BANNARI AMMAN INSTITUTE OF TECHNOLOGY

An Autonomous Institution Affiliated to Anna University - Chennai, Accredited by NAAC with A+ Grade

Sathyamangalam - 638401 Erode District, Tamil Nadu, India

Name	**JAYAMURUGAN VM**
Roll number	**7376221EI117**
Seat number	**309**
Project number	**22**
Problem statement	**Leave Management System**

PROBLEM STATEMENT:

Develop a leave request and approval system for BIT (Bannari Amman Institute of Technology) using the MERN stack. The system should allow students and staff to submit various types of leave requests, ensure validation of leave durations, provide a summary of leave records, and enable designated authorities to approve or reject leave requests efficiently, enhancing the overall leave management process.

Key Objectives:

- Efficient Leave Management: Streamline the process of submitting, tracking, and approving leave requests for students and staff, reducing manual paperwork and administrative burden.
- Role-Based Access: Implement role-specific functionalities, ensuring appropriate access and actions for students, staff, and approvers to maintain system integrity and security.
- **Automated Validation:** Enforce leave duration validation and other rules to maintain compliance and prevent errors, ensuring that all leave requests adhere to institutional policies.
- **Comprehensive Record Keeping:** Maintain detailed leave summaries and histories for transparent and easy reference, allowing for efficient auditing and review of leave activities.

STACK:

Component	Tech Stack
Frontend	React js
Backend	Node.js with Express.js
Database	MongoDB
Authentication	JWT (JSON web Tokens) for secure login
Styling	CSS/Material-UI/Bootstrap

PROGRESS - TIMELINE:

Phase	Deadline	Status	Notes
Stage 1	03/04/2024	In Progress ▾	Planning and Requirement Gathering
Stage 2		Not Started ▾	Design and Prototyping
Stage 3		Not Started ▾	DB Designing
Stage 4		Not Started ▾	Backend Implementation
Stage 5		Not Started ▾	Testing & Implementation
Stage 6		Not Started ▾	Deployment

PROJECT OVERVIEW:

1. Purpose

The purpose of this project is to develop an efficient leave request and approval system for BIT (Bannari Amman Institute of Technology). The system aims to streamline the leave management process, reducing administrative workload and ensuring accurate tracking and approval of leave requests for both students and staff.

2. Scope

The project encompasses the design and implementation of a web application using the MERN stack, facilitating leave requests, validation, approval, and summary views. It includes role-based access control for students, staff, and approvers, ensuring secure and efficient management of leave processes across the institution

3. Business Context

The leave request and approval system will enhance operational efficiency at BIT by automating and streamlining leave management processes. This will reduce administrative overhead, ensure compliance with institutional policies, and provide accurate leave records, ultimately contributing to improved staff and student satisfaction and productivity.

4. Dependencies

Key dependencies for the project include Node.js and Express.js for backend development, React.js for the frontend, MongoDB for the database, and JWT for authentication. Additional libraries and tools like axios for API requests, bcrypt for password hashing, and Material-UI for styling may also be utilized.

5. Database Management System (DBMS)

The project will use MongoDB as the Database Management System. MongoDB is a NoSQL database that allows for flexible schema design, making it ideal for handling varied leave request data. It will store user information, leave requests, and approval statuses, ensuring efficient data retrieval and management.

6. User Personas

- **Students:** Submit leave requests, view leave summaries, and track approval statuses. They need a user-friendly interface to select leave types, enter details, and see the current status of their requests.
- **Staff:** Submit their own leave requests, view summaries, and approve or reject student leave requests. They require access to both personal leave management and student leave approval functionalities.
- **Approvers:** Designated authorities (e.g., wardens, department heads) responsible for approving or rejecting leave requests. They need a clear overview of pending requests and tools to manage approvals efficiently.

FUNCTIONAL REQUIREMENTS:

1. **User Authentication:**
   Description: Securely authenticate users to ensure that only authorized individuals can access the system.
   Features: Login with email and password, role-based access control for students and staff, JWT-based authentication for secure sessions.

2. **Idea Submission:**
   Description: Allow users to submit leave requests with necessary details.
   Features:
   Forms for leave type selection, date and time input, reason for leave, real-time validation to ensure compliance with leave policies.

