EMPLOYEE REWARD MANAGEMENT SYSTEM

A MINI-PROJECT REPORT

Submitted by

ADHESH. M (220701012)

AKASH NARAYAN. P (220701020)

in partial fulfillment of the award of the degree

of

BACHELOR OF ENGINEERING

IN

COMPUTER SCIENCE AND ENGINEERING

RAJALAKSHMI ENGINEERING COLLEGE

AUTONOMOUS, CHENNAI

NOV/DEC, 2023

BONAFIDE CERTIFICATE

Certified that this mini project **"EMPLOYEE REWARD MANAGEMENT SYSTEM"** is the bonafide work of "**ADHESH. M(220701012), AKASH NARAYAN. P(220701020)"** who carried out the project work under my supervision.

SIGNATURE

Mr.Duraimurugan,

Professor,

Computer Science & Engineering

Rajalakshmi Engineering College

Thandalam, Chennai -602105.

Submitted for the End semester practical examination to be held on

INTERNAL EXAMINER **EXTERNAL EXAMINER**

ACKNOWLEDGEMENT

I express my sincere thanks to my beloved and honorable chairman **MR.S.MEGANATHAN** and the chairperson **DR.M.THANGAM MEGANATHAN** for their timely support and encouragement.

I am greatly indebted to my respected and honorable principal **Dr. S.N.MURUGESAN** for his able support and guidance.

No words of gratitude will suffice for the unquestioning support extended to us by my head of the department **Dr. P. KUMAR,** and my Academic Head **Dr.N.DURAIMURUGAN,** for being ever supporting force during my project work.

I also extend my sincere and hearty thanks to my internal guide **Mrs. JANANEE V** and **Mrs. BHAVANI M** for her valuable guidance and motivation during the completion of this project.

My sincere thanks to my family members, friends and other staff members of Computer Science and Engineering.

Adhesh M (220701012)

Akash Narayan P (220701020)

ABSTRACT

The Reward Management System is a web-based application designed to streamline and automate the process of tracking, evaluating, and rewarding employee performance within an organization. The system enables managers to assign performance scores to employees, record various achievements, and manage rewards in a centralized database. It provides a user-friendly interface for adding and updating employee performance data, facilitating a transparent and efficient approach to employee recognition and reward distribution. The system supports real-time updates, enabling authorized personnel to access performance metrics, scores, and reward histories from anywhere. By automating the reward process, the system not only enhances organizational transparency but also motivates employees, improving overall productivity and morale. The Reward Management System aims to foster a culture of recognition, ensuring that employees' contributions are consistently acknowledged and rewarded in a fair and systematic manner.

TABLE OF CONTENTS

CHAPTER NO.	TITLE	PAGE NO
	ABSTRACT	**4**
1	**INTRODUCTION**	**6**
	1.1 INTRODUCTION	6
	1.2 SCOPE OF THE WORK	6
	1.3 PROBLEM STATEMENT	6
	1.4 AIM AND OBJECTIVES OF THE PROJECT	7
2	**SYSTEM SPECIFICATIONS**	**8**
	2.1 HARDWARE SPECIFICATIONS	8
	2.2 SOFTWARE SPECIFICATIONS	8
3	**ARCHITECTURE DIAGRAM**	**9**
4	**MODULE DESCRIPTION**	**10**
5	**SYSTEM DESIGN**	**11**
	5.1 USECASE DIAGRAM	11
	5.2 E-R MODEL	12
	5.3 DATAFLOW DIAGRAM	13
	5.4 ACTIVITY DIAGRAM	15
6	**CODING**	**18**
7	**SCREENSHOTS**	**20**
8	**CONCLUSION**	**26**
	REFERENCES	27

CHAPTER 1

1.1 INTRODUCTION

The Reward Management System is a web-based solution designed to simplify the process of tracking and rewarding employee performance in organizations. It allows managers to assign scores, record achievements, and manage rewards, ensuring a fair and transparent recognition system. This system aims to enhance employee motivation, boost morale, and promote a culture of performance-driven growth within the organization.

1.2 SCOPE OF THE WORK

The system is designed to manage employee performance data and reward assignments efficiently. It includes features for recording employee scores, tracking performance over time, and distributing rewards based on merit. The system also allows authorized users to access real-time data on employee performance, ensuring transparency and consistency in the reward process. The goal is to streamline the reward management process and help create a positive and motivating work environment..

1.3 PROBLEM STATEMENT

Many organizations still rely on manual, inconsistent methods to track employee performance and assign rewards. This lack of transparency and efficiency can lead to dissatisfaction among employees and hinder motivation. This project addresses these challenges by developing a web-based Reward Management System that automates the reward process, ensuring fairness, consistency, and ease of access to performance data for managers and employees alike.

1.4 AIM AND OBJECTIVES OF THE PROJECT

The current employee reward system in many organizations is often manual, inconsistent, and lacks transparency. This leads to confusion, dissatisfaction, and demotivation among employees, affecting overall performance and morale. The Reward Management System aims to address these challenges by providing an automated, web-based solution to track employee performance and manage rewards efficiently. This system will be used by managers and authorized personnel to assess and recognize employee contributions in a timely and fair manner, ensuring consistent and transparent reward distribution. The system is designed to improve employee engagement, boost motivation, and foster a positive work environment.

Other objectives of the Reward Management System include enhancing the accessibility of performance data, ensuring data security, and promoting fairness in the reward process. Ultimately, the system seeks to create a transparent culture of recognition within organizations, helping to increase employee satisfaction, retention, and productivity.

