

## Software Requirement Gathering for Leave Management Portal

<b>Name</b>	<b>Bharath Raj G</b>
<b>Roll Number</b>	<b>7376222AD121</b>
<b>Project ID</b>	<b>22</b>
<b>Problem Statement</b>	<b>Leave Management Portal</b>

## Technical Components:

Frontend	React.js
Backend	Node.js
Database	MongoDB
API	RESTful API

-----

## Problem Statement:

To create a Leave management portal for educational institution for managing both faculties and students leave management.

## Document Purpose:

This document outlines the requirements for a Leave Management Portal designed for an educational institution, built using the MERN stack. The portal will cater to both faculty and student leave requests.

## **Stakeholders:**

- System Administrator
- Faculty Members
- Students

## **Functional Requirements:**

- **User Management:**

- System Admin: User creation, role assignment, and access control for faculty and students.
- Faculty: Login, profile management (update contact information), password reset, request leave, access control for students.
- Student: Login, View individual dashboard, request leave.

- **Leave Request Management:**

- Faculty/Student: Submit leave requests specifying type (sick leave, vacation, etc.), start date, end date, and reason for leave. Ability to attach supporting documents (optional).
- Faculty (Approver): View and approve/reject leave requests submitted by students they manage. Ability to approve or reject the request.
- System Admin (Optional): View all leave requests and reports.

- **Leave Tracking:**

- Faculty/Student: Track their leave balance and history of submitted requests.
- System Admin (Optional): Generate reports on leave trends and identify potential issues.

- **Notifications:**

System should send automated notifications to users regarding leave request status (approval/rejection) and upcoming leave expiry.

## **Non-Functional Requirements:**

- **Security:**

- Secure user authentication and authorization mechanisms.
- Data encryption for sensitive information (e.g., leave reasons, medical documents).

- **Performance:**

The portal should be responsive and handle concurrent user requests efficiently.

- **Scalability:**

The system should be scalable to accommodate a growing number of users.

- **Usability:**

- User-friendly interface with clear navigation and intuitive functionalities.
- Responsive design for optimal viewing across different devices (desktop, tablet, mobile).

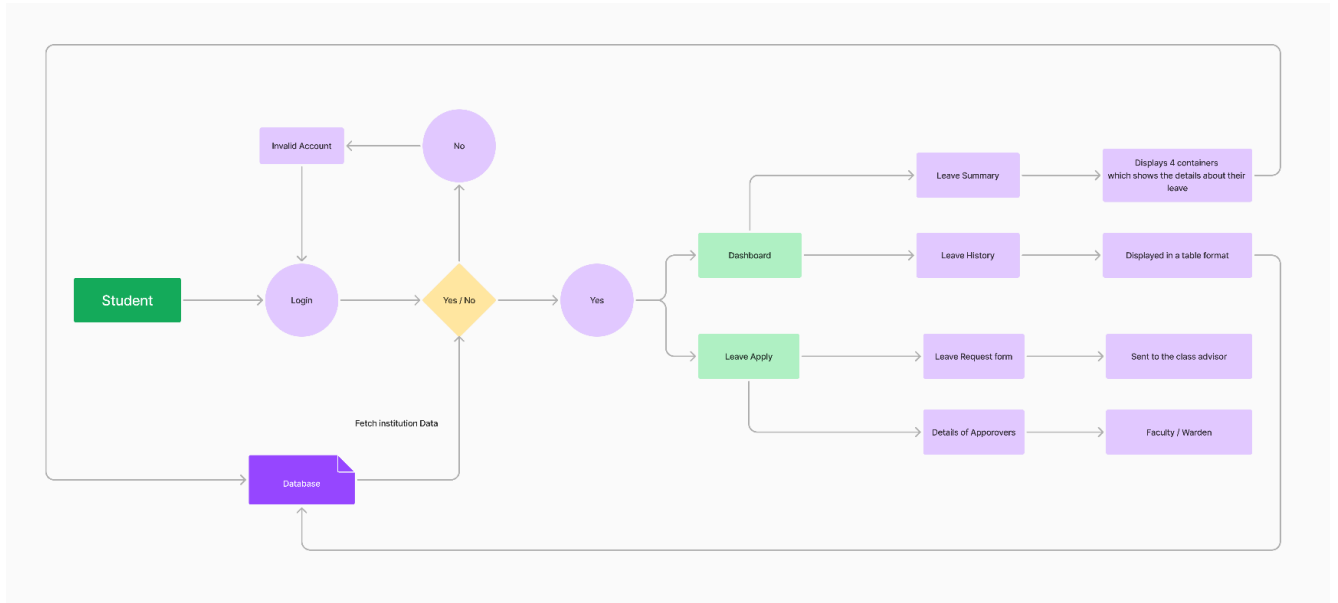
- **Accessibility:**

The portal should be accessible to users with disabilities, following WCAG guidelines.