3. **Evaluation and Approval:**
   Description: Enable designated approvers to review and manage leave requests.
   Features:
   Dashboard for viewing pending requests, options to approve or reject leave, add comments or remarks, automated status updates.

4. **User Interface:**
   Description: Provide a user-friendly interface for all user roles to interact with the system.
   Features:
   Responsive design for desktops and mobiles, intuitive navigation, clear and organized forms for leave requests and approvals, accessible design.

5. **Notification System:**
   Description: Keep users informed about the status of their leave requests and approvals.
   Features:
   Email notifications for request submissions and status changes, in-app notifications for pending approvals or updates, customizable alert settings.

6. **Reporting:**
   Description: Generate and view reports on leave requests and approvals for analysis and record-keeping.
   Features:
   Summary reports of leave taken, approval statistics, detailed records by user or department, export options for data analysis.

PROGRESS AND TOOLS BY STAGE:

Stage 1: Planning and Requirement Gathering

Tech Stack: N/A
Tools:
Trello or **Jira** for task management
Google Docs for documentation.
Slack or Microsoft Teams: For team communication and meetings.

Stage 2: Design and Prototyping

Tech Stack: N/A
Tools:
Figma or **Adobe XD** for creating wireframes and prototypes
Balsamiq for low-fidelity mockups.

Stage 3: Database Designing

Tech stack:
MongoDB
Tools:
MongoDB Atlas for cloud-based database management
Draw.io or **Lucidchart** for database schema design.

Stage 4: Backend Implementation

Tech Stack:
Node.js, **Express.js**, **MongoDB**
Tools:
Postman for API testing
Visual Studio Code for development
Git for version control.

Stage 5: Frontend Implementation

Tech Stack:
React.js, **Material-UI** or **Bootstrap**
Tools:
Visual Studio Code for development
Figma for design integration
Git for version control.
Axios: For handling HTTP requests.

Stage 6: Testing & Implementation

Tech Stack:

Jest or Mocha for unit testing**, Cypress** for end-to-end testing

Tools:

Jenkins for continuous integration
Sentry for error monitoring
Chrome DevTools for debugging.

Stage 7: Deployment

Tech Stack:

Heroku, **AWS**, or **Vercel**

Tools:

GitHub Actions for CI/CD
Docker for containerization
Cloudflare for CDN and security.

