

# EMPLOYEE MANAGEMENT SYSTEM

USING PYTHON AND MYSQL DATABASE

BY

RAHUL GARAI

MD SAHID RAJA

# INTRODUCTION

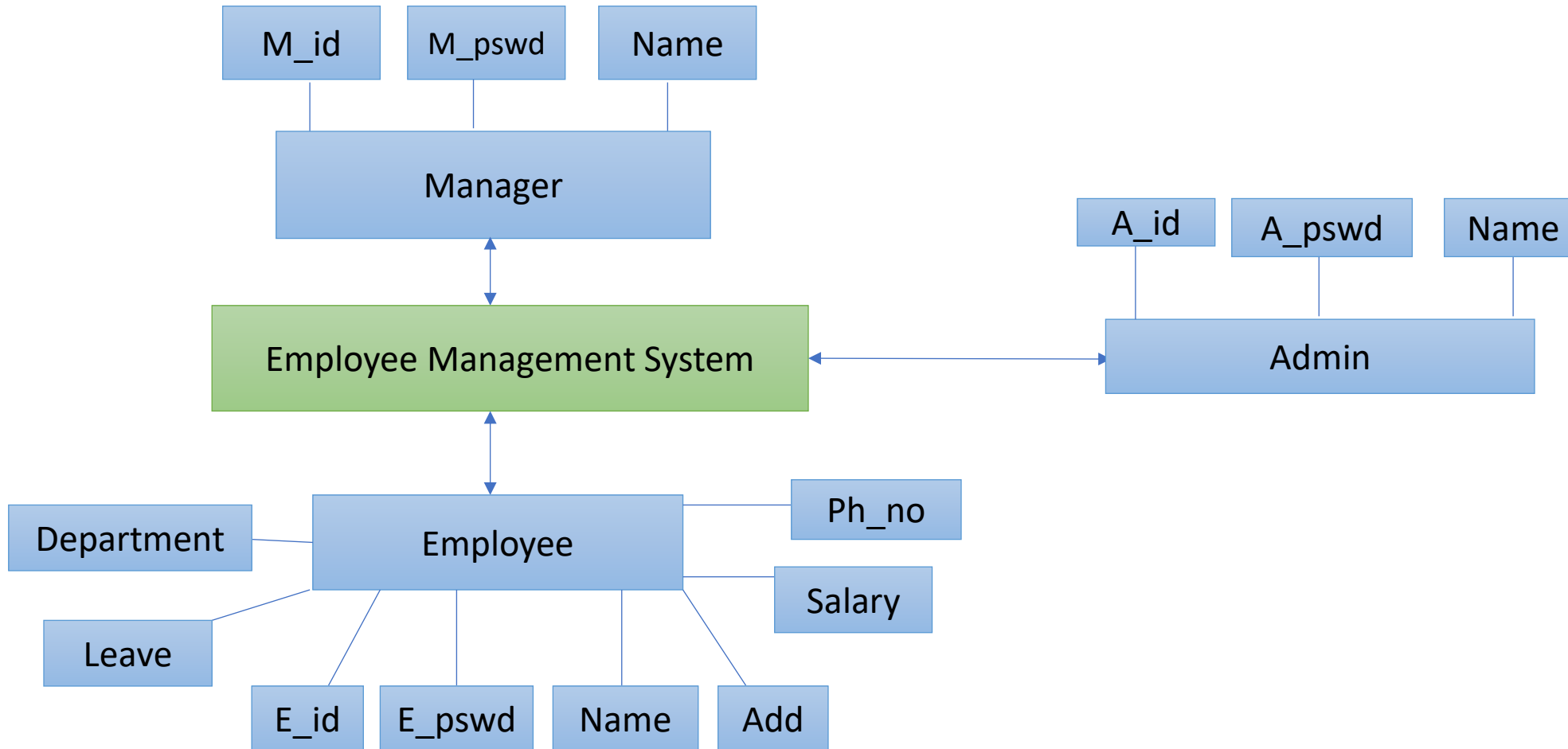
- Employee Management System Project is designed to keep track of employee information in any company. It stores data such as their employees' personal information leave details and salary details.
- The employee management system project gives managers a better idea of their employees and helps them plan and manage their work hours to cut costs and boost productivity. It gives appropriate directions and supervisions for employees. It also secures and manages information that are important to the employees including personal and work-related information.
- Employee management system is developed to manage the data and information of an employee in a company. It is developed to override the problems prevailing in the practicing manual system.

