PROJECT

Employee Payroll Management System

1. Jayashri Shivaji Mankar
2. Jyoti Babu Pagare
3. K PavanKumar
4. Md Faisal
5. Nikhil Ghadge
6. Nilima Subhedar
7. Pallavi Patil
8. Pulluri Pavani
9. Sathish K
10. Uday B C
11. Vasantha Meruva

FUNCTIONAL SPECIFICATION

Project Code:	
Project Name:	**Employee Payroll Management System**

TABLE OF CONTENTS

1. Introduction

Employee Payroll Management System is specially design to manage overall salary related accounting and reduce overhead transactions. It is easy to access and useful for the businesses for the analyzing employee's attendance and performance for increment and promotion.

The main concept we cover in our system is manage employee detail, manage leave, Generate monthly salary and Deduction. Admin is a responsible for operate whole system. Admin has rights to Add new Employee, each employee has a unique username and password to login in to system. After login in to system all employee can apply for a leave with valid reason. The admin can check leave and he can approve and reject employee leave. According to approve leave the leave deduction will be deducted from salary.

The Employee Payroll Management System is a web application made in Angular using Spring Boot with the database. The main aim of the project is to pay the employees perfectly without any delays.

2. System Overview

The "Employee Payroll Management System" should support basic functionalities.

- Admin (A)
- Employee (E)

2.1 Authentication & Authorization

2.1.1 Authentication:

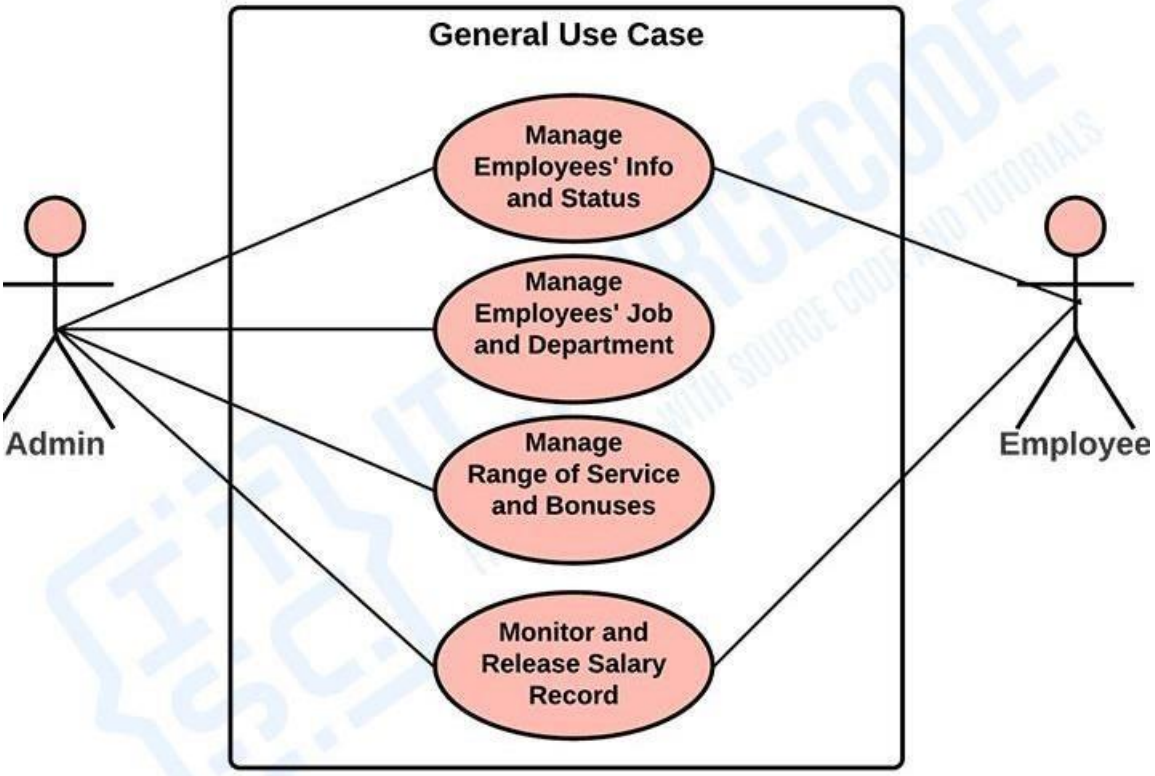
Any end-user should be authenticated using a unique Email ID and Password.

2.1.2 Authorization:

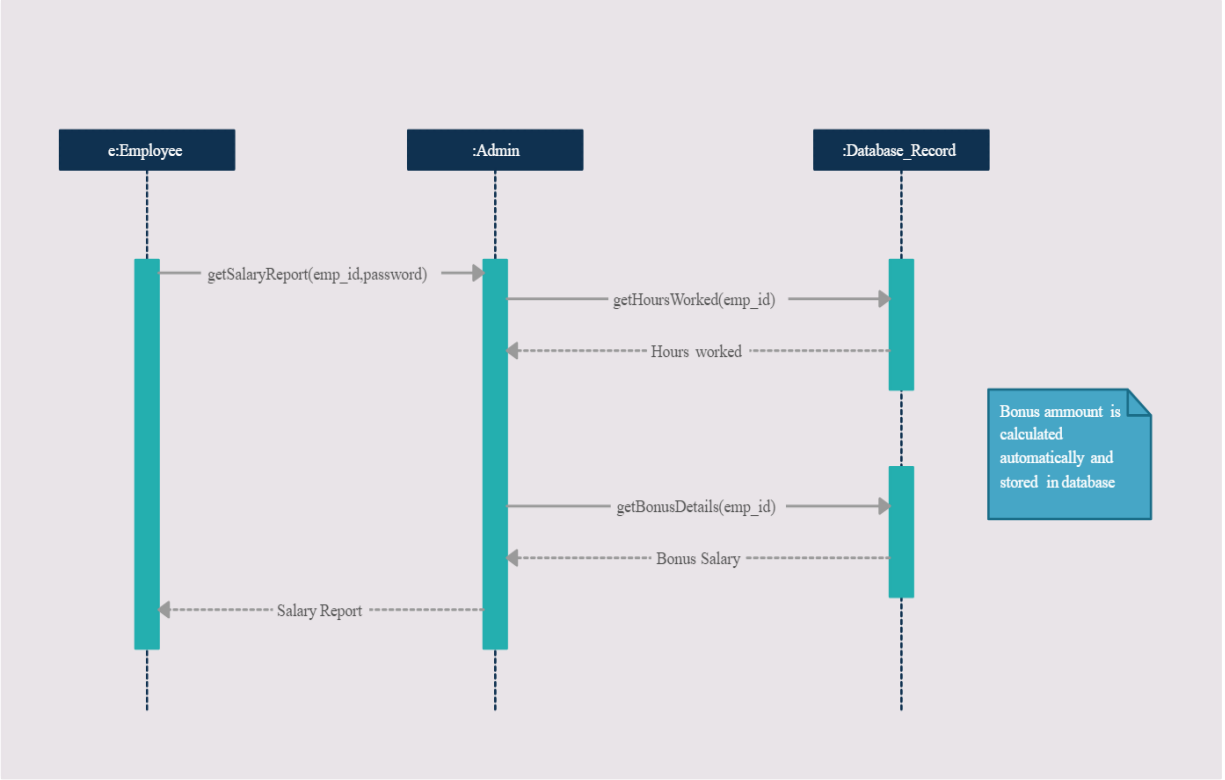
The operations supported and allowed would be based on the user type. For example, Administrator has the rights to add employee information and view employee details. He can also view the employee attendance and leaves details.

Whereas employee can view (Attendance, Time sheet, Salary) and also can update his/her (Profile, Attendance, Leaves).

