Human Resource Management System (HRMS)

1. Introduction

The Human Resource Management System (HRMS) is a modular, web-based application developed using Python's Flask framework. It enables organizations to manage their employee information, attendance tracking, leave applications, and basic HR functions in a centralized platform. It ensures data consistency, real-time updates, and ease of access.

2. Objective

To streamline and automate HR operations such as:

• Employee data management

• Attendance logging

• Leave application and approval

• Profile updates and role-based access

This system eliminates manual processes, reduces errors, and increases operational efficiency.

3. Core Features

- Authentication System

 o Role-based login (Admin/Employee)

 o Password recovery/reset

- Dashboard

 o Admin: Overview of employees, leave stats, and attendance

 o Employee: View attendance, leaves, and announcements

- Employee Management

o Admin: Add, edit, or delete employees

o Employees: Update personal profile

• Attendance Management

o Check-in/Check-out with timestamp

o IP-based logging (optional)

o Picture capture on check-in (optional)

o Manual attendance edit (admin)

o Monthly attendance calendar

• Leave Management

o Apply/Track leave by type (Casual, Sick, Paid, Unpaid)

o Admin: Approve/Reject/Report leaves

4. Technologies Used

Frontend: HTML5, CSS3, JavaScript, Bootstrap

Backend: Python 3.x (Flask Framework)

Database: SQLite (easy migration to MySQL)

Templates: Jinja2 templating engine

ORM: SQLAlchemy

5. Project Directory Structure

/HRMSManager/

├── app.py   # Entry point of the application

├── routes.py   # Flask routes (endpoints)

├── models.py   # SQLAlchemy database models

├── forms.py   # Flask-WTF form classes

```
├── utils.py          # Utility/helper functions

├── setup_mysql.py     # Initial database setup

├── /static/         # CSS, JS, images

├── /templates/      # HTML templates using Jinja2

└── requirements.txt   # Python package dependencies
```

6. Database Models

• User: Stores user profile (name, email, phone, department, role, etc.)

• Attendance: Tracks daily time logs with timestamps and image if captured

• Leave: Stores leave type, reason, status (Pending, Approved, Rejected)

7. Roles and Permissions

Role   Permissions

Admin Manage users, attendance, leaves, announcements, and dashboard stats

Employee   View dashboard, mark attendance, apply for leave, update personal profile

8. How to Set Up the Project

Prerequisites:

• Python 3.9 or higher

• pip

• Virtualenv (optional but recommended)

Setup Instructions:

Step 1: Create virtual environment

python -m venv venv

```
# Step 2: Activate virtual environment

source venv/bin/activate  # macOS/Linux

venv\Scripts\activate    # Windows


# Step 3: Install dependencies

pip install -r requirements.txt


# Step 4: Setup database (run only once)

python setup_mysql.py


# Step 5: Start the application

python app.py
```

Then visit: http://127.0.0.1:5000 in your browser.

9. Future Enhancements

• Face recognition-based attendance

• SMS/Email alerts for leave updates

• Payroll system integration

• Analytics dashboard (charts/graphs)