# Overview of the Problem

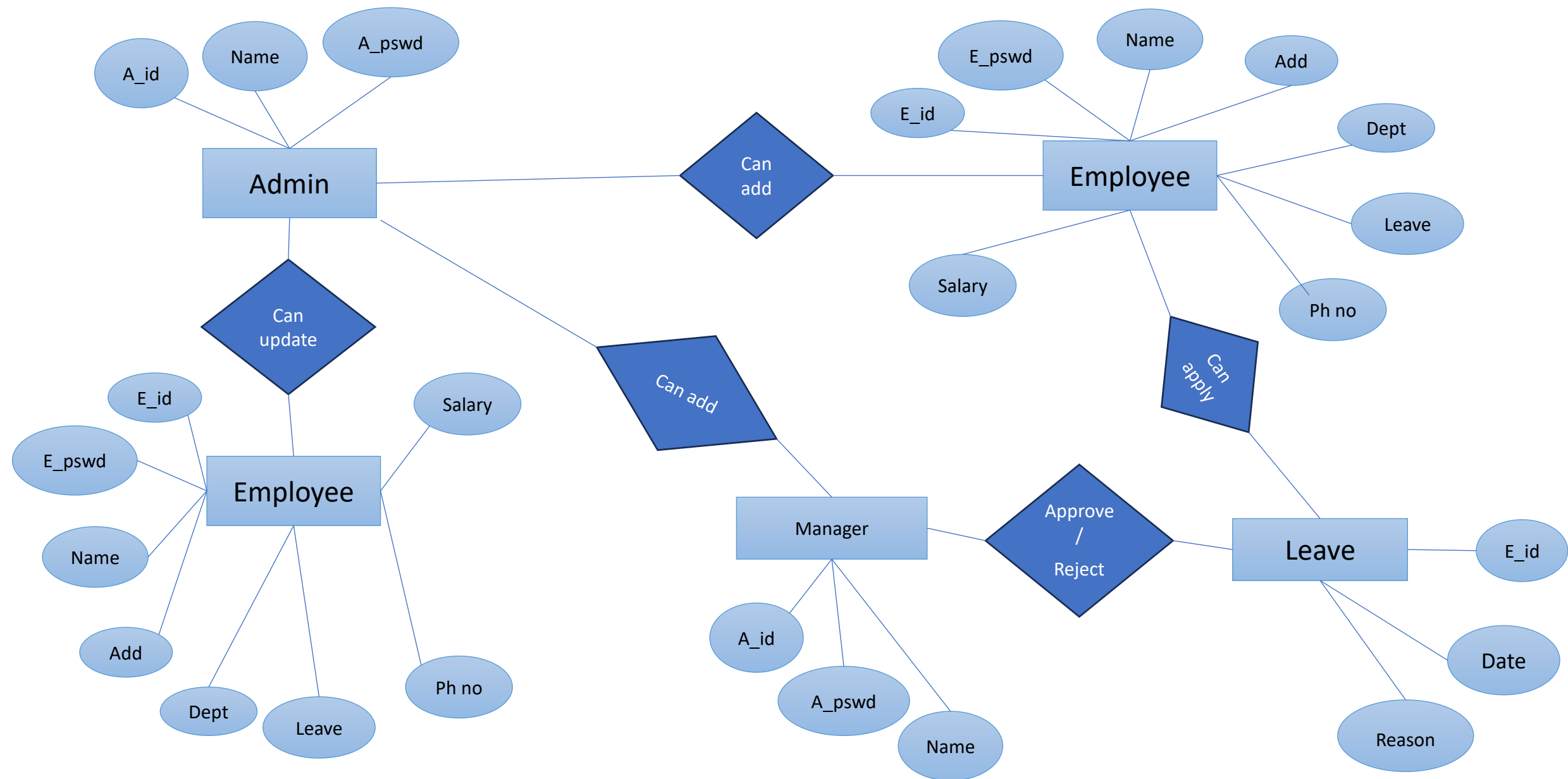
Due too many data and paperwork that needed to record the employee data could consume a lot of space in the filling cabinet. The retrieval of data can time consuming because it must be searched from the filling cabinet. This will cause waste of resource in term of time and money. In addition, it would also cause inconvenience and ineffectiveness in daily work. Plus, the manger will face difficulties when need to update employee working schedule, report and leave request.