CHAPTER 2

SYSTEM SPECIFICATIONS

2.1 HARDWARE SPECIFICATIONS

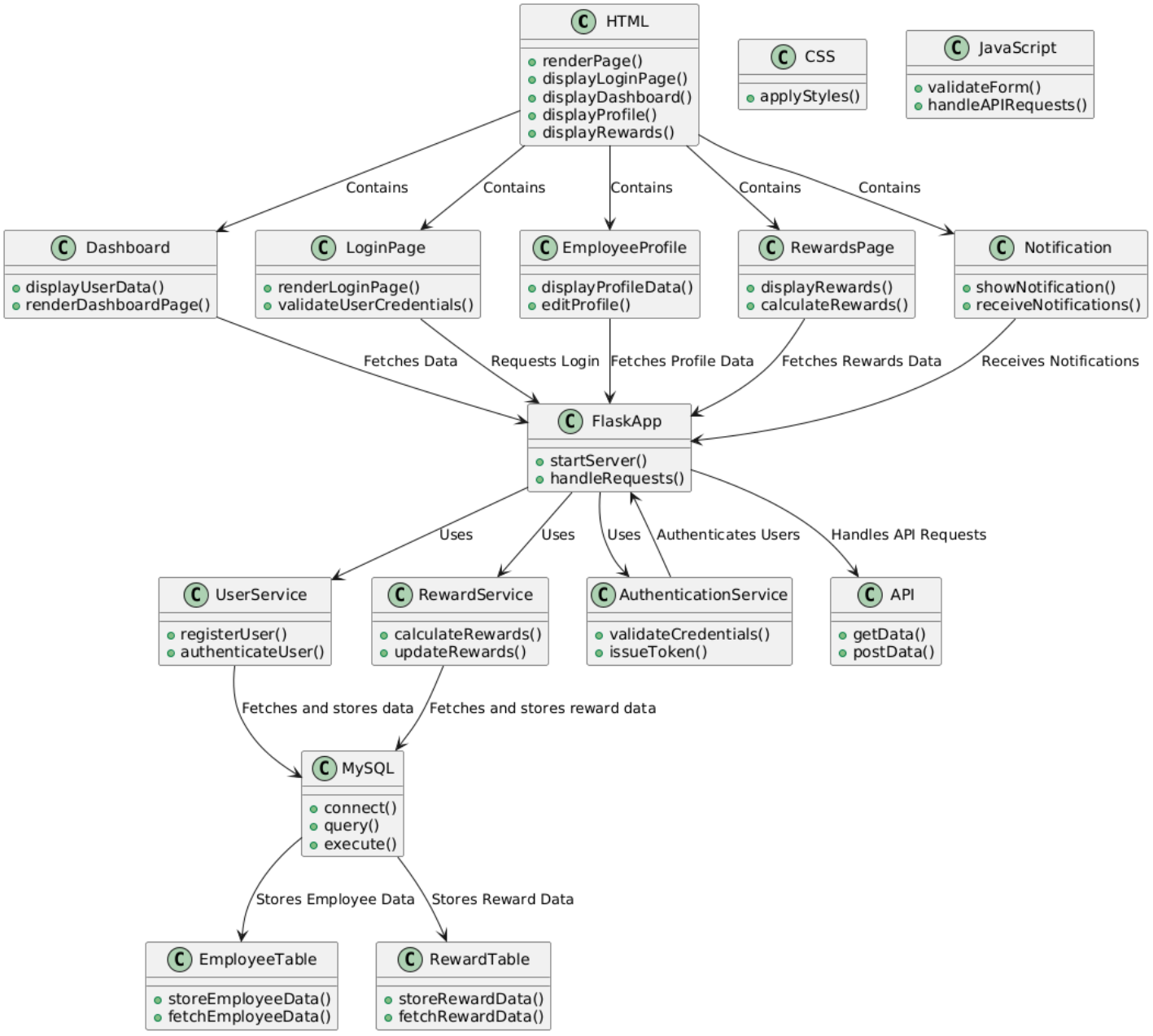
Processor	:	Intel Core i3 or Higher
Memory Size	:	128 GB (Minimum)
HDD	:	40 GB (Minimum)

2.2 SOFTWARE SPECIFICATIONS

Operating System	:	WINDOWS 7 AND PLUS
Front – End	:	HTML, CSS, JAVASCRIPT
Back – End	:	FLASK, MYSQL

CHAPTER 3

ARCHITECTURE DIAGRAM



CHAPTER 4

MODULE DESCRIPTION

4.1. User Registration and Login Module:

This module allows users to create an account and log in to the Reward Management System. Users provide basic details (name, email, password) to register, and can log in securely using these credentials.

4.2. Reward Management Module:

Managers and admins use this module to track and assign rewards based on employee performance and achievements. Employees can view their rewards and progress, while managers and admins can modify reward criteria.

4.3. Employee Profile Module:

This module allows employees to view and update their personal information, such as name, department, and reward history. Employees can track their performance, monitor rewards, and make changes to their profiles.

4.4. Admin Management Module

The Admin Management Module gives admins full control over user accounts, employee data, and reward criteria. Admins can manage system functionality, update reward rules, and generate reports

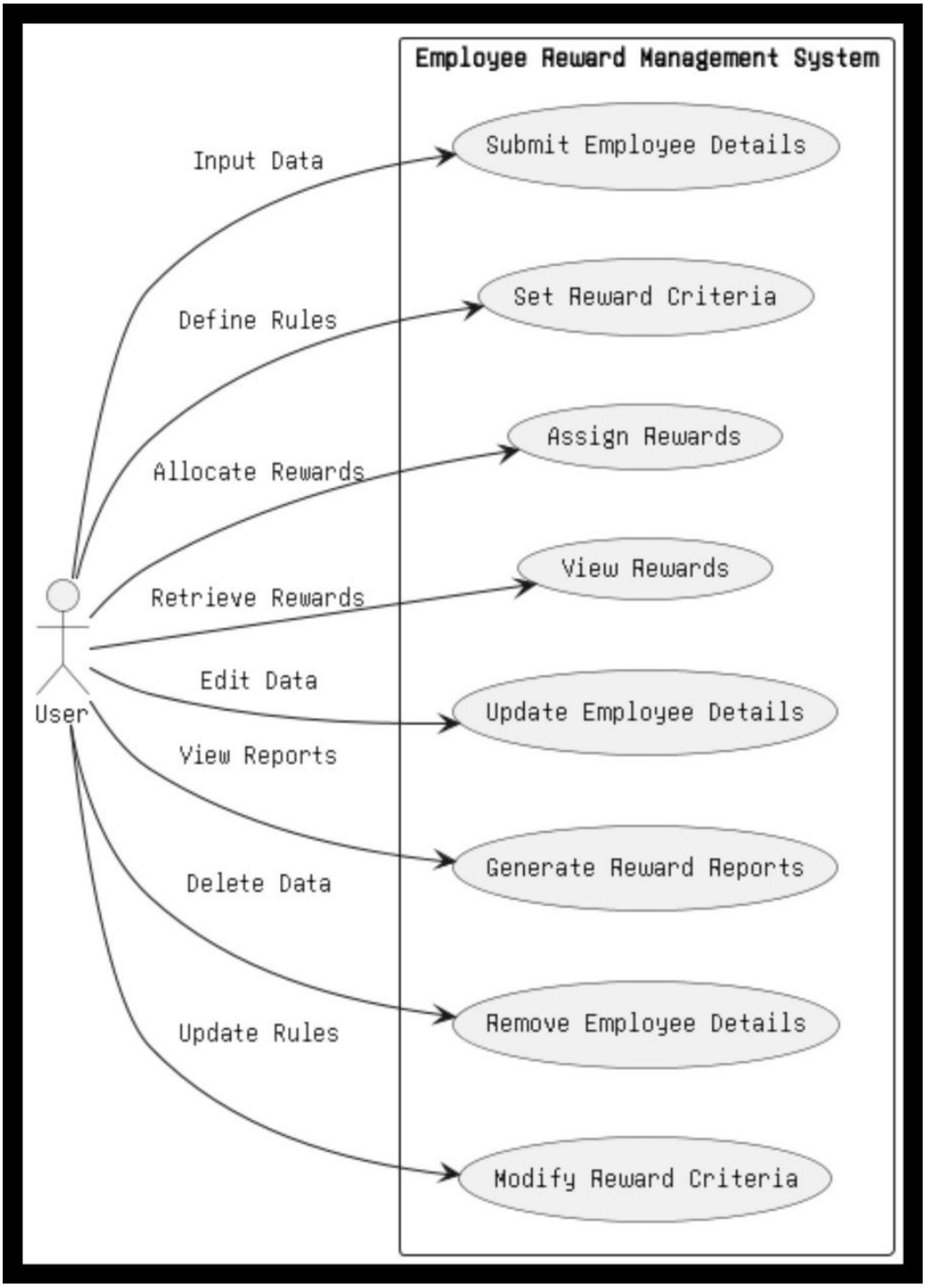
4.5 Admin Dashboard Module:

The Admin Dashboard module provides the website's admin with complete control over the website's content and user management. The module allows the admin to add products, categories, and brands. The admin can also view products, user details, manage orders, and track the website's performance.

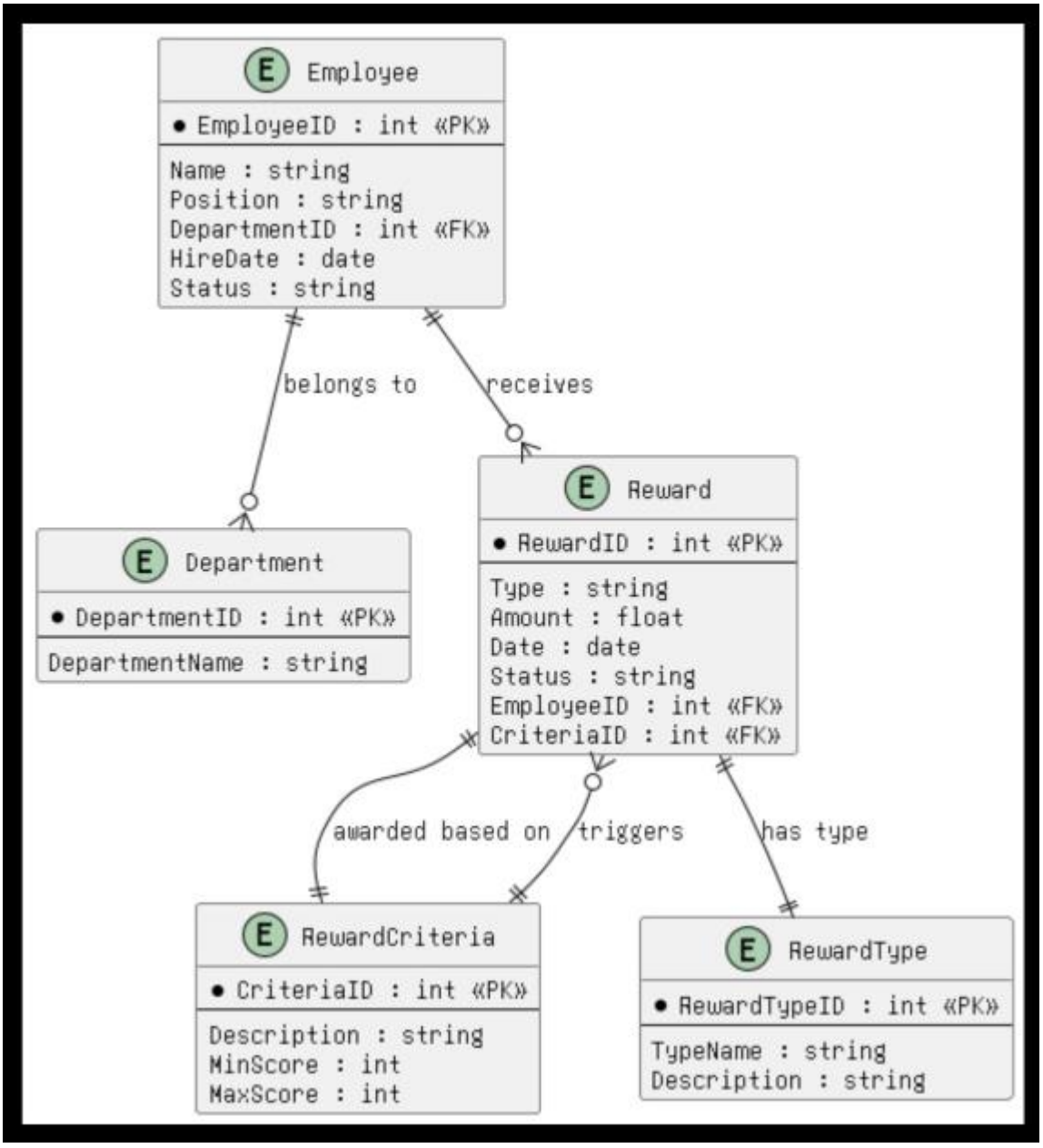
CHAPTER 5

SYSTEM DESIGN

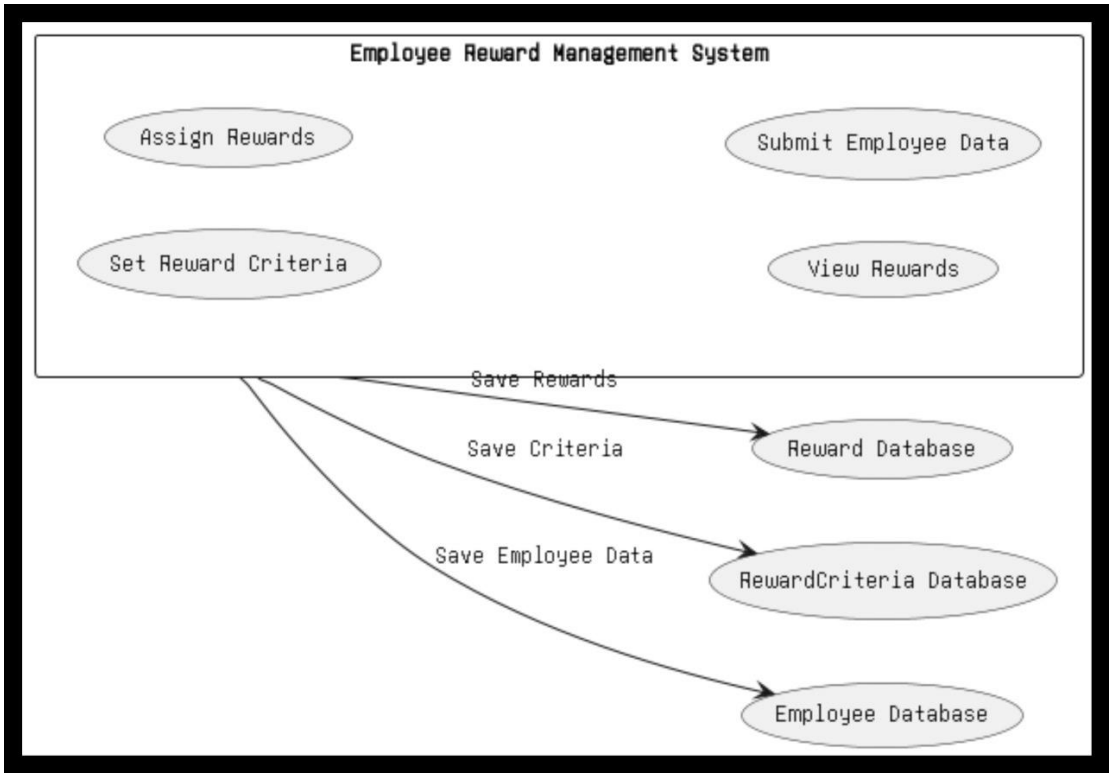
5.1 USE CASE DIAGRAM



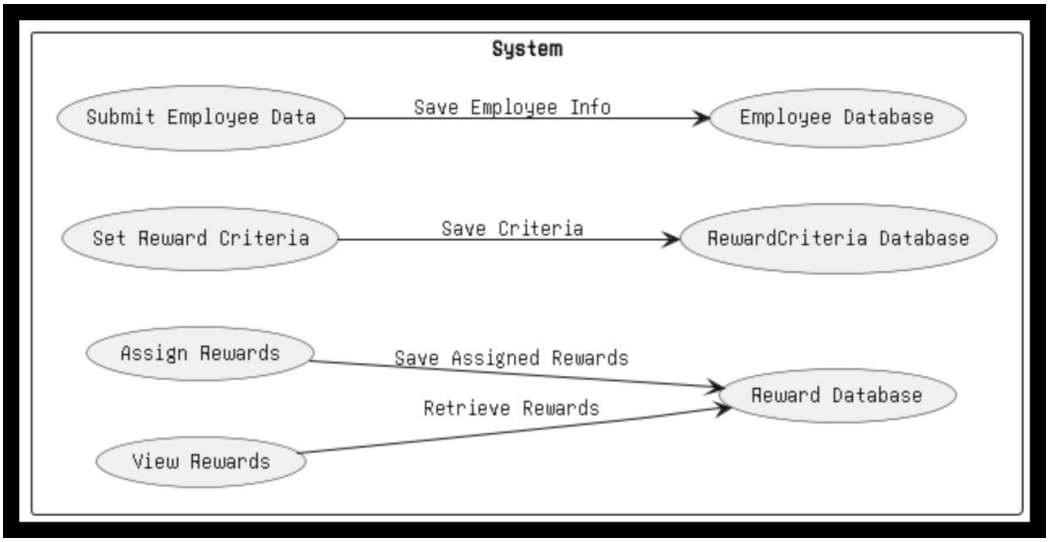
5.2 ER DIAGRAM



5.3 DFD DIAGRAM

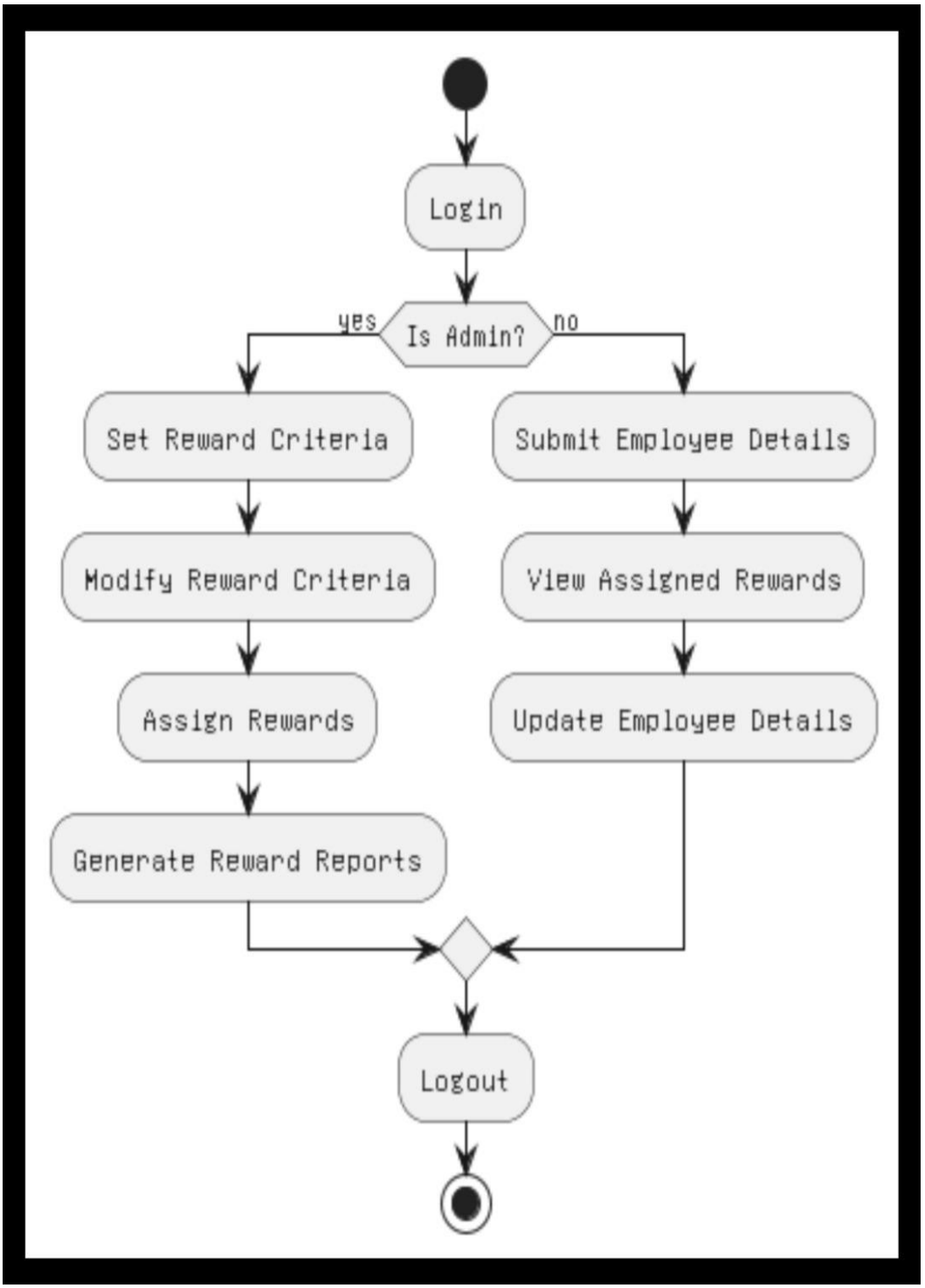


DFD Level-0 Diagram



DFD Level-1 Diagram

5.4 ACTIVITY DIAGRAM



CHAPTER 6

SAMPLE CODING

app.py

python app.py

```
from flask import Flask, render_template, request, redirect, url_for, session
import os
import mysql.connector
from mysql.connector import Error

app = Flask(_name_)
secret_key = os.urandom(24)
app.secret_key = secret_key

# Database connection function
def create_connection():
    connection = None
    try:
        connection = mysql.connector.connect(
            host='localhost',
            user='root',
            password='Falalala06??',
            database='employee_reward_management'
        )
        if connection.is_connected():
            print("Connection to MySQL DB successful")
    except Error as e:
        print(f"The error '{e}' occurred")
    return connection

@app.route('/')
def index():
    return render_template('index.html')

@app.route('/index.html')
def bindex():
    return render_template('index.html')

@app.route('/login.html')
def bogin():
    return render_template('login.html')

@app.route('/main.html')
def bain():
    return render_template('main.html')
@app.route('/about.html')
def board():
    return render_template('about.html')
```

```
@app.route('/adminlogin.html')
def bard():
    return render_template('adminlogin.html')