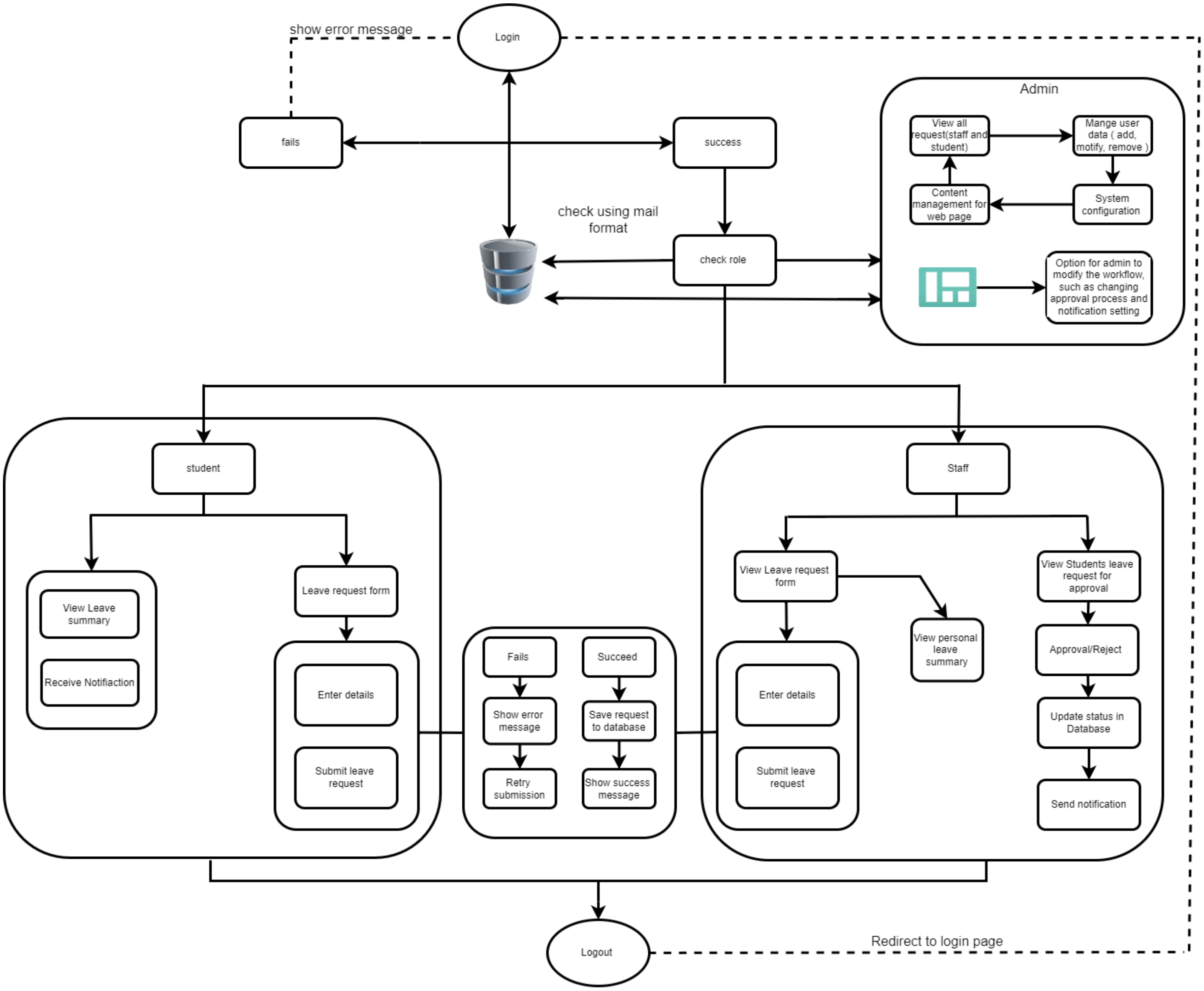
Strategies to Enhance Workflow:

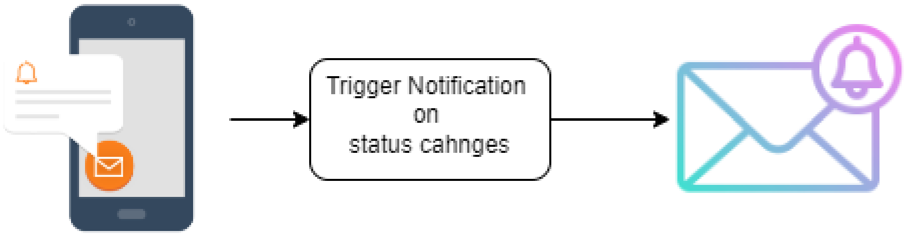
- **Implement Agile Methodology**: Adopt Agile practices to ensure iterative development, regular feedback, and continuous improvement. Use tools like **Jira** or **Trello** to manage tasks and track progress, which helps in adapting to changes quickly and maintaining a steady workflow.

- **Automate Testing and Deployment**: Utilize CI/CD pipelines with tools like **Jenkins** or **GitHub Actions** to automate testing and deployment processes. This reduces manual errors, speeds up delivery, and ensures that code changes are continuously integrated and deployed efficiently.

- **Enhance Communication and Collaboration**: Foster effective communication within the team using platforms like **Slack** or **Microsoft Teams**. Regularly hold meetings and updates to address issues promptly, share progress, and collaborate on problem-solving, which streamlines the workflow and keeps everyone aligned.

FlowChart :


show error message
Login
fails
success
check using mail format
check role
Admin
View all request(staff and student)
Mange user data (add, motify, remove)
Content management for web page
System configuration
Option for admin to modify the workflow, such as changing approval process and notification setting
student
View Leave summary
Receive Notifiaction
Leave request form
Enter details
Submit leave request
Fails
Show error message
Retry submission
Succeed
Save request to database
Show success message
Staff
View Leave request form
Enter details
Submit leave request
View personal leave summary
View Students leave request for approval
Approval/Reject
Update status in Database
Send notification
Logout
Redirect to login page


Trigger Notification on status cahnges