Pincode	Action
	2	Manager 1	Demo	manager1@demo.com	1234567890	Software Development	10	35	Male	Demo Street 1	Demo City	000000	Remove Update
	3	Manager2	Demo Last	manager2@gmail.com	0000000000	Software Development	15	40	Male	Demo Street 10	Demo City1	111111	Remove Update
	4	Manager 3	Demo Last	manager3@demo.com	2222222222	Software Development	24	45	Male	Demo Street 20	Demo City10	333333	Remove Update

viii) Delete or Update Managers

Admin can delete the Managers by clicking on the remove button as shown above. And also he can update the Managers by clicking on the update button, after clicking on the update button, the admin will be able to see the below page.

Update manager

First Name

Last Name

Email Id

Contact No

Age

Experience

Street

City

Pincode

Select Profile Image




Choose File

No file chosen

Update User

ix) View All Employees

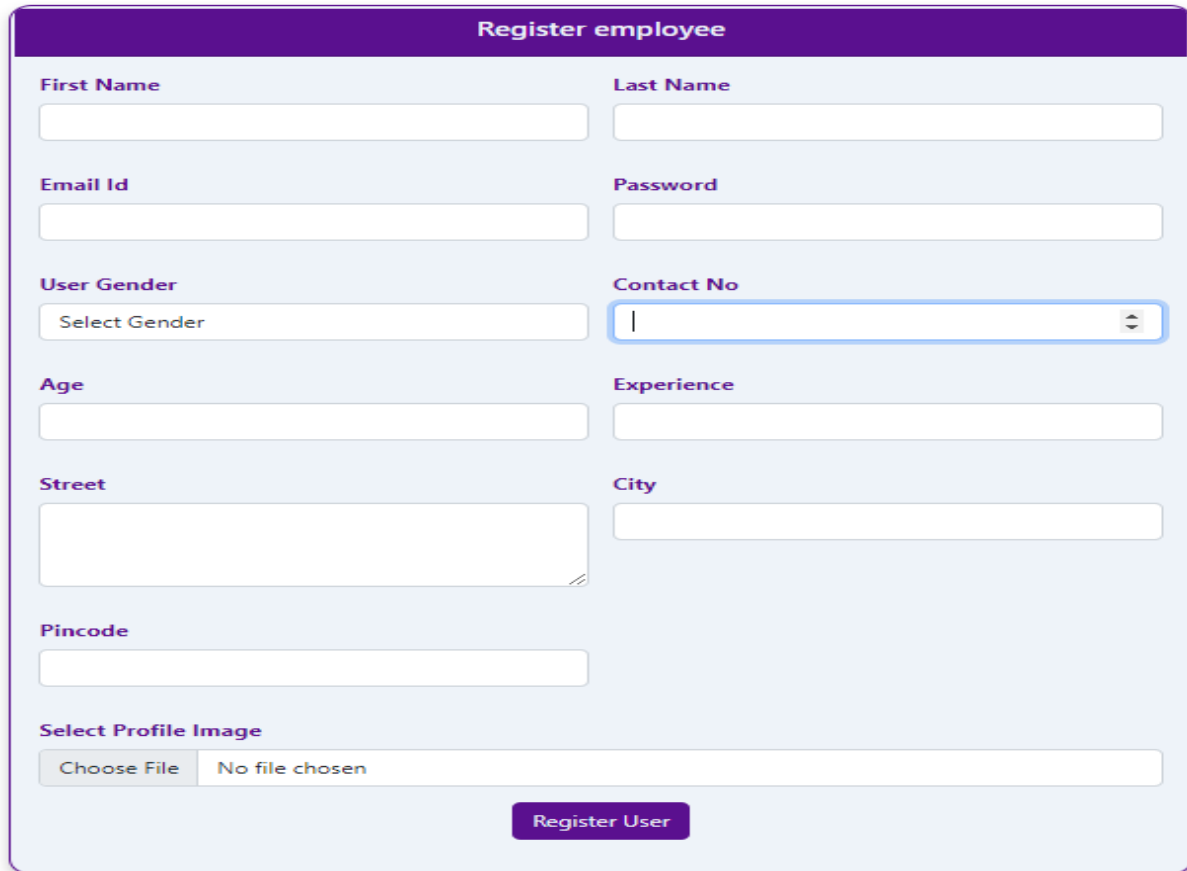
Admin can view all the employees in the departments present in the EMS application as shown below.

All Employee													
Employee	Employee Id	First Name	Last Name	Email Id	Contact	Department	Experience	Age	Gender	Street	City	Pincode	Salary Details
	5	Employee 1	Demo Name	employee1@demo.com	3333333333	Software Development	2	25	Male	Demo Street 1	Demo City	000000	View Remove Update
	6	Employee 2	Demo name	employee2@demo.com	7777777777	Software Development	3	27	Male	Demo Street	Demo City	999999	View Remove Update
	7	Employee 3	Demo name	employee3@demo.com	4444444444	Software Development	2	25	Female	Demo Street	Demo City	666666	View Remove Update

17.2) Manager Role

i) Register Employee

Managers can register employees into the system by using the below register employee form.



The form is titled "Register employee" and contains the following fields and controls:

- First Name**: Text input field.
- Last Name**: Text input field.
- Email Id**: Text input field.
- Password**: Text input field.
- User Gender**: Dropdown menu with "Select Gender" as the placeholder.
- Contact No**: Text input field with a numeric keypad icon on the right.
- Age**: Text input field.
- Experience**: Text input field.
- Street**: Text input field.
- City**: Text input field.
- Pincode**: Text input field.
- Select Profile Image**: File upload control with a "Choose File" button and "No file chosen" text.
- Register User**: Purple button at the bottom right.

ii) Delete & Update Employees




Managers can delete the Employees by clicking on the remove button as shown above. And also he can update the Employees by clicking on the update button, after clicking on the update button, the managers will be able to see the below page.

