Leave Management System

Capstone Project Case Study

By

HARSHITA SRI R

Leave Management System

Introduction

Traditional leave management processes, often reliant on physical forms, scattered email chains, or error-prone spreadsheets, present significant operational inefficiencies and a notable lack of transparency for modern organizations. This manual and fragmented approach typically results in prolonged approval cycles, frequent miscommunication between staff and management, and considerable administrative overhead in tracking and calculating leave balances. Furthermore, the absence of a centralized, auditable record system complicates reporting and compliance.

The development of this system followed a structured methodology, beginning with a thorough analysis of user requirements and culminating in a functional prototype. Key considerations throughout the process included ensuring data security through role-based access control, designing an intuitive user experience to encourage adoption, and building a scalable architecture to accommodate future enhancements such as calendar integrations, automated balance accruals, and multi-level approval chains.

Abstract

This report details the development of a web-based Leave Management System (LMS) designed to replace inefficient, manual leave-tracking methods. The system provides a centralized digital platform featuring distinct interfaces for employees and administrators. Employees can seamlessly submit leave requests and track their status and balances, while administrators can review, approve, reject, or hold applications through a dedicated dashboard. By automating the entire workflow, the LMS eliminates paperwork, reduces errors, ensures policy compliance, and provides real-time transparency. The project demonstrates a functional solution that enhances operational efficiency, improves communication, and simplifies leave administration for modern organizations.

Project Objectives:

1. **Automate Core Workflow:** To digitalize the end-to-end leave application and approval process, eliminating reliance on physical forms, emails, or spreadsheets.
2. **Enhance User Experience:** To provide an intuitive interface for employees to easily submit leave requests and view their real-time status and balance history.
3. **Empower Administrative Control:** To equip managers/admins with a centralized dashboard to efficiently review, approve, reject, or hold pending leave applications and manage user accounts.
4. **Ensure Accuracy & Transparency:** To maintain an accurate, automated record of all leave transactions, providing clear audit trails and preventing disputes over balances and approvals.
5. **Establish a Scalable Foundation:** To build a modular system with a secure REST API backend that can be extended with future features like calendar integration, multi-level approvals, and detailed reporting.

Client Requirements

Based on initial stakeholder consultations and analysis of existing manual processes, the following functional and non-functional requirements were defined for the Leave Management System (LMS).

Functional Requirements (What the system must do)

1. User Authentication & Role Management:

- The system must allow new users to register with details (name, email, password, role).
- The system must allow registered users to log in and log out securely.
- The system must enforce two distinct user roles: **Employee** and **Manager (Admin)**.
- System functionality and data access must be determined by the user's role.

2. Employee Portal Requirements:

- An employee must be able to submit a new leave application, providing details such as leave type, start date, end date, and reason.
- An employee must be able to view a list of all their own past and current leave requests.
- An employee must be able to see the current status (e.g., Pending, Approved, Rejected) of each request.

3. Manager/Admin Portal Requirements:

- A manager must be able to view a dashboard listing **all** leave requests from all employees.
- A manager must have the ability to **approve** or **reject** any pending leave request.
- A manager must be able to view and manage all user accounts (view list, update details, delete users).

Technologies and Tools Used

- **Programming Language:** Java – used for backend logic and application development
- **Framework:** Spring Boot – used to build the web application with MVC architecture and embedded Tomcat server
- **Database Connectivity:** JDBC (Java Database Connectivity) – used to connect the application with the MySQL database and perform CRUD operations
- **Database:** MySQL – used as the relational database to store employee and user information
- **Frontend Technologies:** HTML and CSS – used to design and style the user interface
- **IDE / Development Tool:** Spring Tool Suite (STS) / Eclipse – used for coding, debugging, and project management
- **Server:** Embedded Apache Tomcat – used to deploy and run the web application
- **Build Tool:** Maven – used for dependency management and project build automation
- **Browser:** Google Chrome – used for testing and accessing the application

Software Requirements:

- Operating System: Windows / Linux
- Programming Language: Java (JDK 8 or above)
- Framework: Spring Boot
- Database: MySQL
- Database Connectivity: JDBC (Java Database Connectivity)
- Frontend Technologies: HTML, CSS
- IDE / Development Tool: Spring Tool Suite (STS) / Eclipse
- Build Tool: Maven
- Server: Embedded Apache Tomcat
- Web Browser: Google Chrome

System Modules

The Leave Management System is architecturally divided into four core modules, each handling a distinct set of functionalities.

Module 1: Authentication & Authorization Module

- **Purpose:** To manage user identity and control system access.
- **Key Functions:**
  - User Registration (for Employees and Managers).
  - User Login and Logout (Session Management).
  - Role-based permission enforcement across the application.

Module 2: User Management Module (Admin)

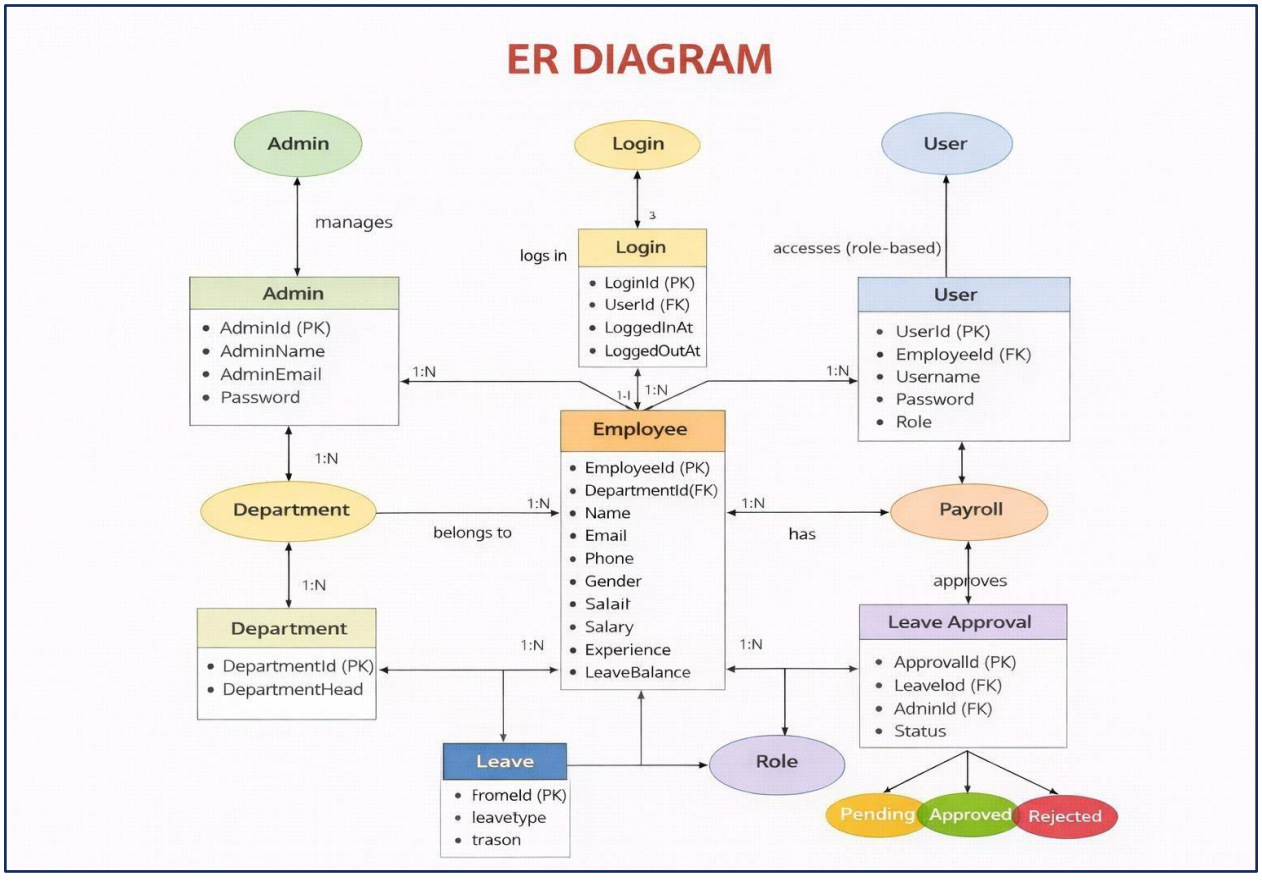
- **Purpose:** To allow administrators to oversee and manage all user accounts.
- **Key Functions:**
  - View a complete list of all system users.
  - Update user profiles (e.g., name, email, role).
  - Delete user accounts from the system.

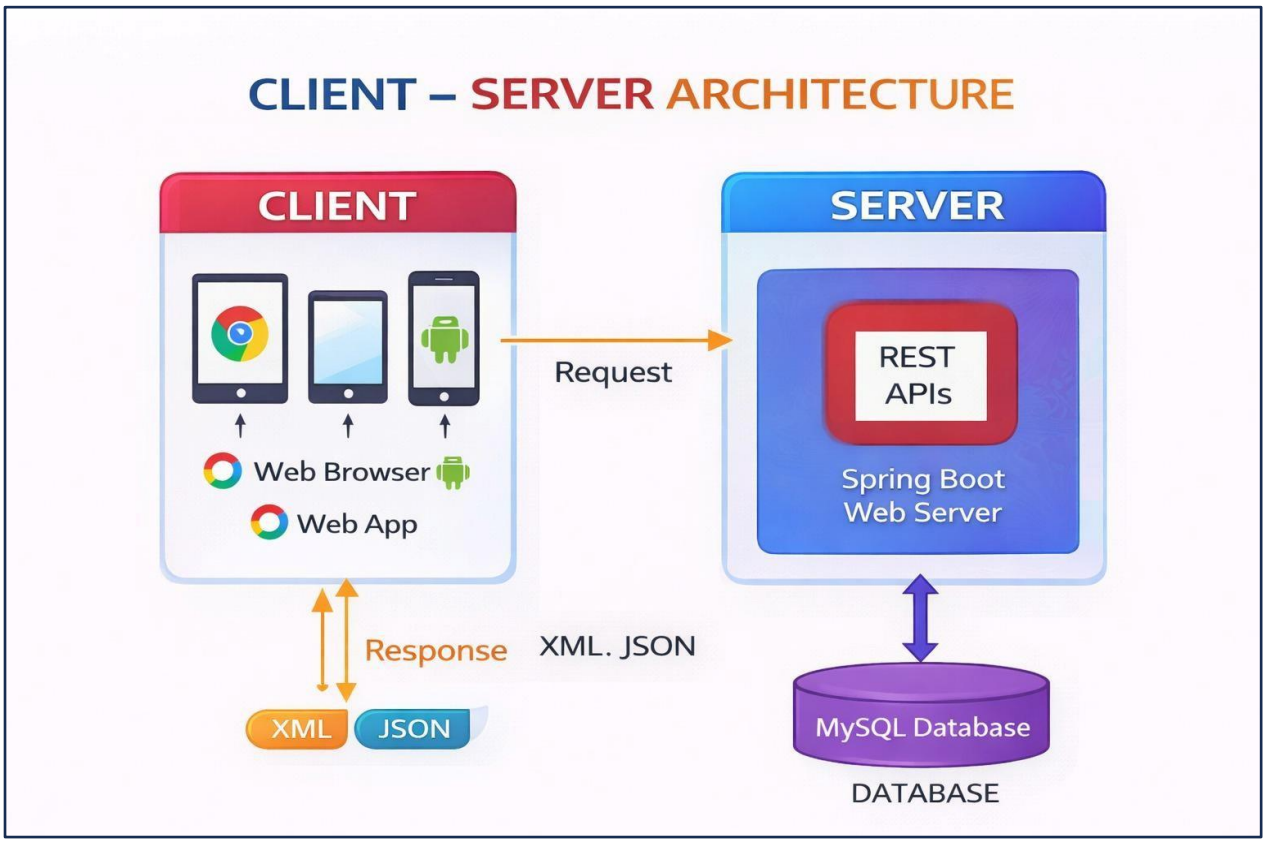
Module 3: Employee Leave Module

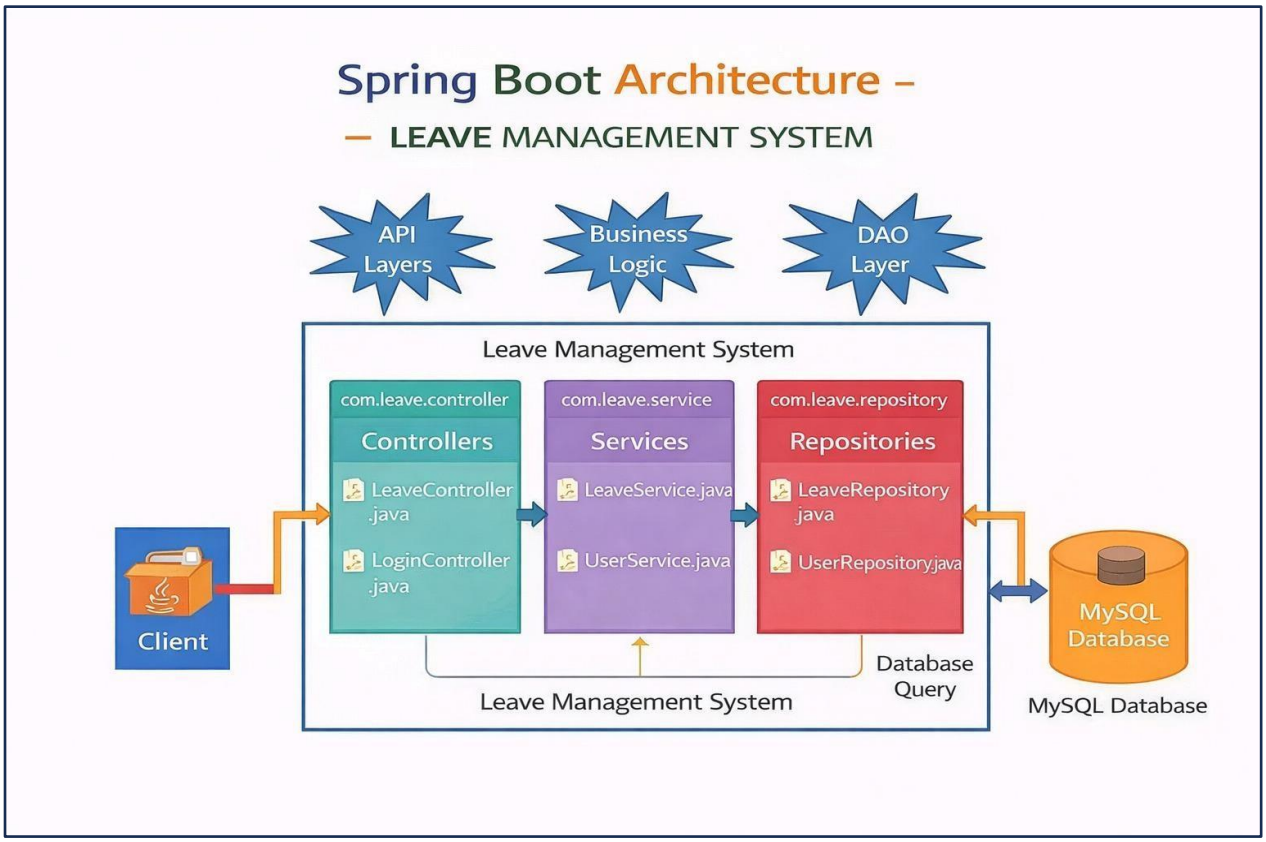
- **Purpose:** To provide employees with self-service leave management.
- **Key Functions:**
    - Submit a new leave application form.
    - View a personal dashboard listing all their own leave requests (history and current status).

