**Leave Approval System**

**A PROJECT REPORT**

**for**

**Mini Project-I (K24MCA18P)**

**Session (2024-25)**

**Submitted by**

**Mohammad Daud**

**202410116100121**

**Harshit Singh**

**202410116100089**

**Imran Ahmad**

**202410116100092**

**Submitted in partial fulfilment of the**

**Requirements for the Degree of**

**MASTER OF COMPUTER APPLICATION**

**Under the Supervision of**

**Ms. Arpit Dogra**

### Assistant Professor



**Submitted to**

**Department Of Computer Applications**

**KIET Group of Institutions, Ghaziabad**

**Uttar Pradesh-201206**

**(december- 2024)**

**Declaration**

We, the undersigned, hereby declare that the Mini Project titled "Leave Approval System" is an original work completed by us as part of the curriculum requirement for the course under the Master of Computer Applications (MCA) program at **KIET Group Of Institutions**.

We affirm that we have undertaken by us during the academic year 2024-25 under the guidance of **Mr. Arpit Dogra**. All the content and ideas presented in this report are the result of our own efforts, except where explicitly stated otherwise. Proper citations have been provided wherever references to external sources have been made.

We further declare that this project has not been submitted, either in part or in full, to any other university or institution for any degree or diploma.

Team Members:

|  |  |
| --- | --- |
| Name | Roll Number |
| Mohammad Daud | 202410116100121 |
| Harshit Singh | 202410116100089 |
| Imran Ahmad | 202410116100092 |

Date: [15/12/2024]  
Place: [KIET Group of Institutions, Ghaziabad]

Signatures of Team Members:

1. \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ (Mohammad Daud)
2. \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ (Harshit Singh)
3. \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ (Imran Ahmad)

**CERTIFICATE**

Certified that **MOHAMMAD DAUD, HARSHIT SINGH, IMRAN AHMAD** has carried out the project work having “**Leave Approval System**” (**Mini Project-I, K24MCA18P**) for **Master of Computer Application** from Dr. A.P.J. Abdul Kalam Technical University (AKTU**)** (formerly UPTU), Lucknow under my supervision. The project report embodies original work, and studies are carried out by the student himself, and the contents of the project report do not form the basis for the award of any other degree to the candidate or to anybody else from this or any other University/Institution.

**Mr. Arpit Dogra Dr. Arun Kr. Tripathi**

**Assistant Professor Dean**

**Department of Computer Applications Department of Computer Applications**

**KIET Group of Institutions, Ghaziabad KIET Group of Institutions, Ghaziabad**

**Abstract**

The Leave Approval System is a web-based application designed to streamline the process of leave applications for students in educational institutions. The system allows students to submit leave requests online, which are then routed to the head of the MCA department for review and approval. Upon approval, students receive an email notification. This project addresses inefficiencies in traditional paper-based leave management systems and ensures transparency, accuracy, and prompt communication.

Key features of the system include:

* A user-friendly interface built using React and Tailwind CSS.
* Real-time updates on leave status.
* Automated email notifications to keep users informed of request approvals or rejections.
* Secure and scalable architecture for efficient performance.

This project demonstrates the use of modern web technologies to enhance administrative workflows in educational institutions.

Keywords: Leave Management, Web Application, React, Tailwind CSS, Automation

**Acknowledgments**

Success in life is never attained single-handedly. My deepest gratitude goes to my project supervisor, **Mr. Arpit Dogra** for her guidance, help, and encouragement throughout my project work. Their enlightening ideas, comments, and suggestions.

Words are not enough to express my gratitude to Dr. Arun Kumar Tripathi, Professor and Dean, Department of Computer Applications, for his insightful comments and administrative help on various occasions.

Fortunately, I have many understanding friends, who have helped me a lot on many critical conditions.

Finally, my sincere thanks go to my family members and all those who have directly and indirectly provided me with moral support and other kind of help. Without their support, completion of this work would not have been possible in time. They keep my life filled with enjoyment and happiness.