Update employee

First Name	Last Name
<input type="text" value="Employee 1"/>	<input type="text" value="Demo Name"/>
Email Id	Contact No
<input type="text" value="employee1@demo.com"/>	<input type="text" value="3333333333"/>
Age	Experience
<input type="text" value="25"/>	<input type="text" value="2"/>
Street	City
<input type="text" value="Demo Street 1"/>	<input type="text" value="Demo City"/>
Pincode	
<input type="text" value="000000"/>	
Select Profile Image	
<input type="button" value="Choose File"/> <input type="text" value="No file chosen"/>	
<input type="button" value="Update User"/>	

iii) View Department Employees

Managers can view all the active employees present in his department in the EMS application.

All Employee													
Employee	Employee Id	First Name	Last Name	Email Id	Contact	Department	Experience	Age	Gender	Street	City	Pincode	Salary Details
	5	Employee 1	Demo Name	employee1@demo.com	3333333333	Software Development	2	25	Male	Demo Street 1	Demo City	000000	View Remove Update
	6	Employee 2	Demo name	employee2@demo.com	7777777777	Software Development	3	27	Male	Demo Street	Demo City	999999	View Remove Update
	7	Employee 3	Demo name	employee3@demo.com	4444444444	Software Development	2	25	Female	Demo Street	Demo City	666666	View Remove Update

iv) Add Employee Salary

Managers can add the salary details for the active employees present in their department in EMS.

Add Employee Salary

User Id

Bank Name

Bank Account Number

Bank Ifsc

Salary

Pay Cycle

Payment Mode

From Date

To Date

Add Salary

v) View Employee Salary

Managers can view their departments with any employee's salary details as shown below.

Salary Detail								
Salary	From Date	To Date	Payment Mode	Pay Cycle	Bank	Account No	Ifsc Code	Action
50000	2022-01-01	2022-12-31	Bank Transfer	Month	World Bank	1234567890	WORLD12345	Update
70000	2023-01-01	2023-12-31	Bank Transfer	Month	World Bank	1234567890	WORLD12345	Update

vi) Update Employee Salary

Managers can update their departments with any employee's salary details as shown below.

Update Employee Salary

User Id	Bank Name
<input type="text" value="5"/>	<input type="text" value="World Bank"/>
Bank Account Number	Bank Ifsc
<input type="text" value="1234567890"/>	<input type="text" value="WORLD12345"/>
Salary	Pay Cycle
<input type="text" value="50000"/>	<input type="text" value="Select Pay Cycle"/>
Payment Mode	From Date
<input type="text" value="Select Payment Mode"/>	<input type="text" value="01-01-2022"/>
To Date	
<input type="text" value="31-12-2022"/>	
<input type="button" value="Update Salary"/>	

17.3) Employee Role

i) View Profile

After logging in to the EMS system, Employee can see their profile as shown below.