• Multi-language and timezone support

• Two-factor authentication

10. Conclusion

The HRMS project provides a full-stack web application with role-based control and a clean, responsive interface. It is designed to support future upgrades like facial
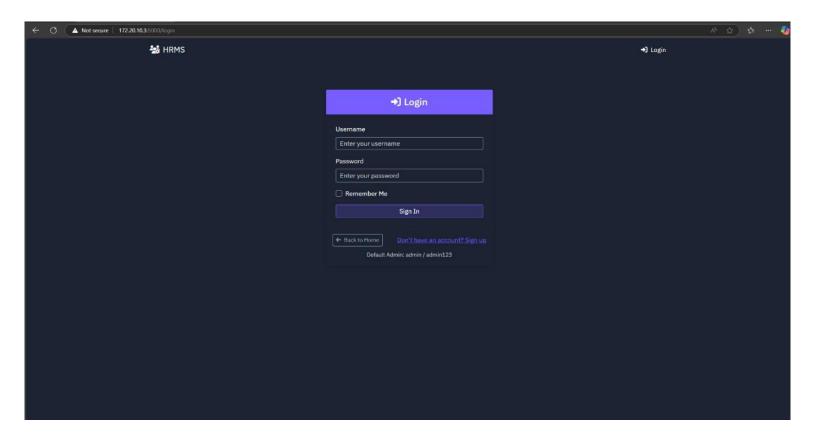
recognition or notification systems. This solution serves as an effective starting point for any modern HR digital system.
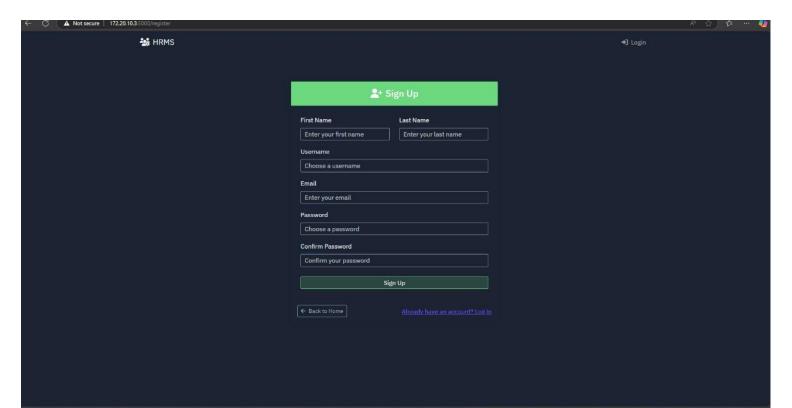
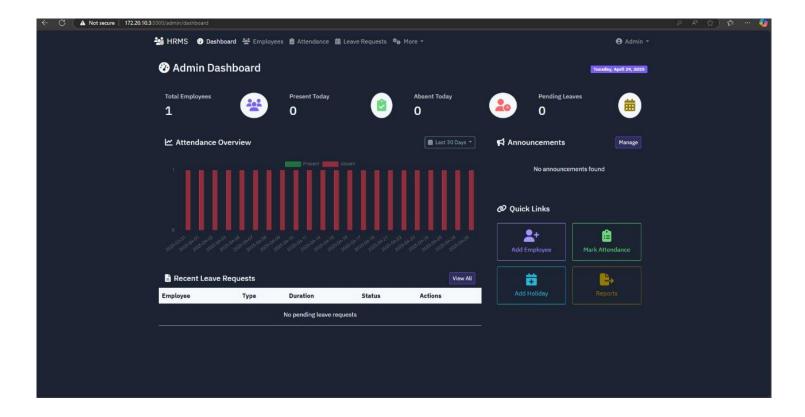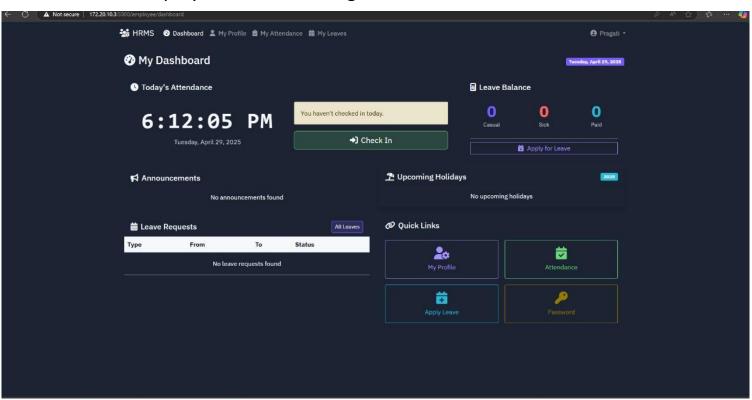## 11. Screenshots of the project

1. Welcome Page.

2. Login Page

3. Sign-up Page

4. Admin Dashboard Page

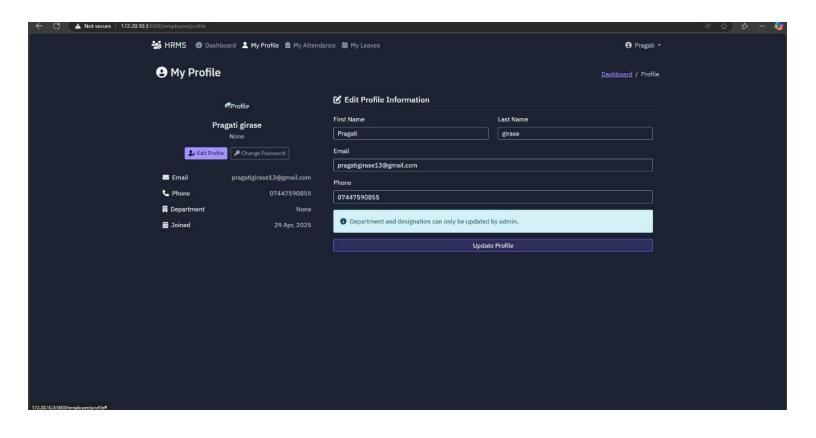## 5. Employee Dashboard Page

6. My Profile Page

7. Employee Management Page