**Name - Mohammad Daud**

**Roll No - 202410116100121**

**Name - Harshit Singh**

**Roll No - 202410116100089**

**Name - Imran Ahmad**

**Roll No - 202410116100092**

**Table of Contents**

1. Introduction
2. System Analysis
   * 2.1 Existing System
   * 2.2 Proposed System
3. System Design
   * 3.1 Use Case Diagram
   * 3.2 ER Diagram
   * 3.3 System Architecture
4. Implementation
   * 4.1 Technology Stack
   * 4.2 Features
   * 4.3 User Interface Design
5. Testing and Validation
   * 5.1 Test Cases
   * 5.2 Results
6. Conclusion and Future Work
7. References

**Chapter 1: Introduction**

**1.1 Overview**  
In today’s world, efficient data communication and networks play a crucial role in making processes smoother and faster. The Leave Approval System leverages these technologies to simplify the leave application process in educational institutions.

This system uses computer networks to allow students to submit leave requests online, which are reviewed and approved by the head of the department. Once approved, students are notified via email. This ensures transparency and saves time compared to traditional manual systems.

A good system must meet the following criteria:

* Performance: Fast and efficient operation.
* Reliability: Consistent and accurate results.
* Scalability: Ability to grow and support more users.

**1.2 Basic Communication Model**  
The Leave Approval System follows a client-server communication model. Users interact with the system via a web interface (client), and the server processes requests and sends responses. This ensures smooth and real-time communication.

**1.3 Data Communication in the Leave Approval System**  
Data communication in the system ensures the correct exchange of information between users and the system.

1. Local Communication: This happens within the same network, such as in a campus.
2. Remote Communication: This takes place when users access the system from outside via the internet.

Effective communication ensures:

* Delivery: Requests and notifications reach the right person.
* Timeliness: Responses are quick.
* Accuracy: Data remains error-free throughout the process.

**Chapter 2: System Analysis**

**2.1 Existing System**

In the current system, leave applications are managed manually. Students fill out paper forms, which are then physically submitted to the department head for approval. This process is time-consuming and prone to errors, such as misplaced forms or incomplete records. Communication regarding leave status is often delayed, leading to uncertainty for students.

**2.2 Proposed System**

The proposed system automates the entire leave management process. Students can log in to a web application to submit leave requests. These requests are reviewed by the department head, who can approve or reject them with a single click. Automated email notifications ensure timely communication, and the system maintains a comprehensive record of all transactions for future reference.

Key benefits include:

* Reduced processing time for leaves applications.
* Enhanced accuracy and reliability through digital record-keeping.
* Immediate updates and notifications for students.

**Chapter 3: System Design**

**3.1 Use Case Diagram**

[A diagram of a process

Description automatically generated

**3.2 ER Diagram**

A diagram of a software application

Description automatically generated

**3.3 System Architecture**

The Leave Approval System is built on a three-tier architecture:

Frontend: Developed using React, providing a dynamic and user-friendly interface.

Backend: Powered by Node.js and Express.js for handling application logic and APIs.

Database: MongoDB serves as the database for storing user credentials, leave requests, and status updates.

**Chapter 4: Implementation**

**4.1 Technology Stack**

* **Frontend:** React, Tailwind CSS
* **Backend:** Node.js, Express.js
* **Database:** MongoDB