Employee Salary Details



Employee Name:Employee 1 Demo Name

Age:25

Gender:Male

Email Id:employee1@demo.com

Contact:3333333333

Address:Demo Street 1, Demo City, 000000

Department:Software Development

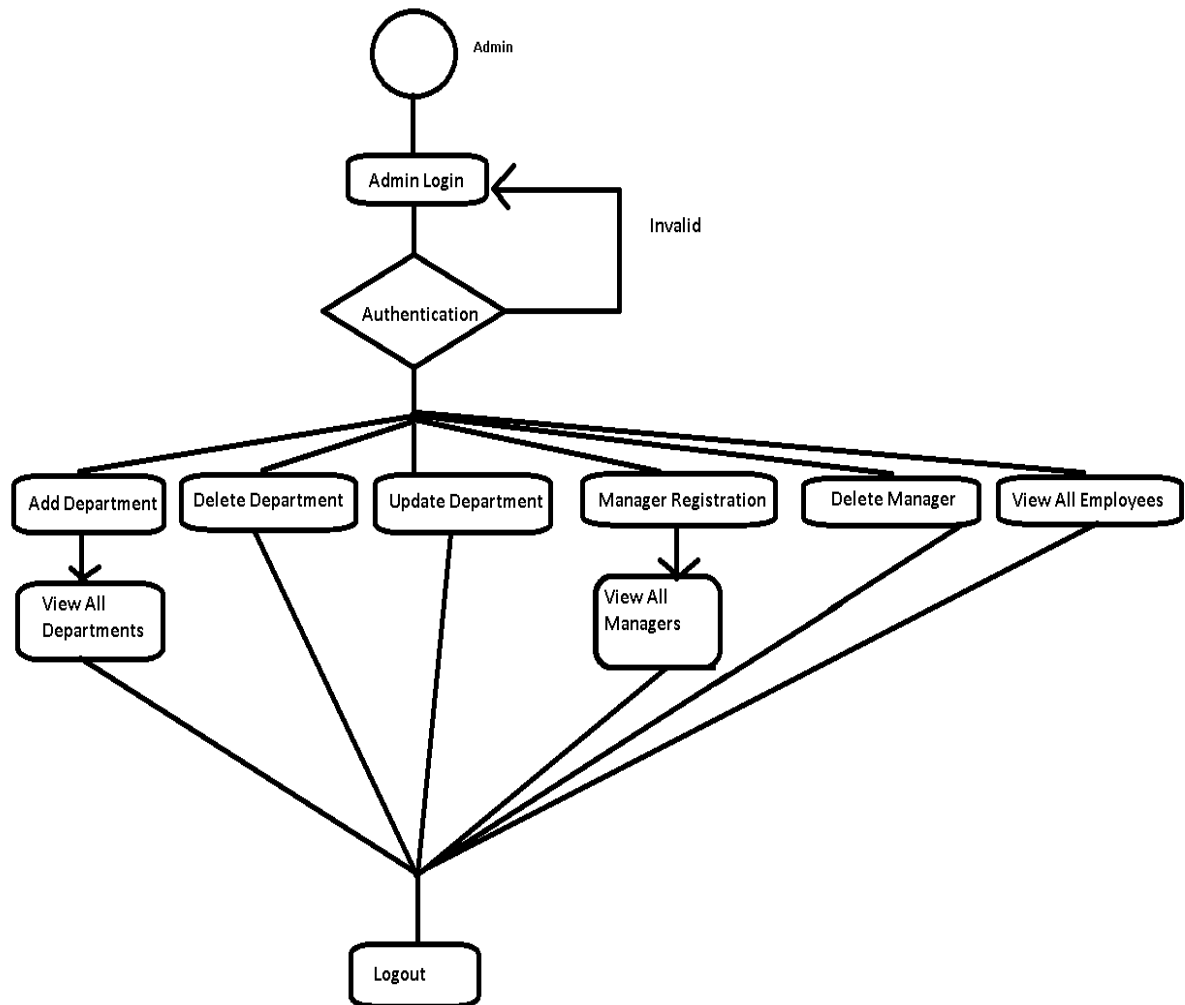
Experience:2 years

i) View Salary Details

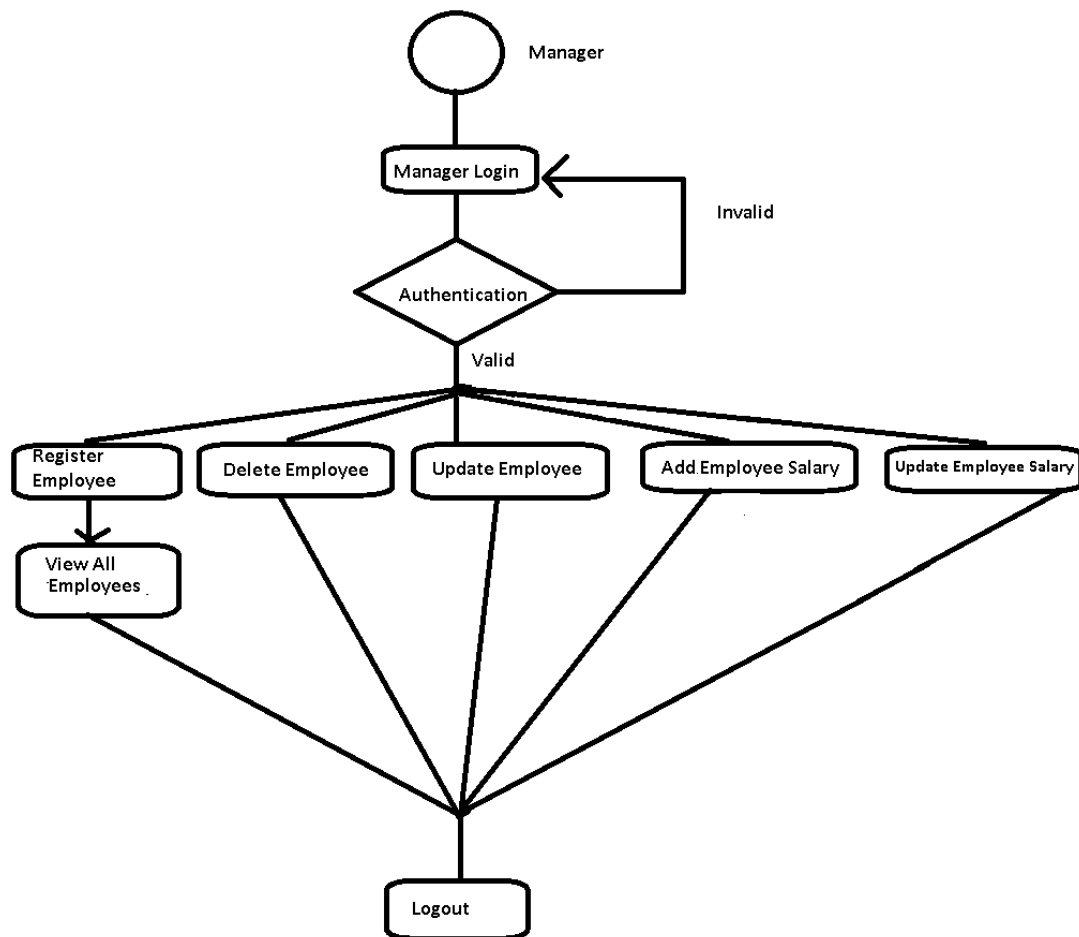
An employee can see their salary details from start to end after logging into the EMS application as shown below.