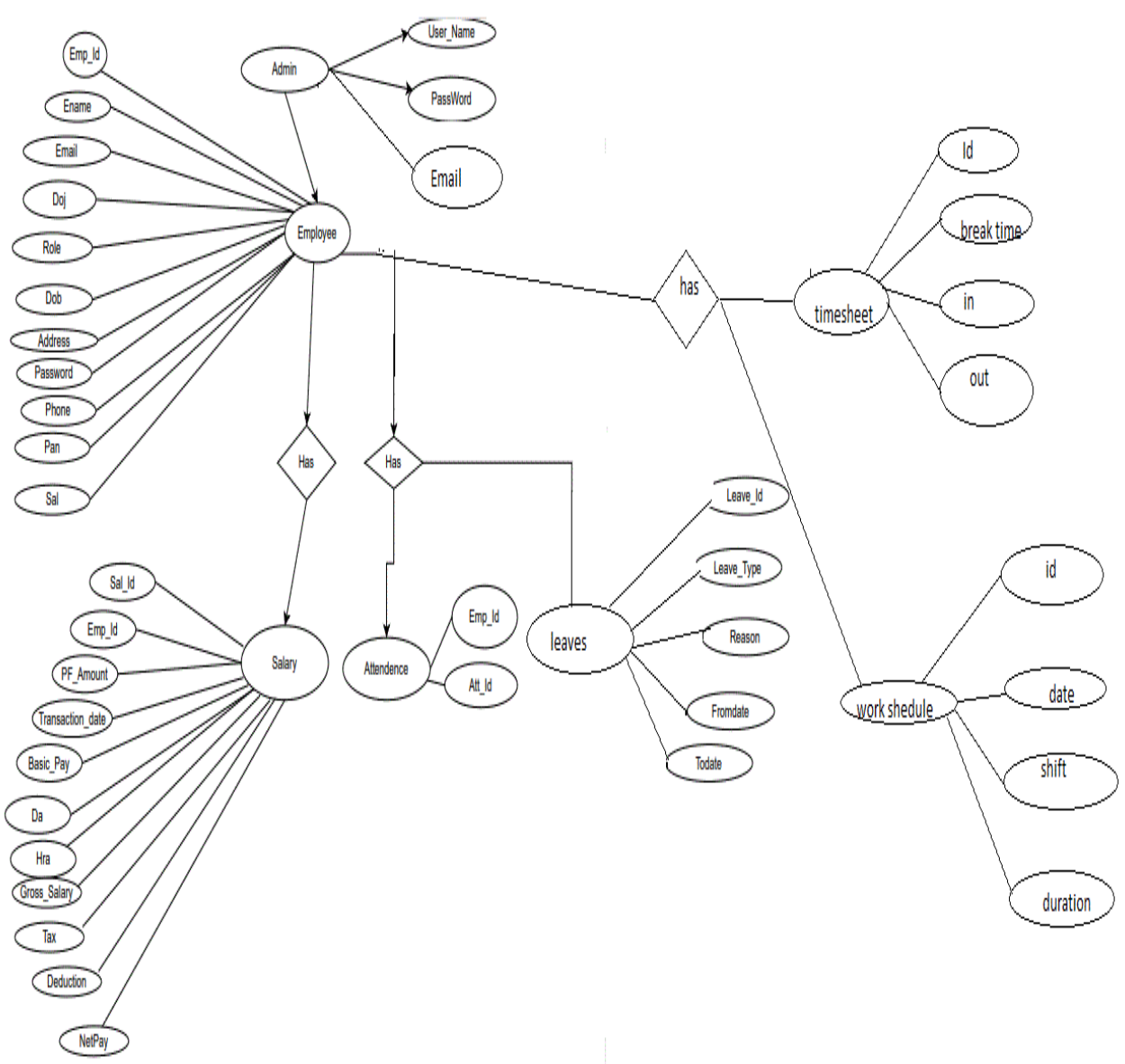
2.2 Functional Flow:



2.2.1 Sequence Diagram:



ER Diagram:


Emp_Id
Ename
Email
Doj
Role
Dob
Address
Password
Phone
Pan
Sal
Admin
User_Name
PassWord
Email
Employee
has
timesheet
Id
break time
in
out
Has
Has
Sal_Id
Emp_Id
PF_Amount
Transaction_date
Basic_Pay
Da
Hra
Gross_Salary
Tax
Deduction
NetPay
Salary
Attendence
Emp_Id
Att_Id
leaves
Leave_Id
Leave_Type
Reason
Fromdate
Todate
work shedule
id
date
shift
duration

2.3 Environment:

The system will be developed on any Windows OS machine using J2EE, Hibernate and Spring.

- Angular
- Server – Apache Tomcat 8
- Database – MySQL
- JRE 8
- Spring Tool Suite
- Postman & JUnit

3. Sub-system Details

The Employee Payroll Management System is defined, wherein all users need to login successfully before performing any of their respective operations.

Find below (section 3.1 & 3.2) tables that provides functionality descriptions for each type of user / sub-system. Against each requirement, indicative data is listed in column 'Data to include'. Further, suggested to add/modify more details wherever required with an approval from Administrator.

3.1 Administrator

The administrator as a user is defined to perform below listed operations after successful login.

S.No	Objects	Operations	Data to include	Remarks
1.	Employee	Add View Update Delete		

3.2 Employee

The employee as a user is defined to perform below listed operations after successful login.

S.No	Objects	Operations	Data to include	Remarks
1.	Employee	Register	Emailid password	
2.	Employee	View Update		

3.3 Login | Logout

[Web Application - J2EE, Hibernate, Spring]

- Go to Registration screen when you click on Register link.

- Go to Success screen when you login successfully after entering valid username &password fetched from the database.

- Redirect back to same login screen if username & password are not matching.

- Implement Session tracking for all logged in users before allowing access toapplication features. Anonymous users should be checked, unless explicitly mentioned.

4. REST APIs to be Built

Create following REST resources which are required in the application,

4.1 Creating User Entity:

Create Spring Boot with Microservices Application with Spring Data JPA.

Technology stack:

- Spring Boot
- Spring Data JPA
- Spring REST

4.2 Creating Employee Entity:

Build a RESTful resource for Employee manipulation, where CRUD operations to be carried out. Here will have multiple layers into the application:

1. Create an Entity: Employee

2. Create a Employee interface and will make use of Spring Data JPA

a) Will have getAllEmployees method.

b) Add the Employee details method.

c) Will have getEmpById method.

d) Will have deleteEmpById method.

3. Create a EmployeeService class and will expose all these services.

4. Finally, create a EmployeeController will have the following Uri's:

URI	METHODS	DESCRIPTION	FORMAT
/employees	GET	Get all the employees	JSON

/employees	POST	Add the employees details	JSON
/employees/{id}	GET	Give a single employee description searched based on employee by id	JSON
/employees/{id}	DELETE	Delete a employee based on employee by id	JSON
/employees/{id}	PUT	Update a employee based on employee by id	JSON

4.3 Creating Attendance Entity:

Build a RESTful resource for Attendance manipulation, where CRUD operations to be carried out. Here will have multiple layers into the application:

1. Create an Entity: Attendance

2. Create a Attendance interface and will make use of Spring Data JPA.

a) Will have getAllAttendances method.

b) Add the Attendance details method.

c) Will have getAttendanceById method.

d) Will have getAttendanceByEmployeeId method.

3. Create a AttendanceService class and will expose all these services.

4. Finally, create a AttendanceController will have the following Uri's:

URI	METHODS	DESCRIPTION	FORMAT

/attendances	GET	Get all the attendances	JSON
/attendances	POST	Add all the attendances	JSON
/attendances/{id}	GET	Give a single employee description searched based on attendance by id	JSON
/attendance/{id}	GET	Give a single employee description searched based on employee by id	JSON

4.4 Creating Leavedetail entity:

Build a RESTful resource for Leaves manipulation, where CRUD operations to be carried out. Here will have multiple layers into the application:

1. Create an Entity: Leaves

2. Create a Leaves interface and will make use of Spring Data JPA.

a) Will have getLeavesById method.

b) Add the Leaves details method.

c) Will have getAllLeaves method.

d) Will have deleteLeaveById method.

e) Will have fetchLeavesByEmployeeId method.