Module 4: Manager Leave Dashboard Module

- **Purpose:** To provide managers with a control panel for overseeing the leave workflow.
- **Key Functions:**
    - View a centralized list of **all** leave requests from every employee.
    - Review pending requests and take action: **Approve** or **Reject**.




CLIENT – SERVER ARCHITECTURE
CLIENT
SERVER
Request
REST APIs
Spring Boot Web Server
Web Browser
Web App
Response
XML. JSON
XML
JSON
MySQL Database
DATABASE


Spring Boot Architecture –
– LEAVE MANAGEMENT SYSTEM
API Layers
Business Logic
DAO Layer
Leave Management System
com.leave.controller
Controllers
LeaveController .java
LoginController .java
com.leave.service
Services
LeaveService.java
UserService.java
com.leave.repository
Repositories
LeaveRepository .java
UserRepository.java
Client
Database Query
Leave Management System
MySQL Database
MySQL Database

Manager Module

The Manager Module is a secure, role-restricted control panel that empowers managers to oversee the entire leave ecosystem. It features a comprehensive User Management Dashboard to view, edit, and delete all system users, and a Leave Administration Dashboard to review every leave request, with direct action buttons to approve or reject applications. Additionally, it includes tools for configuring leave policies and generating organizational reports, all accessed through a centralized, intuitive interface that consolidates full administrative control.

Employee Module

The Employee Module is a self-service portal designed for staff to manage their leave simply and efficiently. It centers on a user-friendly Leave Application Form for submitting new requests and a personalized "My Leave" Dashboard to track the real-time status of all their applications. This focused interface provides employees with complete transparency and control over their own leave process

HTTP / HTTPS Request Methods

Login Module

HTTP Method	URL	Description
POST (Employee/Manager)	http://localhost:8084/auth/register	Registers a new user
POST	http://localhost:8084/auth/login	Authenticates user and starts session
GET	http://localhost:8084/auth/logout	Logout the current user

User Management Module

HTTP Method	URL	Description
GET	http://localhost:8084/users/all	Retrives all registered users
POST	http://localhost:8084/users/update	Updates user details
POST	http://localhost:8084/users/change-password	Changes password for logged in users

Leave Management Module - Employee

HTTP Method	URL	Description
POST	http://localhost:8084/leave/apply	Employee applies for leave
GET	http://localhost:8080/leave/my leaves	Retrives logged in employees leave

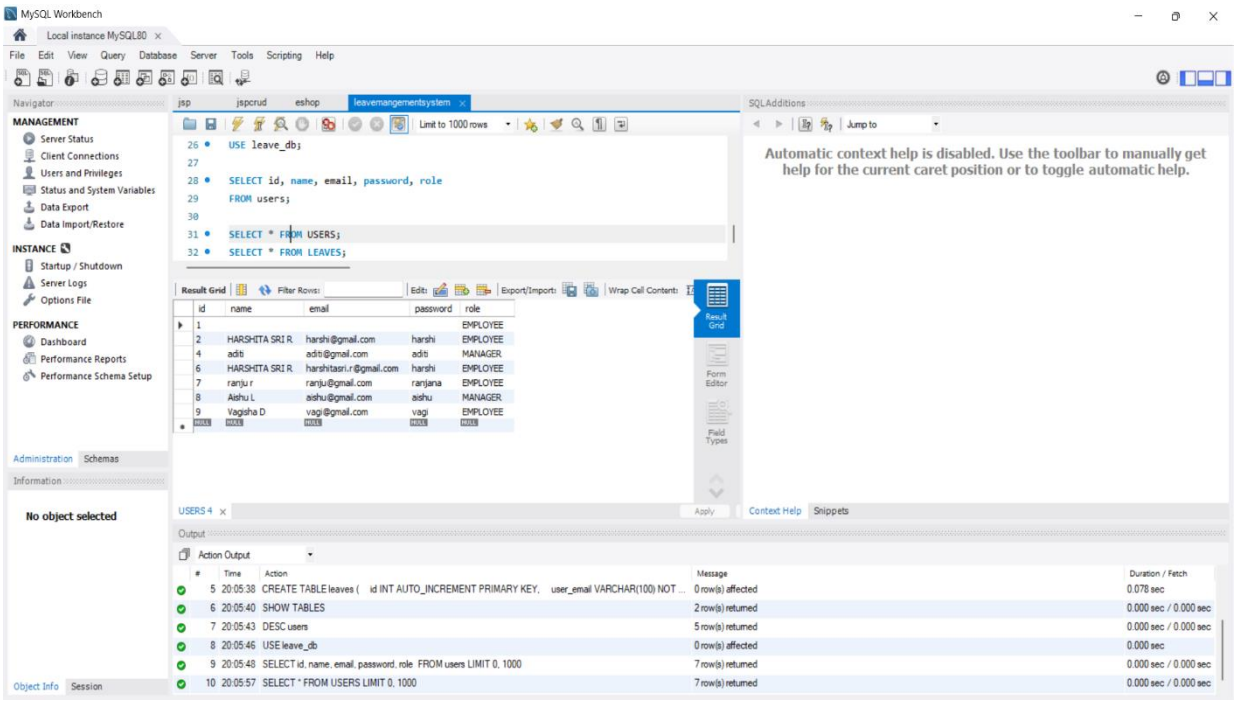
Leave Management Module - Manager

HTTP Method	URL	Description
GET	http://localhost:8084/leave/all	Reterives all leave requests
GET	http://localhost:8084/leave/approve	Approves a leave requests