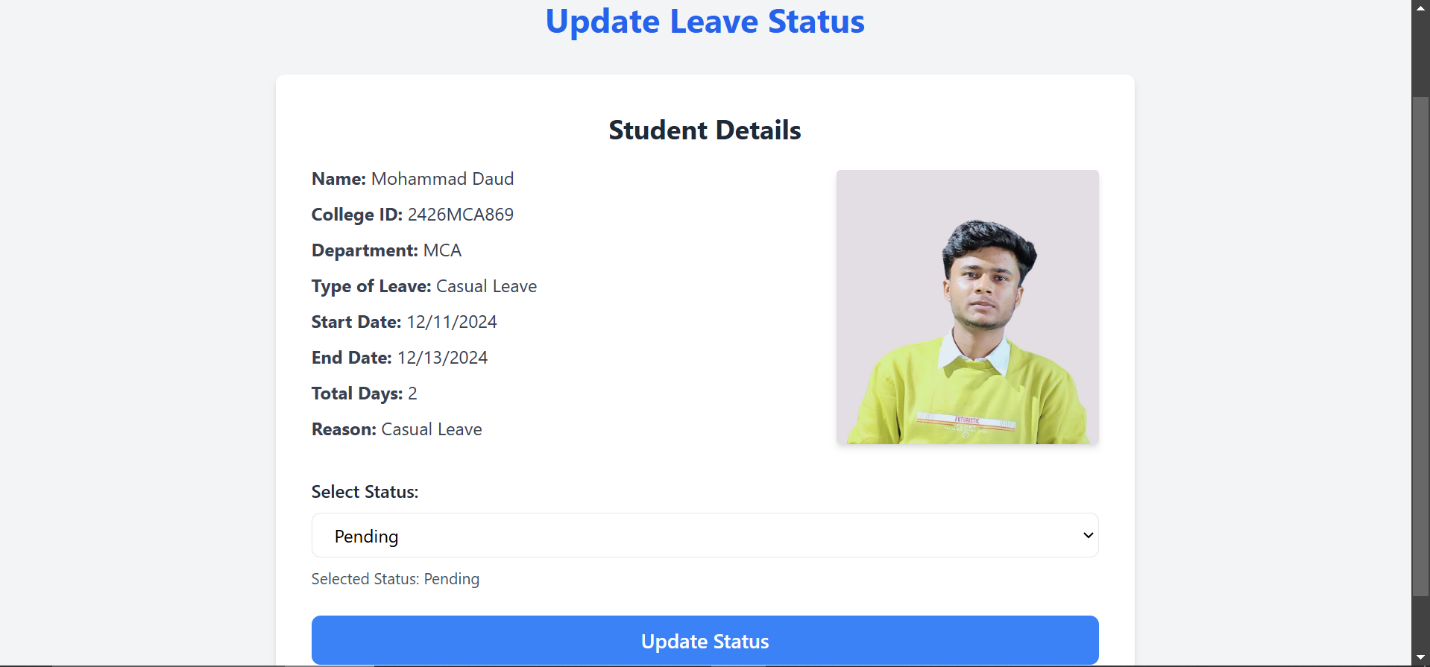
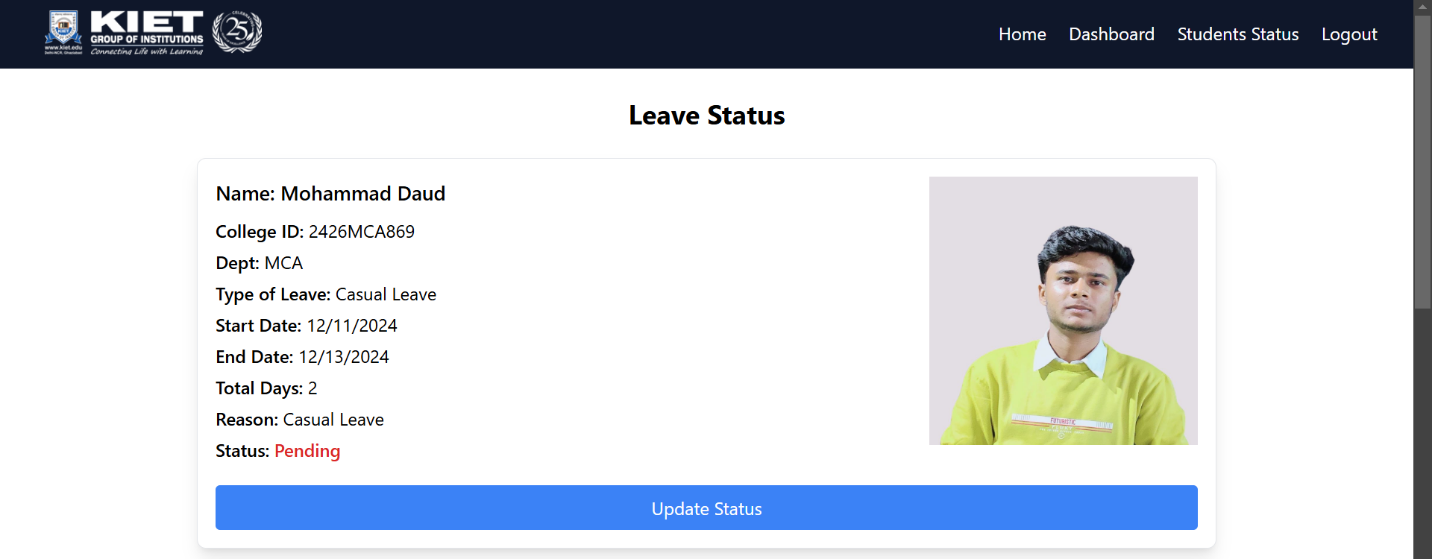