3. Create a LeavesService class and will expose all theseservices.

4. Finally, create a LeavesController will have the followingUri's:

URI	METHODS	DESCRIPTION	FORMAT
/leaves	GET	Get all the leaves	JSON
/leaves	POST	Add all the leaves	JSON
/leaves/{id}	GET	Give a single employee description searched based on leaves by id	JSON
/leave/{id}	GET	Give a single employee description searched based on employee by id	JSON
/leaves/{id}	DELETE	Delete a employee based on leaves by id	JSON
/leaves/{id}	PUT	Update a employee based on leaves by id	JSON

4.5 Creating Salary Entity:

Build a RESTful resource for Salary manipulation, where CRUD operations to be carried out. Here will have multiple layers into the application:

1. Create an Entity: Salary

2. Create a Salary interface and will make use of Spring Data JPA.

a) Will have getAllSalaries method.

b) Add the Salary details method.

c) Will have getSalById method.

d) Will have deleteSalById method.

e) Will have fetchSalByEmployeeId method.

3. Create a SalaryService class and will expose all these services.

4. Finally, create a SalaryController will have the followingUri's:

URI	METHODS	DESCRIPTION	FORMAT
/salaries	GET	Get all the salaries	JSON
/salaries	POST	Add all the salaries	JSON
/salaries/{eId}	GET	Give a single employee description searched based on employee by id	JSON
/salariesid/{id}	GET	Give a single employee description searched based on salary by id	JSON
/salaries/{id}	DELETE	Delete a employee based on salary by id	JSON
/salaries/{id}	PUT	Update a employee based on salary by id	JSON

5. Assumptions

- User Interface: The type of client interface (front-end) to be supported - Angular based.
- The administrator can add and remove employees into the database.
- You must not allow user to add same employee twice.

6. Expectations

- Participants must create the Class Diagram, Sequence Diagram and ER Diagram.
- Participants must do Unit testing and Functional Testing using POSTMAN tool.
- Integration of Angular and Spring Boot with Microservices should be done.

- The server should be a concurrent server servicing multipleclients.
- Database can be implemented using MYSQL.
- Compilation and Build should be done using STS.
- Source-code and all documents must be maintained (checked-in) in configuration management system (subversion)
- Coding standards (for Java) should be followed.

7. Functional Requirements

7.1 Employee:

This module helps to add the details of the employee like the personal detail and the employee detail.

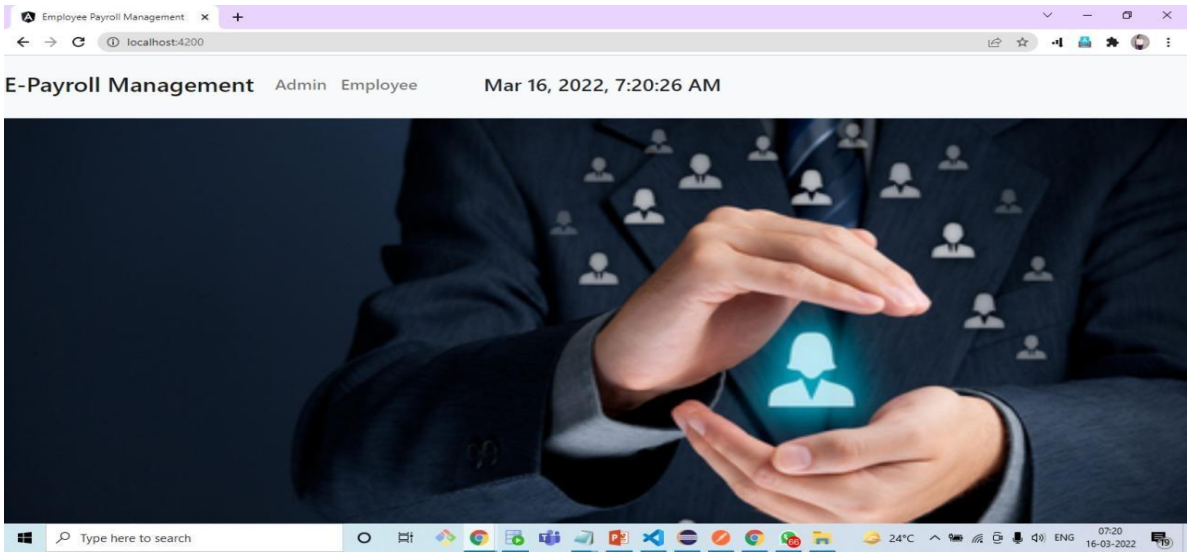
7.2 Attendance:

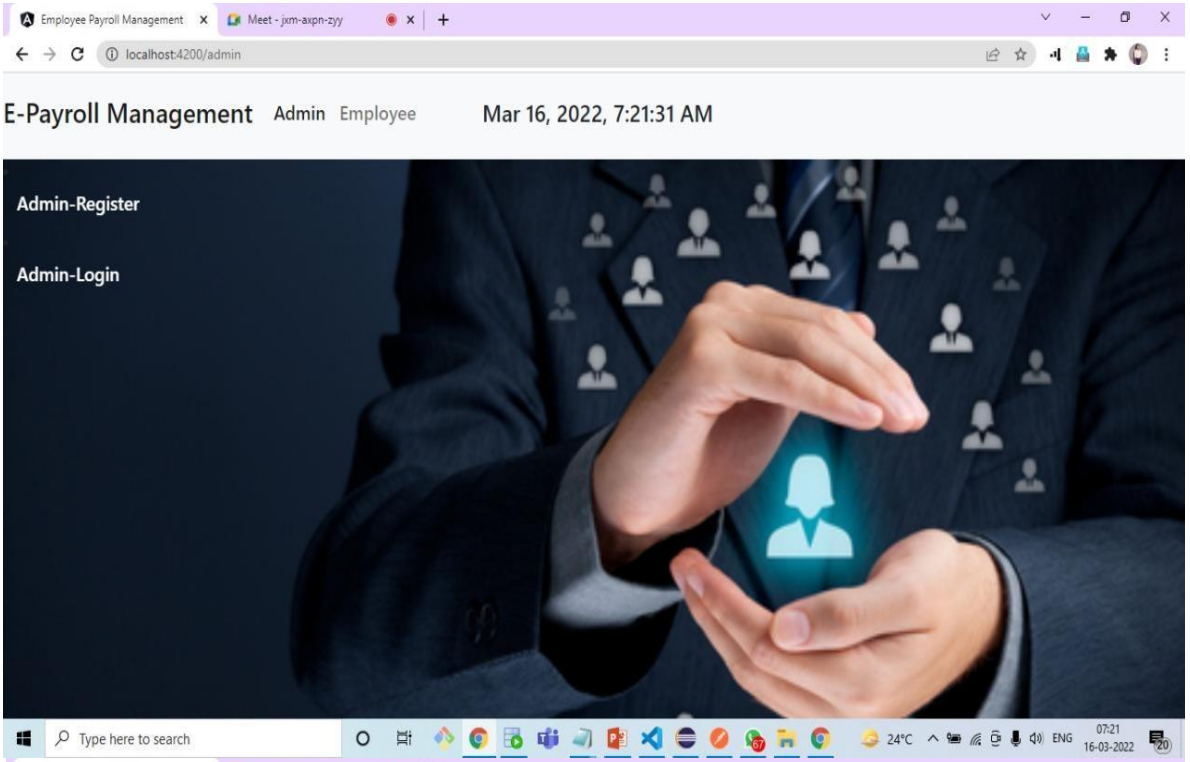
This module helps to different types of leave for different year. It also helps the employee to entire their entry and exit time. Using the attendance module the employee can also check their remaining leaves and also apply for the leave.

