

UNIVERSITY OF MUMBAI

PROJECT REPORT ON

**Employee Management
System**

SUBMITTED

BY

VIKRAM SURAJ PAL



DTSS COLLEGE OF COMMERCE

**UNDER THE GUIDANCE OF
Prof. UTKARSHA.BHAT**

M. Sc. IT [2024-2025]

CERTIFICATE

DEPARTMENT OF IT-DATA SCIENCE

Ref No:

Date:

CERTIFICATE

This is to certify that the project entitled EMPLOYEE MANAGEMENT SYSTEM Undertaken at the D.T.S.S. COLLEGE OF COMMERC & SCIENCE.

By

VIKRAM SURAJ PAL

In partial full filament of M.Sc. IT (Semester III) Examination had not been submitted for any other examination and does not form part of any other course undergone by the candidate.

It is further certified that she has completed all required phases of project.

Signature of Internal Guide

Signature of External

Signature of the co-ordinate

Performa for the Approval Project Proposal

PNR NO: 2020016401575657

ROLL NO: 04

1. Name of the Student

❖ **Vikram Suraj Pal**

2. Title of the Project

❖ **Employee Management System**

3. Name of the Guide.

❖ **Prof. Utkarsha. Bhat**

4. Teaching experience of the Guide.

❖ **15 Years**

5. Is this your first Submission?

Yes, _____ NO _____

Signature of the student

Signature of Guide

Date: -

Date: -

Signature of the co-ordinate

ABSTRACT

The actual problem is to maintain different database for an organization whose main purpose is to issue pay-slips for their employees every month working in various departments of the organization, and maintain details of all the departments, employees with various grades, their designations and address details.

In the manual system it is difficult to maintain data and generating different reports according to requesting transaction. In the present system it is becoming difficult to issue pay-slip for the entire employee every month by manually going through the various record of the organization. i.e. the manger has to go through all the records of the organization of various departments of the and find out the employee working in a particular department and go through his grade, and he have to check the employee leaves of that month, his earnings and his deductions along with his pf and all other deduction including his IT and savings. So, to perform all these activities it is becoming difficult the admin/manager every month.

Hence in order to overcome the difficulties of the organization the present system is automated to perform all the activities of the organization.

ACKNOWLEDGEMENT

In performing this project “**EMPLOYEE MANAGEMENT SYSTEM**”, I had to take the help and guideline of some respected persons, who deserve our greatest gratitude.

The completion of this assignment gives me much pleasure. We would like to show our gratitude to **Prof. Utkarsha. Bhat** for giving us a good guideline for project throughout numerous consultations.

We would also like to expand our deepest gratitude to all those who have directly and indirectly guided us in writing this project report.

Many people, especially our classmates and friends itself, have made valuable comments and suggestion on this proposal which gave an inspiration to improve my project.

Here I thank all the people for their help directly and indirectly to complete this project.

Table of Contents

| Sr. No | Title | Page No |
|---------------|----------------------------------|----------------|
| 01 | Introduction | 1 |
| | 1.1 Abstract | 2 |
| | 1.2 Objectives | 3 |
| | 1.3 Purpose | 4 |
| | 1.4 Scope | 5 |
| 02 | Modules | 6 |
| | 2.1 Admin | 7 |
| | 2.2 Advantage | 7 |
| 03 | Survey of Technology | 8 |
| 04 | Feasibility Study | 9 |
| 05 | Project Category | 10 |
| | 5.1 Web Based Application | 10 |
| 06 | System Design | 11 |
| | 6.1 ER Diagram | 11 |
| | 6.2 Activity Diagram | 12 |
| | 6.3 Class Diagram | 13 |
| | 6.4 Use Case Diagram | 14 |
| | 6.5 Sequence Diagram | 15 |
| | 6.6 Data Flow Diagram | 16 |
| | 6.7 Component Diagram | 17 |
| | 6.8 Deployment Diagram | 18 |
| 07 | Source Code | 19 |
| 08 | Screen Short | 29 |
| 09 | Features of the Project | 33 |
| 10 | Future Scope Of this Work | 34 |
| 11 | Conclusion | 35 |
| 12 | Bibliography | 36 |

EMPLOYEE MANAGEMENT SYSTEM

INTRODUCTION

The “Employee Management System” has been developed to override the problems prevailing in the practicing manual system. This software is supported to eliminate and, in some cases, reduce the hardship faced by this existing system. Moreover, this system is designed for the particular need of the company to carry out operations in a smooth and effective manner.

Every organization, whether big or small, has challenges to overcome and managing the information's of Employee, Employee, Appraisals, Payments, and Working Points. Every Employee Management System has different Employee needs; therefore, we design exclusive employee management systems that are adapted to your managerial requirements. This is designed to assist in strategic planning, and will help you ensure that your organization is equipped with the right level of information and details for your future goals. Also, for those busy executive who are always on the go, our systems come with remote access features, which will allow you to manage your workforce anytime, at all times. These systems will ultimately allow you to better manage resources.

The application is reduced as much as possible to avoid errors while entering the data. It also provides error message while entering invalid data. No formal knowledge is needed for the user to use this system. Thus by this all it proves it is user-friendly. Employee Management System, as described above, can lead to error free, secure, reliable and fast management system. It can assist the user to concentrate on their other activities rather to concentrate on the record keeping. Thus, it will help organization in better utilization pf resources.

1.1 ABSTRACT

The purpose of Employee Management System is to automate the existing manual system by the help of computerized equipment and full-fledged computer software, fulfilling their requirements, so that their valuable data/information can be stored for a longer period with easy accessing and manipulation of the same. The required software and hardware are easily available and easy to work with.

Employee Management System, as described above, can lead to error free, secure, reliable and fast management system. It can assist the user to concentrate in their other activities rather than concentrate on the record keeping. Thus, it will help organization in better utilization of resources. The organization can maintain computerized records without redundant entries. That means that one need not be distracted by information that is not relevant, while being able to reach the information.

The aim is to automate its existing manual system by the help of computerized equipment's and full-fledged computer software, fulfilled their requirement, so that their valuable data/information can be stored for a longer period with easy accessing and manipulation of the same. Basically, the project describes how to manage for good performance and better services for the clients.

The required software is easily available and easy to work with. This web application can maintain and view computerized records without getting redundant entries. The project describes how to manage user data for good performance and provide better services for the client.

1.2 OBJECTIVE

The main objective of the project on Employee Management System is to manage the details of Employee. Employee, Salary, Appraisals, Working Points. It manages all the information about Employee, Payments, Working Points, and Employee. The project is totally built at administrative end and thus only the administrator is guaranteed the access. The purpose of the project is to projectis to build an application program to reduce the manual work for managing the Employee. Employee, Payments, Salary, it tracks all the details about the Salary, Appraisals, Working Points.

To assist ease the works of the Retails Outlet of Any Type of Companies, in particular, Employee Management System is being developed. This would comprise the features that can be operated easily. Employee Management System would take care of the Day-to-day Attendance of AllEmployee. It covers activities from keeping the details of day to day In, Out, OD etc.

The master and transaction activities are divided in modules so that the activities can be operated easily. The regular backup to the data can be taken and the backup data can be restored effectively. So, the Employee Management System assists the employees of Company in each and every aspect of In and Out. The Employee Management System is customized software and developed according tothe needs of Any Type Company.

1.3 PURPOSE

The purpose of this document is to describe the functionality and specifications of the design of a web application for Managing Employees and their Employee. The expected audiences of this document are the developers and the admin of the web application. Now with the help of this system the admin has the information on his finger tips and can easily prepare a good record based on their requirements.

Finally, we can say that this system will not only automate the process but save the valuable time of the manager or the admin, which can be well utilized by his institute. This will be an additional advantage and management of power based on their free time from his normal duty.

The purpose of the project is to maintain the information regarding the employees and generates the pay slip. Main aim of developing Employee Management System is to provide an easy way not only to automate all functionalities involved managing leaves and Employee for the employee of Company, But also to provide full functional reports to management of Company with the details about usage of leave facility and Salaries Paid or to be paid to employees.

- ❖ We are committed to bring the best way to management in the various forms of EPM.
- ❖ we understand that EPM is not just a product to be sold, it is a tool to manage the inner operation of Company related to employee leave and Employee.

1.4 Scope

It may help collecting perfect management in details. In a very short time, the collection will be obvious, Simple and sensible. It will be helping a person to know the management of passed year perfectly and vividly. It also helps n current all works relative to Employee Management System. It will be also reducing the cost of collecting the management & collection procedure will go on smoothly.

Our project aims at Business process automation, i.e. we have tried to computerize various processes of Employee Management System.

- In computer system the person has to fill the various forms & number of copies of the forms can be easily generated at a time.
- In computer system, it is not necessary to create the manifest but we can directly print it, which saves our time.
- To assist the staff in capturing the effort spent on their respective working areas.
- To utilize resources in an efficient manner by increasing their productivity through automation.
- The system generates types of information that can be used for various purposes.
- It satisfies the user requirement
- Be easy to understand by the user and operator
- Be easy to operate
- Have a good user interface
- BE expandable
- Delivered on schedule within the budget

02. MODULES

- **Employee Management Module:** Used for managing the Employee details.
- **Working Points Modules:** Used for managing the details of working Points.
- **Payments Modules:** Used for managing the details of Payments.
- **Employee Management Modules:** Used for managing the information and details of the Employee.
- **Salary Module:** Used for managing the Salary details.
- **Appraisals Modules:** Used for managing the Appraisals information.
- **Login Module:** Used for managing the login details.
- **Users Module:** Used for managing the users of the system.