In the employee point of view, when they need to request for leave, they need to fill in a leave request form manually and submit to manager personally and wait for confirmation, this is time consuming. Other than that, if there are any changes in working schedule, employee might have wrong information in the working schedule because the schedule might not update immediately, therefore the employee might not satisfy with the working schedule.

# Architecture diagram



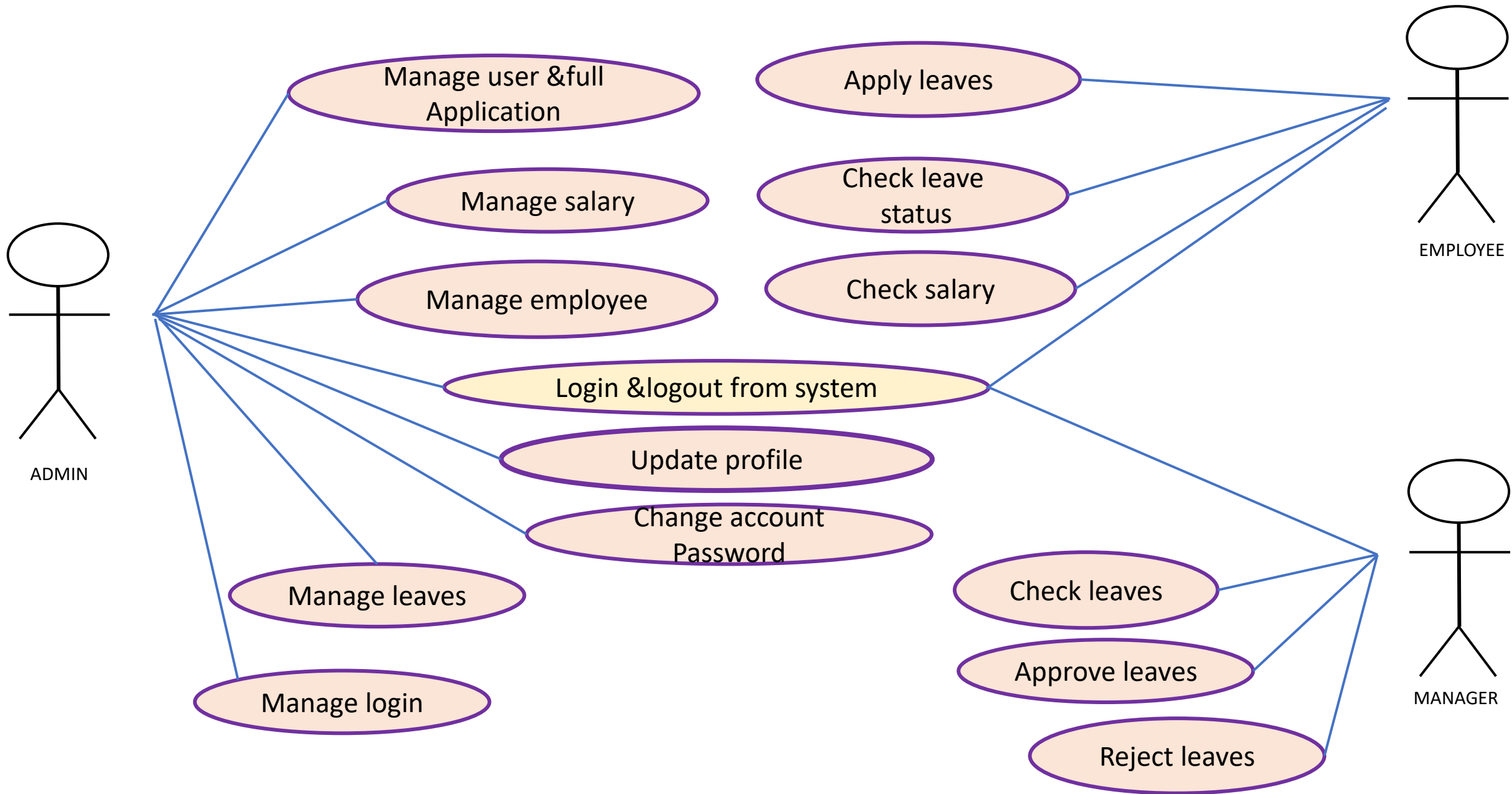
# Entry relationship diagram:



In the figure ER diagram of employee management system.

We have 5 different entities and each entity have its own-own attributes and relationship between them. The first entities are admin, and its attribute are admin id, admin name and admin password. The second entities we have employee, and its attribute are employee name, employee proof, employee phone number, employee address, employee department, employee salary, employee leave, employee password. And the relationship admin and employee are “can add” that is admin can add employee. The third entities are employee details and the relationship between admin and employee details “can update” that is admin can update the employee details. The fourth entities we have is manager and its attribute are manager id, manager password, manger name and the relationship between admin and manager is “can add” that is admin can add any manger. The last entities we have leave and its attribute are leave id, leave reason, leave type and leave date and the relationship between employee and leave are “can apply’ that is employee can apply for leave and the relationship between manager and leave is “can approve/reject” that is manager can either approve or reject employee leave.

# UML USE CASE DIAGRAM



In the figure UML Use Case Diagram of Employee Management System we have admin, manager and employee

- Admin: Admin can manage the full application and user. It can manage salary, employee, and leave. It can update employee details and change the password.
- Employee: Employee can apply for leave, check leave status and check salary. Employee can also login and logout from the system, update their profile and change account password.
- Manager: Manager can check leave, check timesheet and can approve or reject leaves.



# HOW TO RUN THIS PROJECT

- Open mysql and create database named **employee\_management\_system**.
- Create the following table- **admin, employee, manager, leave\_applications**.
- Open visual studio code or any other source code editor, go to the terminal and run following commands –
  - 1.pip install tk
  - 2.pip install mysql-connector-python
- Open **Employee Management** folder in source code editor and Update this line of code according to your mysql database.

```
conn = mysql.connector.connect(  
    host='localhost',  
    username='root',  
    password='your password ',  
    database='employee_management_system'  
)
```

- Then open login.py file and run this code.

# SQL QUARY FOR CREATE EMPLOYEE TABLE:

```
CREATE TABLE employee (  
    Emp_ID INT AUTO_INCREMENT PRIMARY KEY, -- Auto-incrementing primary key for employee ID  
    Manager_ID INT, -- Foreign key or reference to Manager (optional)  
    Department VARCHAR(50),  
    Designation VARCHAR(50),  
    Name VARCHAR(50),  
    Mobile_Number VARCHAR(15),  
    DOJ DATE, -- Date of joining  
    Email VARCHAR(100),  
    Country VARCHAR(50),  
    City VARCHAR(50),  
    Married_Status VARCHAR(50),  
    DOB DATE, -- Date of birth  
    ID_Type VARCHAR(50),  
    ID_Proof VARCHAR(100),  
    Gender VARCHAR(10),  
    Salary DECIMAL(10, 2), -- Salary with two decimal places  
    Password VARCHAR(50) -- Password field  
);  
  
CREATE TRIGGER generate_password_before_insert  
BEFORE INSERT ON employee  
FOR EACH ROW  
SET NEW.Password = CONCAT(UPPER(SUBSTRING(NEW.Name, 1, 4)), YEAR(NEW.DOB));
```