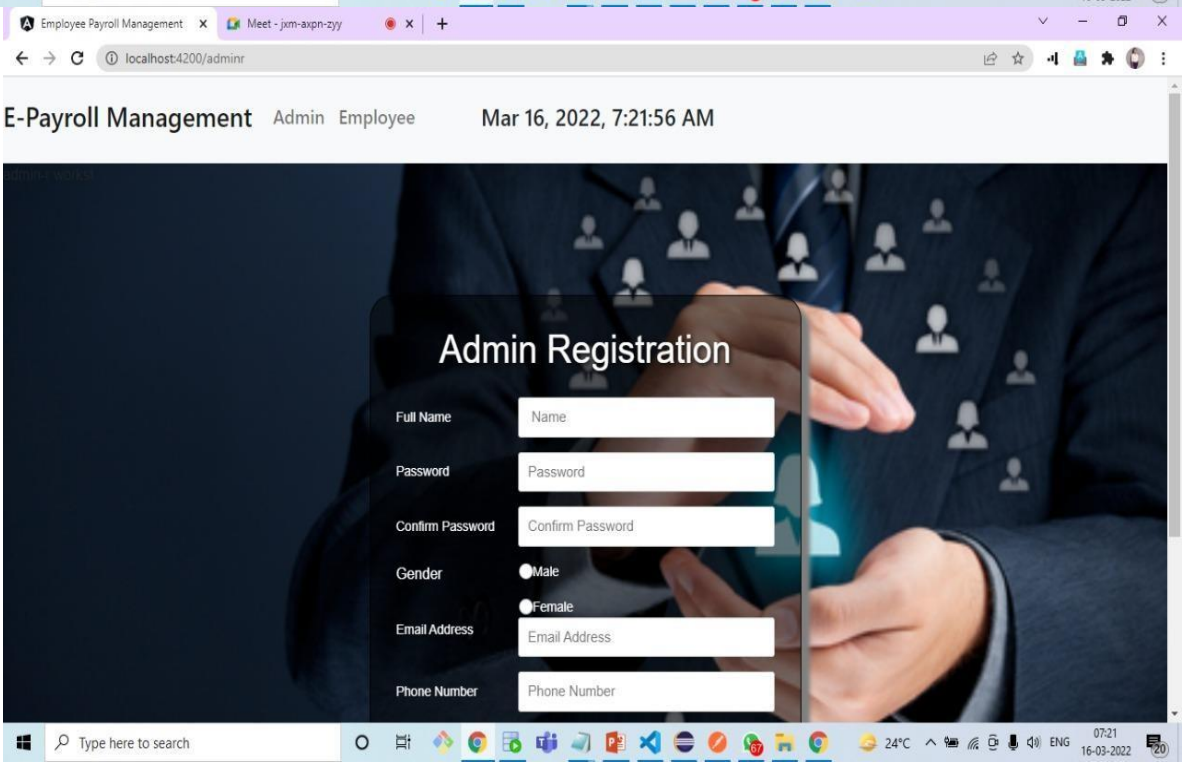
7.3 Salary:

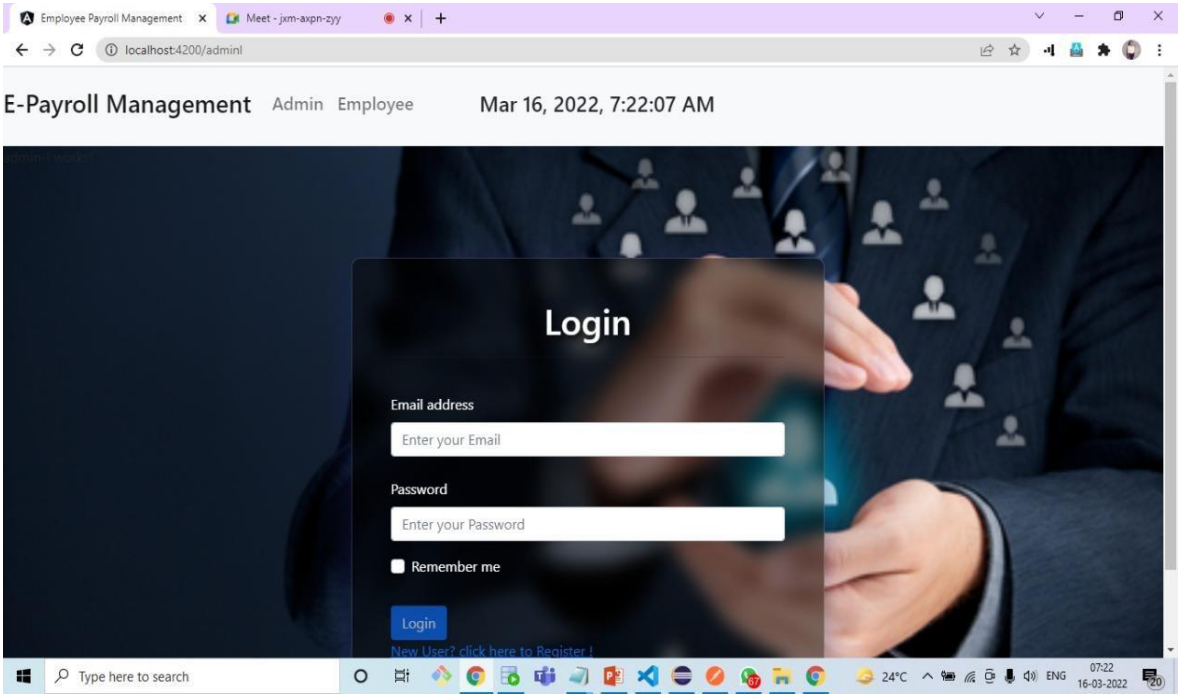
This module helps to calculate the salary by adding allowances and the basic salary and by deducting the deductions based on the leaves and also the PF, ESI. It also helps to generate the employee pay slip.

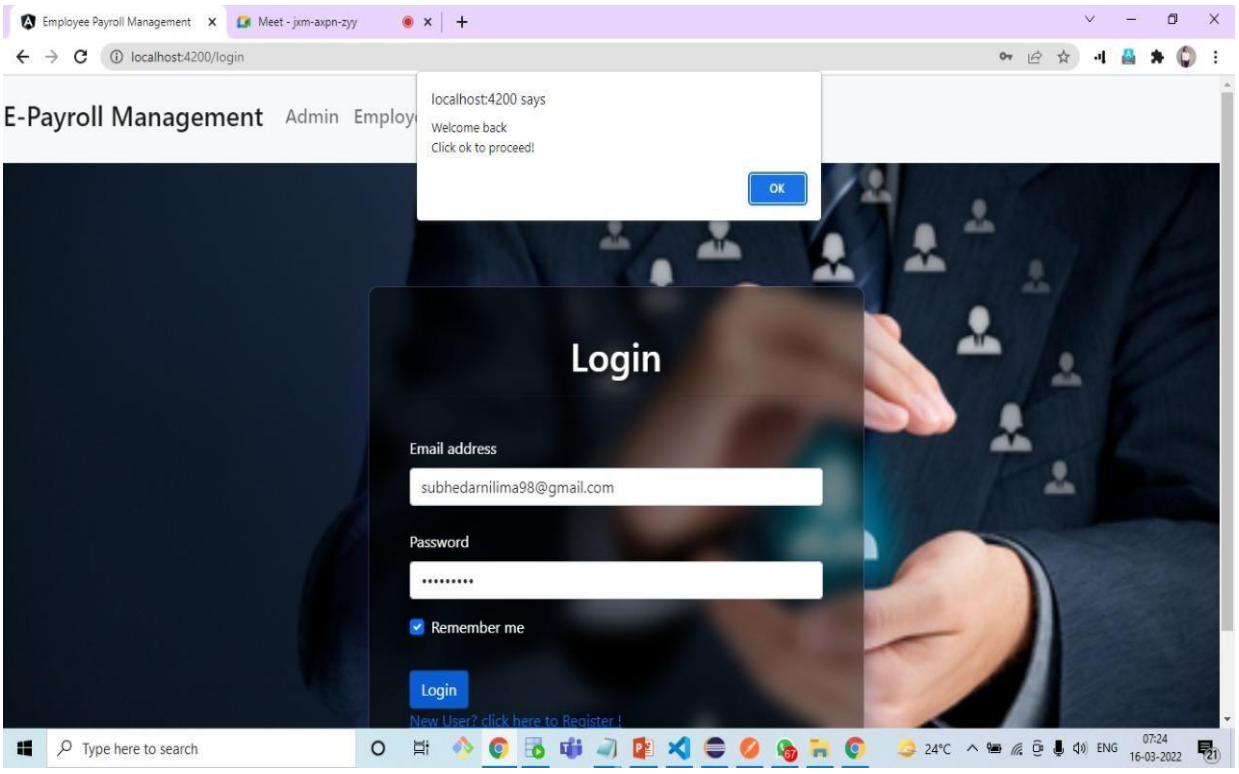
8. Output Pictures:



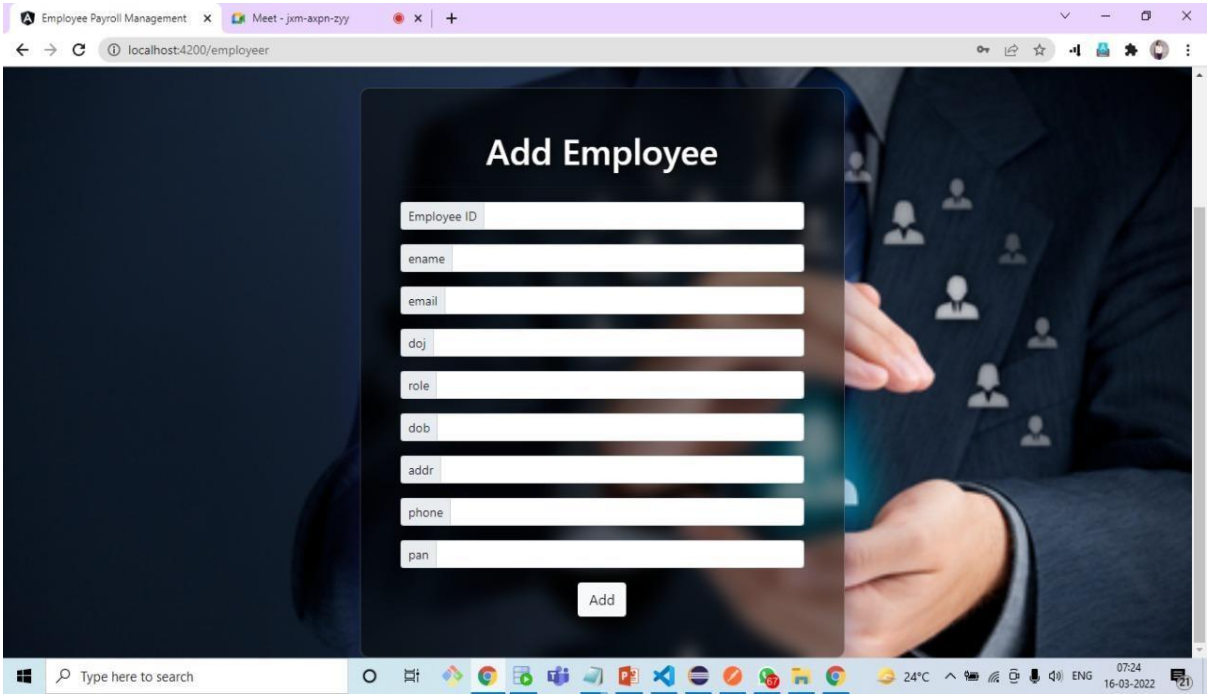

Employee Payroll Management
Meet - jxm-axpn-zyy
localhost:4200/admin
E-Payroll Management
Admin
Employee
Mar 16, 2022, 7:21:31 AM
Admin-Register
Admin-Login
Type here to search
24°C
ENG
07:21
16-03-2022

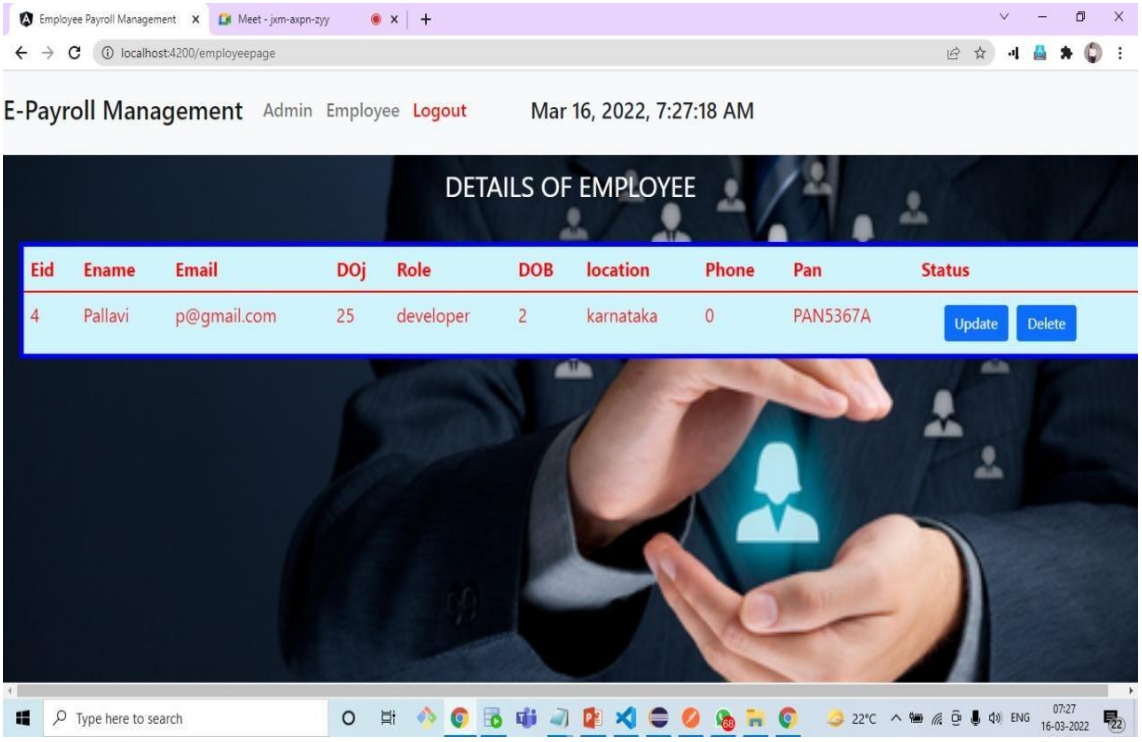

Employee Payroll Management
Meet - jxm-axpn-zyy
localhost:4200/adminr
E-Payroll Management
Admin
Employee
Mar 16, 2022, 7:21:56 AM
Admin Registration
Full Name
Name
Password
Password
Confirm Password
Confirm Password
Gender
Male
Female
Email Address
Email Address
Phone Number
Phone Number
Type here to search
24°C
ENG
07:21
16-03-2022