@app.route('/login', methods=['GET', 'POST'])
def login():
    connection = create_connection()
    cursor = connection.cursor(dictionary=True)

    if request.method == 'POST':
        email = request.form.get('username')
        password = request.form.get('password')  # You can implement password checking

        query = "SELECT * FROM employees WHERE email=%s"
        cursor.execute(query, (email,))
        user = cursor.fetchone()

        if user:
            session['user'] = user
            return redirect('/main')  # Redirect to the main page
        else:
            return "Invalid login"  # You can redirect to a login error page

    return render_template('login.html')

@app.route('/signup', methods=['POST'])
def signup():
    name = request.form.get('name')
    username = request.form.get('email')
    designation = request.form.get('designation')
    branch_office = request.form.get('branch_office')
    department = request.form.get('department')
    annual_income = request.form.get('annual_income')

    connection = create_connection()
    cursor = connection.cursor(dictionary=True)
    query = """
    INSERT INTO employees (name, email, designation, branch_office, department, annual_income, points)
    VALUES (%s, %s, %s, %s, %s, %s, %s)
    """
    data = (name, username, designation, branch_office, department, annual_income, 0)

    try:
        cursor.execute(query, data)
        connection.commit()

        # Retrieve the newly signed-up user data
        cursor.execute("SELECT * FROM employees WHERE email = %s", (username,))
        user = cursor.fetchone()

        # Store the user data in the session
        session['user'] = user

        print("User signed up successfully")
```

```
    except Error as e:
        print(f"The error '{e}' occurred")
    finally:
        cursor.close()
        connection.close()