2.1 ADMIN

The admin gets logged in by valid username and password. Admin can add new Employee, add new Department, adds new Pay Grade for the employees. Admin can set the 'from' and 'to' date worked by an employee in a department with specific pay grade. The admin can generate an automated monthly salary of an employee. The admin can view all the past records of any recorded employee.

2.2 ADVANTAGE

- ❖ It is cost effective as the user control the web application himself and does not go for professional service.
- ❖ It saves time as it speeds up every aspect of the employee database management and Employee process with a range of automated features.
- ❖ It is secure as the employee database and the Employee process is managed by the admin inhouse rather than sending private information to a third party.
- ❖ Validating procedures and checks restrict user from making mistakes.
- ❖ The software is easy to use and is user friendly so no expertise is required.
- ❖ The calculations are automated so no chance of error.

2.3 DISADVANTAGE

- ❖ It requires an internet connection.
- ❖ It requires large database.

03. Survey of Technology

❖ Front end: React.JS, Vite, Node

1. **React JS:** React JS is a JavaScript library used for building interactive and dynamic user interfaces, particularly for single-page applications. It follows a component-based architecture and uses a virtual DOM to enhance performance, making it efficient for modern web development.
2. **Vite:** Vite is a modern frontend build tool that provides a fast development experience by using native ES modules for development and optimized bundling for production. It is lightweight, highly efficient, and significantly reduces startup and build times.
3. **Node JS:** Node.js is a server-side JavaScript runtime built on Chrome's V8 engine, allowing developers to build scalable and fast backend applications. It is designed for non-blocking, event-driven I/O, making it ideal for real-time and data-intensive applications.

❖ Back end: PHP, MySQL

1. **PHP:** Hypertext Pre-processors (PHP) is a technology that allows software developers to create dynamically generated web pages, in HTML, XML, or other document types, as per client request. PHP is open-source software.
2. **MySQL:** MySQL is a database, widely used for accessing querying, updating and managing data in databases.

04. FEASIBILITY STUDY

After identifying the scope of the project, the feasibility study is needed to be carried out. It is basically keeping the following points in mind.

- ❖ **Building the software for meeting the scope:** This software has met the scope. As there is no data involved in the system, processing on the file, and the behavior of this project is already identified and bundled in quantitative manner.
The processing of this software is very simple as it has been designed in php and it has been well divided into several functions according to the need.
- ❖ **Technically feasible:** This software is very much technically feasible. This software is very much concerned with specifying equipment and the software will successfully satisfy almost all the adman's requirements. The technical need for this system may vary considerably but might include:
 - A. The facility to produce output in a given time.
 - B. Response time under certain conditions.
 - C. Ability to process data at a particular speed
- ❖ **State of Art:** The project is very much within the state of art since the project is a window based; it uses very modern and common technique.
 - Beside it is very much modern and user friendly. It also works as middleware.
 - i.e. only inbetween the user and the file. So, it is completely a state of art project.
- ❖ **Financially Feasible:** The project is very much financially feasible. The implementation and development cost of this software under the reach of any college. Moreover, it requires some training for the use. So, training cost can be neglected and the resources of this software are very much available. It also reduces the labour and extra cost to be paid for labor. So indeed, it is financially feasible.
- ❖ **Resources:** As motioned earlier that the resources are easily available and the cost of training is almost negligible. Sometimes situations may arise when it may not be so much easy. For aperson completely unaware of using a computer system could result in a training cost or for a very small organization the purchase of a computer, instalment of the system and other charges may lead to a difficult matter.

05.PROJECT CATEGORY

❖ WEB BASED APPLICATION

Available Technologies:

Languages: React JS, Vite, Node JS

Database: MYSQL, PHP

Tools Used:

Editor Used: Visual Studio

Operating System: Windows 11

Hardware Used:

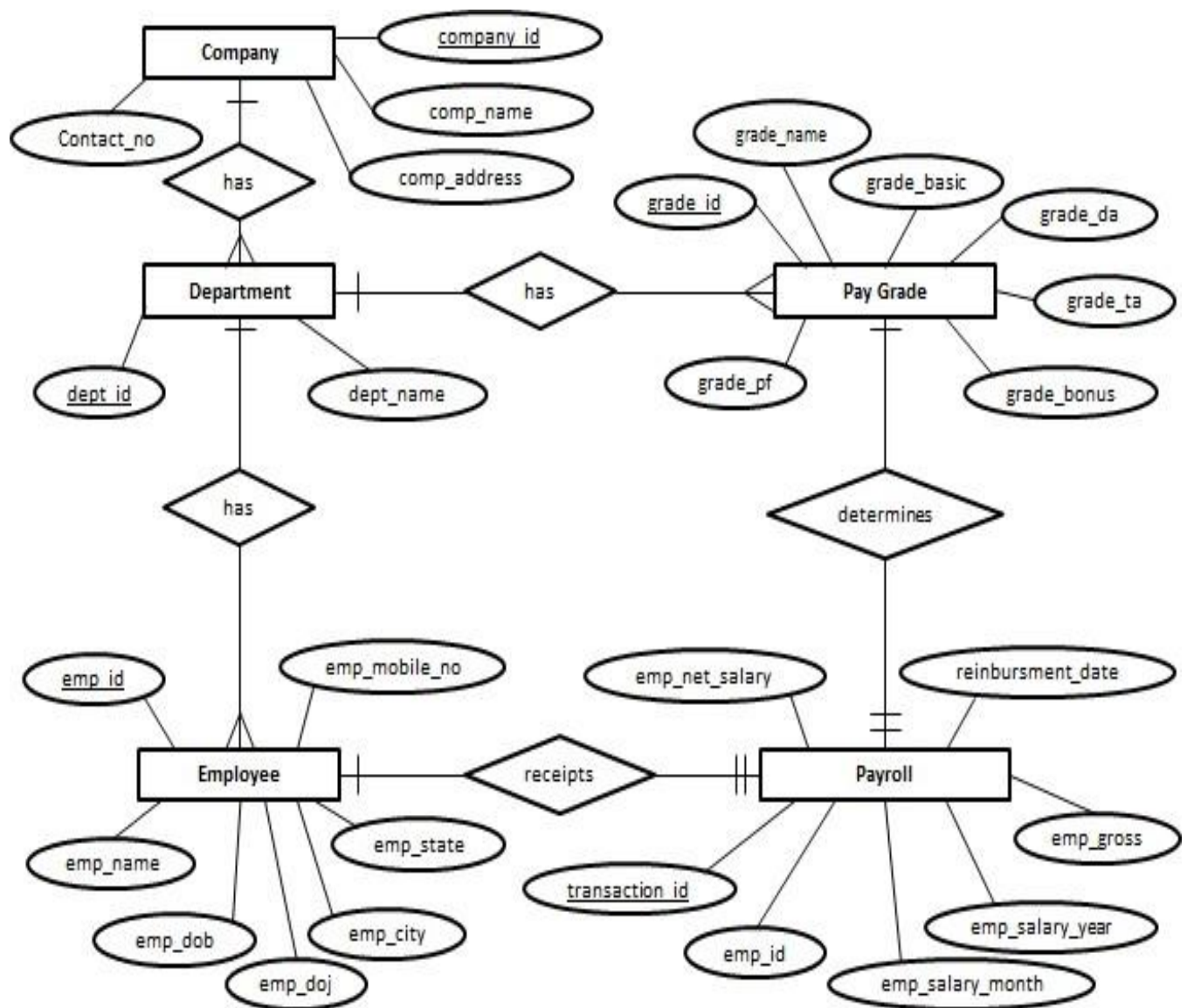
Processor: Intel core i5

RAM: GB

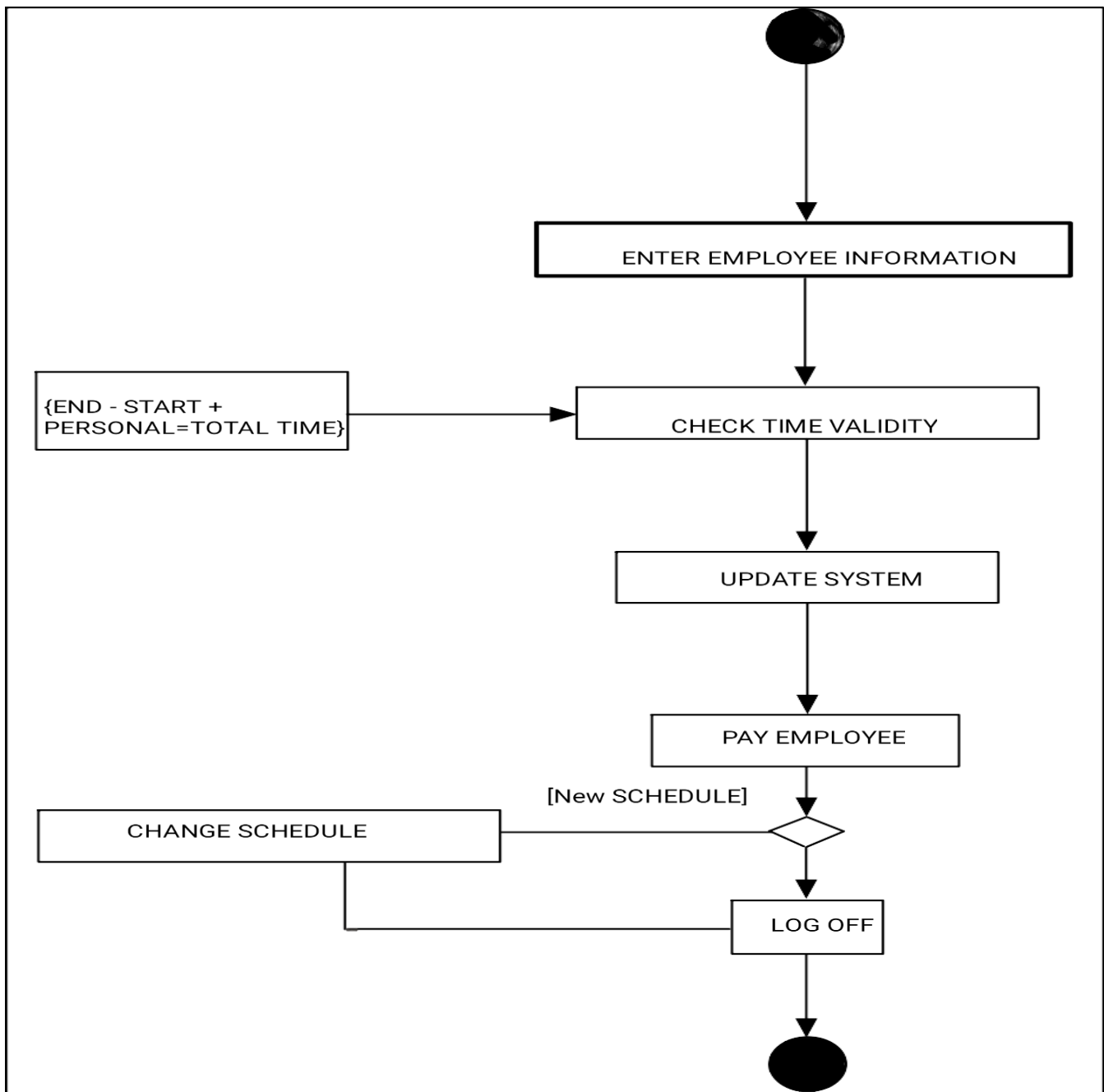
Hard Disk: 1TB

06.SYSTEM DESIGN

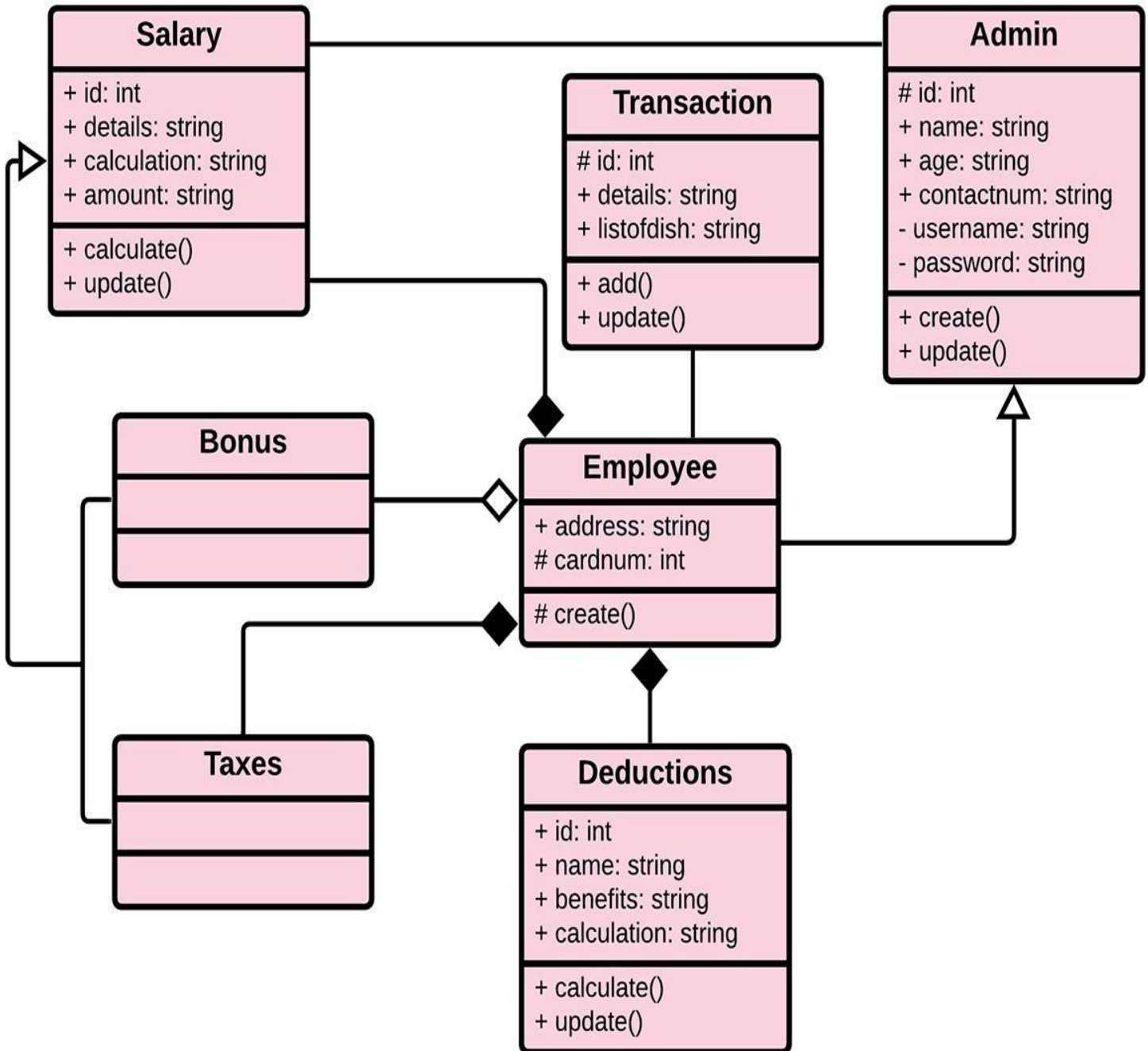
❖ 6.1 Entity Relationship Diagram (ERD):



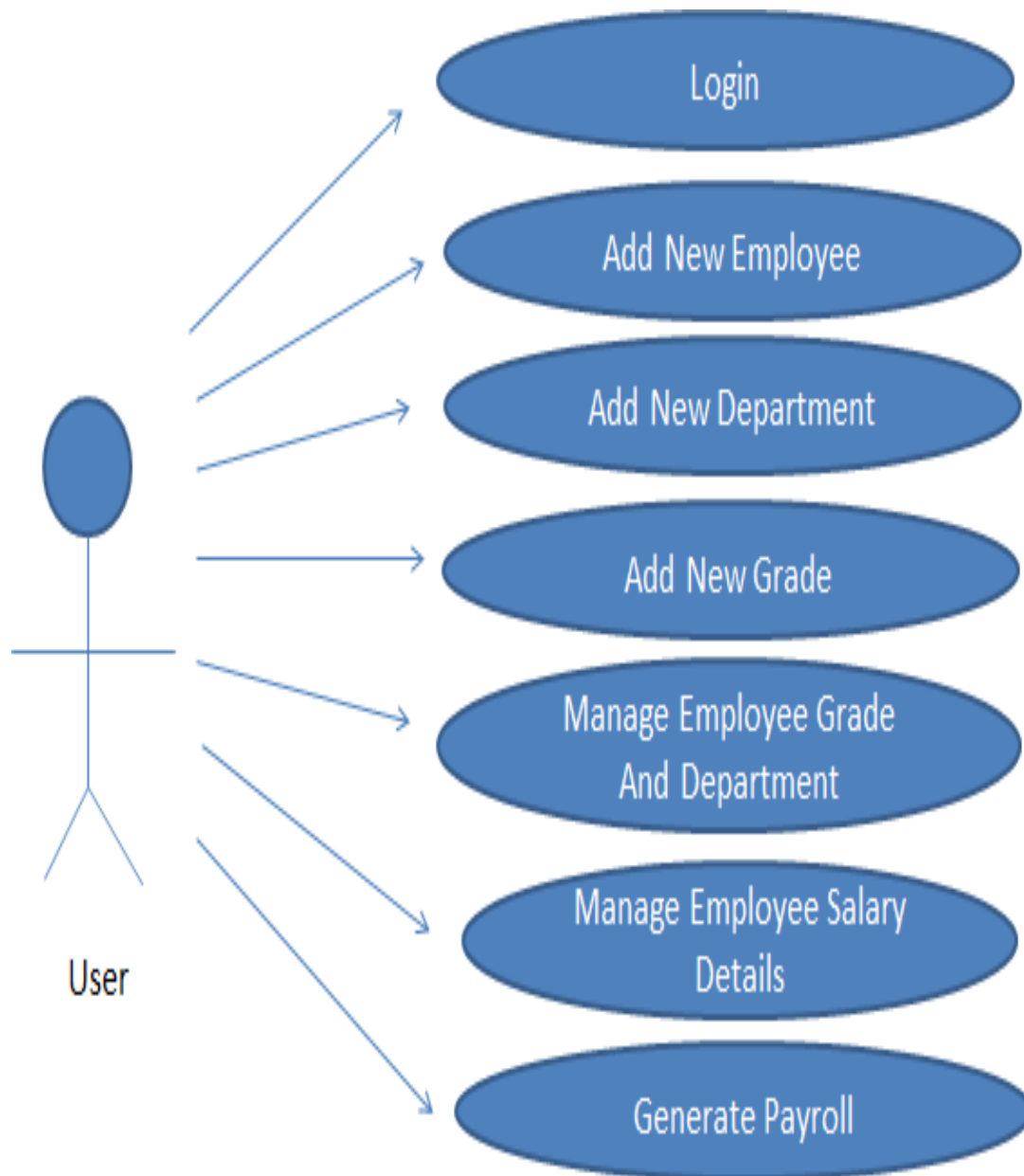
❖ 6.2 ACTIVITY DIAGRAM:



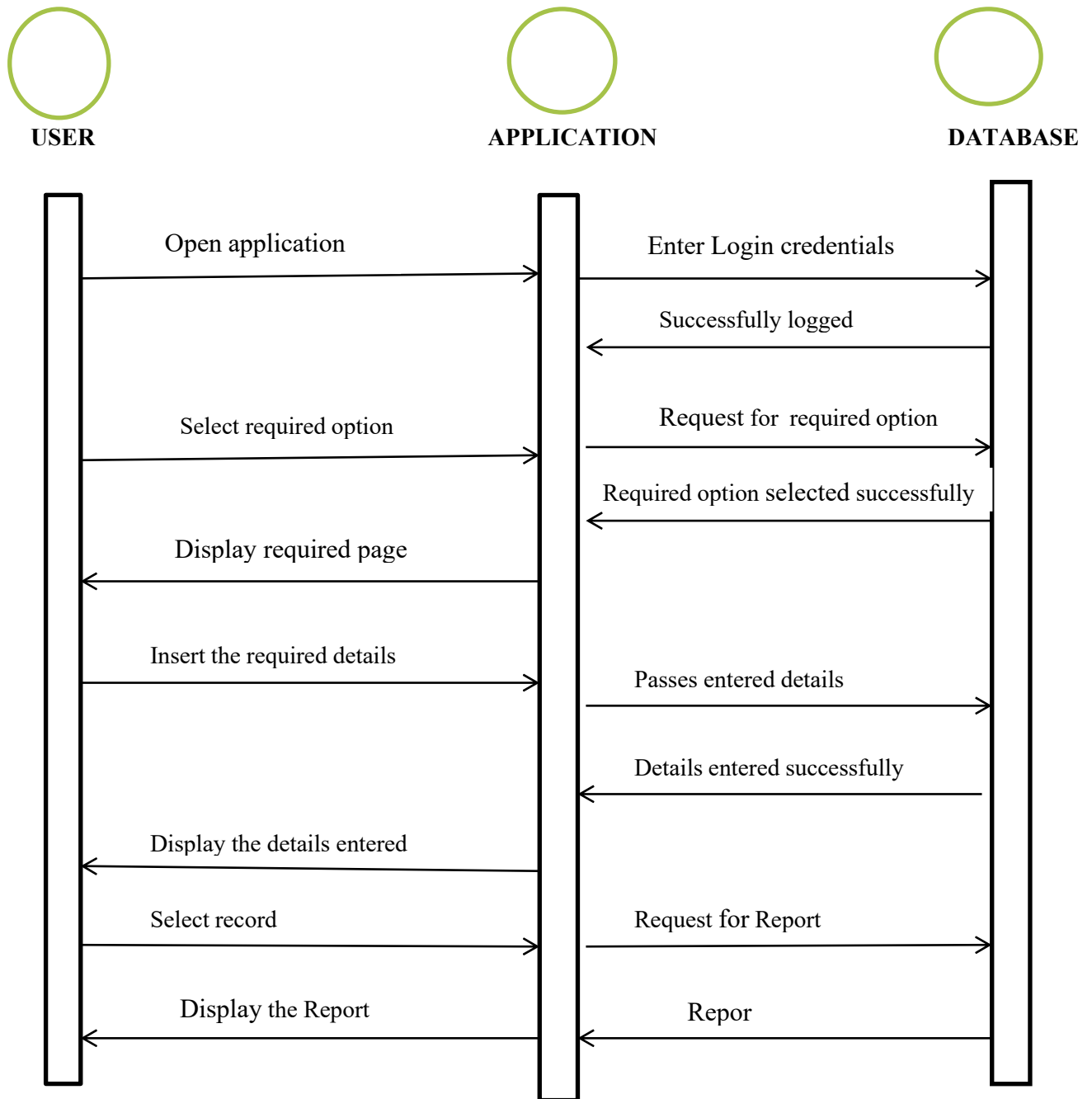
❖ 6.3 CLASS DIAGRAM:



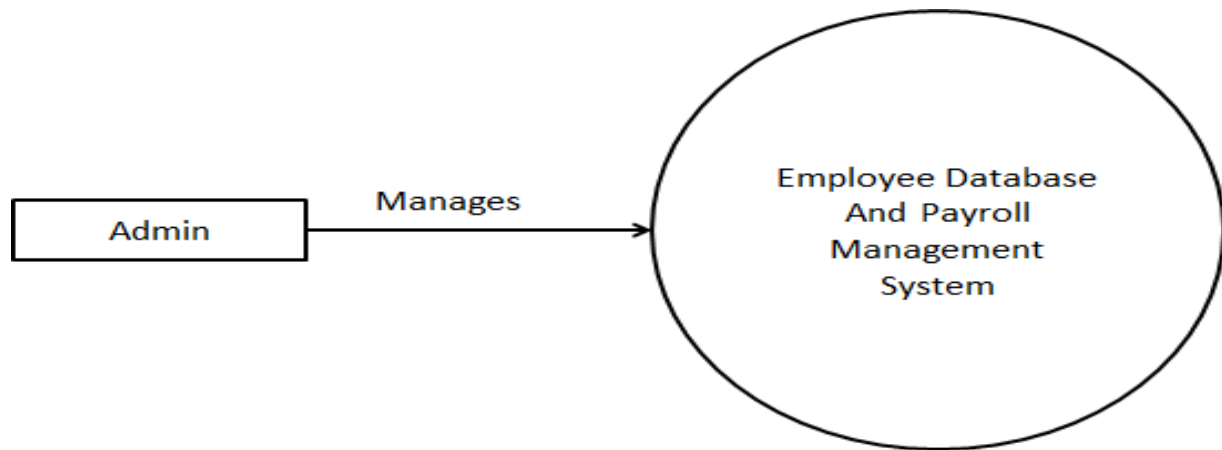
❖ **6.4 USE CASE DIAGRAM:**



❖ 6.5 SEQUENCE DIAGRAM:

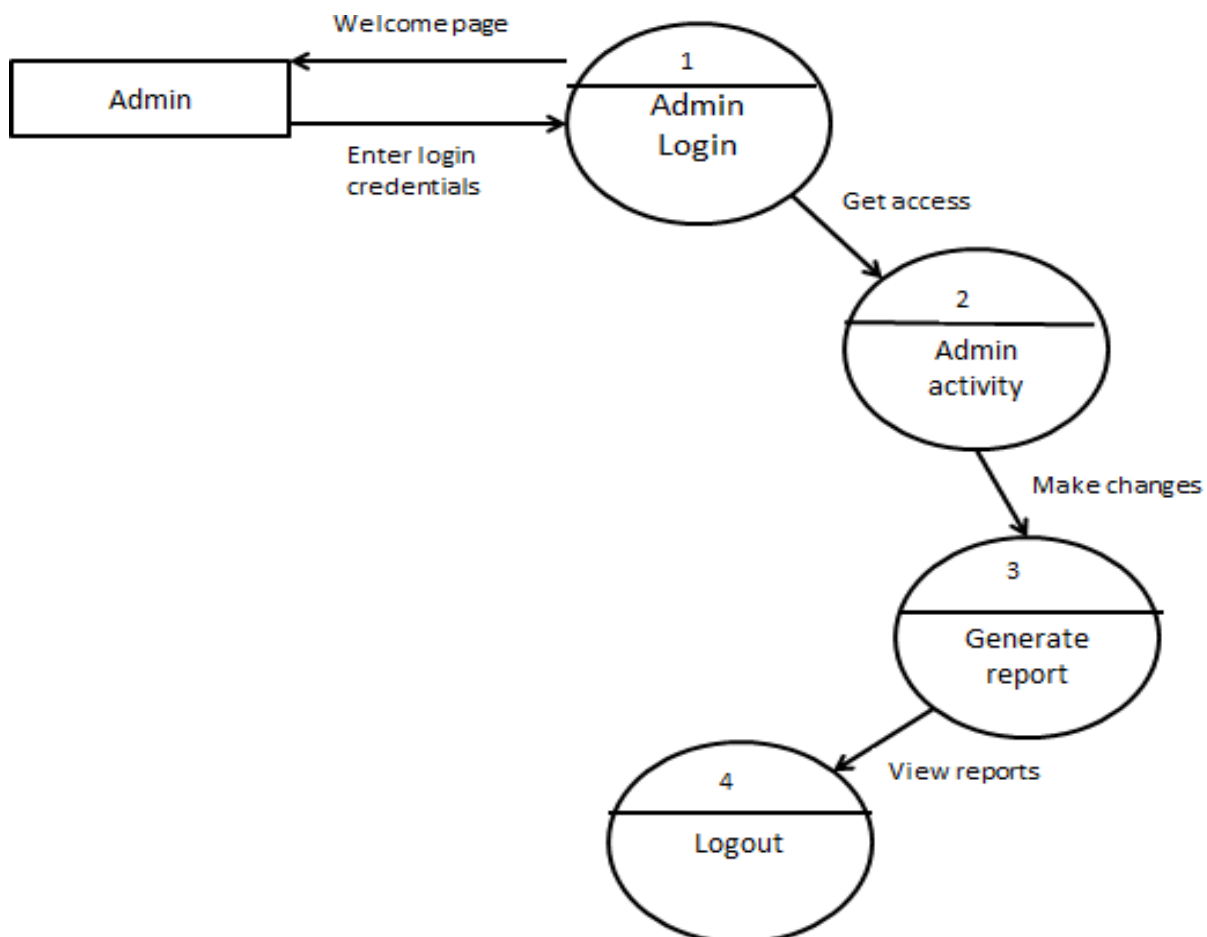


❖ 6.6 DATA FLOW DIAGRAM:

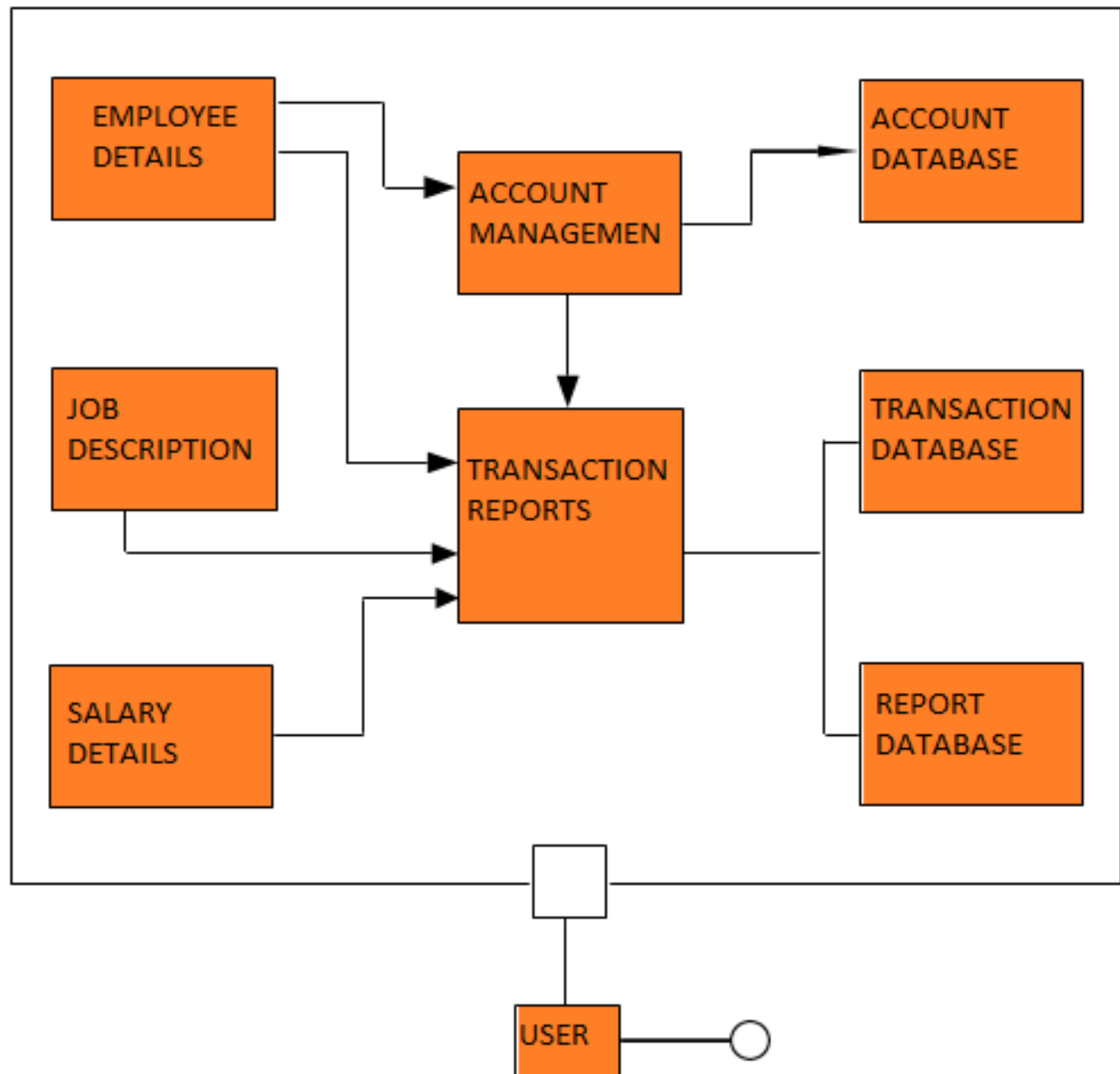


Level 0

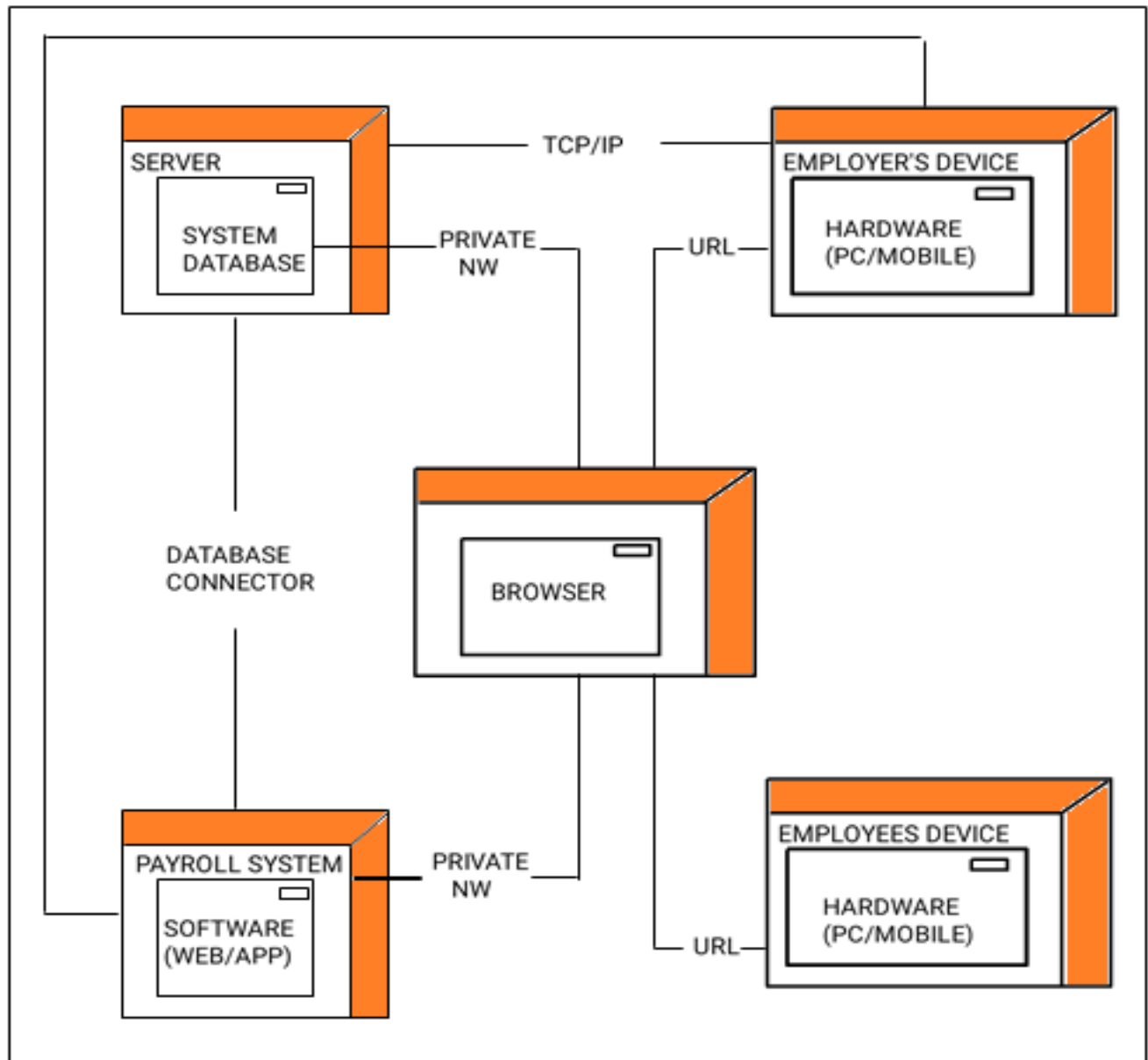
Level 1



❖ 6.7 COMPONENT DIAGRAM:



❖ 6.8 DEPLOYMENT DIAGRAM:



07. Source Code

❖ SIGN-IN PAGE:-

```
import { ErrorPopup } from "../error-popup"
import { Link } from "react-router-dom"

export const SignIn = ({ image, handlesigninform, handlesigninsubmit, targetedstate,
statevalue, redirectpath }) => {
  return (
    <
      {targetedstate.error.status ? <ErrorPopup error={targetedstate.error.message} /> : null}
      <div className="flex min-h-full flex-1 min-[200px]:flex-col md:flex-row md:items-
center justify-center px-6 py-12 lg:px-8 md:gap-5">

        <div className="sm:mx-auto sm:w-full sm:max-w-sm lg:mx-10">
          <img
            alt="Your Company"
            src={image}
            className="mx-auto h-auto w-auto"
          />
        </div>
        <div className="my-5 sm:mx-auto sm:w-full sm:max-w-sm lg:mx-10">
          <h2 className="mb-5 text-center text-2xl/9 font-bold tracking-tight text-gray-
900">
            Sign in to your account
          </h2>
          <form className="space-y-6" onSubmit={handlesigninsubmit}>
            <div>
              <label htmlFor="email" className="block text-sm/6 font-medium text-gray-
900">
                Email address
              </label>
              <div className="mt-2">
                <input
                  id="email"
                  name="email"
                  type="email"
                  required
                  autoComplete="email"
                  value={statevalue.email}
                  onChange={handlesigninform}
                  className="block w-full rounded-md border-0 py-1.5 text-gray-900
```

```
shadow-sm ring-1 ring-inset ring-gray-300 placeholder:text-gray-400 focus:ring-2 focus:ring-
inset focus:ring-indigo-600 sm:text-sm/6 p-2"
```

```
/>
```

```
</div>
```

```
</div>
```

```
<div>
```

```
<div className="flex items-center justify-between">
```

```
<label htmlFor="password" className="block text-sm/6 font-medium
text-gray-900">
```

```
    Password
```

```
</label>
```

```
<div className="text-sm">
```

```
<Link to={redirectpath}>
```

```
<a className="font-semibold text-indigo-600 hover:text-indigo-500">
```

```
    Forgot password?
```

```
</a>
```

```
</Link>
```

```
</div>
```

```
</div>
```

```
<div className="mt-2">
```

```
<input
```

```
    id="password"
```

```
    name="password"
```

```
    type="password"
```

```
    required
```

```
    autoComplete="current-password"
```

```
    value={statevalue.password}
```

```
    onChange={handlesigninform}
```

```
    className="block w-full rounded-md border-0 py-1.5 text-gray-900
shadow-sm ring-1 ring-inset ring-gray-300 placeholder:text-gray-400 focus:ring-2 focus:ring-
inset focus:ring-indigo-600 sm:text-sm/6 p-2"
```

```
/>
```

```
</div>
```

```
</div>
```

```
<div>
```

```
<button
```

```
    type="submit"
```

```
    className="flex w-full justify-center rounded-md bg-indigo-600 px-3 py-
1.5 text-sm/6 font-semibold text-white shadow-sm hover:bg-indigo-500 focus-visible:outline
focus-visible:outline-2 focus-visible:outline-offset-2 focus-visible:outline-indigo-600"
```

```
>
```

```
    Sign in
```

```
</button>
```

```

        </div>
      </form>
    </div>
  </div>
</>
)