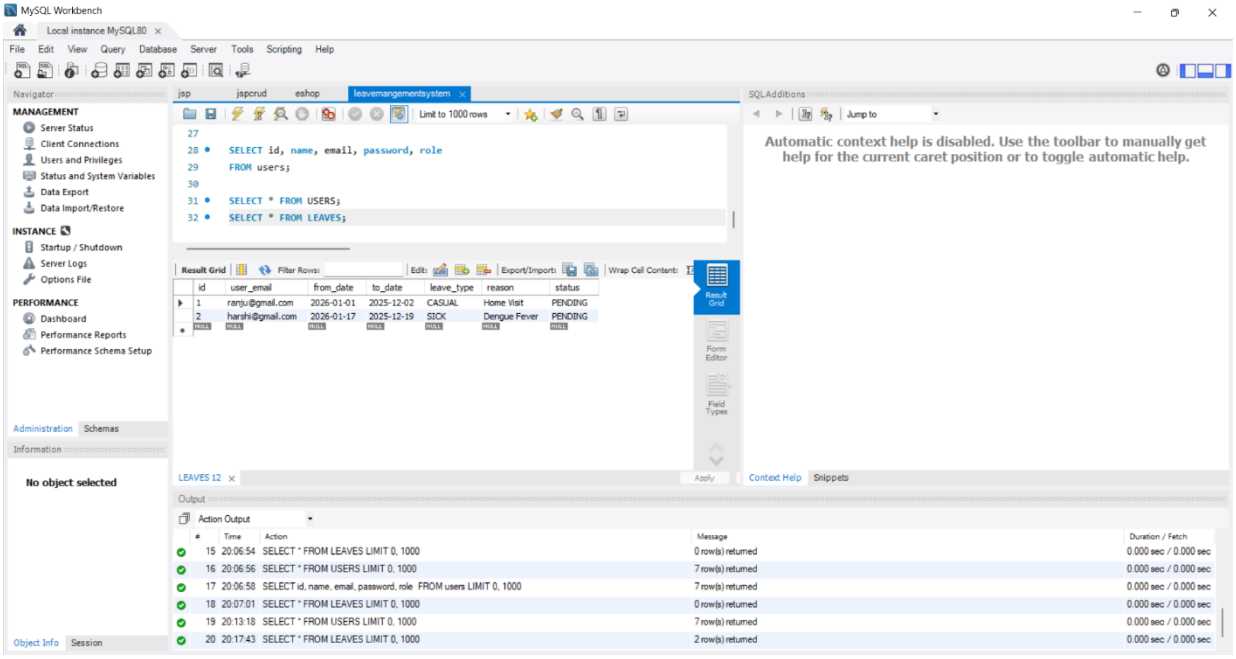
DATA DICTIONARY

DATABASE

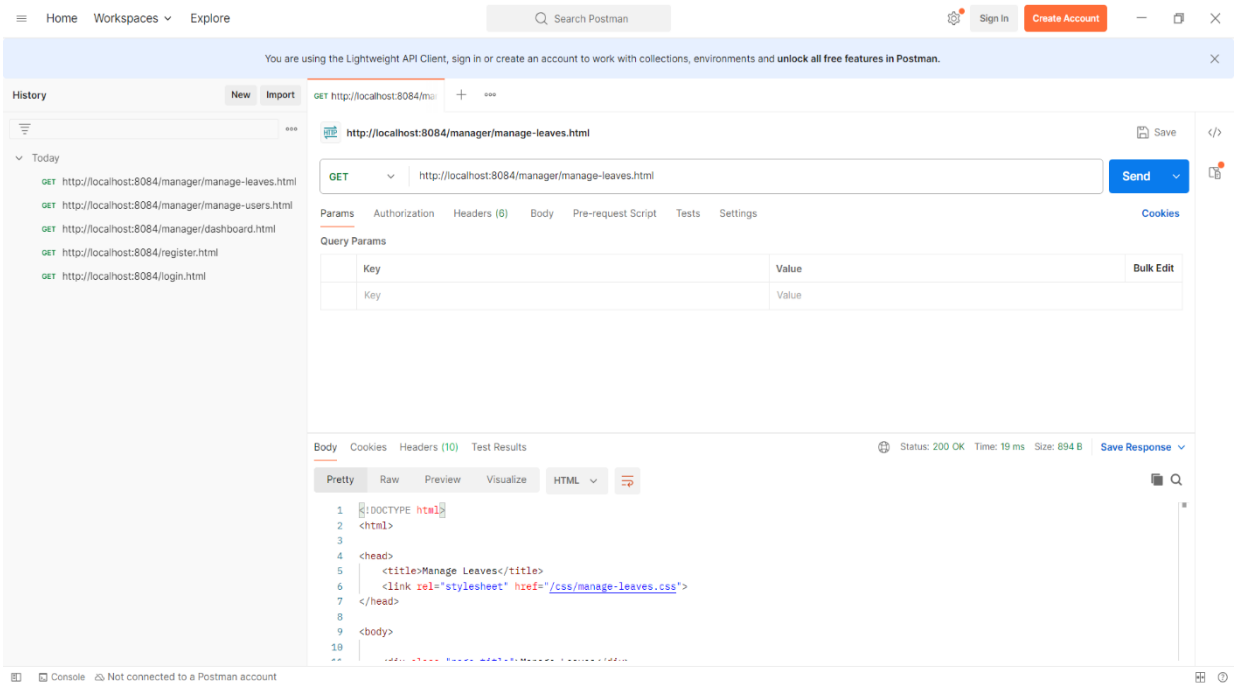
MANAGER / EMPLOYEE DATABASE



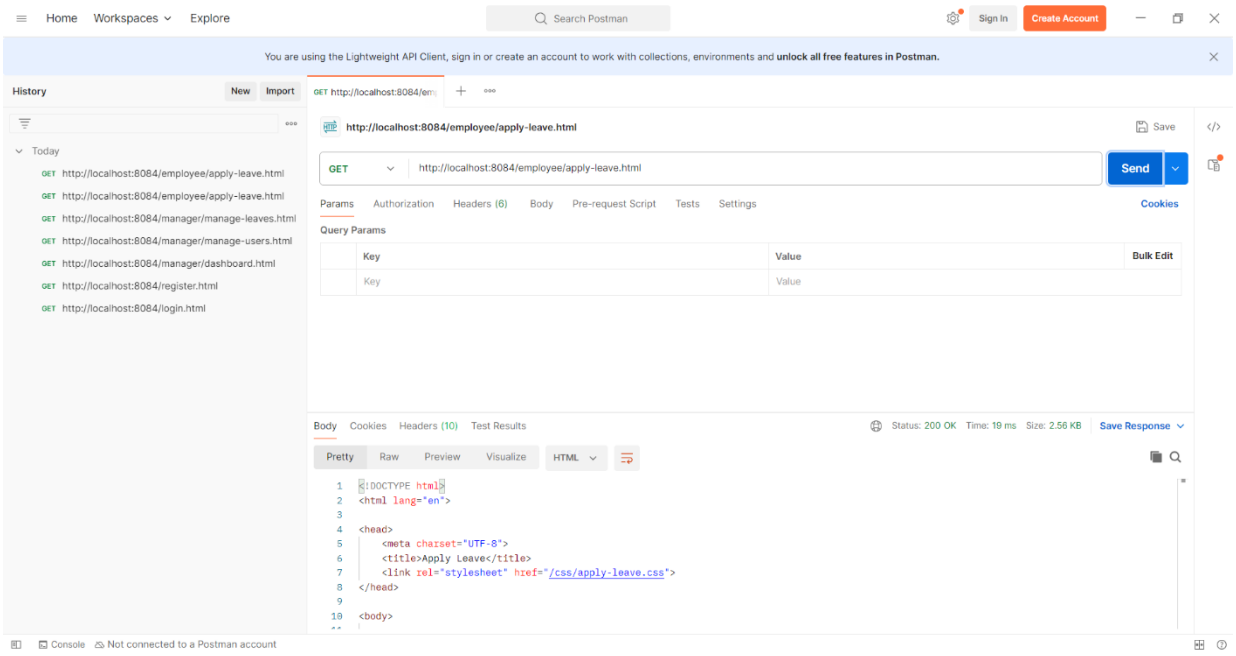
EMPLOYEE LEAVE DATABASE



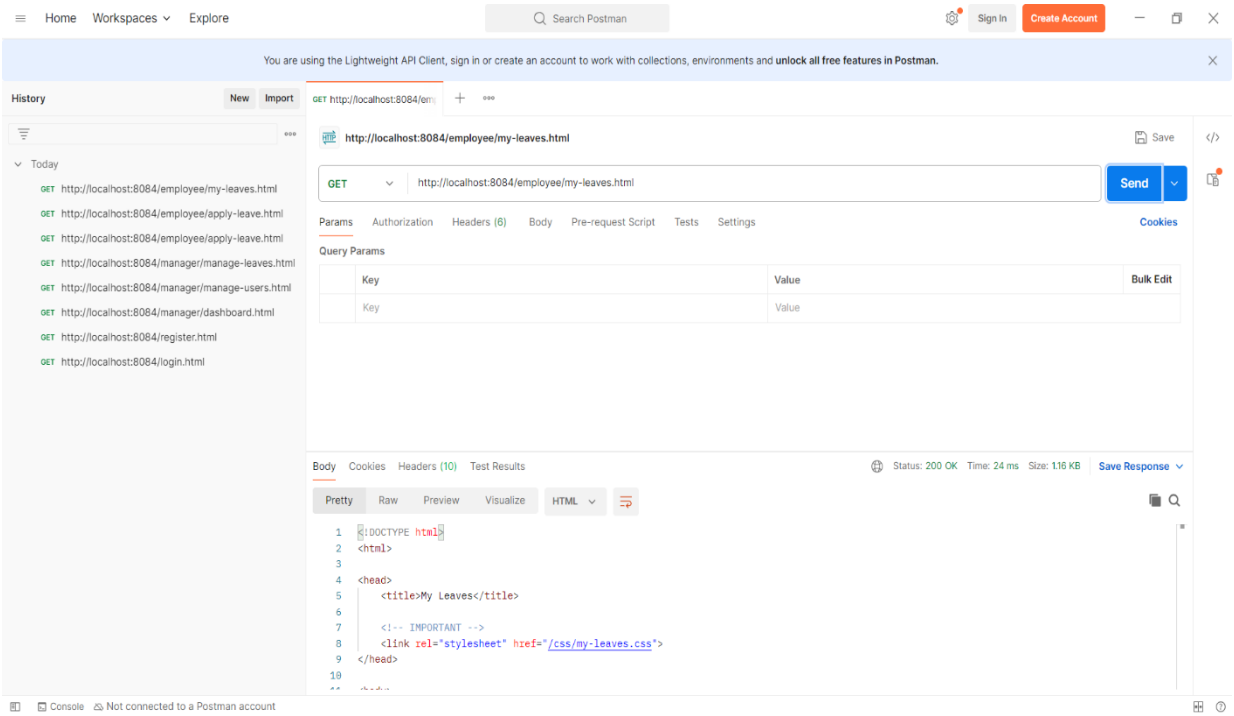
GET METHOD ON MANAGING LEAVES



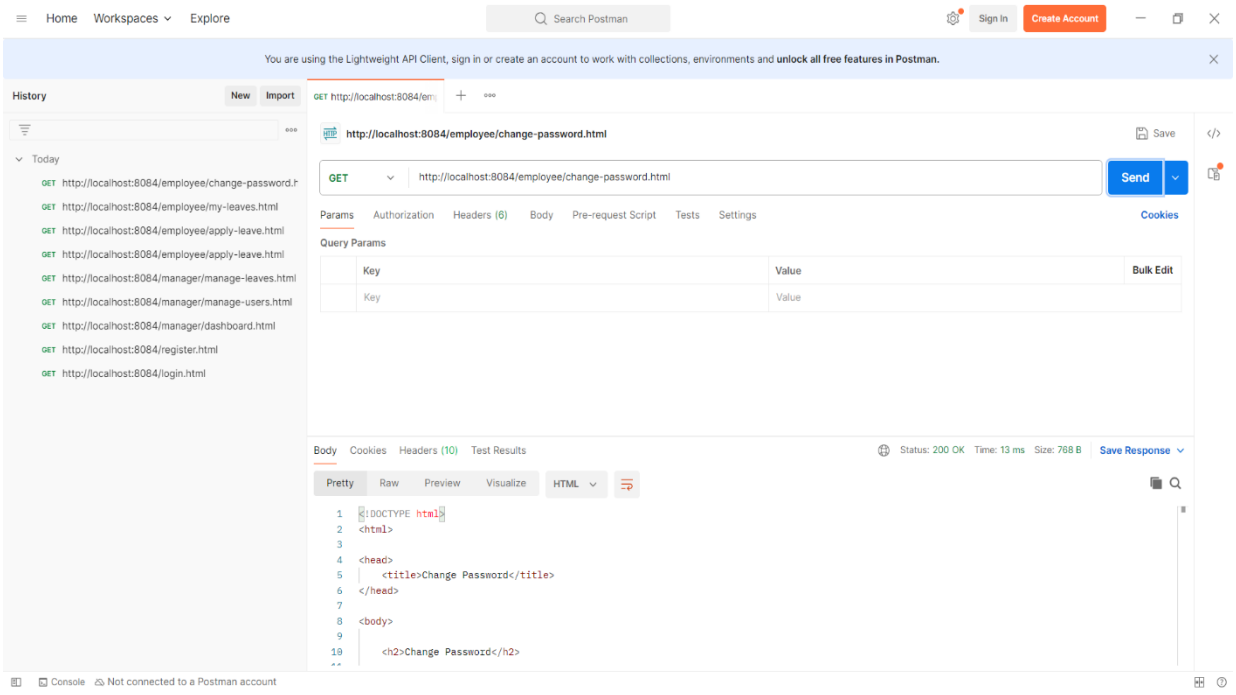
POST METHOD ON APPLYING LEAVES



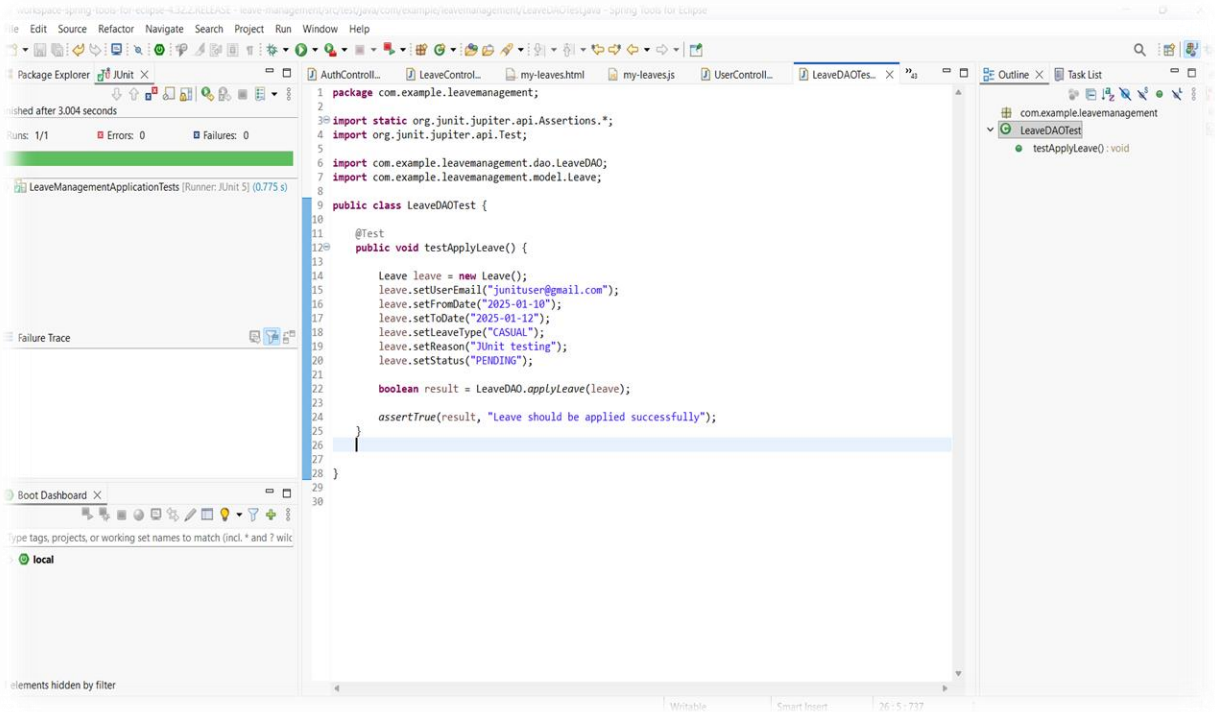
GET METHOD ON MY LEAVES



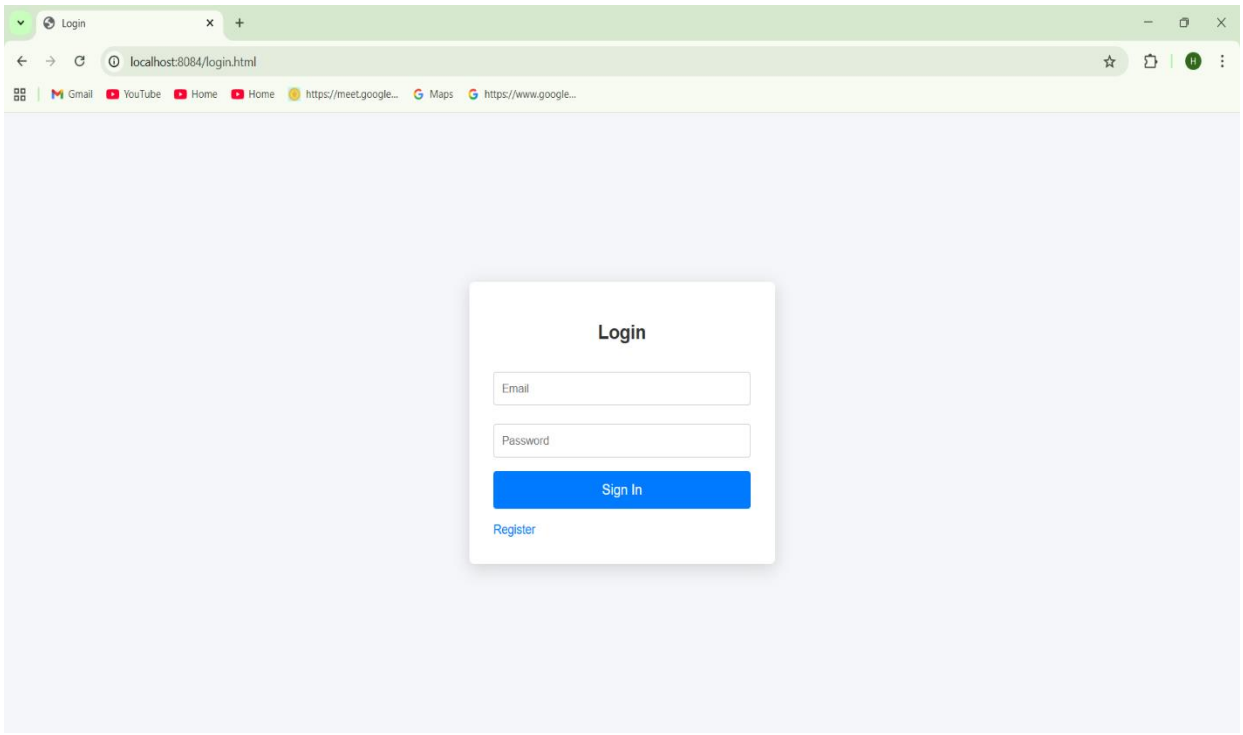
GET METHOD ON CHANGE PASSWORD