    return redirect('/main')  # Redirect to main page after signup

@app.route('/main')
def main():
    if 'user' in session:
        user = session['user']  # Get user data from session
        return render_template('main.html', user=user)
    else:
        return redirect('/login')  # Redirect to login if not logged in


@app.route('/adminmain.html')
def admin_dashboard():
    connection = create_connection()
    cursor = connection.cursor(dictionary=True)

    employees = []
    try:
        query = """
        SELECT id, name, email, designation, branch_office, department, annual_income, points
        FROM employees
        """
        cursor.execute(query)
        employees = cursor.fetchall()
    except Error as e:
        print(f"The error '{e}' occurred")
    finally:
        if cursor:
            cursor.close()
        if connection:
            connection.close()

    return render_template('adminmain.html', employees=employees)


@app.route('/update_points', methods=['POST'])
def update_points():
    employee_id = request.form.get('employee_id')
    action = request.form.get('action')

    connection = create_connection()
    cursor = connection.cursor()

    # Define query based on action
    if action == "add":
        query = "UPDATE employees SET points = points + 1 WHERE id = %s"
    elif action == "subtract":
        query = "UPDATE employees SET points = points - 1 WHERE id = %s"
```

```
    try:
        cursor.execute(query, (employee_id,))
        connection.commit()
    except Error as e:
        print(f"The error '{e}' occurred")
    finally:
        if cursor:
            cursor.close()
        if connection:
            connection.close()

    # Redirect back to admin dashboard
    return redirect('/adminmain.html')

if _name_ == '_main_':
    app.run(debug=True)
python app.py
```