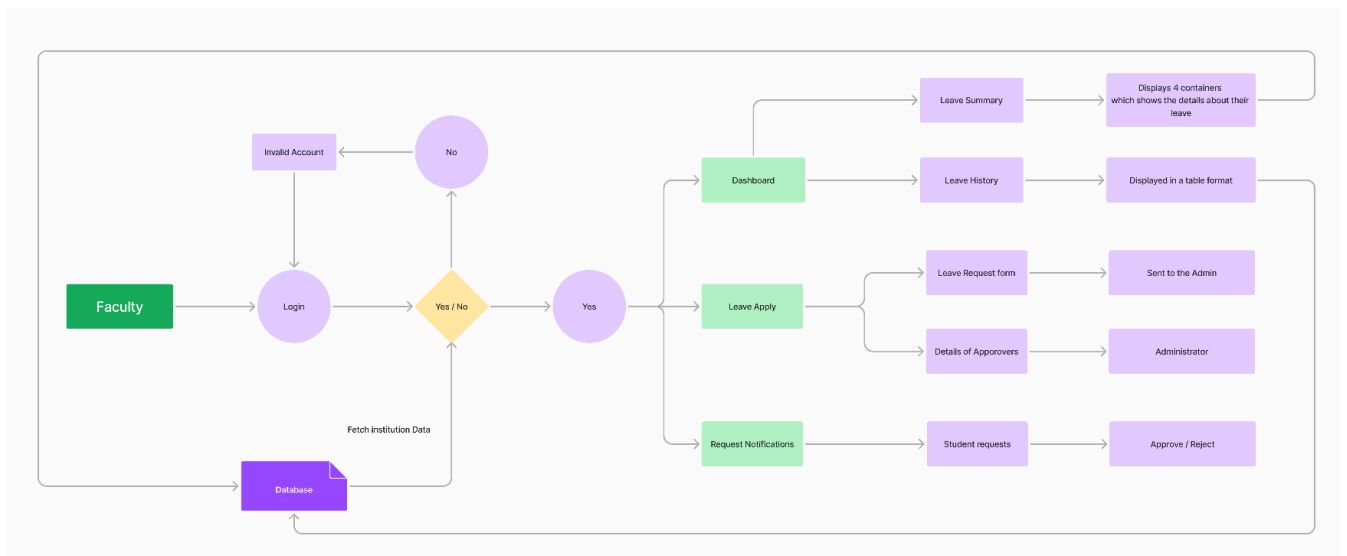
-----

## Flow Chart:

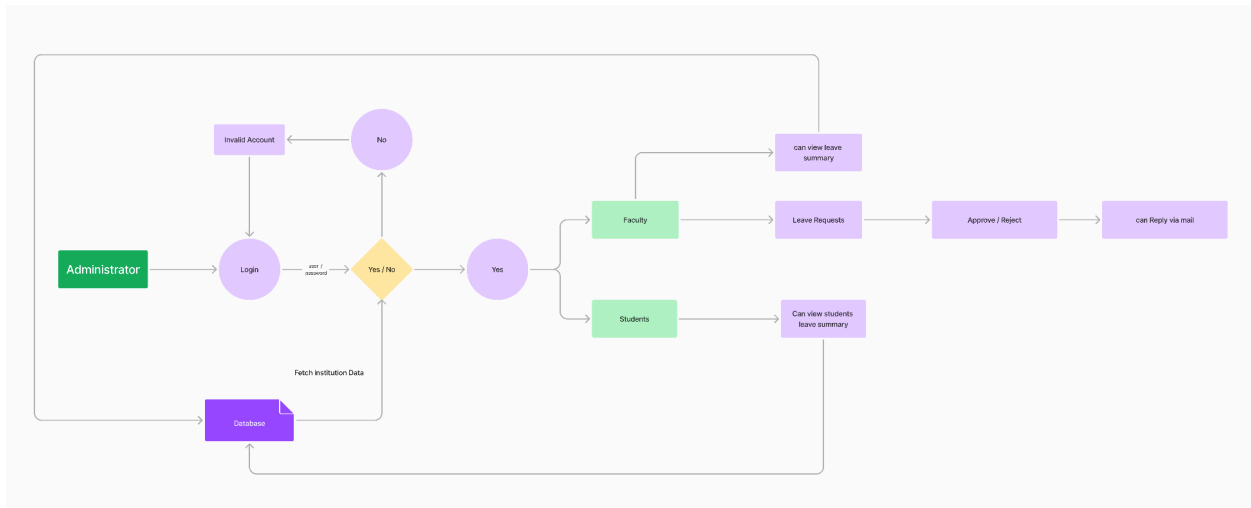
### Student:



### Faculty:



Admin:



-----