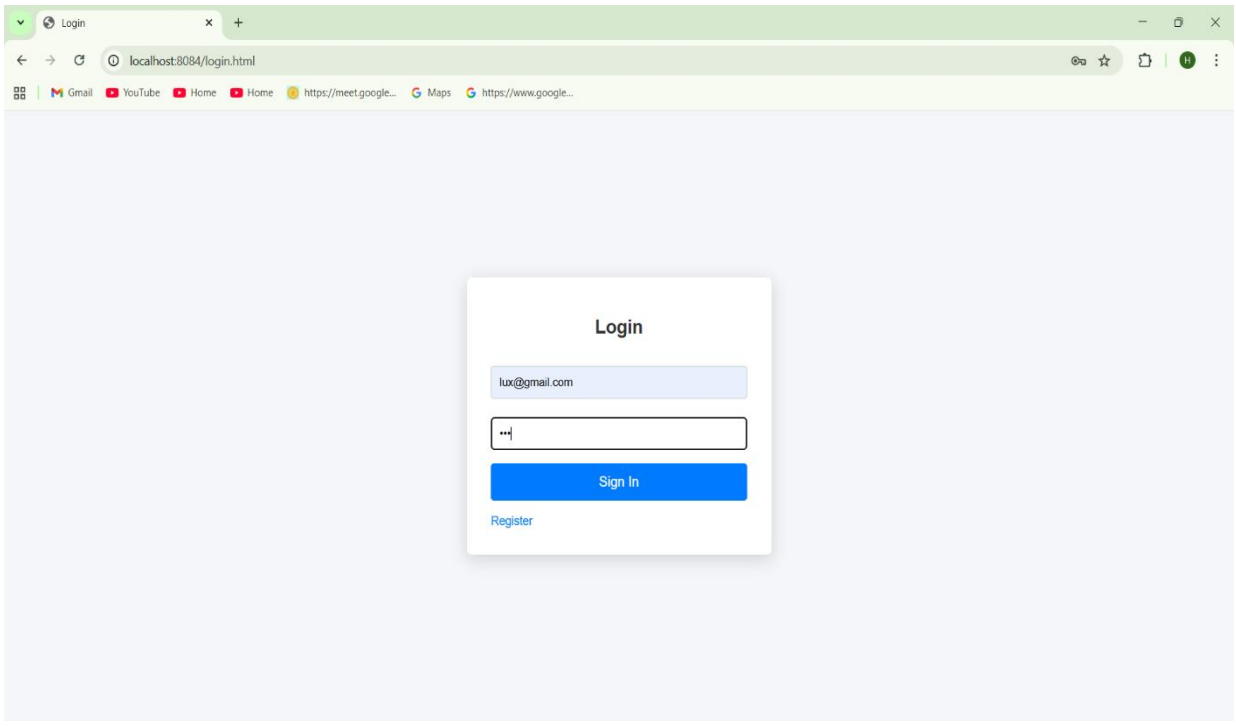
JUNITTEST CASES FOR MANAGER MODULE



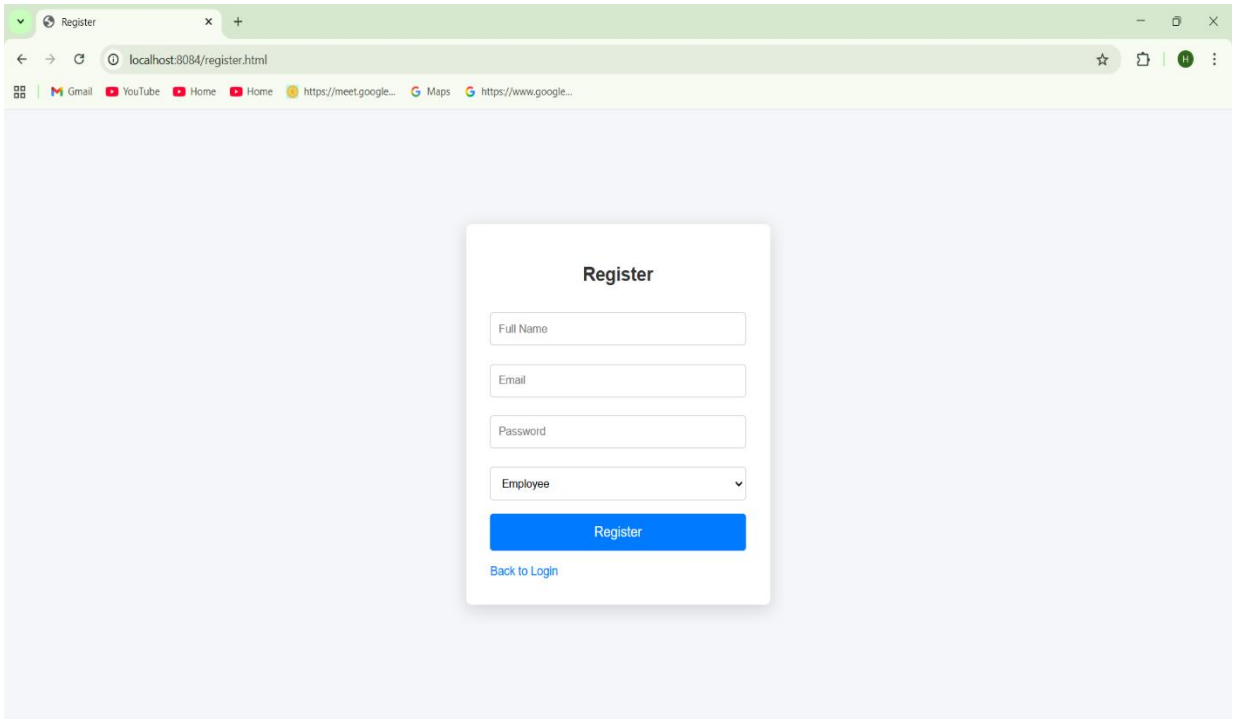
PROJECT OUTPUTS
Login Page



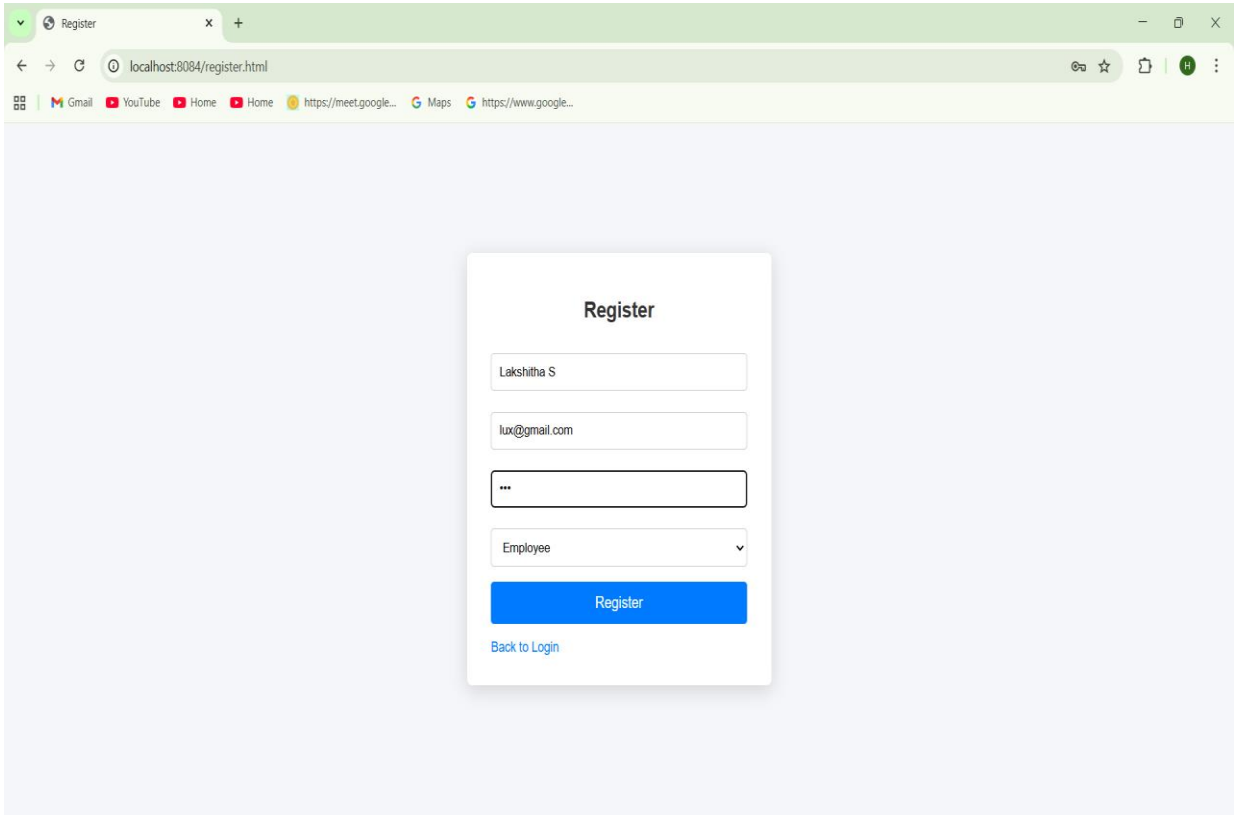
Employee’s Login Page

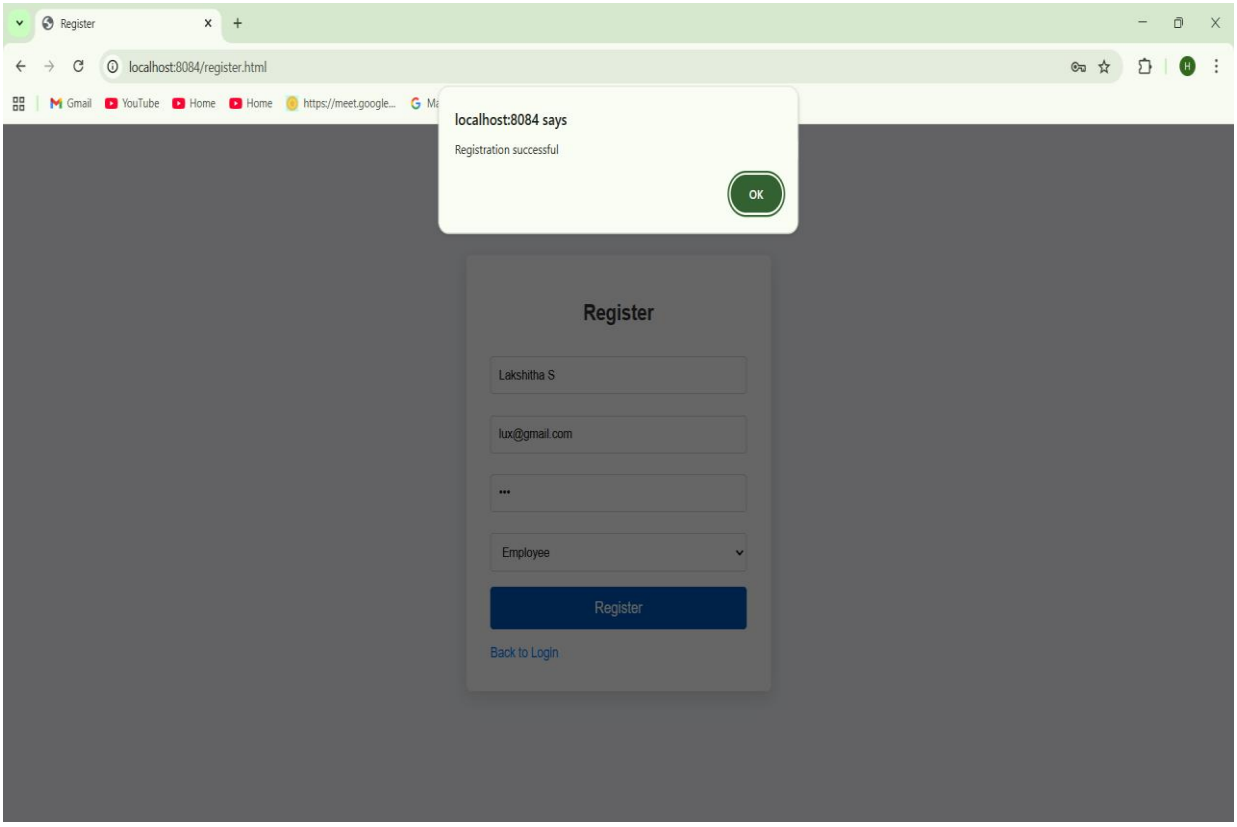


Registration Page

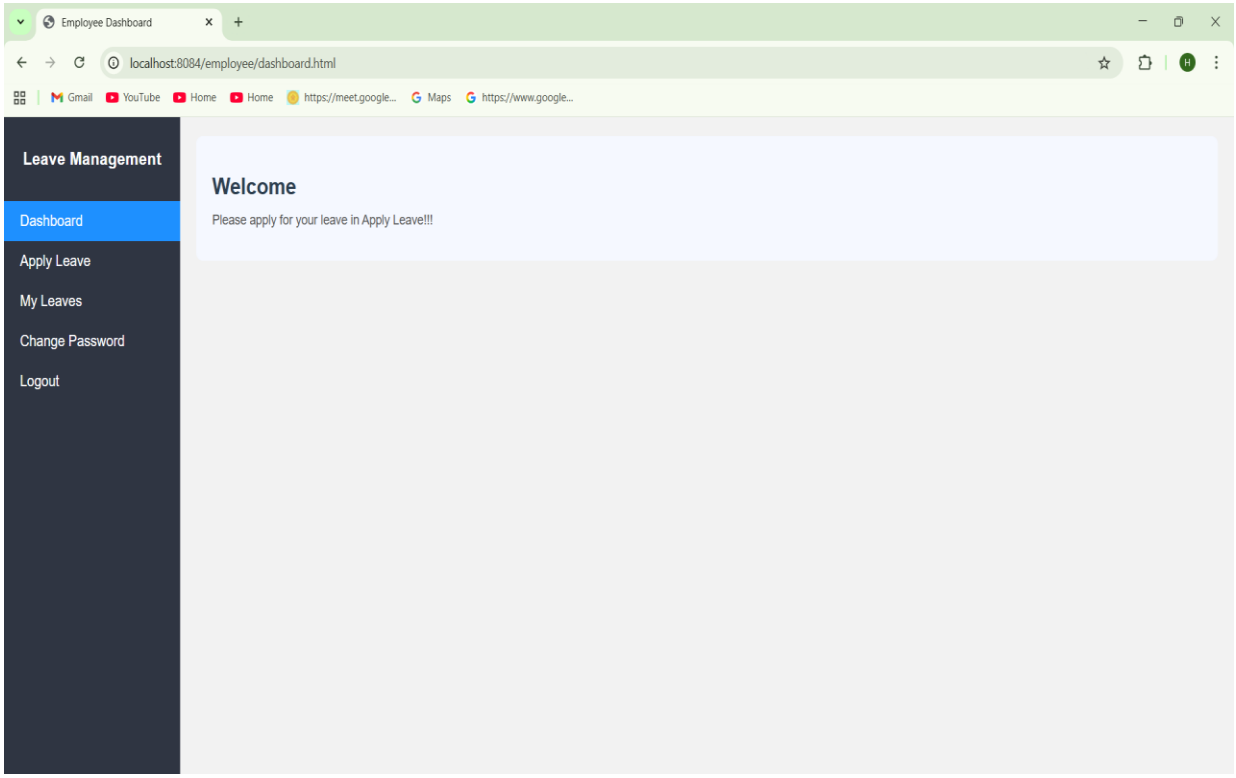


Employee’s Register Page

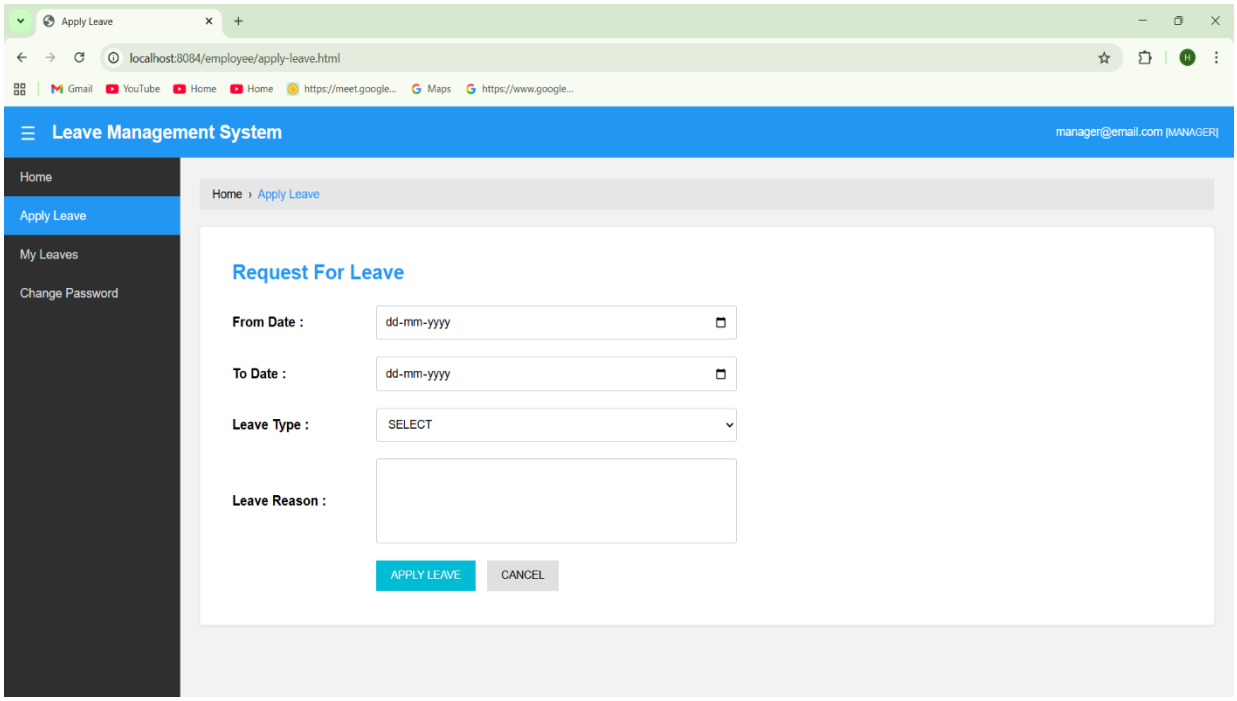




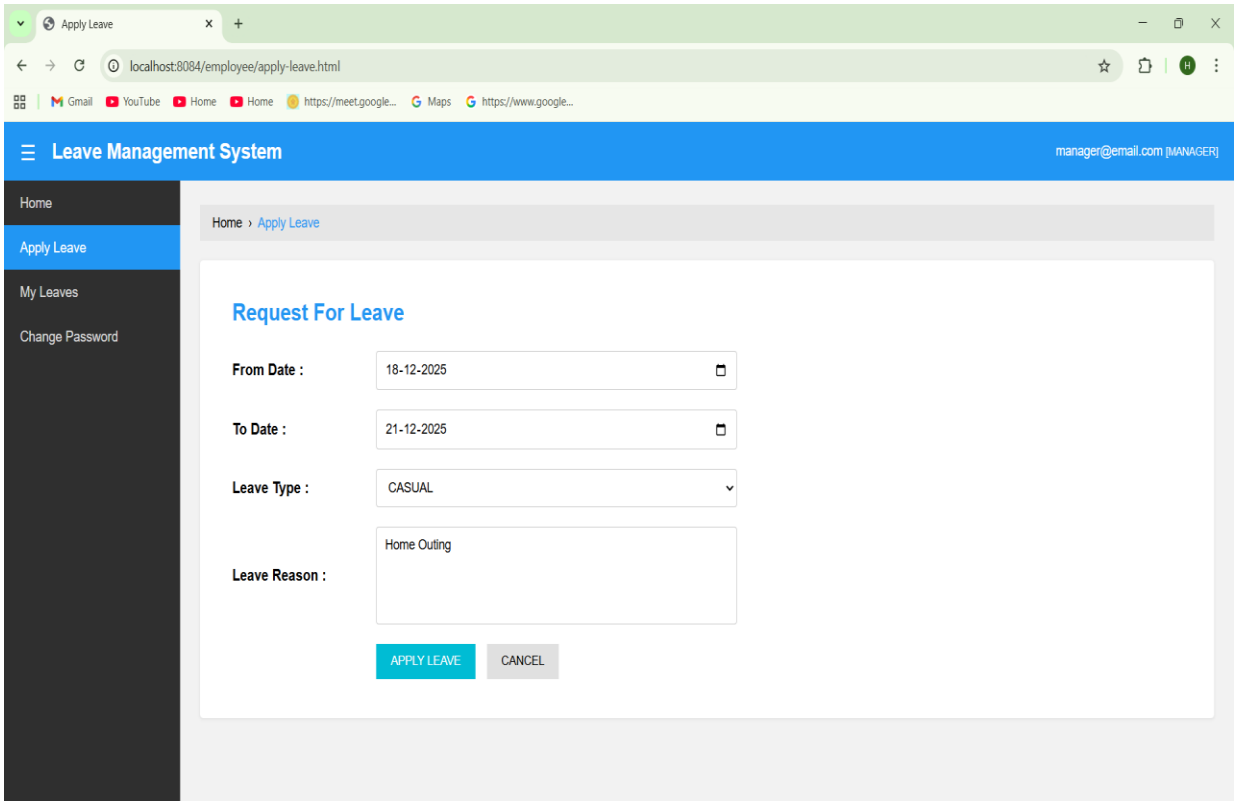
Employee’s Dashboard Page

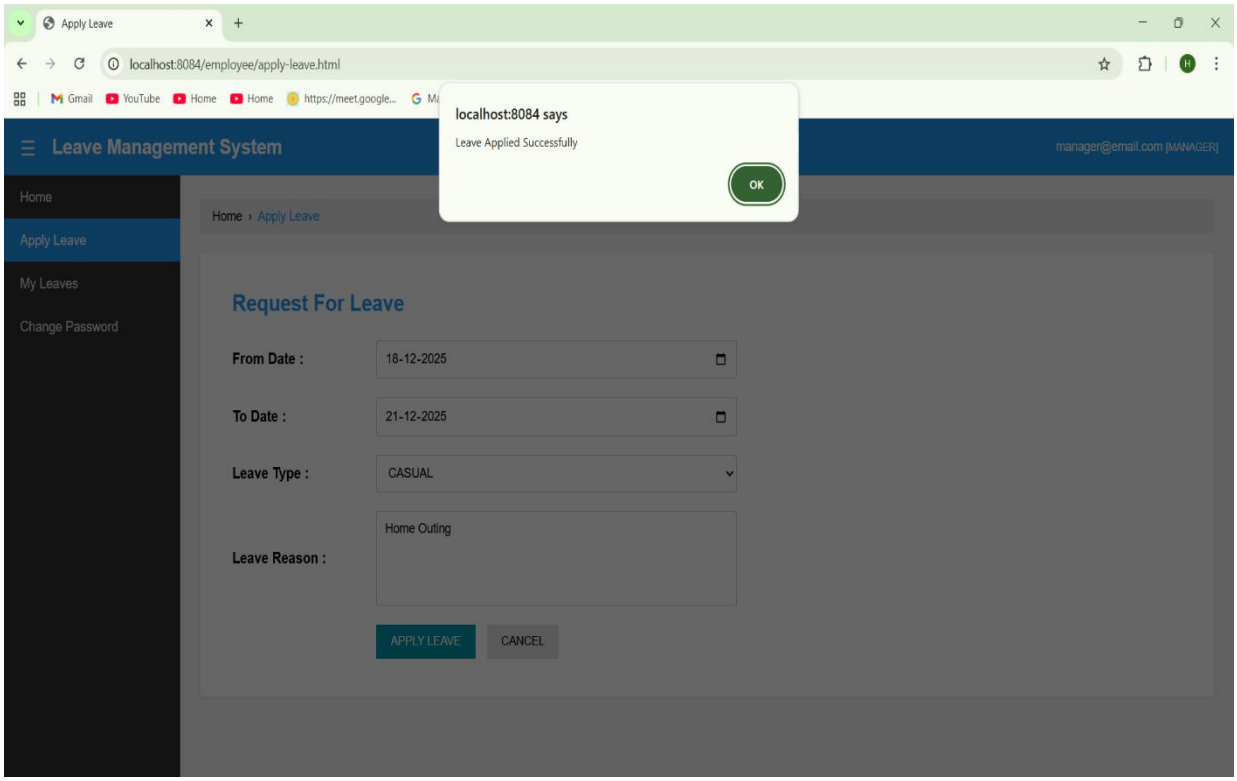


Request For Leave