CHAPTER 7

SCREEN SHOTS

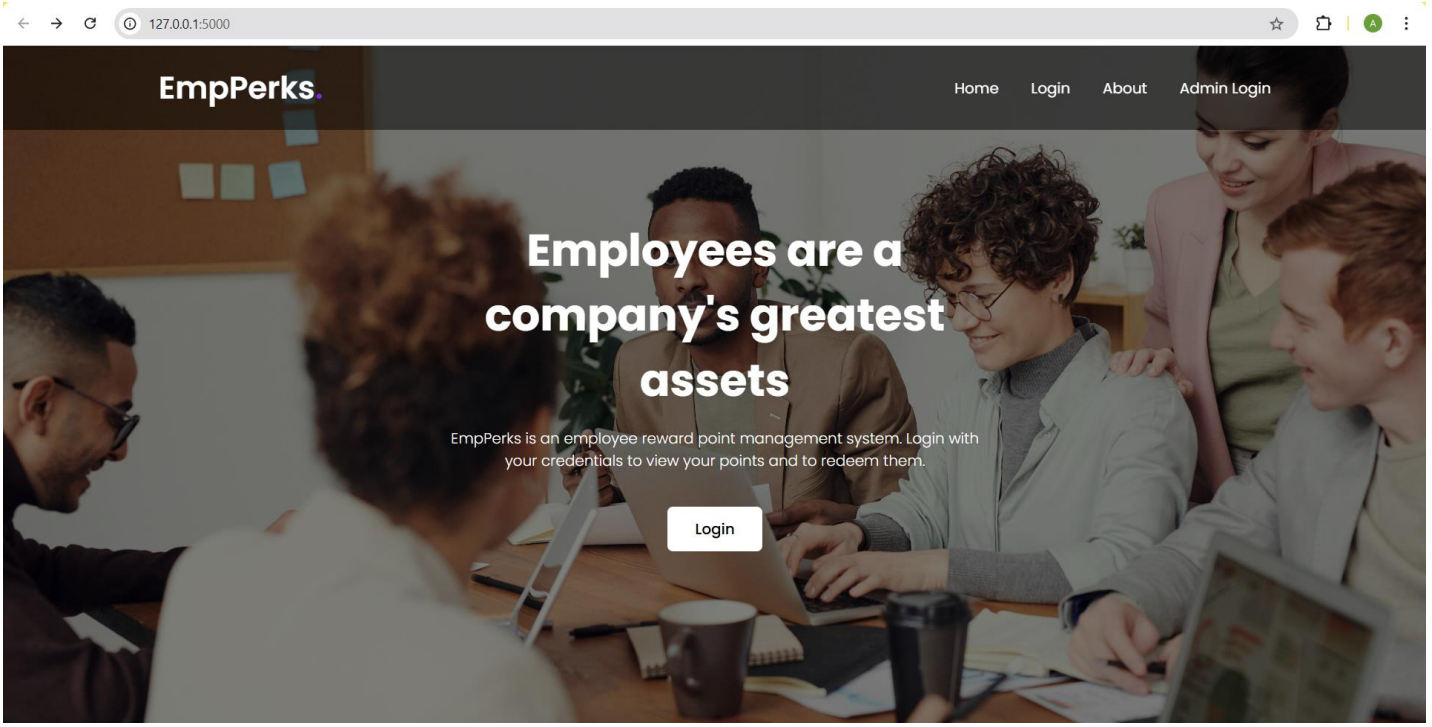


Fig. 7.1. Home Page

This is the home page where the employee can login.