# SQL QUARY FOR CREATE MANAGER TABLE:

```
CREATE TABLE manager (  
    Manager_ID INT AUTO_INCREMENT PRIMARY KEY, -- Auto-incrementing primary key for manager ID  
    Department VARCHAR(50),  
    Name VARCHAR(50),  
    Mobile_Number VARCHAR(15),  
    DOJ DATE, -- Date of joining  
    Email VARCHAR(100),  
    Country VARCHAR(50),  
    City VARCHAR(50),  
    Married_Status VARCHAR(10),  
    DOB DATE, -- Date of birth  
    ID_Type VARCHAR(50),  
    ID_Proof VARCHAR(50),  
    Gender VARCHAR(10),  
    Salary DECIMAL(10, 2), -- Salary with two decimal places  
    Password VARCHAR(50) -- Password field  
);  
CREATE TRIGGER set_password_before_insert  
BEFORE INSERT ON manager  
FOR EACH ROW  
SET NEW.Password = CONCAT(UPPER(SUBSTRING(NEW.Name, 1, 4)), YEAR(NEW.DOB));
```

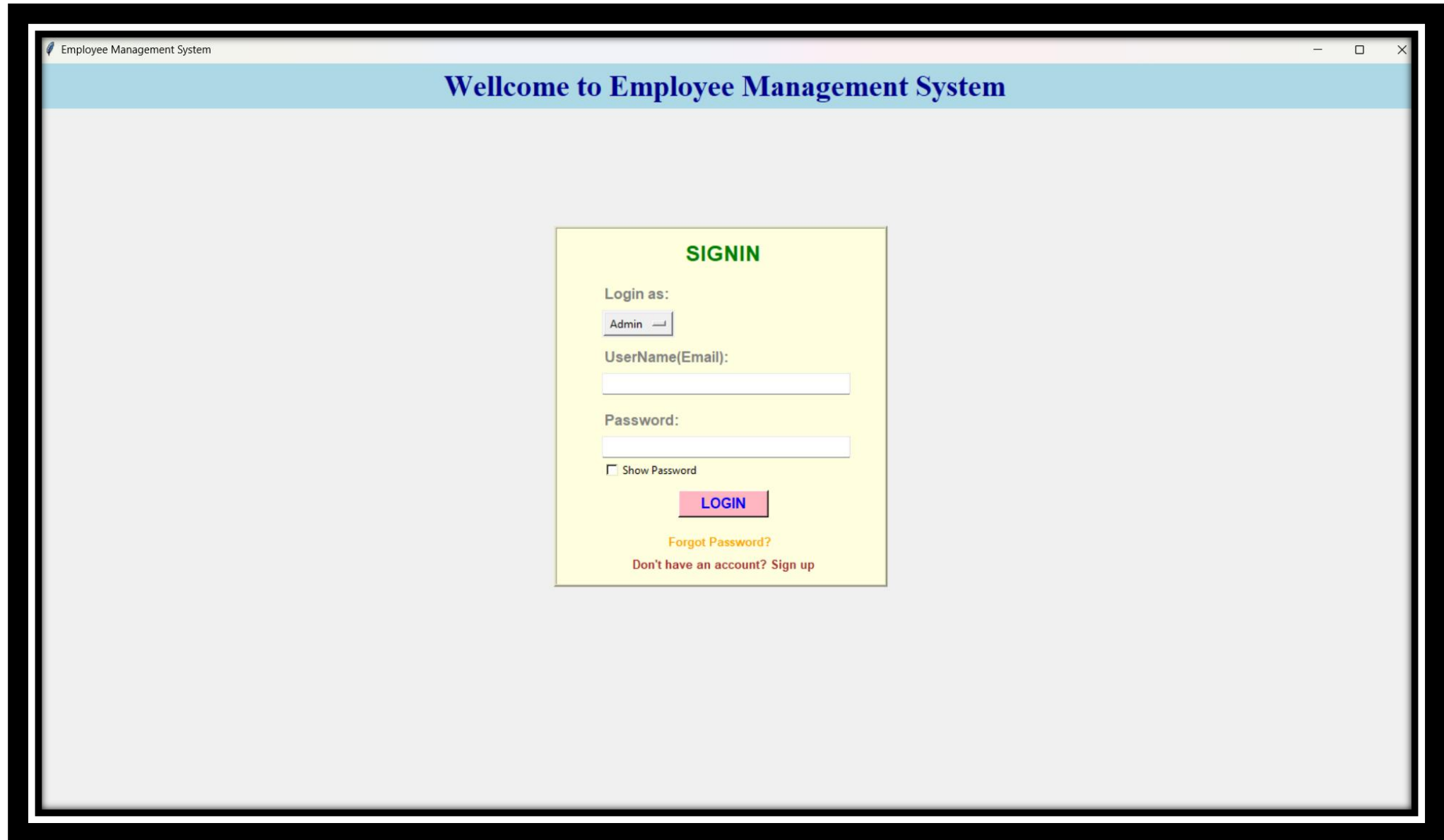
## SQL QUARY FOR CREATE ADMIN TABLE:

```
CREATE TABLE admin (  
    First_Name VARCHAR(50),  
    Last_Name VARCHAR(50),  
    Contact_No VARCHAR(15),  
    Email VARCHAR(100) PRIMARY KEY,  
    ID_Type VARCHAR(50),  
    ID_Proof VARCHAR(50),  
    Password VARCHAR(255)  
);
```