Salary Detail							
Salary	From Date	To Date	Payment Mode	Pay Cycle	Bank	Account No	Ifsc Code
50000	2022-01-01	2022-12-31	Bank Transfer	Month	World Bank	1234567890	WORLD12345
70000	2023-01-01	2023-12-31	Bank Transfer	Month	World Bank	1234567890	WORLD12345

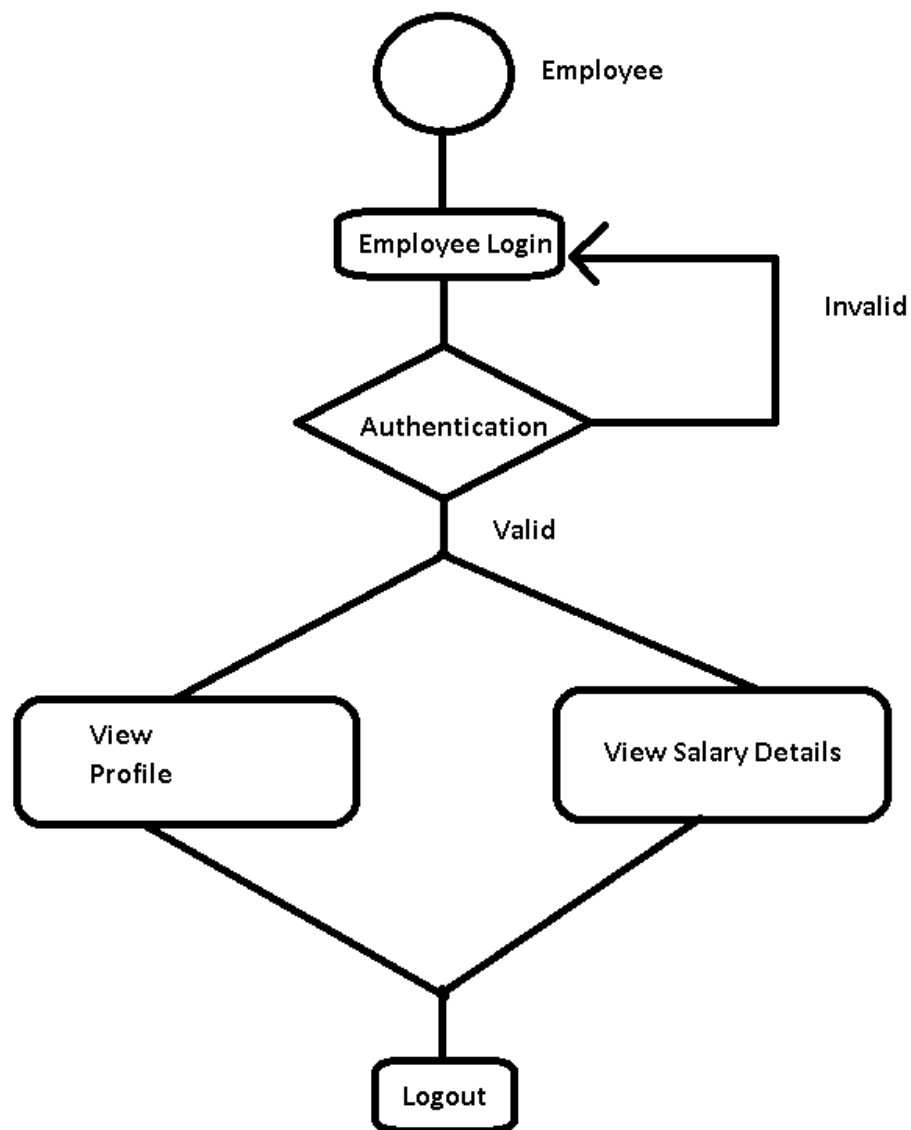
18. DIAGRAMS



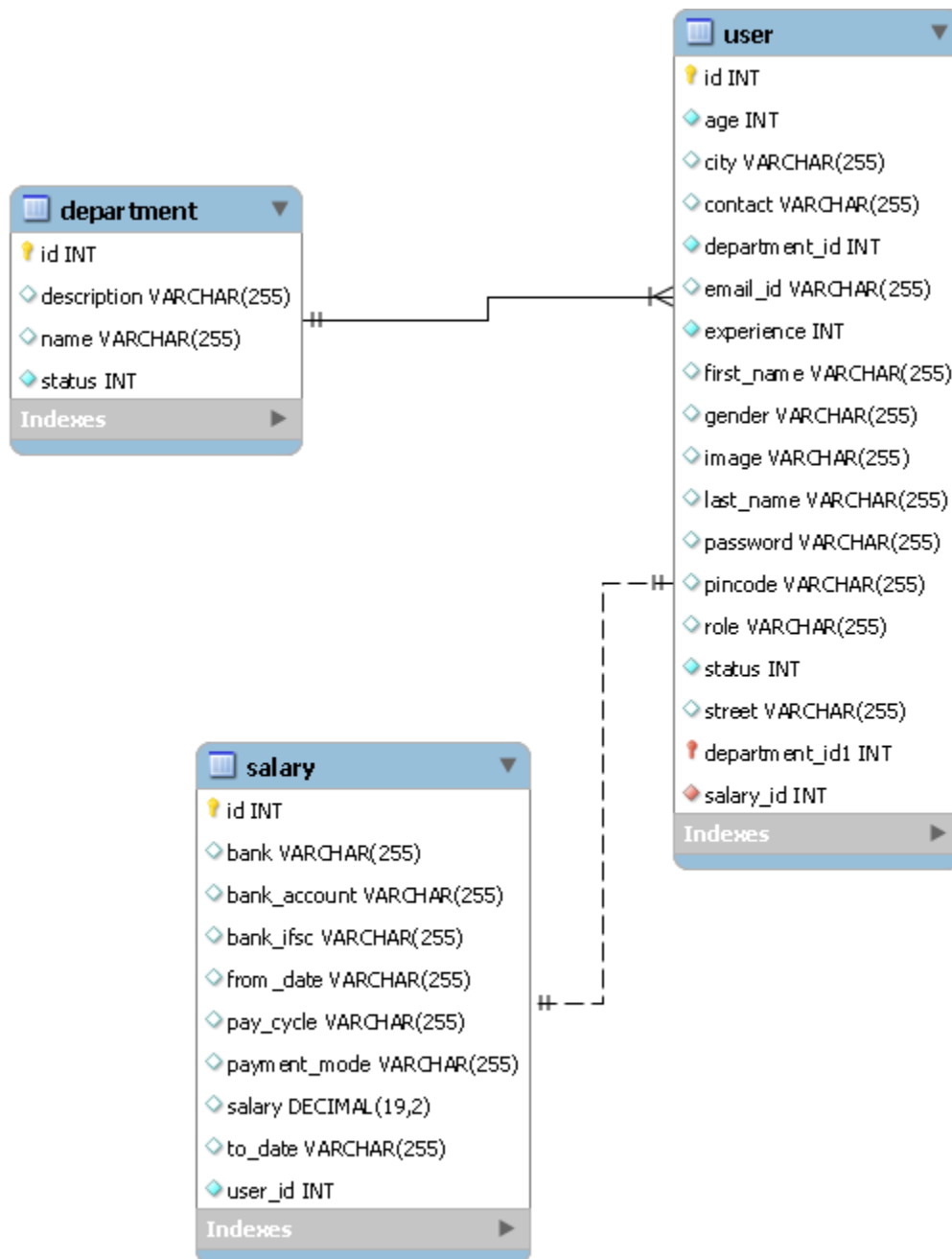
18.1 Admin Activity Diagram



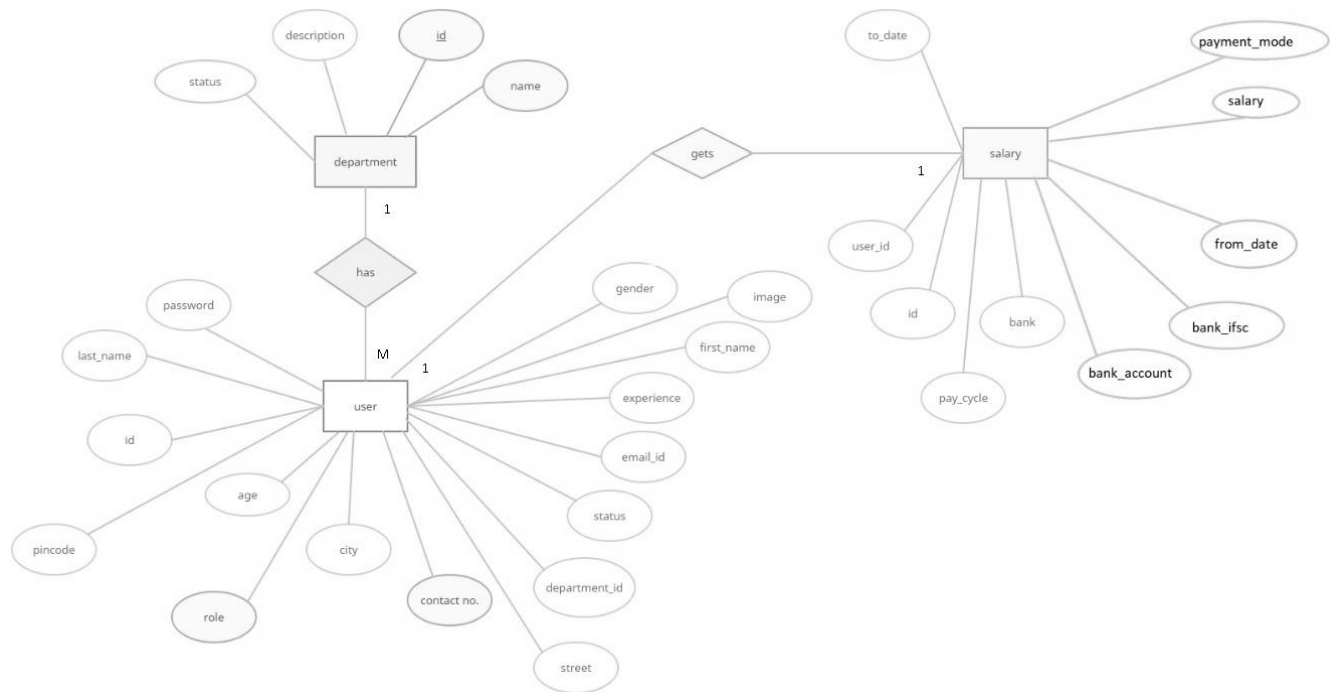
18.2. Manager Activity Diagram



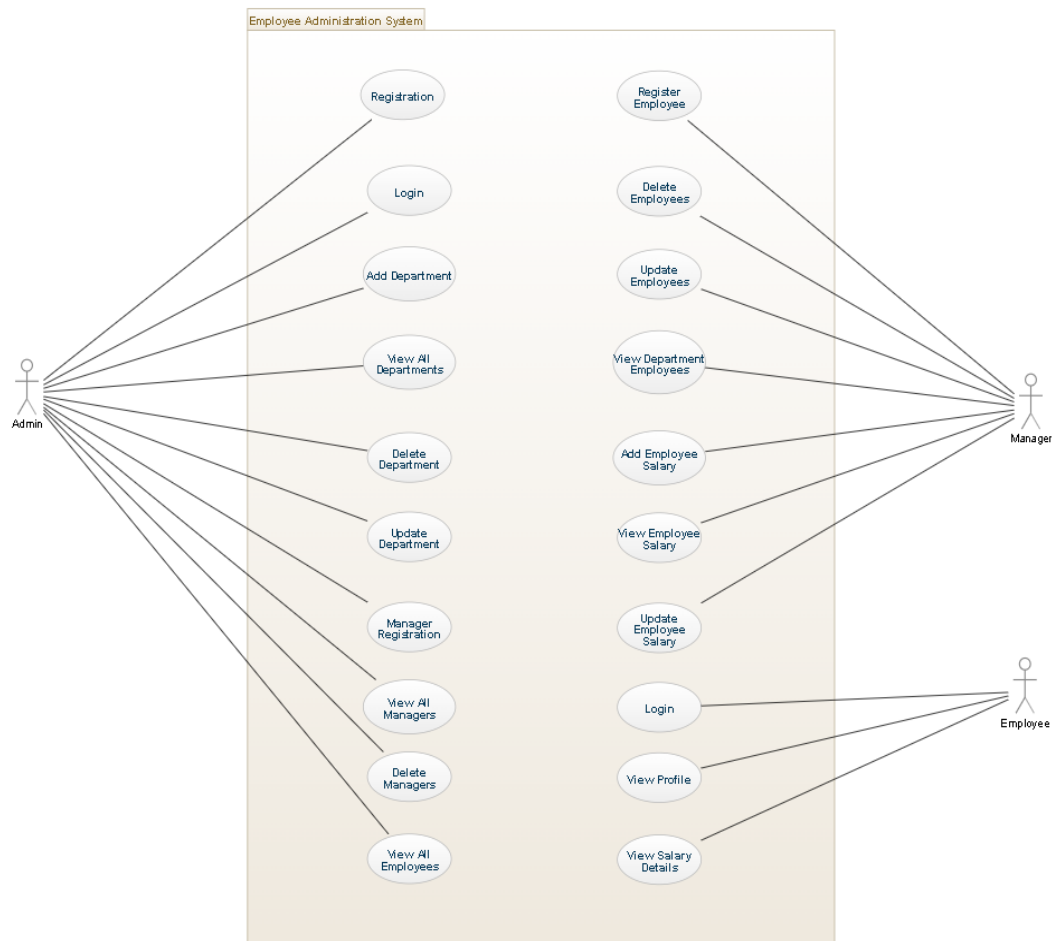