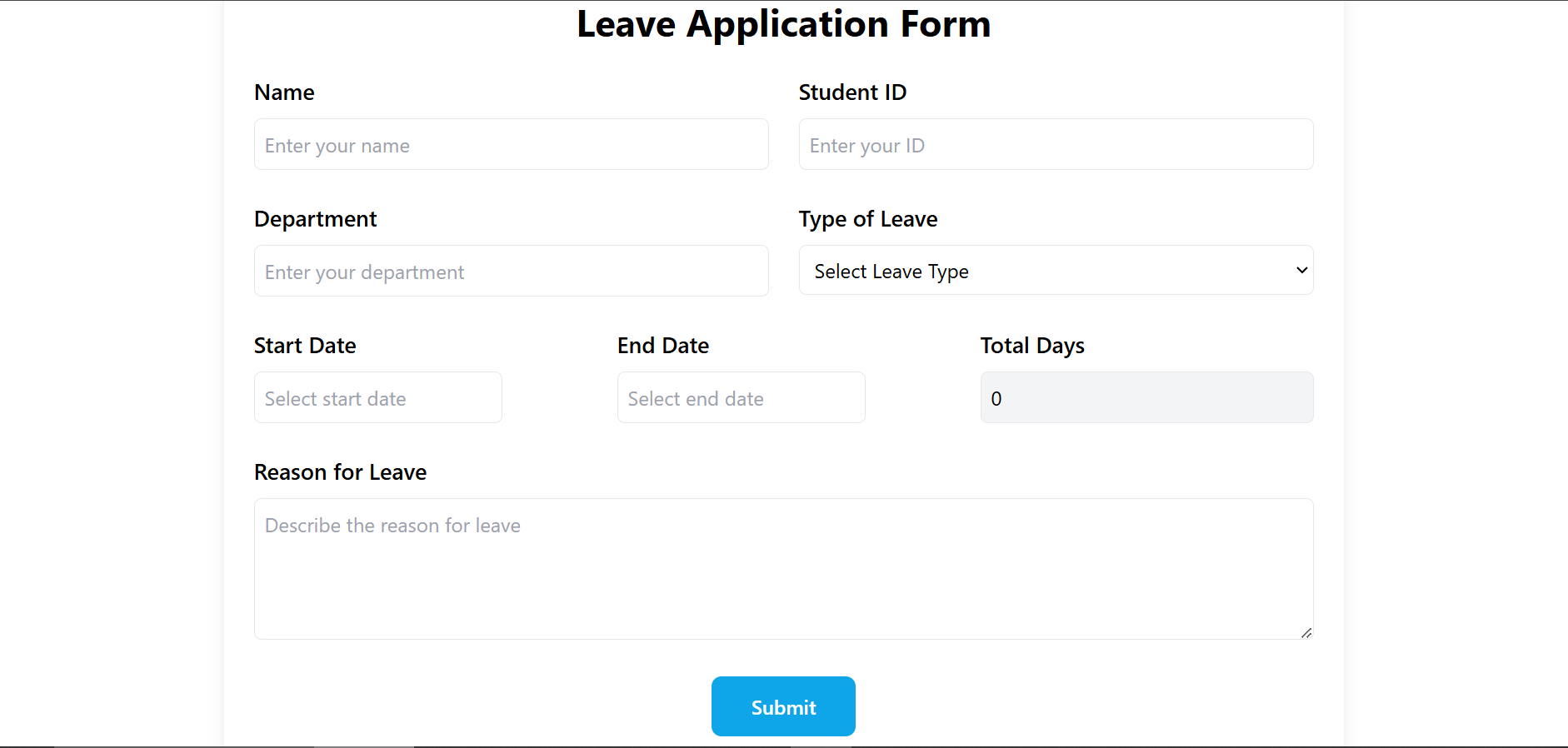