```

❖ SIGN-UP PAGE:-

```

import { ErrorPopup } from "../error-popup"
import { useSelector } from "react-redux"
import { Button } from "@components/ui/button"
import { Link } from "react-router-dom"

export const SignUP = ({ handlesignupform, handlesubmitform, stateformdata, errorpopup }) => {
  const employeeestate = useSelector((state) => state.HRReducer)
  return (
    <div className="HR-form-content justify-center items-center min-[250px]:w-[90%] 2xl:w-[80%] grid grid-cols-1 min-[900px]:grid-cols-2 mx-auto">
      <div className="form-img mx-auto">
        
      </div>
      {/* <div className="form-content flex flex-col gap-6 sm:justify-center min-[250px]:items-center sm:items-center md:items-start md:justify-normal">
        <div className="form-heading my-3">
          <h1 className="text-4xl text-purple-700 font-bold">Sign UP HR</h1>
        </div>
        <div className="form-content flex gap-8 min-[250px]:flex-col sm:flex-row">
          <div className="form-section-first min-[250px]:w-[80vw] sm:w-[30vw] md:w-[20vw] flex flex-col gap-5 sm:text-sm lg:text-md xl:text-lg">
            <div className="form-field">

```

```

900">
    <label htmlFor="firstname" className="block font-medium text-gray-
    First Name
  </label>
  <div className="mt-2">
    <input
      id="firstname"
      name="firstname"
      type="text"
      required
      autoComplete="text"
      value={stateformdata.firstname}
      onChange={handlesignupform}
      className="block w-full rounded-md border-0 py-1.5 text-gray-900
shadow-sm ring-1 ring-inset ring-gray-300 placeholder:text-gray-400 focus:ring-2 focus:ring-
inset focus:ring-indigo-600 sm:text-sm/6 p-2"
    />
  </div>
</div>

<div className="form-field">
  <label htmlFor="lastname" className="block font-medium text-gray-
900">
    Last Name
  </label>
  <div className="mt-2">
    <input
      id="lastname"
      name="lastname"
      type="text"
      required
      autoComplete="text"
      value={stateformdata.lastname}
      onChange={handlesignupform}
      className="block w-full rounded-md border-0 py-1.5 text-gray-900
shadow-sm ring-1 ring-inset ring-gray-300 placeholder:text-gray-400 focus:ring-2 focus:ring-
inset focus:ring-indigo-600 sm:text-sm/6 p-2"
    />
  </div>
</div>

```

❖ DEPARTMENTABS:-

```
import { Button } from "@components/ui/button"
import {
  Card,
  CardContent,
  CardDescription,
  CardFooter,
  CardHeader,
  CardTitle,
} from "@components/ui/card"
import { Input } from "@components/ui/input"
import { Label } from "@components/ui/label"
import {
  Tabs,
  TabsContent,
  TabsList,
  TabsTrigger,
} from "@components/ui/tabs"
import { Check, ChevronsUpDown } from "lucide-react"

import { cn } from "@lib/utils"
import {
  Command,
  CommandEmpty,
  CommandGroup,
  CommandInput,
  CommandItem,
  CommandList,
} from "@components/ui/command"
import {
  Popover,
  PopoverContent,
  PopoverTrigger,
} from "@components/ui/popover"
import { useState, useEffect } from "react"
import {
  DropdownMenu,
  DropdownMenuContent,
  DropdownMenuItem,
  DropdownMenuLabel,
  DropdownMenuSeparator,
  DropdownMenuTrigger,
} from "@components/ui/dropdown-menu"
```

```

import { useSelector, useDispatch } from "react-redux"
import { HandleGetHRDepartments } from
"../../redux/Thunks/HRDepartmentPageThunk"
import { Loading } from "../loading.jsx"
import { HeadingBar } from "../ListDesigns.jsx"
import { DepartmentListItems } from "../ListDesigns.jsx"
import { useToast } from "../../hooks/use-toast.js"
import { EmployeesIDSDialogBox } from "../dialogboxes.jsx"

export const HRDepartmentTabs = () => {
  const { toast } = useToast()
  const HRDepartmentState = useSelector((state) => state.HRDepartmentPageReducer)
  const dispatch = useDispatch()
  const [department, setdepartment] = useState("All Departments")

  const departments = []

  if (HRDepartmentState.data) {
    for (let index = 0; index < HRDepartmentState.data.length; index++) {
      departments.push({
        value: HRDepartmentState.data[index].name,
        label: HRDepartmentState.data[index].name
      })
    }
  }

  useEffect(() => {
    if (HRDepartmentState.fetchData) {
      dispatch(HandleGetHRDepartments({ apiRoute: "GETALL" }))
    }

    if (HRDepartmentState.error.status) {
      toast({
        variant: "destructive",
        title: "Uh oh! Something went wrong.",
        description: ` ${HRDepartmentState.error.message}`,
      })
    }

    if (HRDepartmentState.success.status) {
      toast({
        title: <p className="text-xl m-1">Success!</p>,
        description: <div className="flex justify-center items-center gap-2">
          
      <p className="font-bold">{HRDepartmentState.success.message}</p>
    </div>,
  })
}

console.log("test message")

}, [HRDepartmentState.fetchData, HRDepartmentState.error,
HRDepartmentState.success])

useEffect(() => {
  dispatch(HandleGetHRDepartments({ apiroute: "GETALL" }))
}, [])

if (HRDepartmentState.isLoading) {
  return (
    <Loading />
  )
}

```

❖ FORGOT-PASSWORD:-

```

import { ErrorPopup } from "../error-popup"
import { useSelector } from "react-redux"
import { Link } from "react-router-dom"

export const ForgotPassowrd = ({ handleforgotpasswordsubmit, handlesforgotpasswordform,
targetState, redirectpath }) => {
  const EmployeeState = useSelector((state) => state.employeeReducer)
  return (
    <
      {targetState.error.status ? <ErrorPopup error={targetState.error.message} /> : null}
      <div className="flex min-h-full flex-1 min-[200px]:flex-col md:flex-row md:items-
center justify-center px-6 py-12 lg:px-8 md:gap-5">

        <div className="sm:mx-auto sm:w-full sm:max-w-sm lg:mx-10">
          
        </div>

```

```
<div className="my-5 sm:mx-auto sm:w-full sm:max-w-sm lg:mx-10">
  <h2 className="mb-5 text-center text-2xl/9 font-bold tracking-tight text-gray-
900">
    Enter Your Email Address
  </h2>
  <form className="space-y-6" onSubmit={handleforgotpasswordsubmit}>
    <div>
      <label htmlFor="email" className="block text-sm/6 font-medium text-gray-
900">
        Email address
      </label>
      <div className="mt-2">
        <input
          id="email"
          name="email"
          type="email"
          required
          autoComplete="email"
          onChange={handlesforgotpasswordform}
          className="block w-full rounded-md border-0 py-1.5 text-gray-900
shadow-sm ring-1 ring-inset ring-gray-300 placeholder:text-gray-400 focus:ring-2 focus:ring-
inset focus:ring-indigo-600 sm:text-sm/6 p-2"
        />
      </div>
    </div>
    <div className="flex flex-col gap-2">
      <button
        type="submit"
        className="flex w-full justify-center rounded-md bg-indigo-600 px-3 py-
1.5 text-sm/6 font-semibold text-white shadow-sm hover:bg-indigo-500 focus-visible:outline
focus-visible:outline-2 focus-visible:outline-offset-2 focus-visible:outline-indigo-600"
      >
        Get Reset Email
      </button>
      <Link to={redirectpath}>
        <button
          className="flex w-full justify-center rounded-md bg-indigo-600 px-3
py-1.5 text-sm/6 font-semibold text-white shadow-sm hover:bg-indigo-500 focus-
visible:outline focus-visible:outline-2 focus-visible:outline-offset-2 focus-visible:outline-
indigo-600"
        >
          Back to Login
        </button>
      </Link>
    </div>
```



```

    </form>
  </div>
</div>
</>
)
}