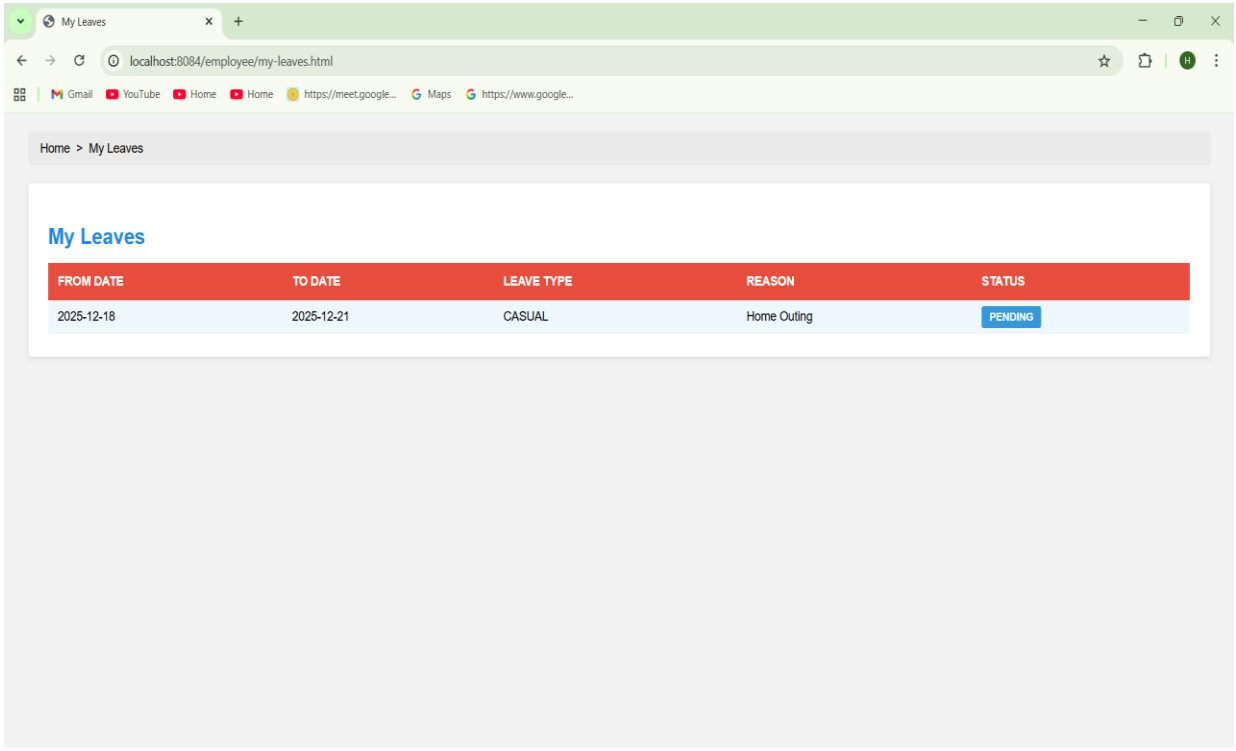


Employee’s Leave Request

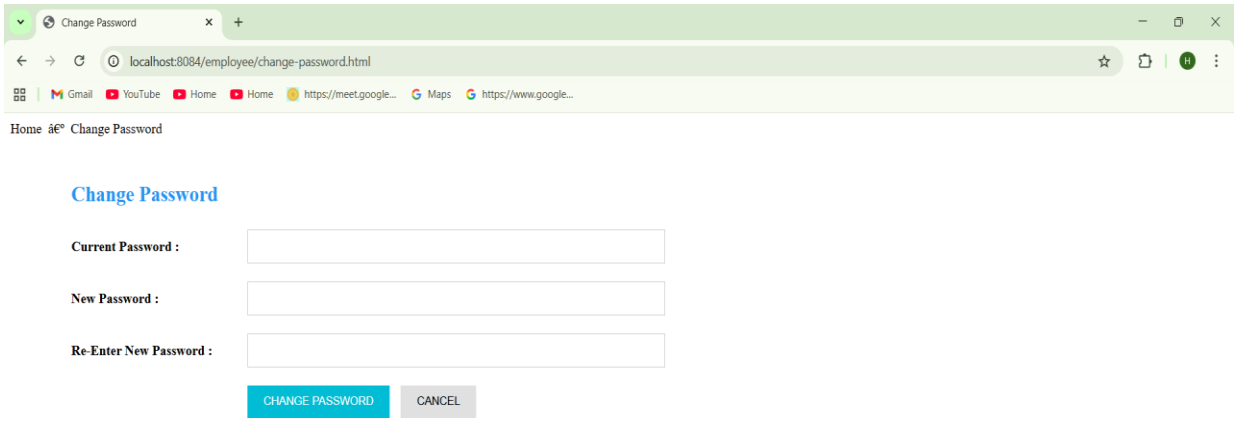




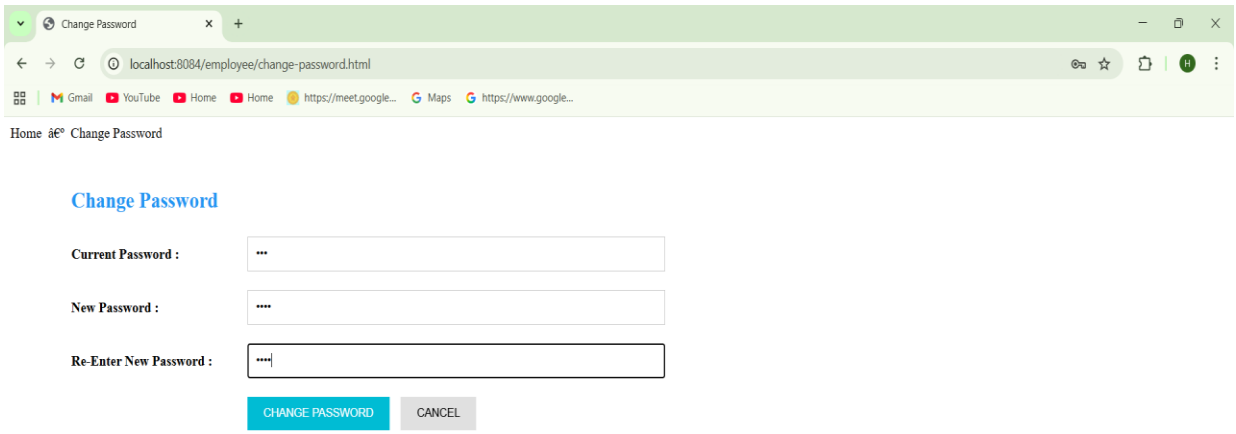
Employee’s - My Leaves Page

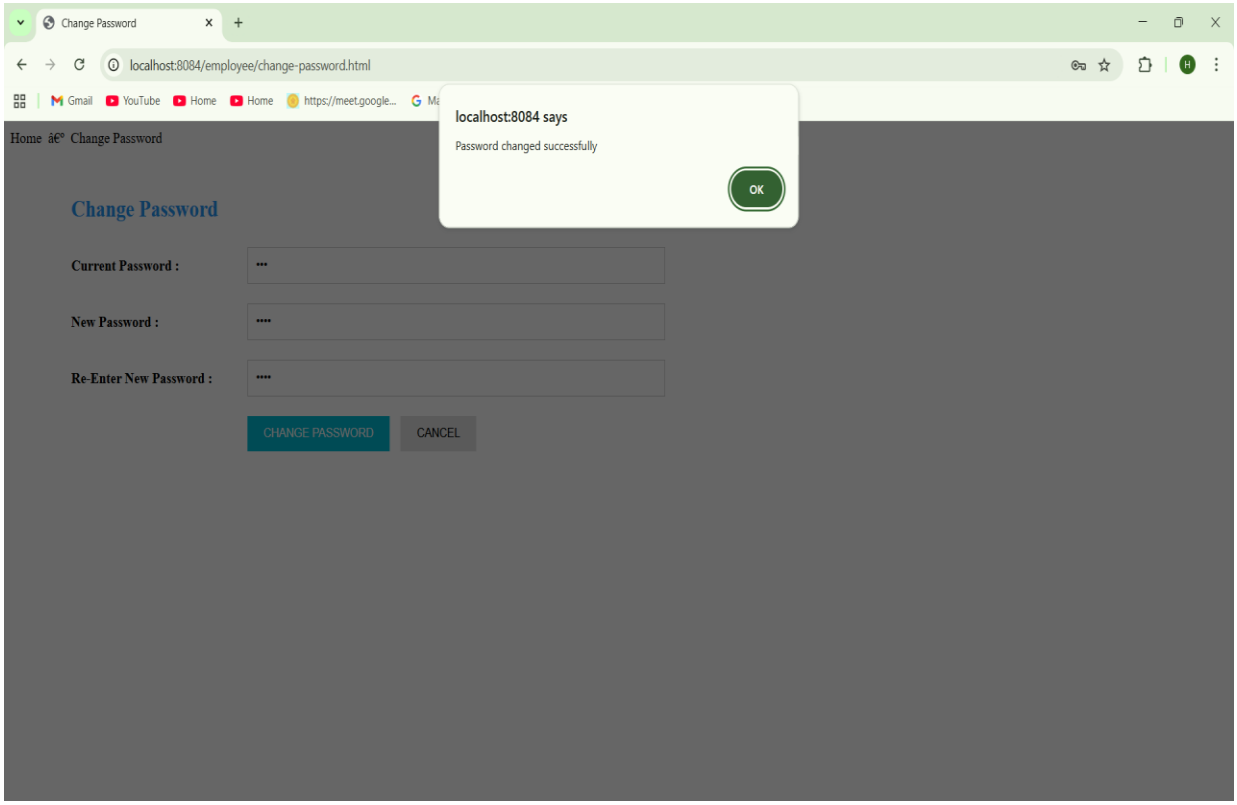


Change Password Page

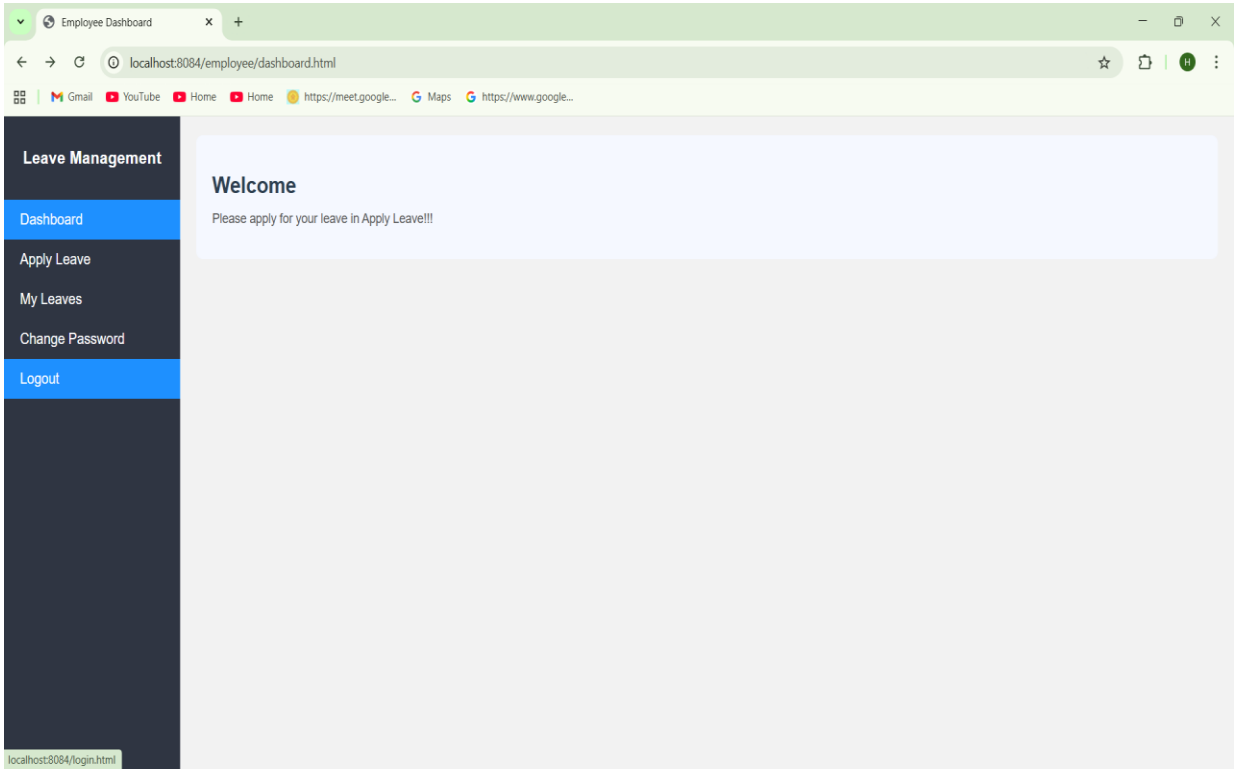


Changing New Password

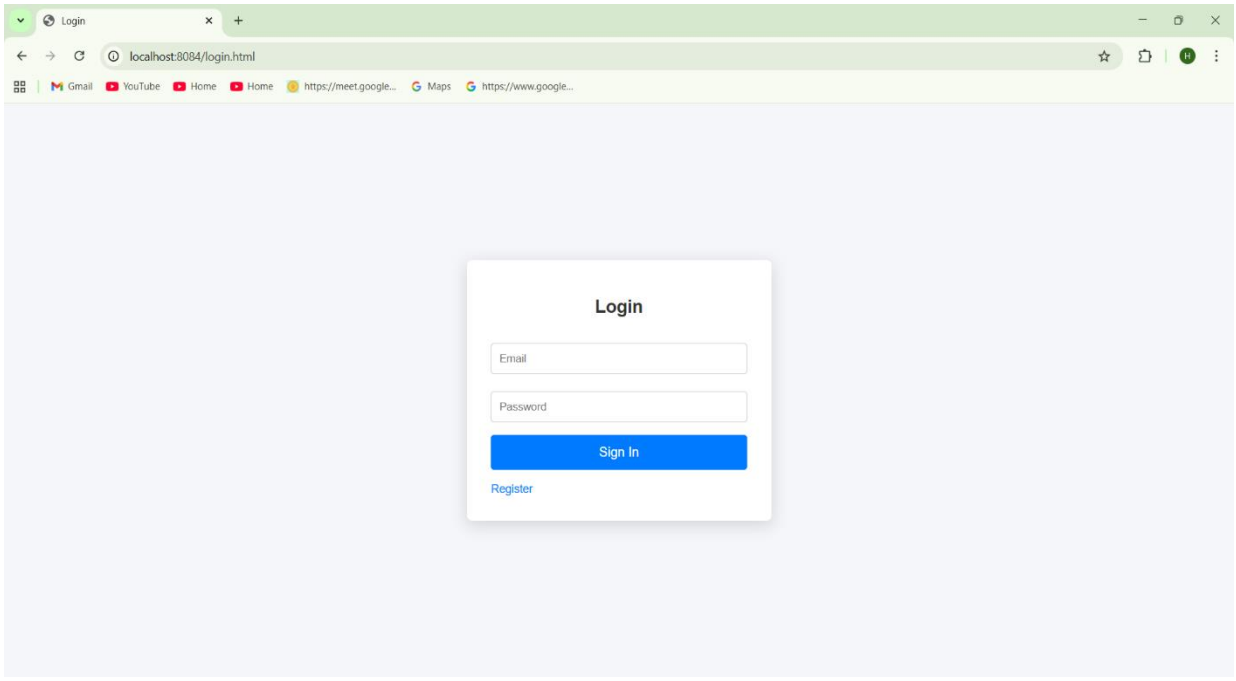




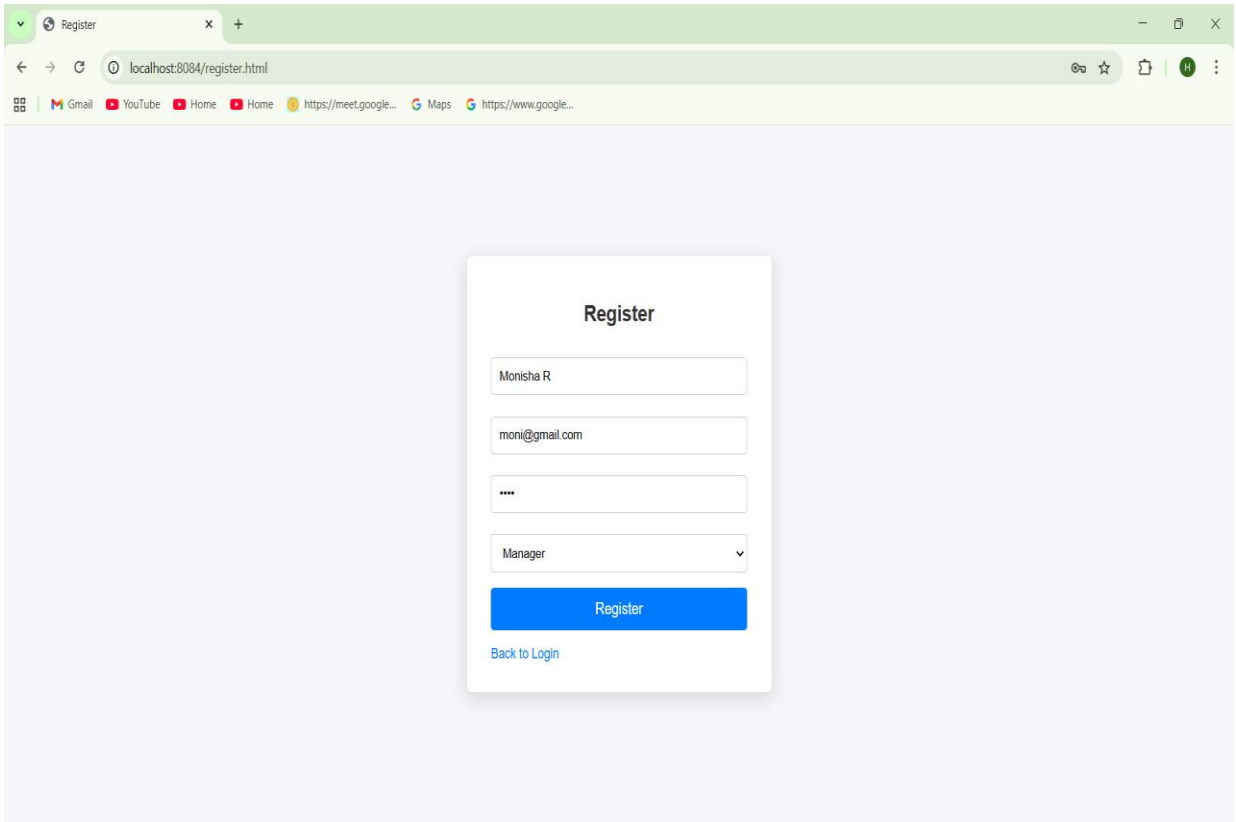
Before Logout

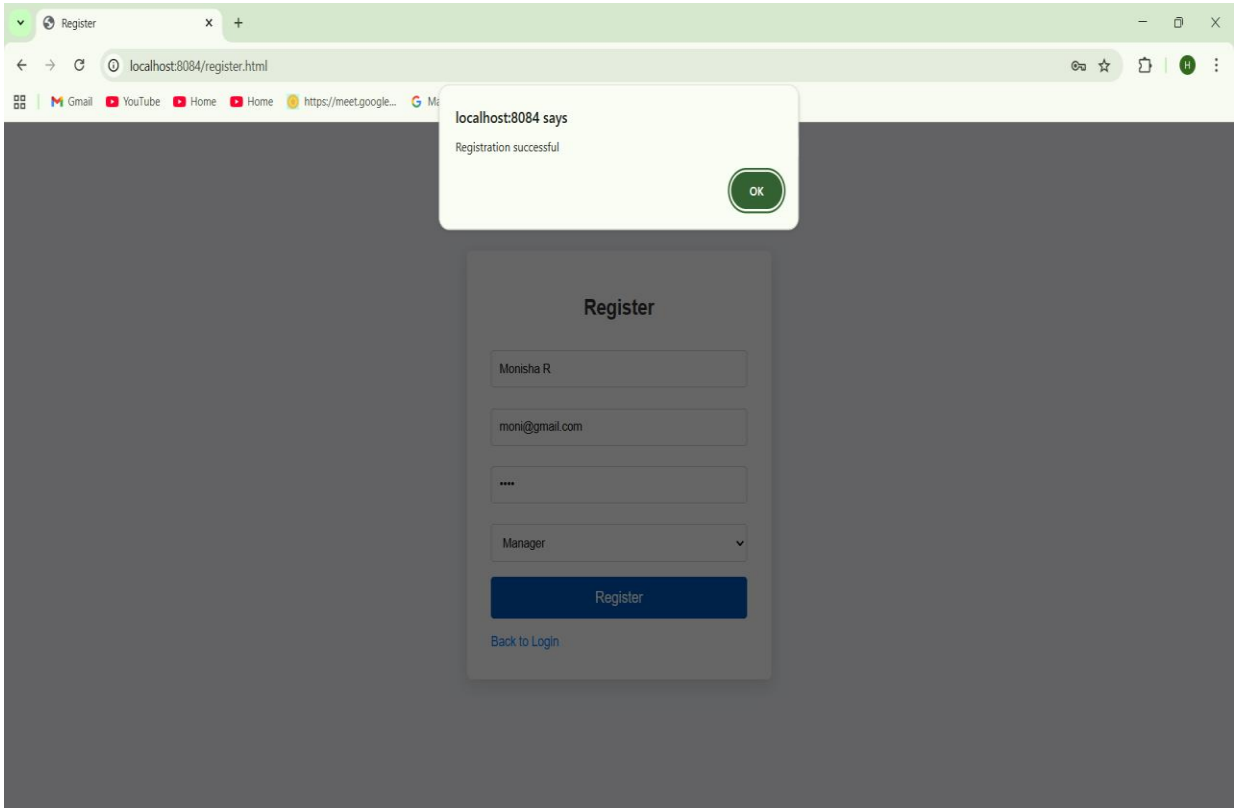


Logged Out

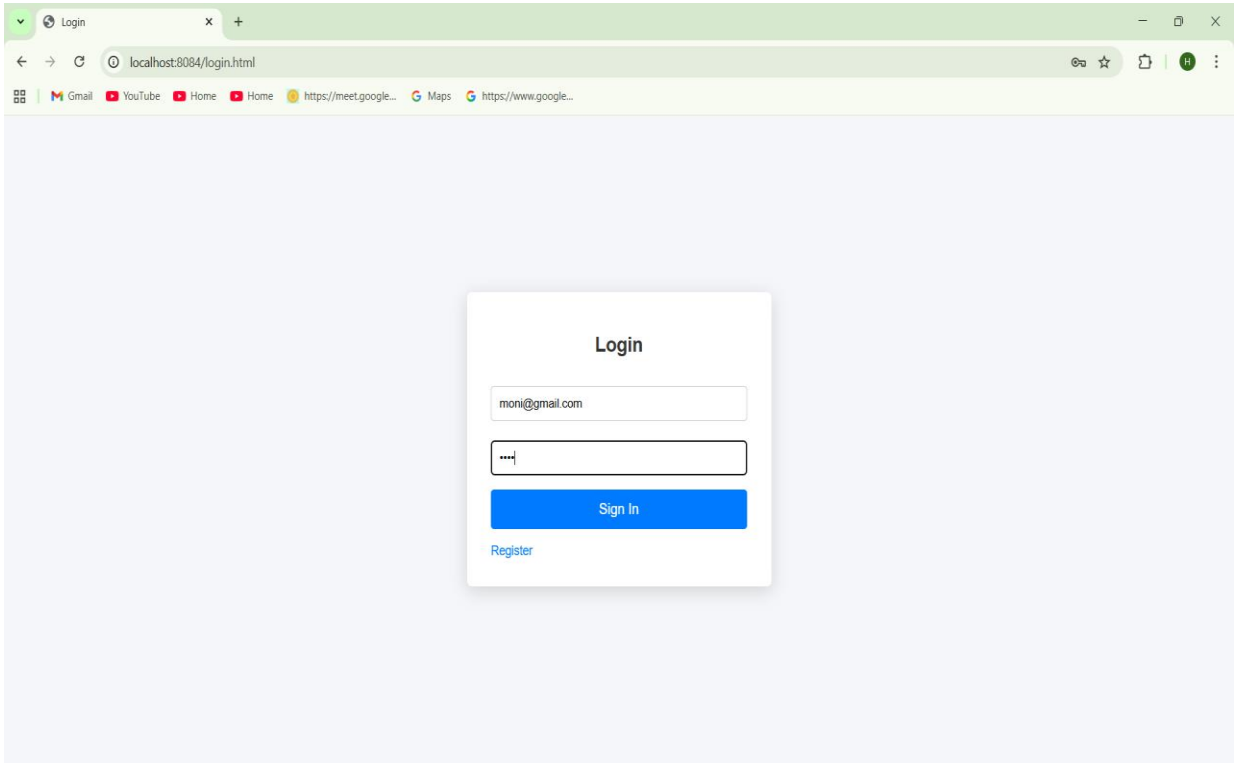


Manager's Register Page