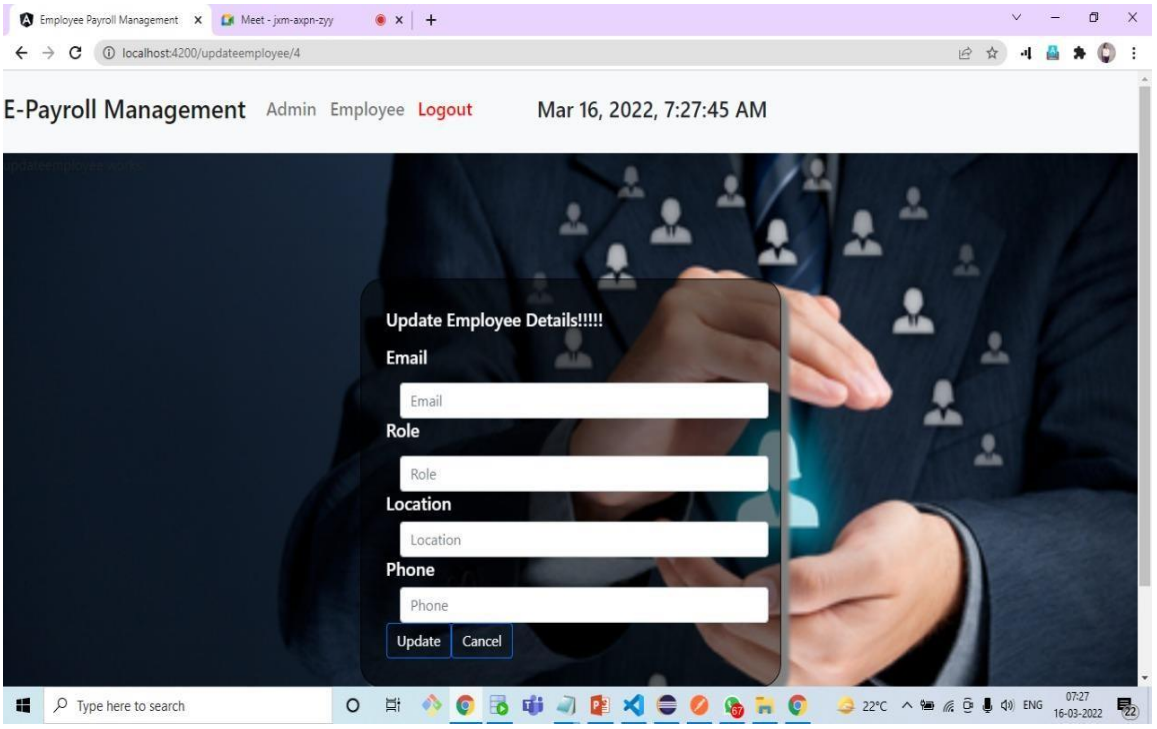

Employee Payroll Management
Meet - jxm-axpn-zyy
localhost:4200/adminl
E-Payroll Management
Admin
Employee
Mar 16, 2022, 7:22:07 AM
Login
Email address
Enter your Email
Password
Enter your Password
Remember me
Login
New User? click here to Register !
Type here to search
24°C
ENG
07:22
16-03-2022

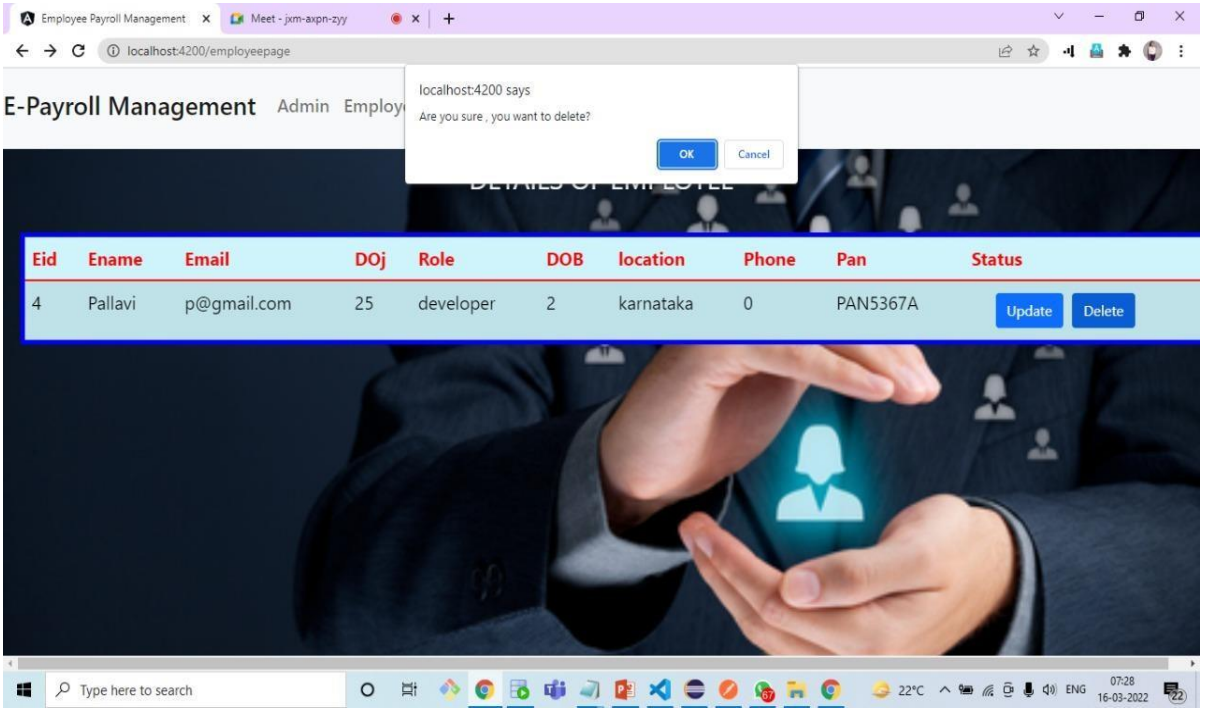

Employee Payroll Management
Meet - jxm-axpn-zyy
localhost:4200/login
E-Payroll Management
Admin
localhost:4200 says
Welcome back
Click ok to proceed!
OK
Login
Email address
subhedarnilima98@gmail.com
Password
Remember me
Login
New User? click here to Register !
Type here to search
24°C
ENG
07:24
16-03-2022

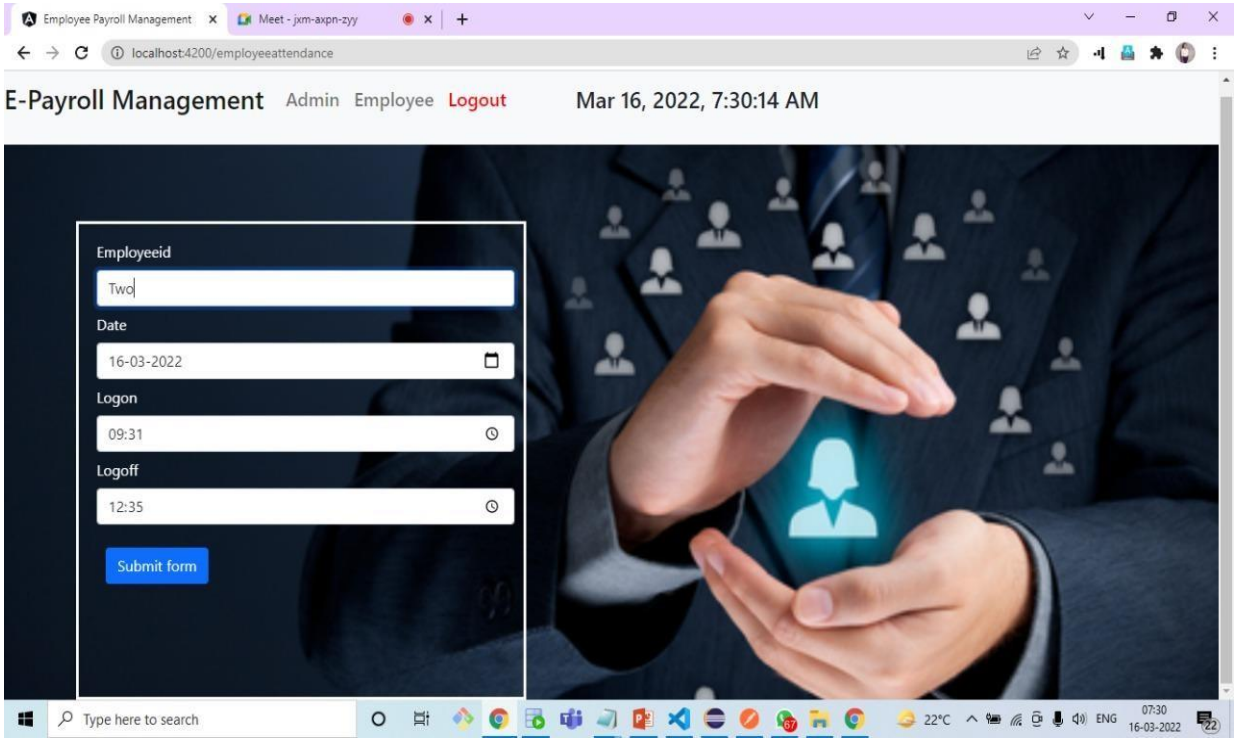

Employee Payroll Management
Meet - jxm-axpn-zyy
localhost:4200/payhome
E-Payroll Management
Admin
Employee
Logout
Mar 16, 2022, 7:24:18 AM
Attendance
Leave
Salary
Reports
Employee
ADD-Employee
Type here to search
24°C
ENG
07:24
16-03-2022