## SQL QUARY FOR CREATE LEAVE\_APPLICATION TABLE:

```
CREATE TABLE leave_applications (  
    Application_ID INT AUTO_INCREMENT PRIMARY KEY,  
    Emp_ID INT,  
    LeaveType VARCHAR(50),  
    StartDate DATE,  
    EndDate DATE,  
    Reason TEXT,  
    Status VARCHAR(20)  
);
```

# RESULT: LOGIN PAGE



The screenshot displays a web browser window titled "Employee Management System". The page features a light blue header with the text "Wellcome to Employee Management System". The main content area is light gray and contains a central yellow box for the login form. The form is titled "SIGNIN" in green. It includes a "Login as:" label, a dropdown menu with "Admin" selected, a "UserName(Email):" label with a text input field, a "Password:" label with a text input field, and a "Show Password" checkbox. A pink "LOGIN" button is positioned below the password field. At the bottom of the form, there are links for "Forgot Password?" and "Don't have an account? Sign up".

Employee Management System

Wellcome to Employee Management System

**SIGNIN**

Login as:

Admin

UserName(Email):

Password:

☐ Show Password

**LOGIN**

[Forgot Password?](#)

[Don't have an account? Sign up](#)

# REGISTRATION PAGE:

Employee Management System

## SIGNUP

<b>First Name:</b>	<b>Last Name:</b>
<input type="text"/>	<input type="text"/>
<b>Contact No:</b>	<b>Email:</b>
<input type="text"/>	<input type="text"/>
<b>Select ID Type:</b>	<b>ID Number:</b>
<input type="text"/>	<input type="text"/>
<b>Password:</b>	<b>Confirm Password:</b>
<input type="text"/>	<input type="text"/>
<input type="checkbox"/> <a href="#">I Agree the Terms &amp; Conditions</a>	<input type="checkbox"/> Show Password

Register Now

LOGIN

# ADMIN PAGE:

Employee Management System

EMPLOYEE MANAGENENT SYSTEM

Log Out

Employee Information

Department:

Name:

Mobaile number:

Go to Manager Portal

Designation:

Email:

Country:

D.O.J:

2024-10-24

D.O.B:

2024-10-24

City:

Gender:

Salary:

Select ID Proof

Married status:

Manager ID:

0

Save

Update

Delete

Reset

Employee Information Table

Search Employee Information

Search By

Search By

Search

Show All

Emp_ID	Manager_ID	Department	Designation	Name	Mobaile number	D.O.J	Email	Country	City	Married Statu	D.O.B	ID Type	ID Pro
20	3	Software Development	Software Developer	Rajib Dey	8767899889	2020-06-19	deyrajibdey@gmail.com	India	Durgapur	Married	1997-12-10	Adhar Card	223366554
21	3	Software Development	Front-End Developer	Sakshi Agarwal	2233221133	2020-05-22	agarwalsakshi@gmail.com	India	Mumbai	Single	1994-09-23	Votar Card	GX060953S
22	3	Software Development	Front-End Developer	Irrfan Khan	2233224455	2015-05-29	irrfankhan@gmail.com	India	Chennai	Married	1991-09-20	Votar Card	OI904826BI
23	3	Software Development	Back-End Developer	Raj Kumar	2233224389	2015-05-29	rajkumar@gmail.com	India	Pune	Married	1994-07-15	Votar Card	HF445038S
24	4	Quality Assurance(QA)	Software Tester	Samyuktha Mennon	2233554389	2015-05-29	amyukthamennon@gmail.com	India	Delli	Married	1994-07-15	Votar Card	DW880276'
25	9	Cybersecurity	Security Engineer	Balraj Sahni	9966886533	2022-10-28	sahnibalraj@gmail.com	India	kolkata	Single	1997-07-22	Votar Card	VH004585S

# EMPLOYEE PAGE:

Employee Dashboard

EMPLOYEE DASHBOARD

Log Out

Employee Information

Employee ID: 20

Manager ID: 3

Department: Software Development

Designation: Software Developer

Email: deyrajibdey@gmail.com

Name: Rajib Dey

Mobile\_Number: 8767899889

D.O.J: 2020-06-19

Country: India

City: Durgapur

Married Status: Married

D.O.B: 1997-12-10

ID Type: Adhar Card

ID Proof: 223366554455

Gender: M

Salary: 60000.00

Password: RAJI1997

Apply for Leave

Leave Type:

Start Date (YYYY-MM-DD): 2024-10-24

End Date (YYYY-MM-DD): 2024-10-24

Reason for Leave:

Apply Leave

Application History

Application_ID	Emp_ID	LeaveType	StartDate	EndDate	Reason	Status
3	20	Casual Leave	2024-09-24	2024-09-30	Death of a family member	Approved
10	20	Sick Leave	2024-09-26	2024-09-26	High Feaver	Rejected
13	20	Sick Leave	2024-09-27	2024-09-27	High Feaver	Pending
14	20	Casual Leave	2024-09-28	2024-09-28	personal	Approved
15	20	Casual Leave	2024-10-31	2024-11-05	DIWALI CELEBRATION	Pending



# MANAGER PAGE:

Manager Dashboard

MANAGER DASHBOARD

Log Out

Manager Information

Manager ID:3

Department:Software Development

Email:john@example.com

D.O.J:2022-01-15

Name:John Doe

Mobile\_Number:1234567890

Country:USA

City:New York

Married Status:Single

D.O.B:1990-05-25

ID Type:Passport

ID Proof:ABC12345

Gender:M

Salary:70000.00

Password:JOHN1990

Employee Information

Emp_ID	Manager_ID	Department	Designation	Name	Mobaile number	D.O.J	Email	Country	City	Married Status	D.O.B	ID Type
20	3	Software Development	Software Developer	Rajib Dey	8767899889	2020-06-19	deyrajibdey@gmail.com	India	Durgapur	Married	1997-12-10	Adhar Card
21	3	Software Development	Front-End Developer	Sakshi Agarwal	2233221133	2020-05-22	agarwalsakshi@gmail.com	India	Mumbai	Single	1994-09-23	Votar Card
22	3	Software Development	Front-End Developer	Irrfan Khan	2233224455	2015-05-29	irrfankhan@gmail.com	India	Chennai	Married	1991-09-20	Votar Card
23	3	Software Development	Back-End Developer	Raj Kumar	2233224389	2015-05-29	rajkumar@gmail.com	India	Pune	Married	1994-07-15	Votar Card

Application History

Application_ID	Emp_ID	LeaveType	StartDate	EndDate	Reason	Status
3	20	Casual Leave	2024-09-24	2024-09-30	Death of a family member	Approved
10	20	Sick Leave	2024-09-26	2024-09-26	High Feaver	Rejected
11	21	Sick Leave	2024-09-26	2024-09-26	High Feaver	Approved
13	20	Sick Leave	2024-09-27	2024-09-27	High Feaver	Pending
14	20	Casual Leave	2024-09-28	2024-09-28	personal	Approved
15	20	Casual Leave	2024-10-31	2024-11-05	DIWALI CELEBRATION	Pending

Approve

Reject

THANK YOU