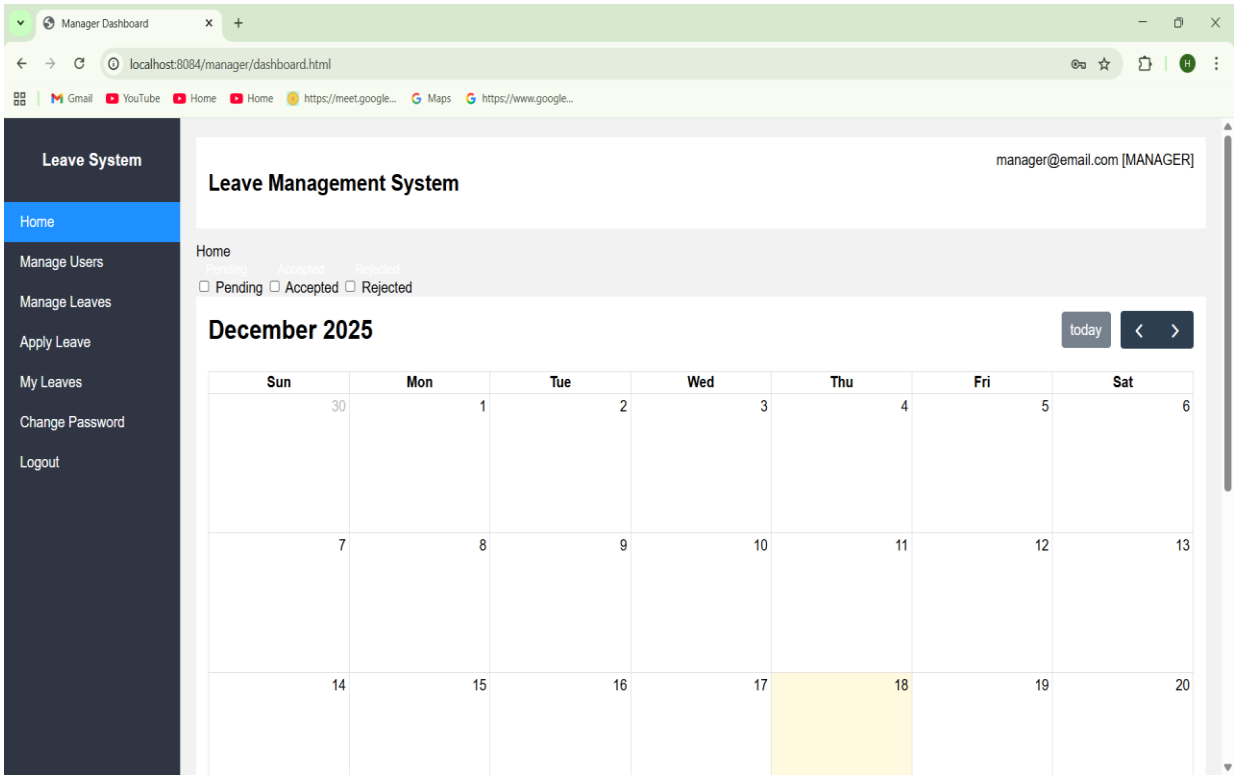




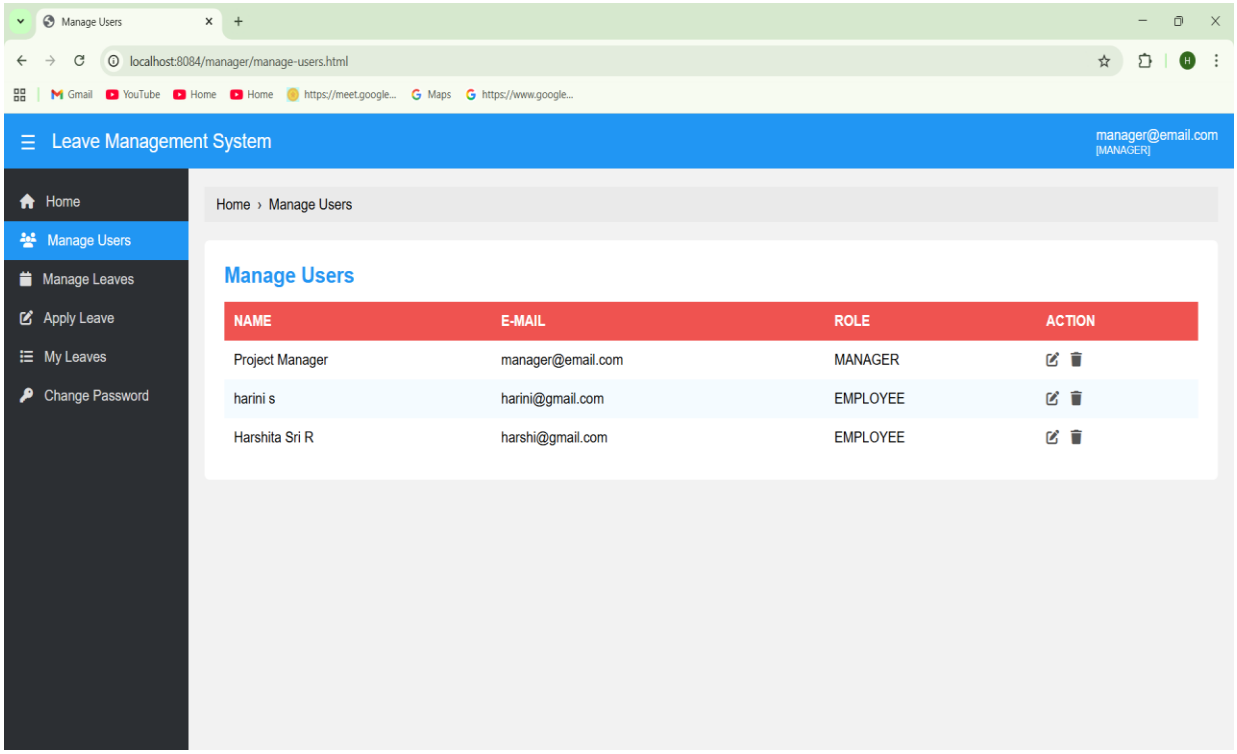
Manager’s Login Page



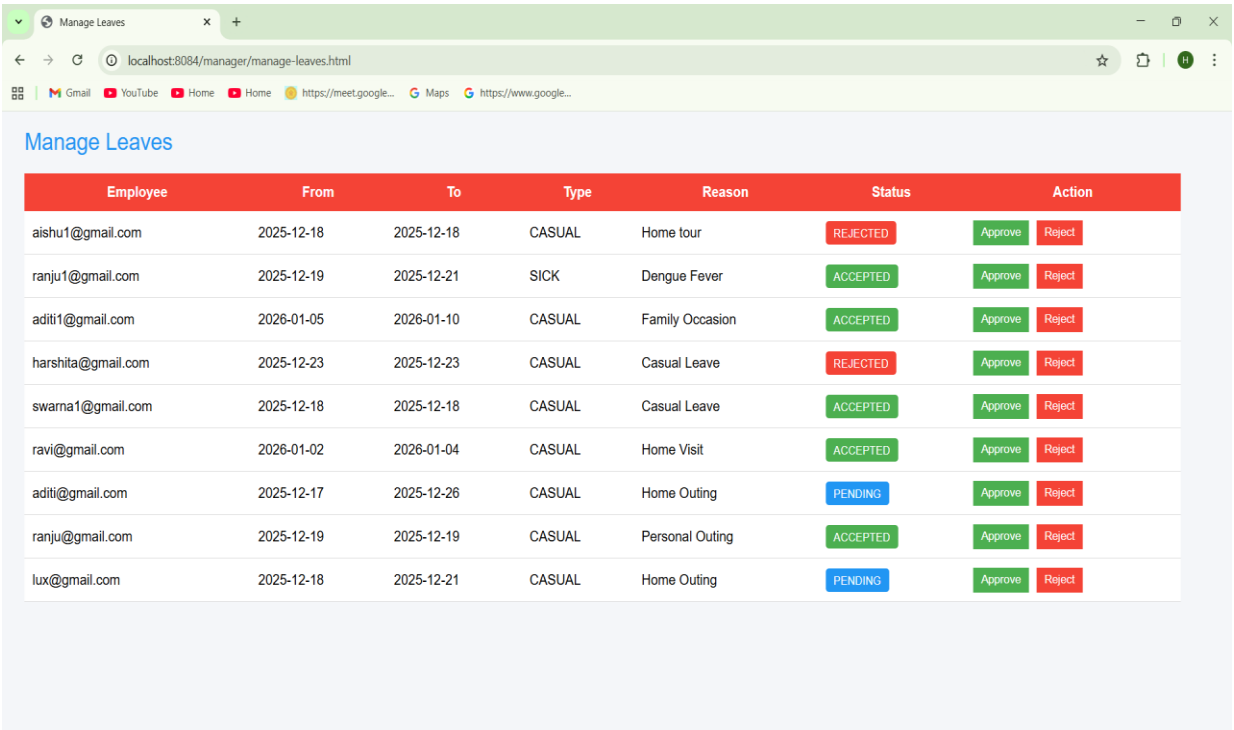
Manager's Home Page



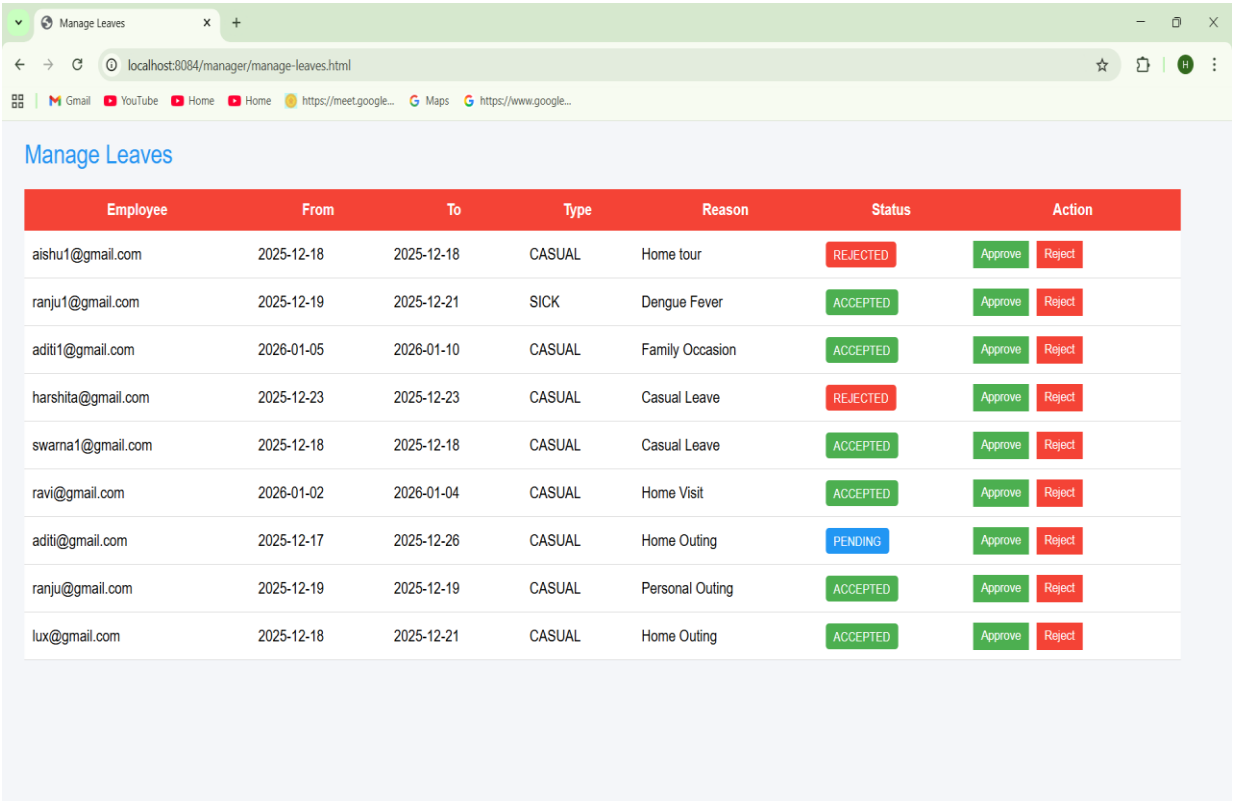
Manage Users Page



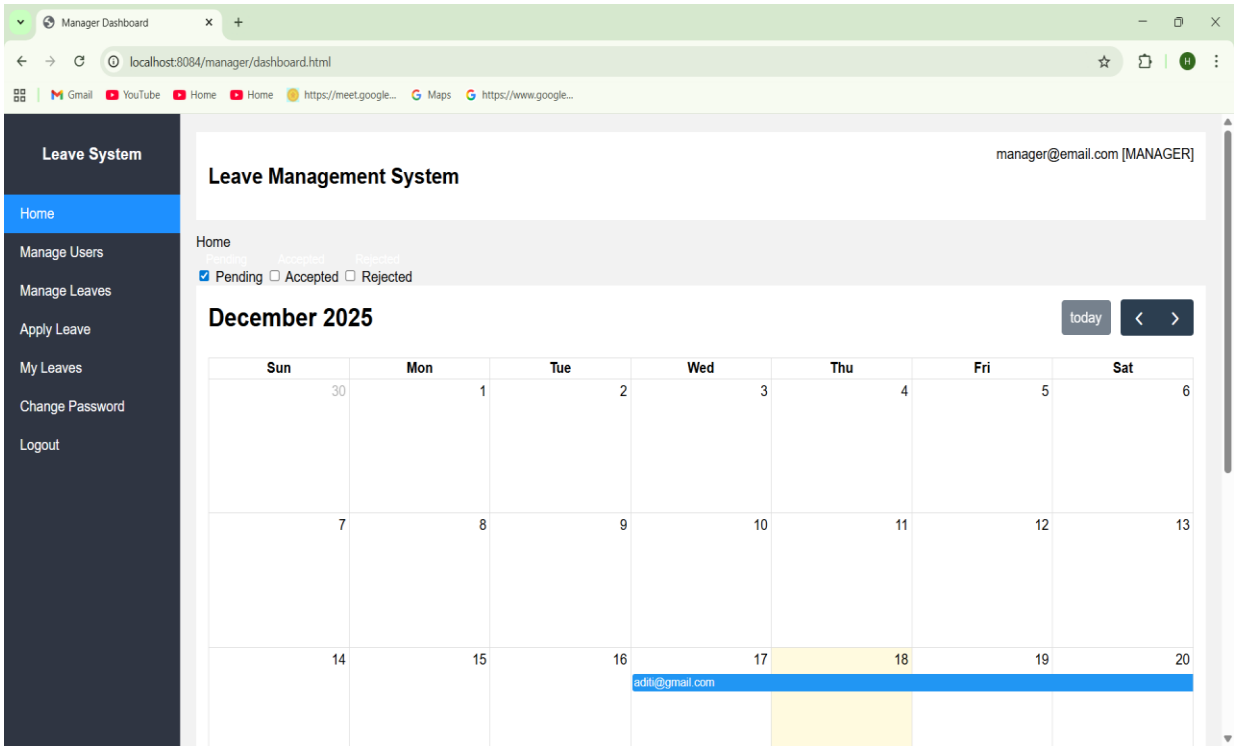
Manage Leaves Page



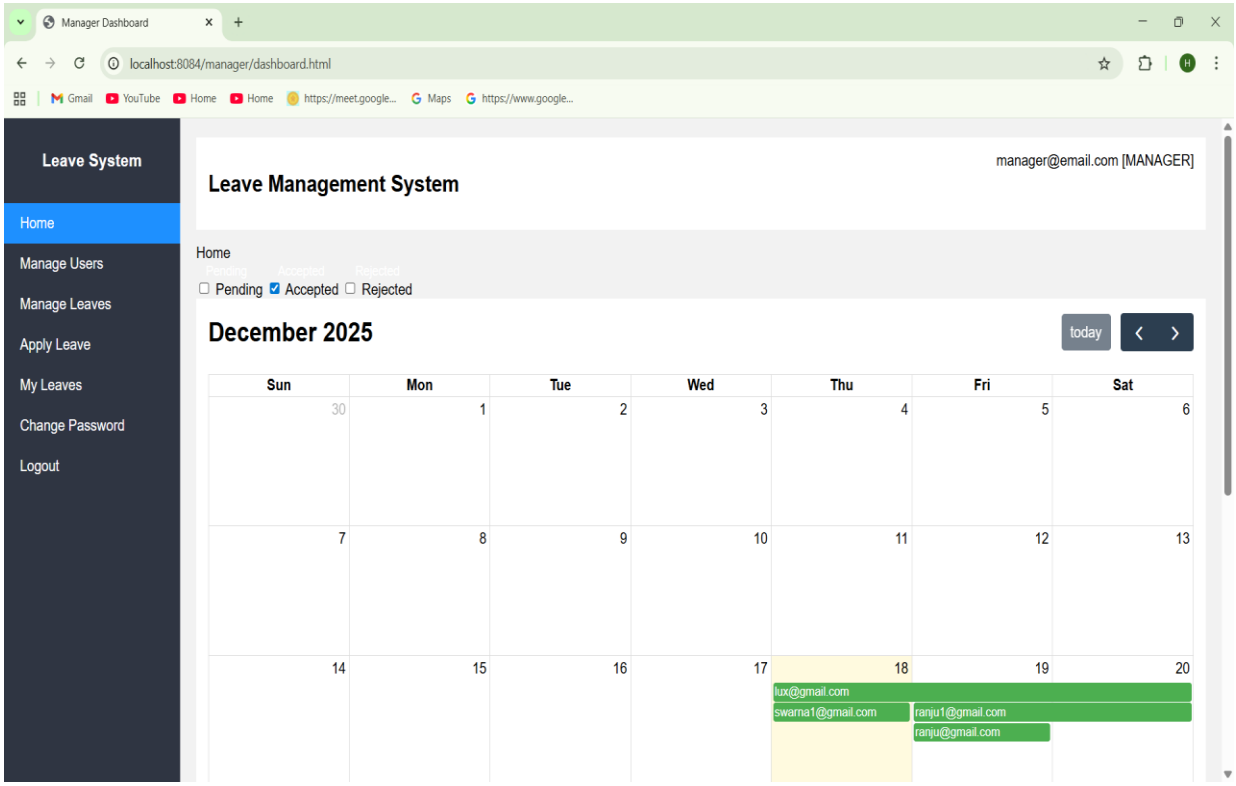
Manager – Approve/Reject Leaves



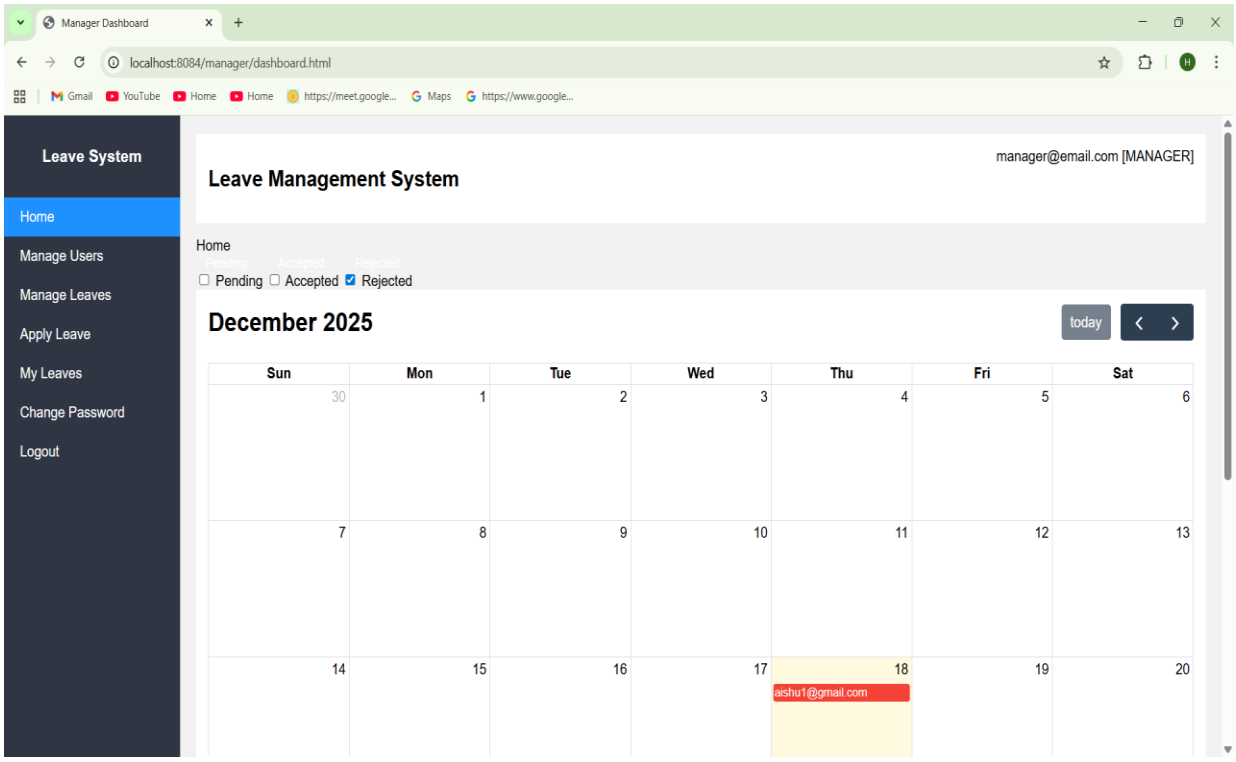
Leave Pending Status in Calender



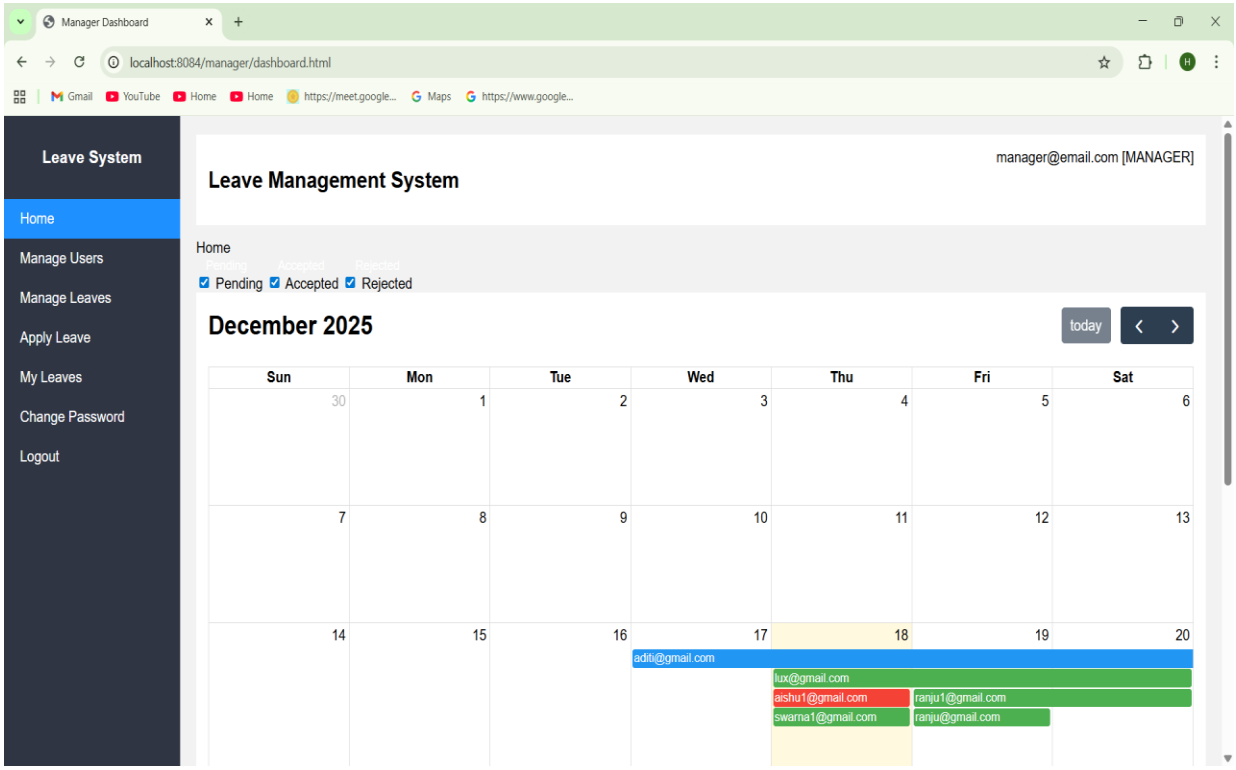
Leave Approved Status in Calender