Employee Payroll Management
Meet - jxm-axpn-zyy
localhost:4200/employeer
Add Employee
Employee ID
ename
email
doj
role
dob
addr
phone
pan
Add
Type here to search
24°C
ENG
07:24
16-03-2022

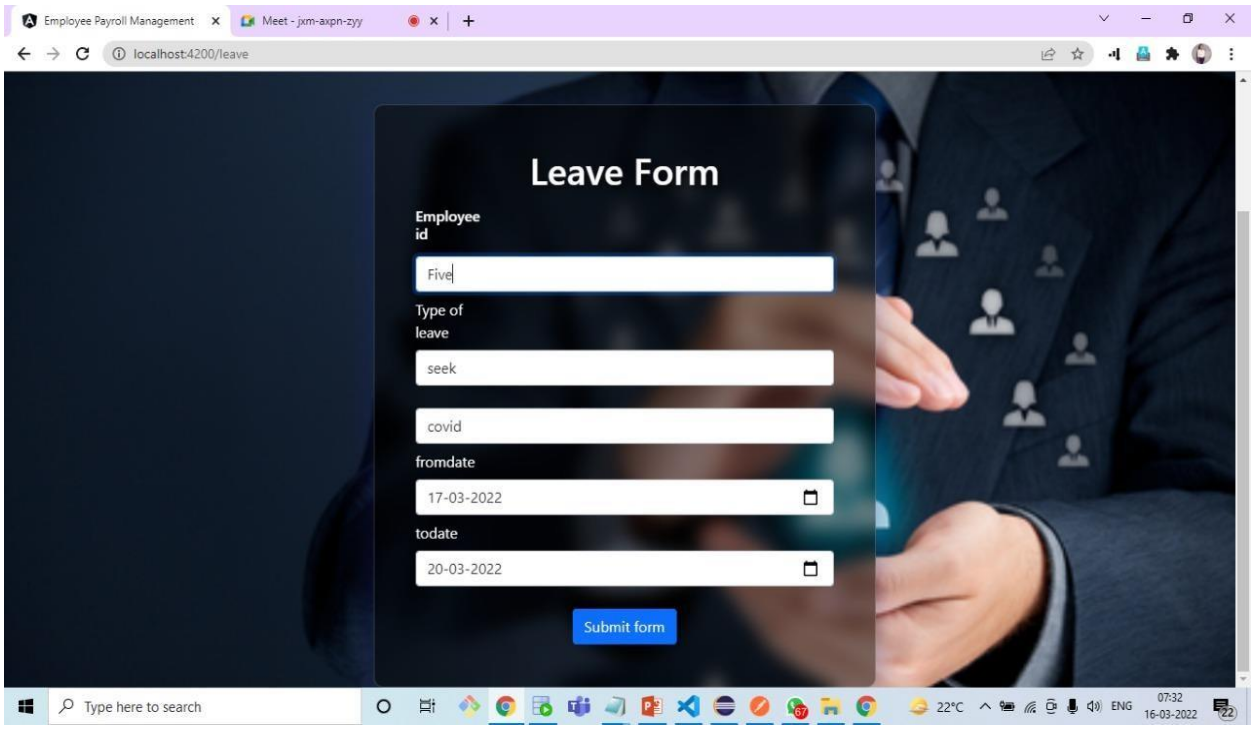

Employee Payroll Management
Meet - jxm-axpn-zyy
localhost:4200/employeepage
E-Payroll Management
Admin
Employee
Logout
Mar 16, 2022, 7:27:18 AM
DETAILS OF EMPLOYEE
Eid
Ename
Email
DOj
Role
DOB
location
Phone
Pan
Status
4
Pallavi
p@gmail.com
25
developer
2
karnataka
0
PAN5367A
Update
Delete
Type here to search
22°C
ENG
07:27
16-03-2022

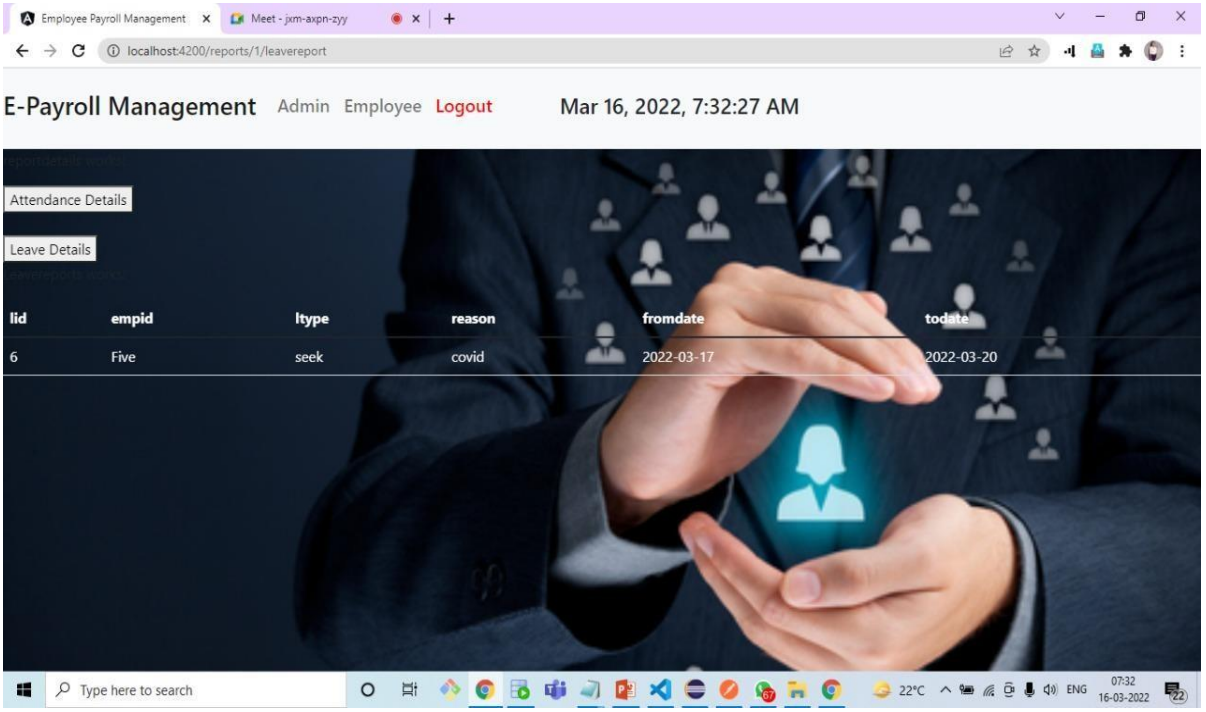

Employee Payroll Management
Meet - jxm-axpn-zyy
localhost:4200/updateemployee/4
E-Payroll Management
Admin
Employee
Logout
Mar 16, 2022, 7:27:45 AM
Update Employee Details!!!!!
Email
Email
Role
Role
Location
Location
Phone
Phone
Update
Cancel
Type here to search
22°C
ENG
07:27
16-03-2022