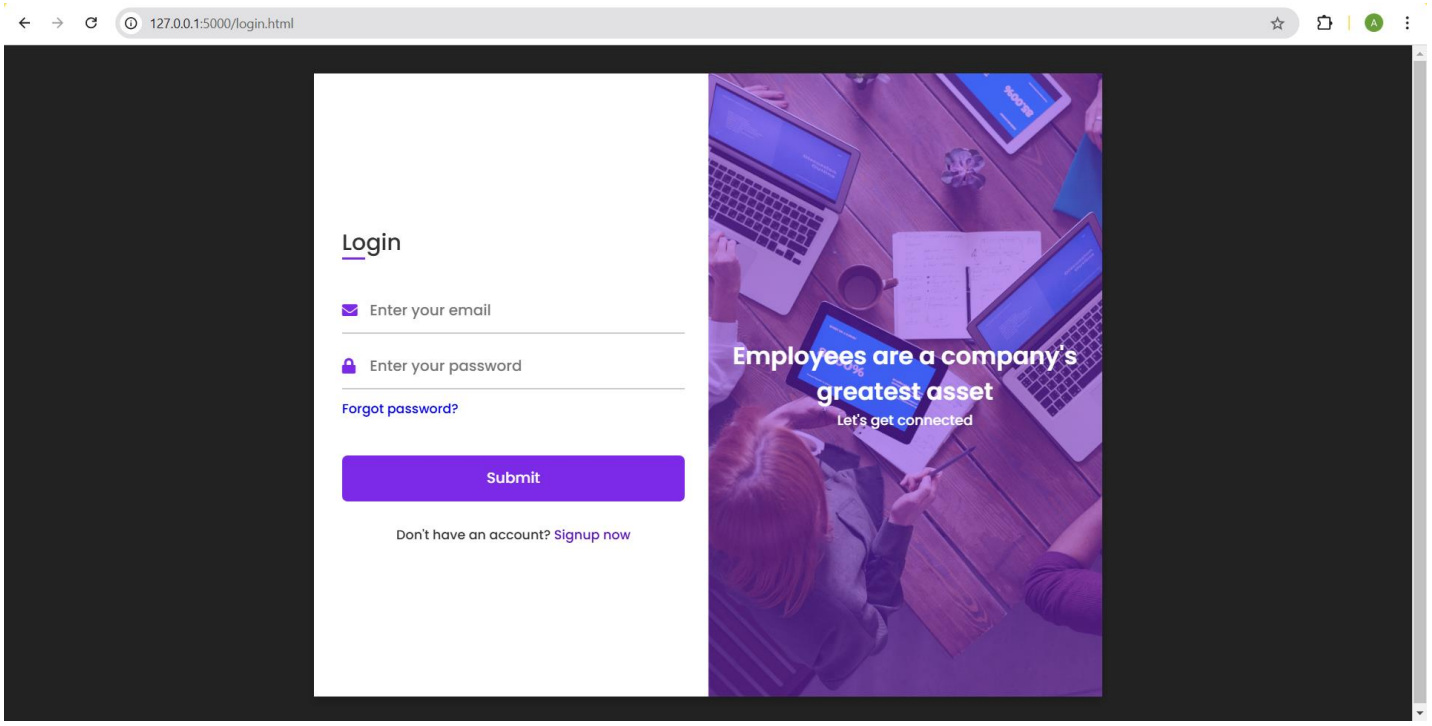


Fig. 7.2. Login page Section

Employee can login the page by using their credentials

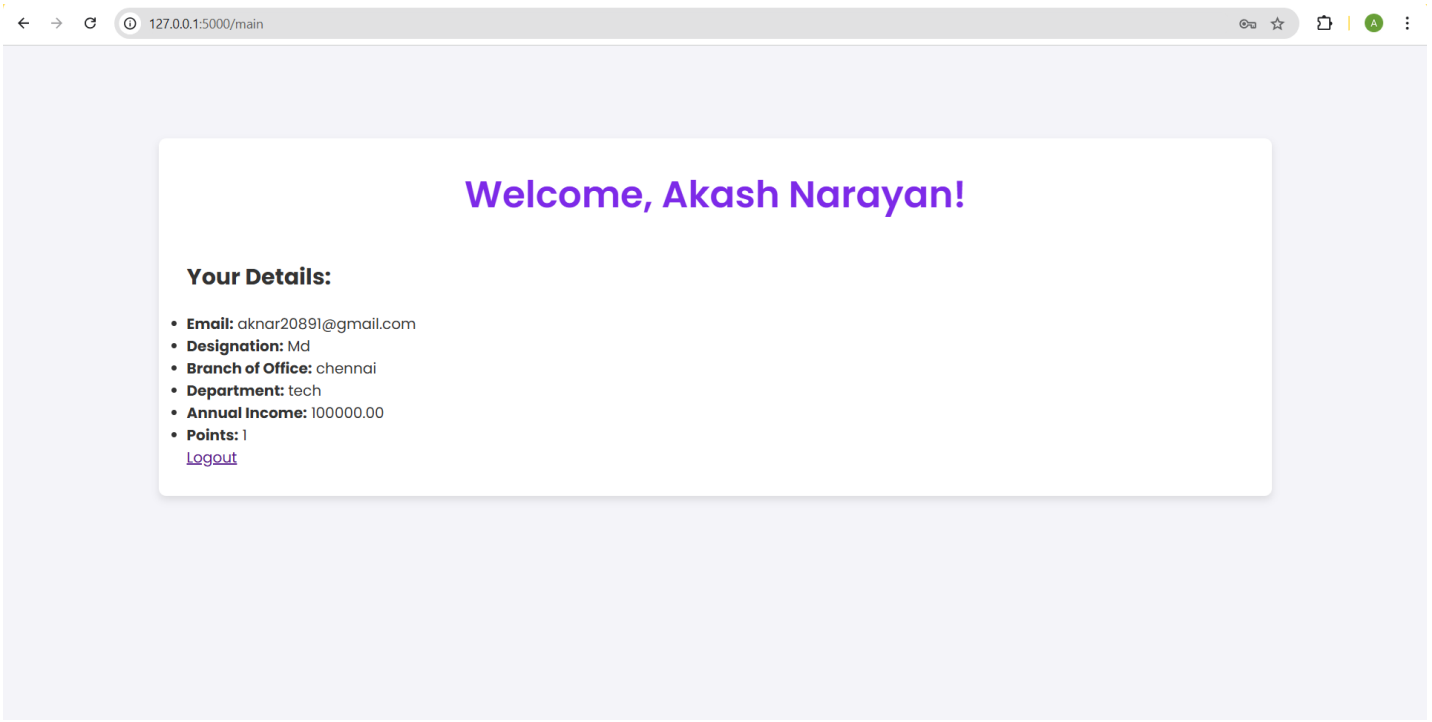


Fig. 7.3. Employee Details Page

Employee can see the detailed information about the departments.

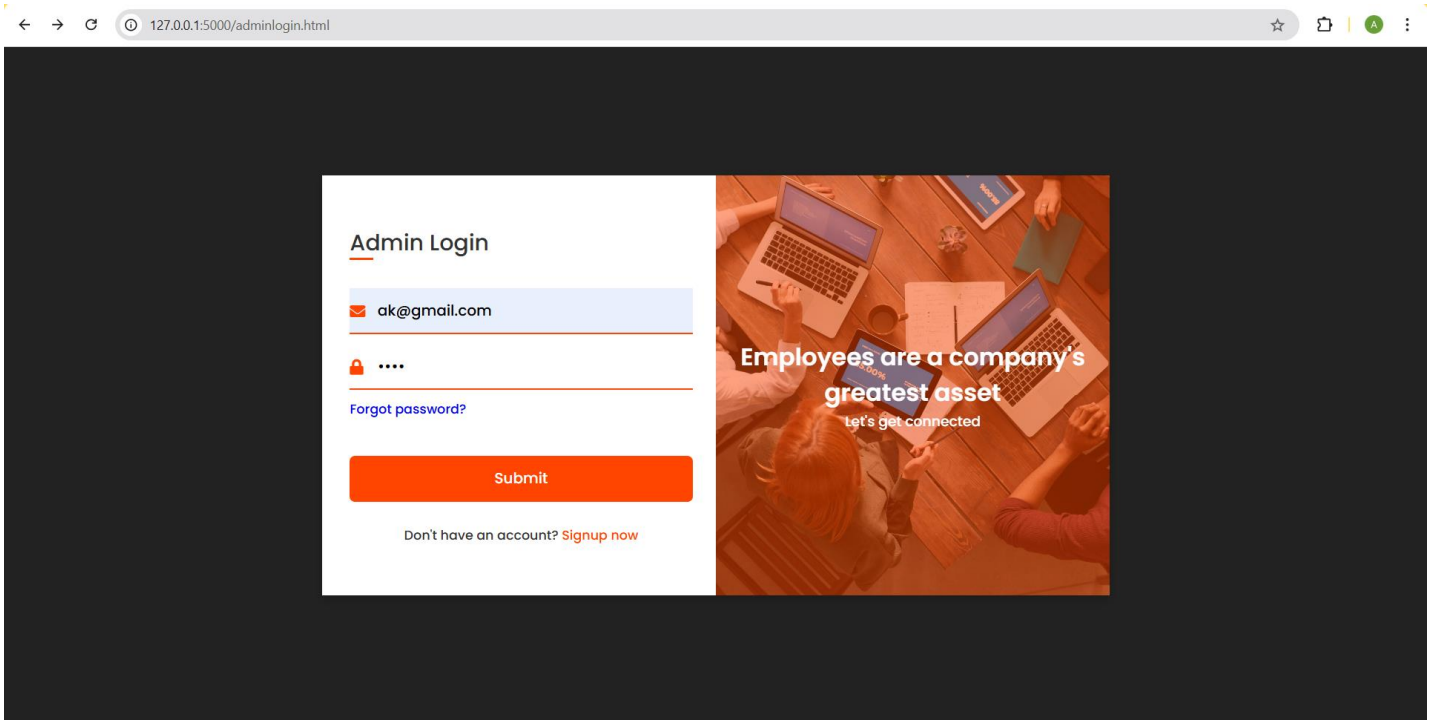


Fig. 7.4. Admin login page

Admin can login in to the site.