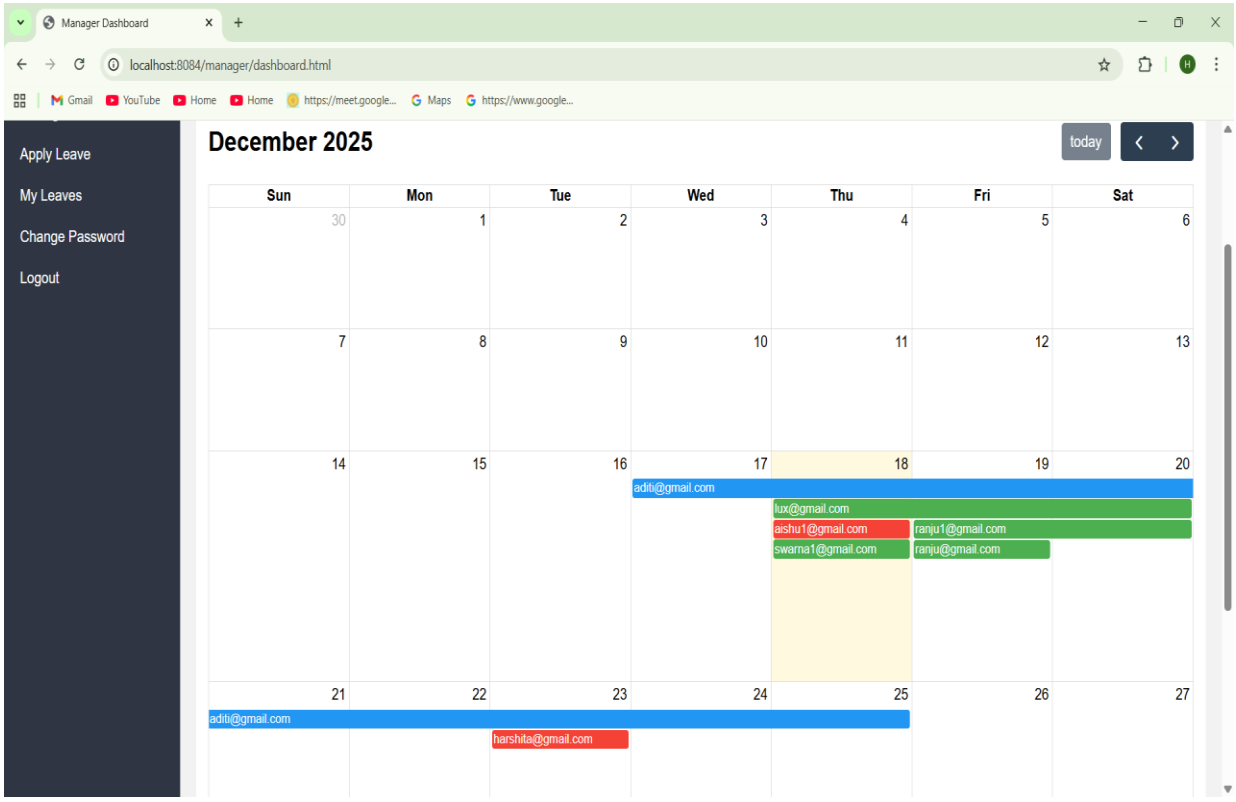


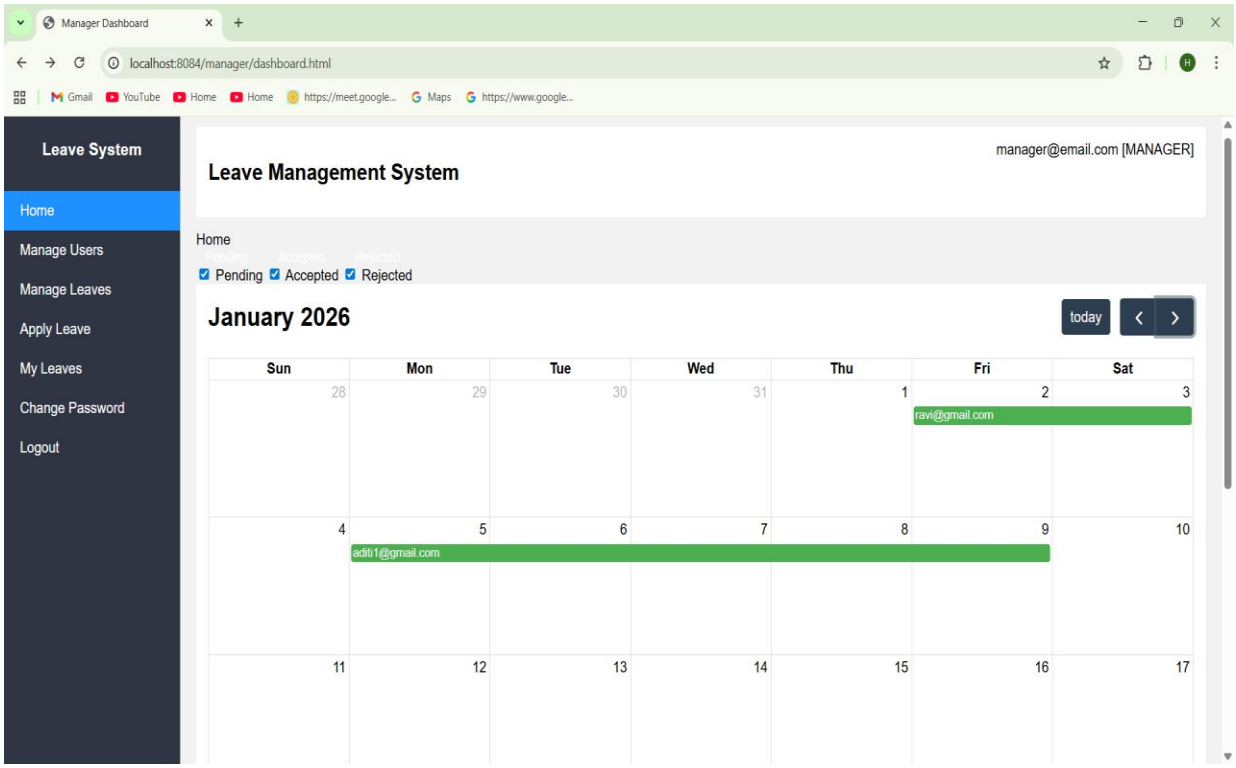
Leave Rejected Status in Calender



Pending/Approved/Rejected Status in Calender




Manager Dashboard
localhost:8084/manager/dashboard.html
Gmail
YouTube
Home
Home
https://meet.google...
Maps
https://www.google...
Apply Leave
My Leaves
Change Password
Logout
December 2025
today
Sun
Mon
Tue
Wed
Thu
Fri
Sat
30
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
aditi@gmail.com
lux@gmail.com
aishu1@gmail.com
ranju1@gmail.com
swarna1@gmail.com
ranju@gmail.com
aditi@gmail.com
harshita@gmail.com


Manager Dashboard
localhost:8084/manager/dashboard.html
Gmail
YouTube
Home
Home
https://meet.google...
Maps
https://www.google...
Leave System
Home
Manage Users
Manage Leaves
Apply Leave
My Leaves
Change Password
Logout
manager@email.com [MANAGER]
Leave Management System
Home
Pending
Accepted
Rejected
January 2026
today
Sun
Mon
Tue
Wed
Thu
Fri
Sat
28
29
30
31
1
2
3
ravi@gmail.com
4
5
6
7
8
9
10
aditi1@gmail.com
11
12
13
14
15
16
17