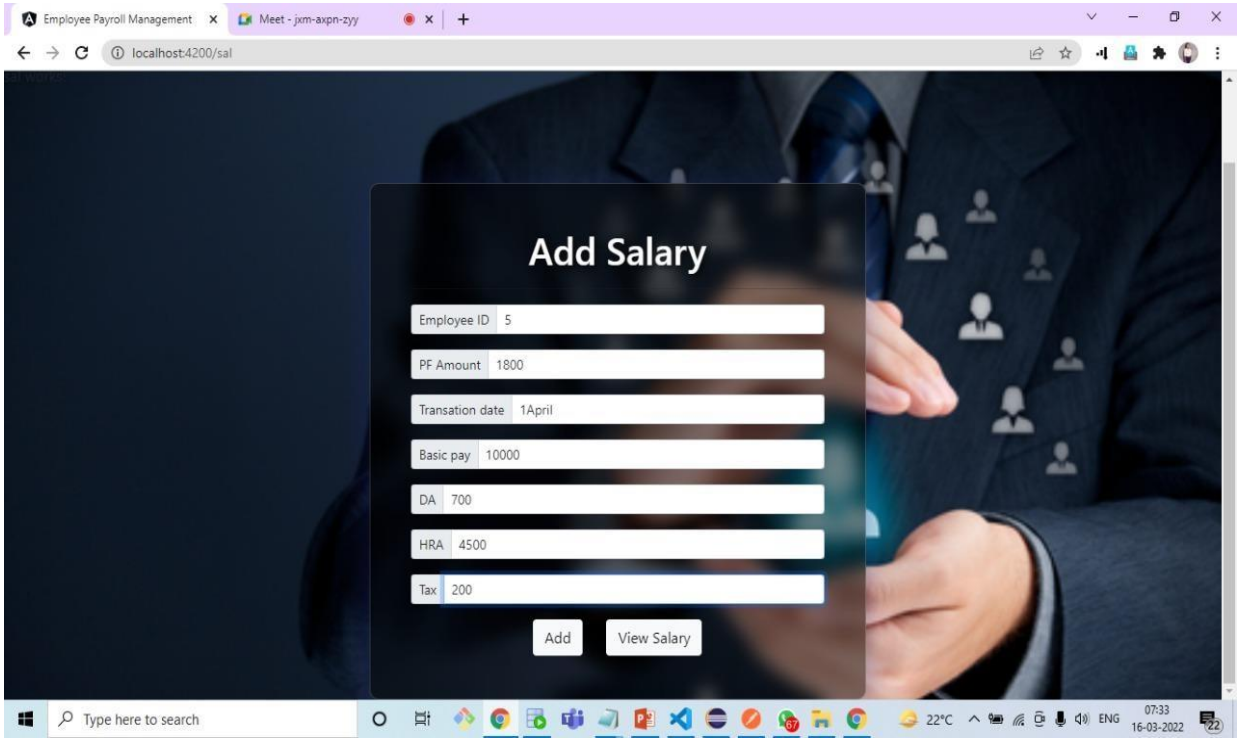

Employee Payroll Management
Meet - jxm-axpn-zyy
localhost:4200/employeepage
E-Payroll Management
Admin
localhost:4200 says
Are you sure , you want to delete?
OK
Cancel
Eid
Ename
Email
DOj
Role
DOB
location
Phone
Pan
Status
4
Pallavi
p@gmail.com
25
developer
2
karnataka
0
PAN5367A
Update
Delete
Type here to search
22°C
ENG
07:28
16-03-2022

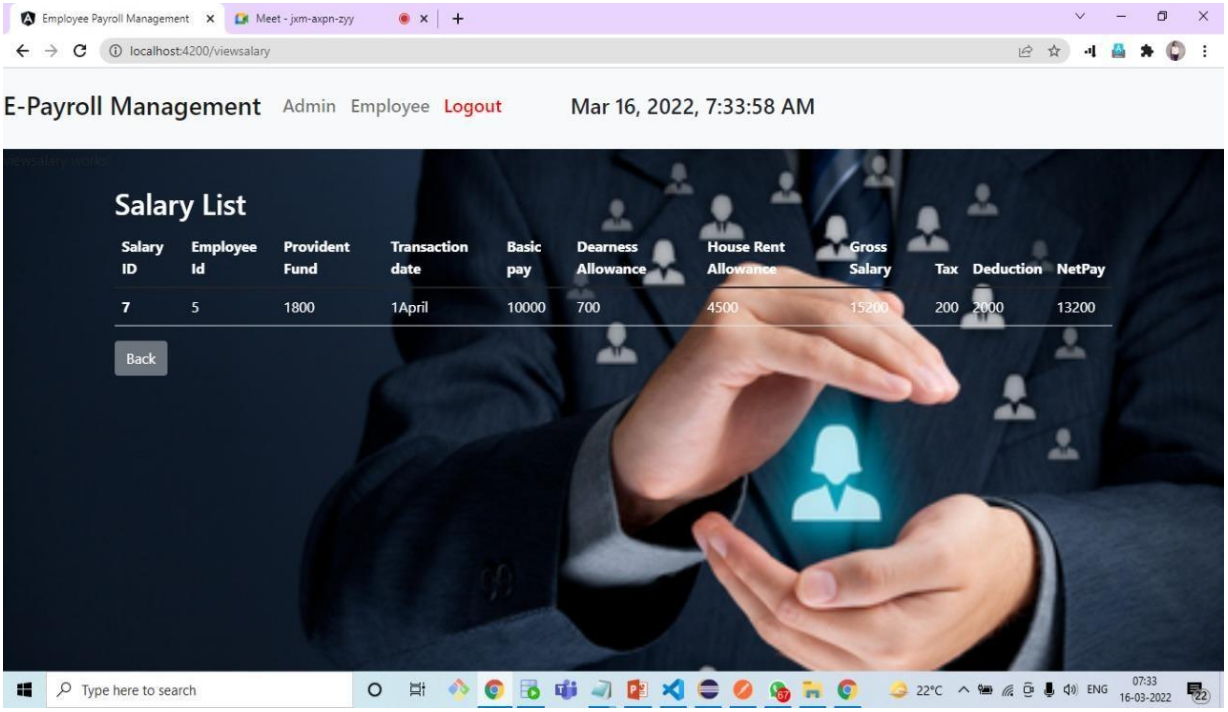

Employee Payroll Management
Meet - jxm-axpn-zyy
localhost:4200/employeeattendance
E-Payroll Management
Admin
Employee
Logout
Mar 16, 2022, 7:30:14 AM
Employeeid
Two
Date
16-03-2022
Logon
09:31
Logoff
12:35
Submit form
Type here to search
22°C
ENG
07:30
16-03-2022


Employee Payroll Management
Meet - jxm-axpn-zyy
localhost:4200/reports/1/attendancereport
E-Payroll Management
Admin
Employee
Logout
Mar 16, 2022, 7:30:42 AM
Attendance Details
Leave Details
aid
employeeid
adate
logon
logoff
5
Two
2022-03-16
09:31
12:35
Type here to search
22°C
ENG
07:30
16-03-2022


Employee Payroll Management
Meet - jxm-axpn-zyy
localhost:4200/leave
Leave Form
Employee id
Five
Type of leave
seek
covid
fromdate
17-03-2022
todate
20-03-2022
Submit form
Type here to search
22°C
ENG
07:32
16-03-2022


Employee Payroll Management
Meet - jxm-axpn-zyy
localhost:4200/reports/1/leavereport
E-Payroll Management
Admin
Employee
Logout
Mar 16, 2022, 7:32:27 AM
Attendance Details
Leave Details
lid
empid
ltype
reason
fromdate
todate
6
Five
seek
covid
2022-03-17
2022-03-20
Type here to search
22°C
ENG
07:32
16-03-2022


Employee Payroll Management
Meet - jxm-axpn-zyy
localhost:4200/sal
Add Salary
Employee ID
5
PF Amount
1800
Transation date
1April
Basic pay
10000
DA
700
HRA
4500
Tax
200
Add
View Salary
Type here to search
22°C
ENG
07:33
16-03-2022



9. References:

[1] Sonal Kasliwal, H.D.Gadade and Sushma Kotkar (2016), Employee Monitoring System , International Journal of Innovative Research in Advanced Engineering (IJIRAE) SSN: 2349-2763, No03, Vol 3, pp. 1-4

[2] Aparna Chandran (2013), Employee Monitoring System, International Journal of Computer Science & Engineering Technology (IJCSET) ISSN: 2229-3345 Vol. 4 No. 04, page 451-452

[3] Shermin Sultana1, Ishrat Jahan Mouri and Asma Enayet1 (2015), A Smart Time And Attendance Tracking System International Journal Of Computer Science, Engineering And Information Technology (Ijcseit), Vol. 5, No.1,