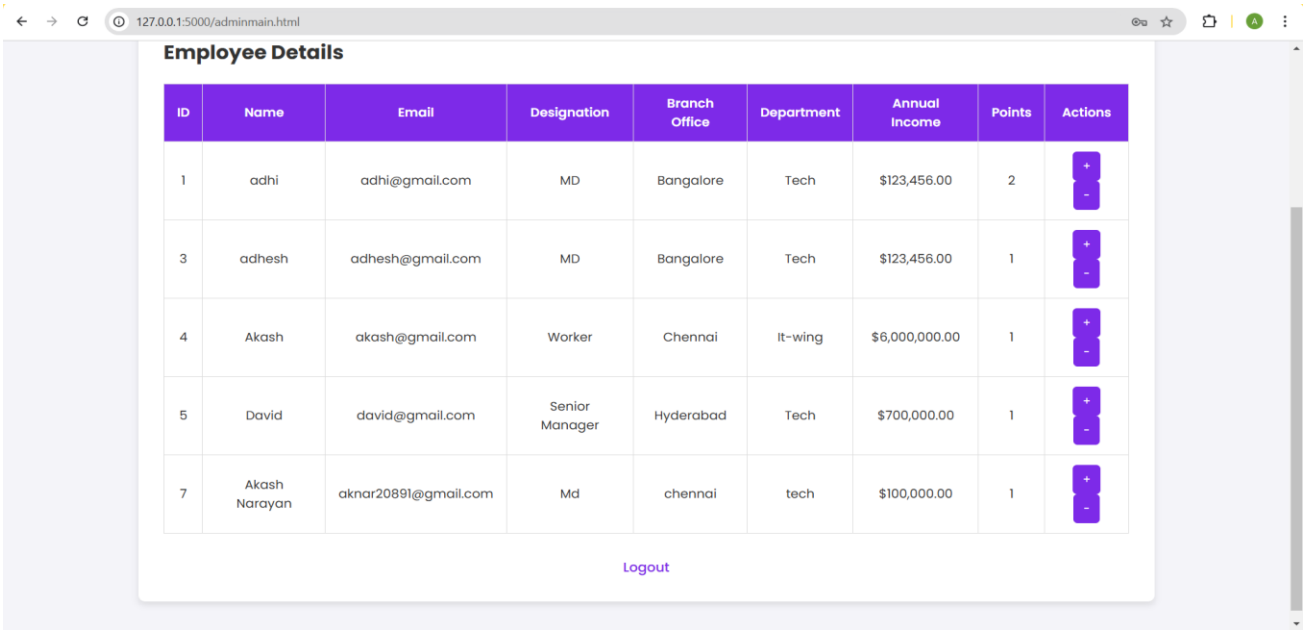


Fig. 7.5. Admin access page

Admin can see detailed information about employees.

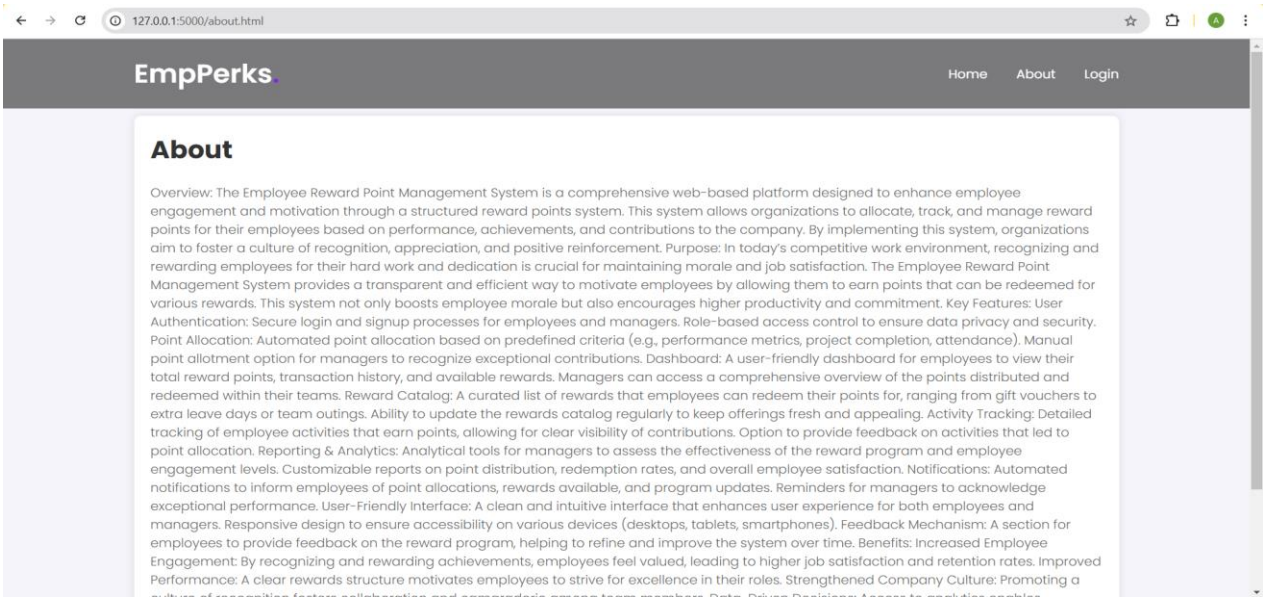


Fig. 7.6. About page section

From here you can access the about section.

CHAPTER 8

8.1 CONCLUSION

In conclusion, the Reward Management System is a valuable tool for modern organizations looking to streamline their employee performance tracking and reward processes. By automating the evaluation and reward distribution, the system enhances transparency, ensures fairness, and improves overall employee engagement and satisfaction.

Implementing a reward management system enables organizations to maintain a consistent and objective approach to recognizing employee achievements, which in turn boosts morale and productivity. As businesses continue to adapt to the evolving work environment, embracing such systems can significantly enhance operational efficiency and employee retention.

In the future, the Reward Management System can be further enhanced by incorporating advanced analytics to better assess employee performance trends and predict future reward needs. Integration with AI and machine learning could automate performance assessments, making the process even more efficient and personalized. Additionally, mobile accessibility could be improved to allow employees to track their own performance and rewards in real-time, enhancing user experience. These advancements would not only streamline the reward process but also foster a more engaged and motivated workforce, ensuring that organizations remain competitive and employee-centric in the years to come.

REFERENCES

HTML , CSS , JS – www.w3schools.com

FLASK, MYSQL – www.youtube.com

Product Details– www.amazon.in

Carousel Slider – www.glidejs.com

Font Awesome Icons – www.fontawesome.com