18.3 Employee Activity Diagram



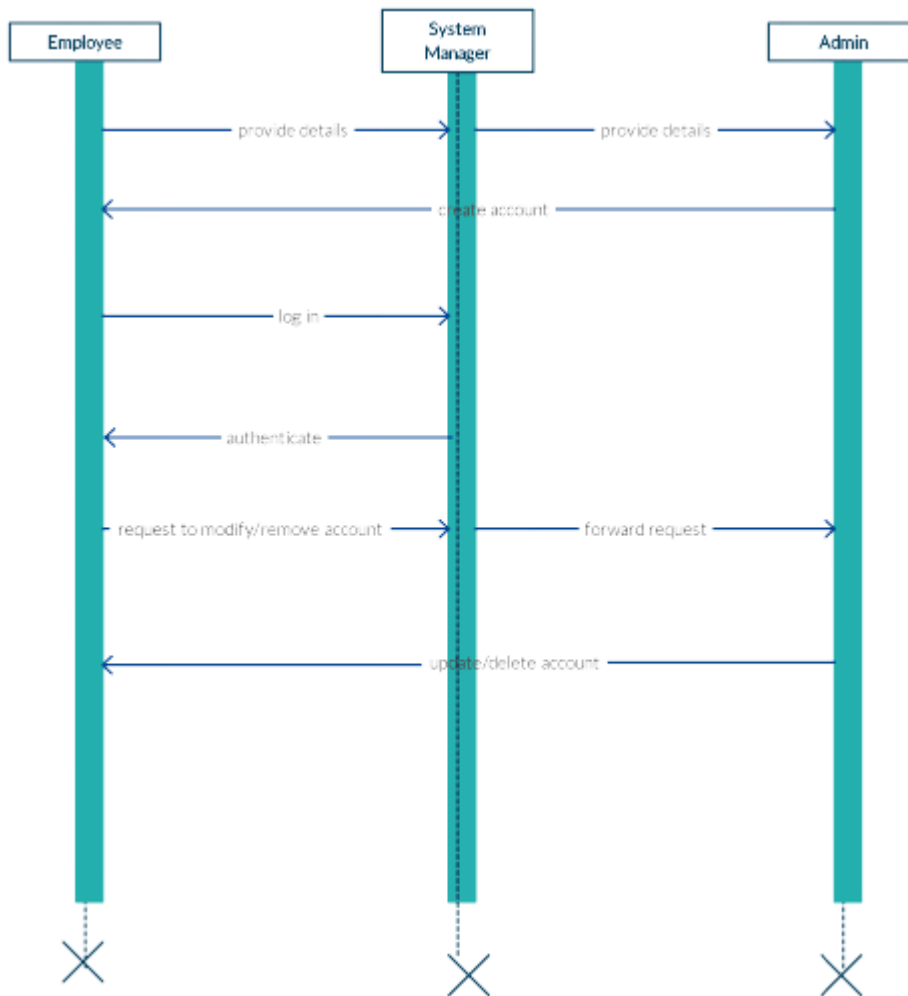
18.4 Class Diagram



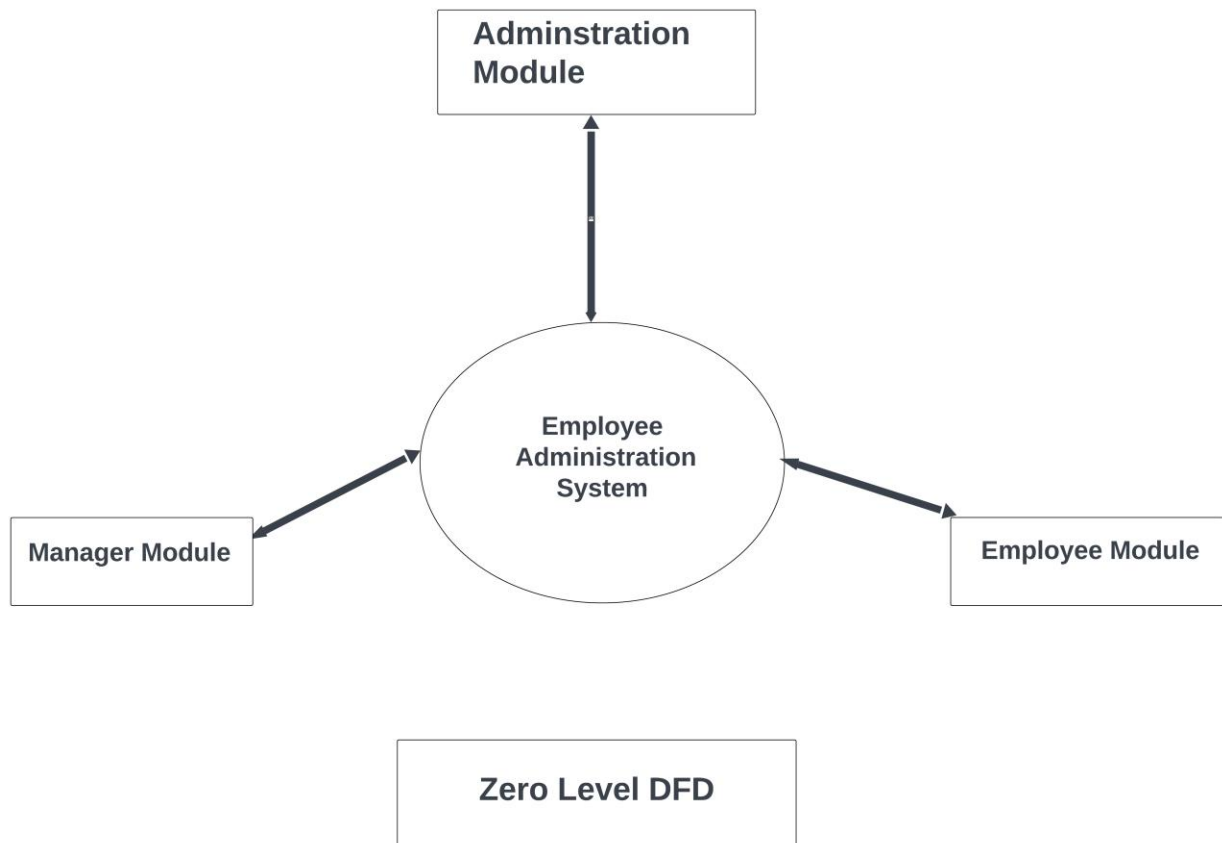
18.5 E-R Diagram



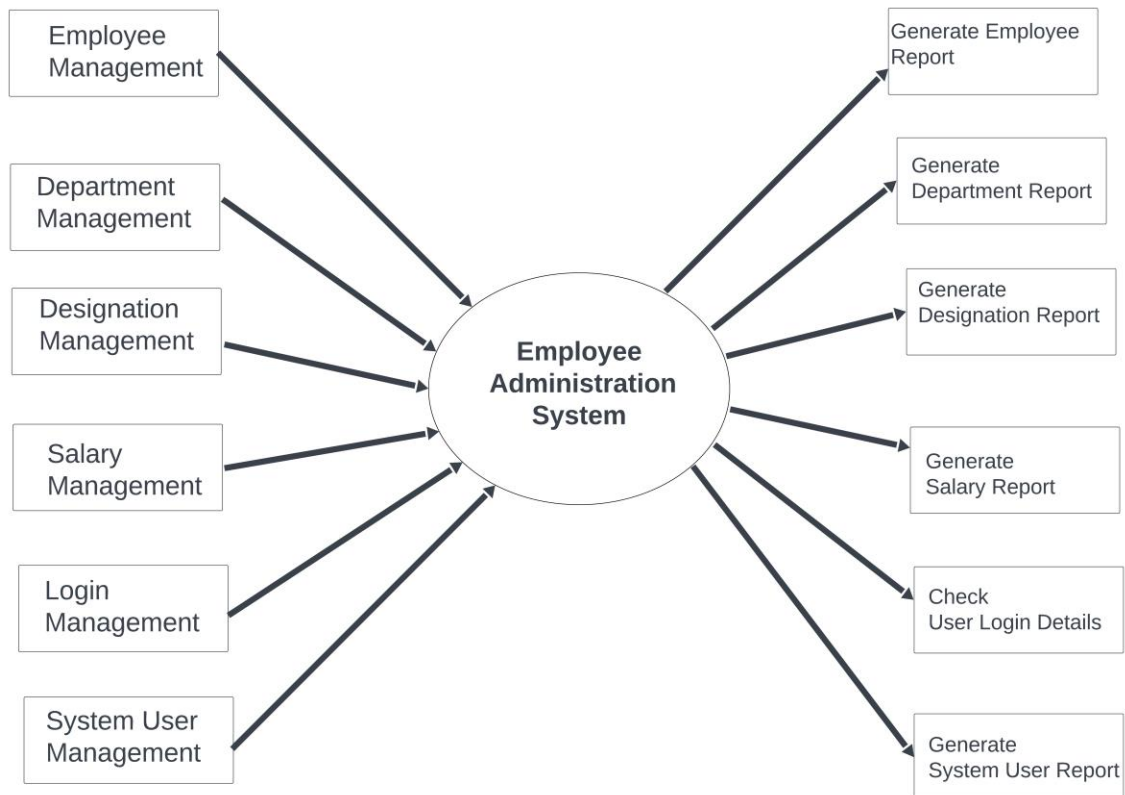
18.6 Use Case Diagram



18.7 Sequence Diagram

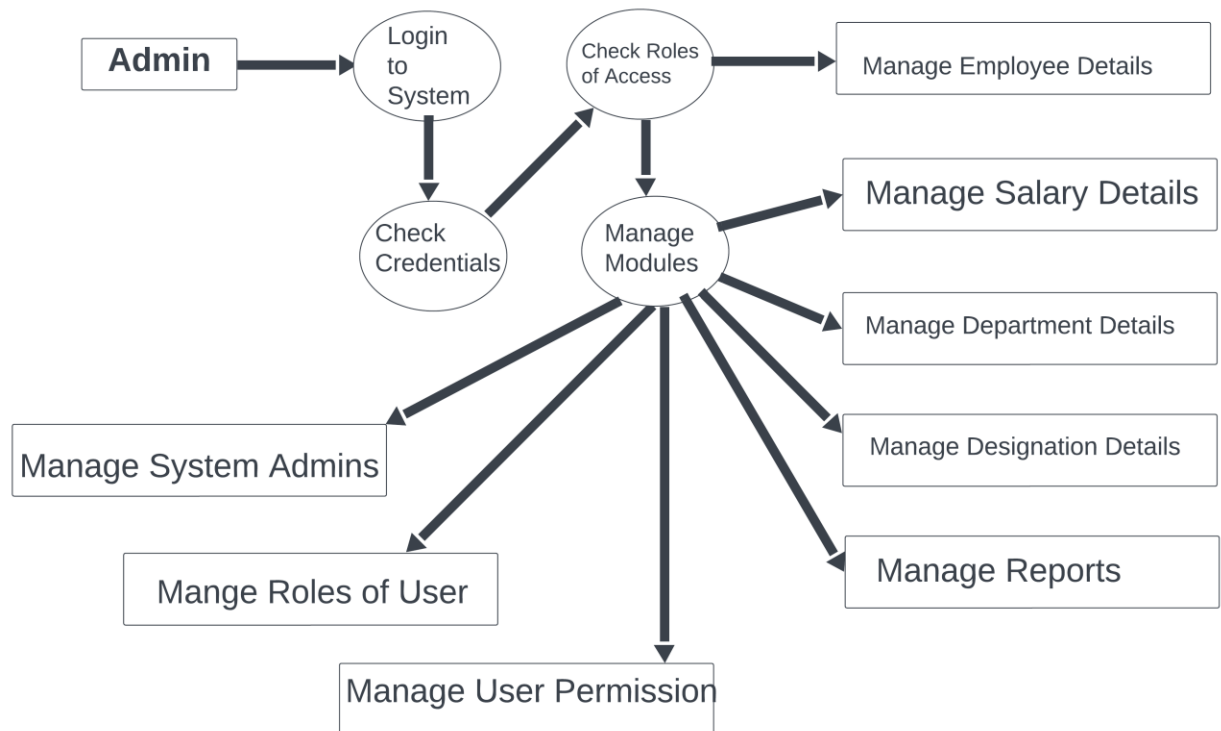


18.8 Zero Level DFD



First Level DFD

18.9 First Level DFD



Second Level DFD

18.10 Second Level DFD

19 .CONCLUSION

The project entitled **Employee Administration System** was completed successfully.

An Employee Administration System Project in Spring Boot and React empowers administrators to efficiently manage departments, administrators, employees, and salary information. This project delves into the functionalities of an Employee Administration System (EAS) wherein administrators can effortlessly add departments and assign administrators liable for overseeing employee administration. Additionally, the system allows the recording and monitoring of salary information for each employee. Discover how this comprehensive solution enhances organizational efficiency, simplifies administrative tasks, and guarantees accurate control of employee records and compensation. Streamline your employee administration tactics and optimize productivity with an Employee Administration System tailored to satisfy the specific desires of your agency.

An Employee Administration System is developed using Spring Boot Microservices, React JS & MySQL. By using this application, Admins can add Managers in the application based on the departments, and Managers can administer the Employee Data, including their Salary and complete details.

20. BIBLIOGRAPHY / REFERENCES :

1. Spring Boot Documentation. (<https://spring.io/projects/spring-boot>)

Reference for Spring Boot framework used in the backend development.

2. React JS Documentation. (<https://legacy.reactjs.org/docs/getting-started.html>)

Reference for React JS framework used in the frontend development.

3. MySQL Documentation. (<https://dev.mysql.com/doc/>)

Reference for the MySQL database management system used in the project