```

❖ LOADING:-

```

import { ErrorPopup } from "../error-popup"
import { useSelector } from "react-redux"
import { Link } from "react-router-dom"

export const ForgotPassowrd = ({ handleforgotpasswordsubmit, handlesforgotpasswordform,
targetState, redirectpath }) => {
  const EmployeeState = useSelector((state) => state.employeeReducer)
  return (
    <
      {targetState.error.status ? <ErrorPopup error={targetState.error.message} /> : null}
      <div className="flex min-h-full flex-1 min-[200px]:flex-col md:flex-row md:items-
center justify-center px-6 py-12 lg:px-8 md:gap-5">

        <div className="sm:mx-auto sm:w-full sm:max-w-sm lg:mx-10">
          
        </div>
        <div className="my-5 sm:mx-auto sm:w-full sm:max-w-sm lg:mx-10">
          <h2 className="mb-5 text-center text-2xl/9 font-bold tracking-tight text-gray-
900">
            Enter Your Email Address
          </h2>
          <form className="space-y-6" onSubmit={handleforgotpasswordsubmit}>
            <div>
              <label htmlFor="email" className="block text-sm/6 font-medium text-gray-
900">

```

❖ RESET-EMAIL-CONFIRM:-

```

import { Link } from 'react-router-dom'
export const ResetEmailConfirmation = ({ redirectpath }) => {
  return (

```

```

<div className="email-reset-container h-[100vh] flex justify-center items-center">
  <div className="reset-email-content flex min-h-full flex-1 min-[200px]:flex-col
md:flex-row md:items-center justify-center px-6 py-12 lg:px-8 md:gap-5">
    <div className="reset-email-img sm:mx-auto sm:w-full sm:max-w-sm lg:mx-10">
      
    </div>
    <div className="reset-email-content my-5 sm:mx-auto sm:w-full sm:max-w-sm
lg:mx-5 md:w-full flex flex-col justify-center min-[200px]:text-center md:text-start min-
[200px]:items-center md:items-start gap-3">
      <p className="md:text-lg text-green-600 font-bold sm:text-sm lg:text-xl">We
have successfully send you a Reset email, Click on The Reset Password To Reset Your
Password.</p>
      <p className="md:text-lg text-red-600 font-bold sm:text-sm lg:text-xl">Please
do not forward reset password email</p>
      <div className="back-login-button">
        <Link to={redirectpath} className="flex w-full justify-center rounded-md bg-
indigo-600 px-3 py-1.5 text-sm/6 font-semibold text-white shadow-sm hover:bg-indigo-500
focus-visible:outline focus-visible:outline-2 focus-visible:outline-offset-2 focus-
visible:outline-indigo-600">
          Back to Login
        </Link>
      </div>
    </div>
  </div>
</div>
)
}

```

❖ Reset-Password:-

```

import { Link } from "react-router-dom"
import { ErrorPopup } from "../error-popup"
import { useSelector } from "react-redux"
export const Reset Password = ({ handlepasswordsubmit, handlepasswordform,
passworderror, targetstate }) => {
  return (
    <
      {targetstate.error.status ? <ErrorPopup error={targetstate.error.message} /> : null}
      {passworderror ? <ErrorPopup error={"Password Does Not Match, Please Try
Again"} /> : null}
    <div className="flex min-h-full flex-1 min-[200px]:flex-col md:flex-row md:items-
center justify-center px-6 py-12 lg:px-8 md:gap-5">

```

08. Screen Short

▪ EMS Entry Page:



Welcome To Employee Management System, Please Select Your Role to Proceed Further

Employee

HR-Admin

▪ EMS HR Sign Up:



First Name

Contact Number

Last Name

Organization Name

Email

Organization Description

Password

Organization URL

Confirm Password


Organization Mail

Sign Up

Already Have an Account?

Sign In

■ EMS HR Login:



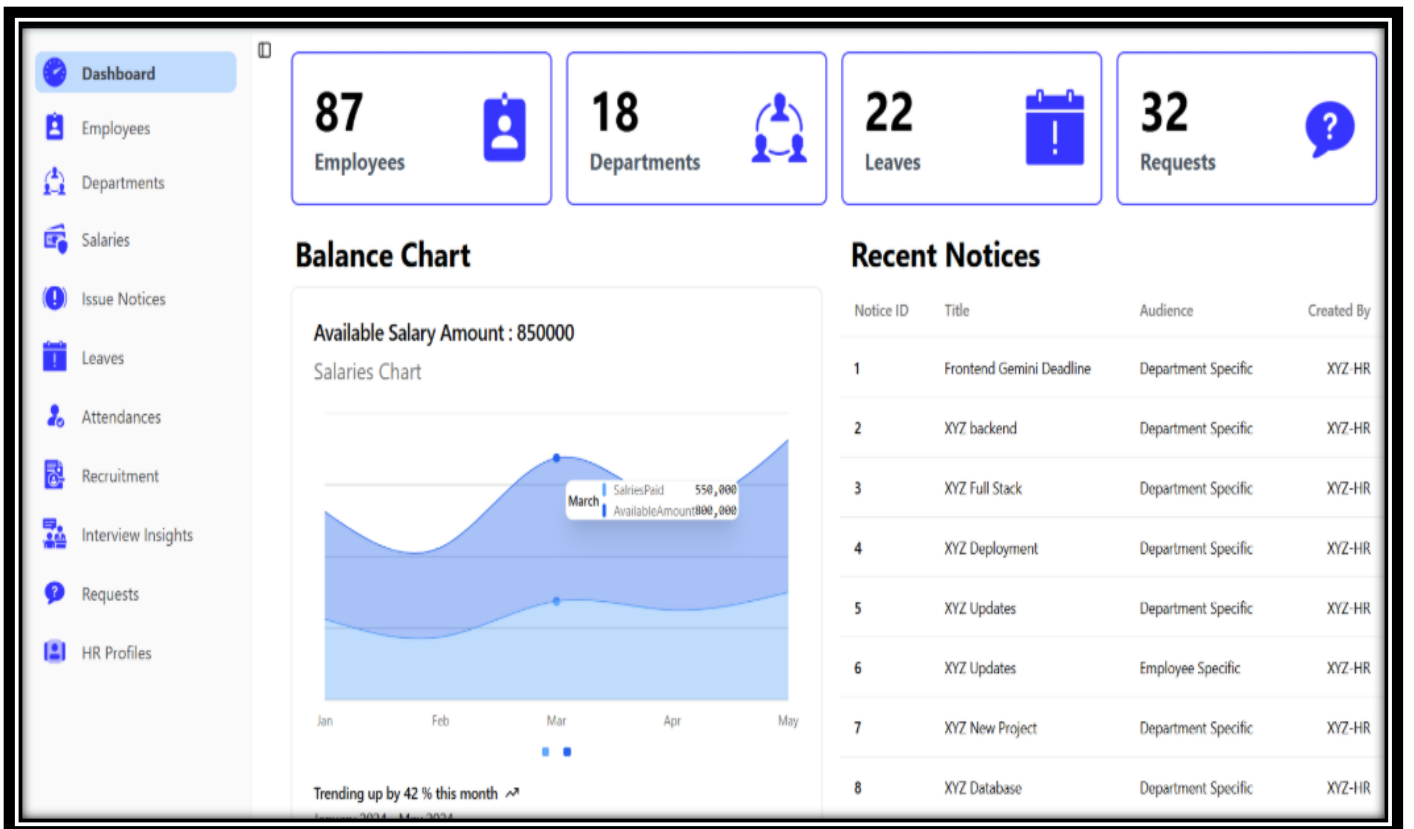
Sign in to your account

Email address

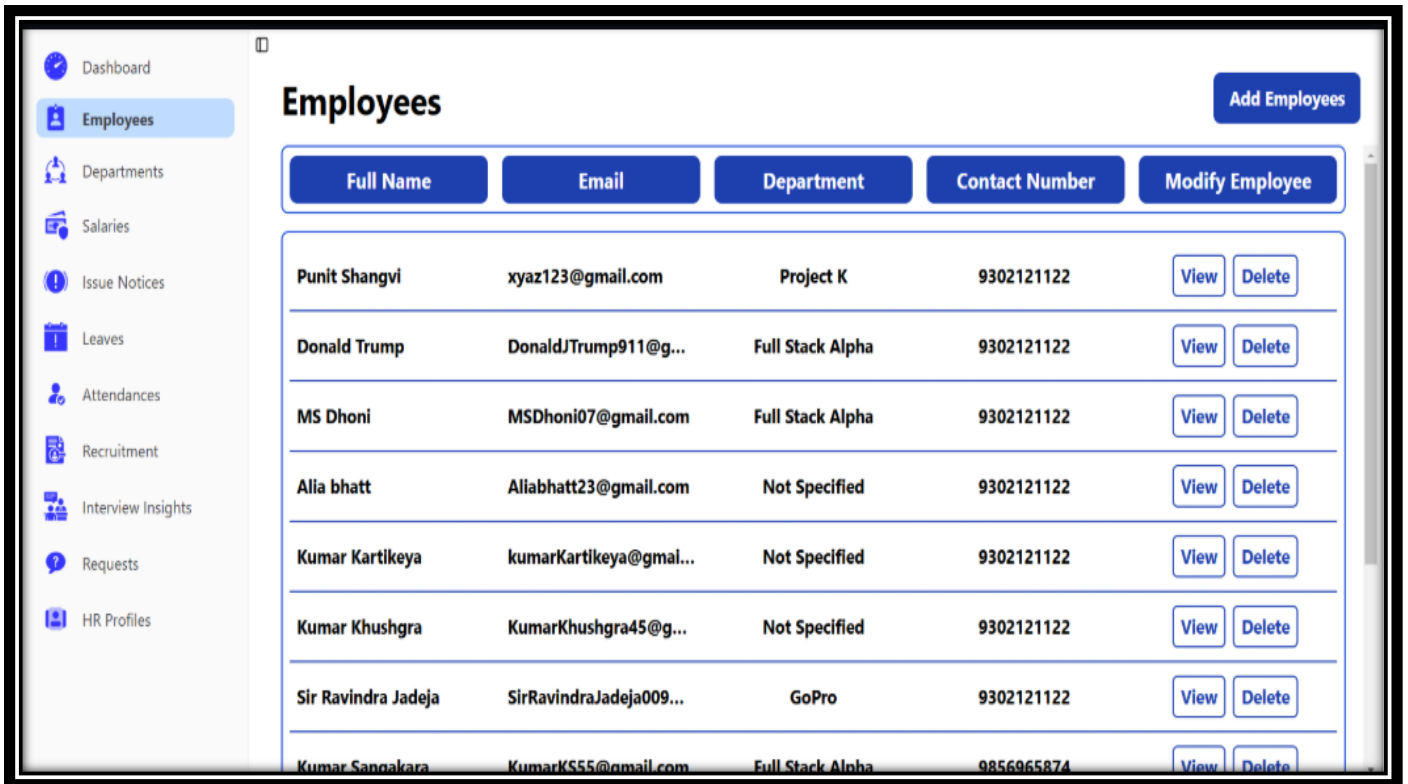
Password [Forgot password?](#)