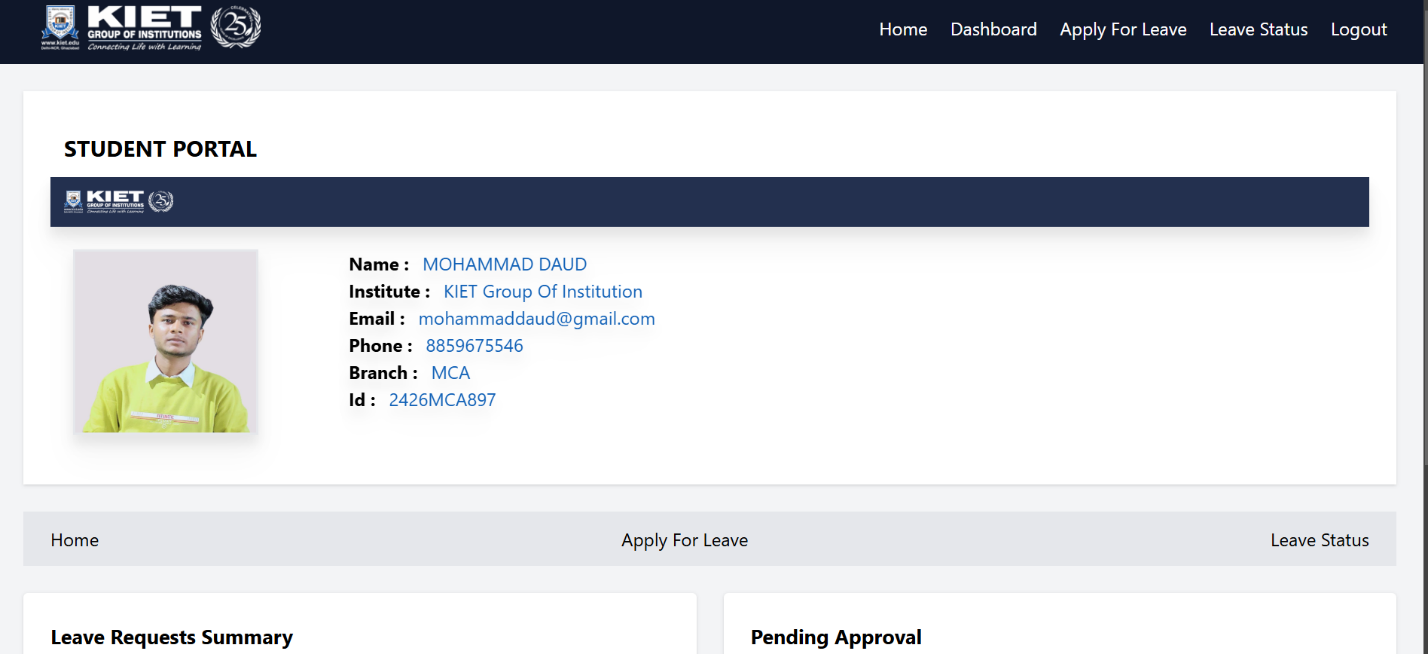
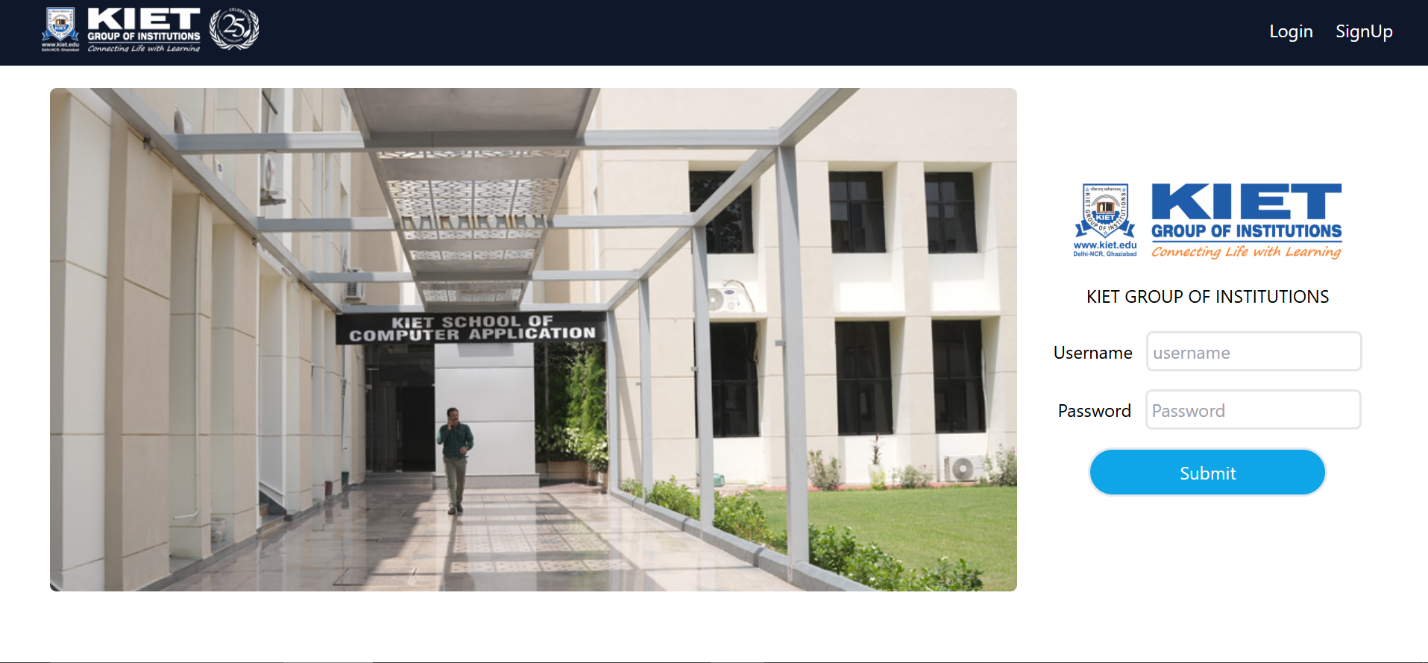