[Sign in](#)

■ EMS HR Dashboard:



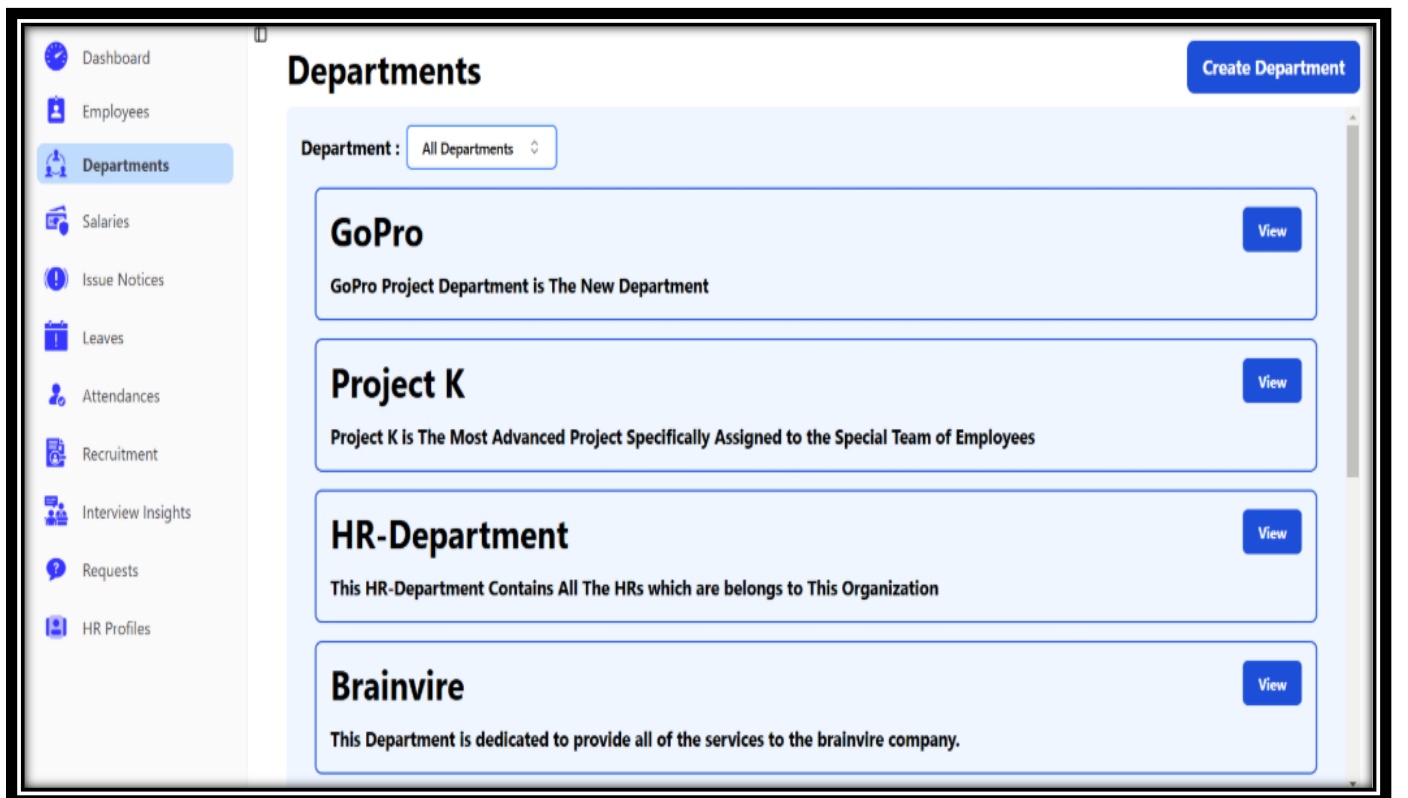
■ EMS HR Employees Page:



The screenshot shows the 'Employees' page in the EMS HR system. On the left is a sidebar with navigation links: Dashboard, Employees (selected), Departments, Salaries, Issue Notices, Leaves, Attendances, Recruitment, Interview Insights, Requests, and HR Profiles. The main content area is titled 'Employees' and features an 'Add Employees' button in the top right. Below the title is a table with columns: Full Name, Email, Department, Contact Number, and Modify Employee. The table contains eight rows of employee data, each with 'View' and 'Delete' buttons in the 'Modify Employee' column.

| Full Name | Email | Department | Contact Number | Modify Employee |
|---------------------|-------------------------|------------------|----------------|---|
| Punit Shangvi | xyz123@gmail.com | Project K | 9302121122 | View Delete |
| Donald Trump | DonaldJTrump911@g... | Full Stack Alpha | 9302121122 | View Delete |
| MS Dhoni | MSDhoni07@gmail.com | Full Stack Alpha | 9302121122 | View Delete |
| Alia bhatt | Aliabhatt23@gmail.com | Not Specified | 9302121122 | View Delete |
| Kumar Kartikeya | kumarKartikeya@gmal... | Not Specified | 9302121122 | View Delete |
| Kumar Khushgra | KumarKhushgra45@g... | Not Specified | 9302121122 | View Delete |
| Sir Ravindra Jadeja | SirRavindraJadeja009... | GoPro | 9302121122 | View Delete |
| Kumar Sangakara | KumarKSS5@gmail.com | Full Stack Alpha | 9856965874 | View Delete |

■ EMS HR Departments Page:



The screenshot shows the 'Departments' page in the EMS HR system. On the left is a sidebar with navigation links: Dashboard, Employees, Departments (selected), Salaries, Issue Notices, Leaves, Attendances, Recruitment, Interview Insights, Requests, and HR Profiles. The main content area is titled 'Departments' and features a 'Create Department' button in the top right. Below the title is a 'Department :' dropdown menu currently set to 'All Departments'. The main content area displays four department cards, each with a title, a description, and a 'View' button.

Department : All Departments

GoPro

GoPro Project Department is The New Department

[View](#)

Project K

Project K is The Most Advanced Project Specifically Assigned to the Special Team of Employees

[View](#)

HR-Department

This HR-Department Contains All The HRs which are belongs to This Organization

[View](#)

Brainvire

This Department is dedicated to provide all of the services to the brainvire company.

[View](#)

■ EMS HR Departments Data:

Dashboard

Employees

Departments

Salaries

Issue Notices

Leaves

Attendances

Recruitment

Interview Insights

Requests

HR Profiles

Departments

Create Department

Department : Project K

Settings

Project K

Project K is The Most Advanced Project Specifically Assigned to the Special Team of Employees

6 Employees0 Notice

Add Employees

| Full Name | Email | Contact Number | Remove Employee |
|-----------------|--------------------------|----------------|-----------------|
| Kane willaimson | KN123@gmail.com | 8585454552 | Remove |
| Alla Bhatt | allabhatt234@gmail.com | 8585454565 | Remove |
| Punit Shangvi | xyaz123@gmail.com | 9302121122 | Remove |
| David Miller | davidmiller23@gmail.com | 8585454574 | Remove |
| Kumar Kartikeya | kumarKartikeya@gmail.com | 9302121122 | Remove |

09. FEATURES OF THE PROJECT

- Easy to use.
- It is completely secure.
- It is completely controlled by admin.
- This system is easily compatible with most of the web browsers.
- It is very interactive and save time.
- Reduces paper works.
- Calculations are automated so it is highly accurate.
- Targets & Milestones for guiding the programmers.
- Attachments & Additional Comments for more information.
- Admin and staff can view all the records whenever necessary with ease.
- Various level of report available with a lot of filters criteria.
- Easy to update information.

10. FUTURE SCOPE OF THIS WORK

- The option to print records in the future.
- I intend to add a leave structure in the future.
- I would like to implement a regular backup mechanism to back up the employee
- Database to avoid disasters.
- The system can be developed in such a way that its existing features can
- Be modified to better versions.

11. CONCLUSION

This project is built keeping in mind that it is to be used by only one user that is the admin. It is built for use in small scale organization where the number of employees is limited. According to the requested requirement the admin can add, manipulate, update and delete all employee data in his organization.

The admin can add new departments and delete them. The Admin can also add predefined pay grades for the employees. The required records can be easily viewed by the admin anytime he wants in an instant. The payment of the employee is based on monthly basis.

Numerous validations implemented would enable the admin to enter accurate data. The main objective of this framework is to save time, make the system cost effective and management records efficiently.

12. BIBLIOGRAPHY

❖ Websites:

- www.w3schools.com
- www.tutorialspoint.com
- www.youtube.com