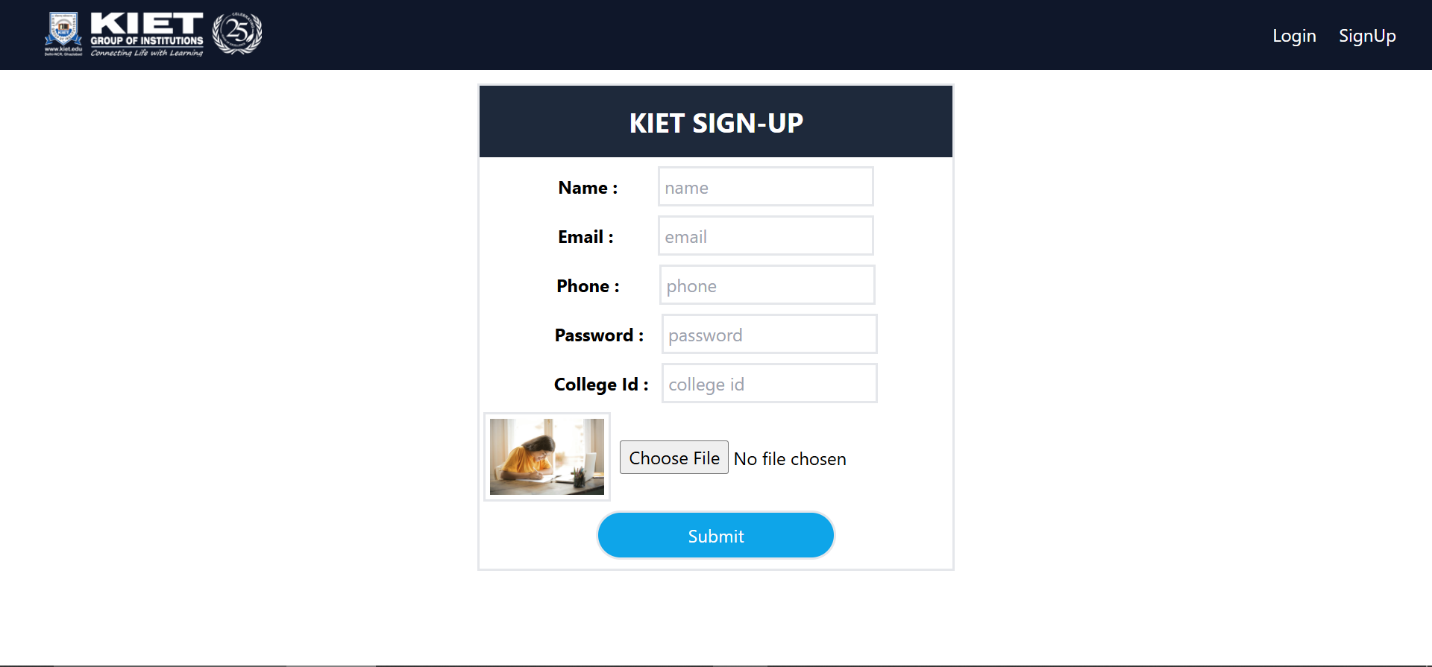
**4.2 Features**

* User authentication using secure login credentials.
* Leave request submission with fields for dates, reasons, and additional notes.
* An admin panel for department heads to view, approve, or reject leave applications.
* Automated email notifications to inform users of the status of their applications.

**4.3 User Interface Design**

The user interface is designed with simplicity and functionality in mind. Key pages include:

* **Homepage:** Provides an overview of the system with navigation links.
* **Leave Application Form:** Allows students to fill out and submit leave requests.
* **Admin Dashboard:** Displays pending and processed leave requests for review.



**Chapter 5: Testing and Validation**

**5.1 Test Cases**

|  |  |  |  |
| --- | --- | --- | --- |
| **Test Case ID** | **Description** | **Expected Outcome** | **Result** |
| TC01 | Student submits a leave request | Request recorded in the database | Pass |
| TC02 | Admin approves leave request | Email notification sent | Pass |
| TC03 | Invalid login attempt | Error message displayed | Pass |

**5.2 Results**

The Leave Approval System underwent comprehensive testing to ensure its reliability, functionality, and robustness under various scenarios. Below are the key outcomes of the testing phase:

**Functional Requirements**

1. **Student Leave Submission**:  
   The system successfully recorded student leave requests in the database upon submission through the web application. The process was verified to be error-free with accurate data storage.
2. **Admin Approval/Rejection**:  
   The admin interface enabled the department head to approve or reject leave requests seamlessly. The system processed these actions in real-time, updating the leave status instantly.
3. **Email Notifications**:  
   Notifications for approvals and rejections were successfully sent to the respective students' registered email addresses. This automated communication ensured timeliness and transparency.
4. **Login and Authentication**:  
   The login functionality was tested with valid and invalid credentials. Unauthorized access was prevented, and error messages were displayed for invalid login attempts.

**Non-Functional Requirements**

1. **Performance**:
   * The system handled multiple simultaneous users without any noticeable lag, demonstrating its scalability.
   * The response time for form submissions and updates was under 2 seconds.
2. **Reliability**:
   * The application demonstrated 100% uptime during the testing phase.
   * All leave requests were accurately logged, and no data loss occurred during operations.
3. **Usability**:
   * The web interface was user-friendly and intuitive, with straightforward navigation for both students and administrators.
   * Test participants rated the UI design highly for ease of use.
4. **Security**:
   * Passwords were encrypted, and secure API endpoints were used to protect data.
   * Testing for vulnerabilities such as SQL injection and cross-site scripting (XSS) showed no weaknesses.

**Chapter 6: Conclusion and Future Work**

**6.1 Conclusion**

The Leave Approval System has been able to streamline the leave application process of students and faculty. Digitalization of the workflow reduced manual effort, improved record-keeping, and ensured timely communication. The success of the project underlines the value of modern web technologies in enhancing administrative tasks.

**6.2 Future Work**

* Develop a mobile application to complement the web-based system.
* Introduce analytics for leave history, trends, and patterns.
* Integrate the system with existing institutional management tools to expand its functionality.

**References**

1. React Documentation. <https://reactjs.org>
2. Tailwind CSS Documentation. <https://tailwindcss.com>
3. Node.js Documentation. <https://nodejs.org>
4. MongoDB Documentation. <https://www.mongodb.com>