FULL STACK PROJECT- FACULTY LOG

Student Name : Afreen S

Seat no : 84

Project ID : 10

Project Title : Faculty Log

TECHNICAL COMPONENTS:

COMPONENT	TECH STACK
Backend	Spring Boot
Frontend	Angular / React js
Database	MySQL
API	RESTful service

PROBLEM STATEMENT:

The decentralized nature of email communication within educational institutions leads to several challenges, including:

● **Inconsistent messaging:** Different departments and administrative units send emails
independently, leading to duplication of information and inconsistent messaging.

● **Schedule conflicts:** Students and faculty receive multiple emails with overlapping schedules and events, leading to confusion and missed opportunities.

● **Fragmented communication:** Important announcements and updates get lost in the volume of emails, making it difficult for recipients to stay informed and engaged.

● **Administrative burden:** Managing email distribution lists, resolving conflicts, and ensuring timely delivery of critical information imposes a significant administrative burden on staff and faculty.

PROJECT-FLOW:

Purpose:

To develop a centralized mailing system that efficiently manages communication regarding Faulty's schedules and activities, resolving existing issues of schedule conflicts and communication inconsistencies.

Scope:

This system includes user authentication, a mailer request form, conflict checks, and a real-time dashboard for viewing and managing schedules. It integrates with existing email systems to ensure scheduled and conflict-free messaging.

Business Context:

The centralized mailing system is aimed at enhancing communication clarity and timeliness across BIT, thus boosting organizational efficiency by minimizing scheduling conflicts. Primary stakeholders include students, faculty, administrative staff, and the IT department.

Consideration:

- All users possess active Google accounts for authentication.
- Users have regular access to internet-enabled devices.

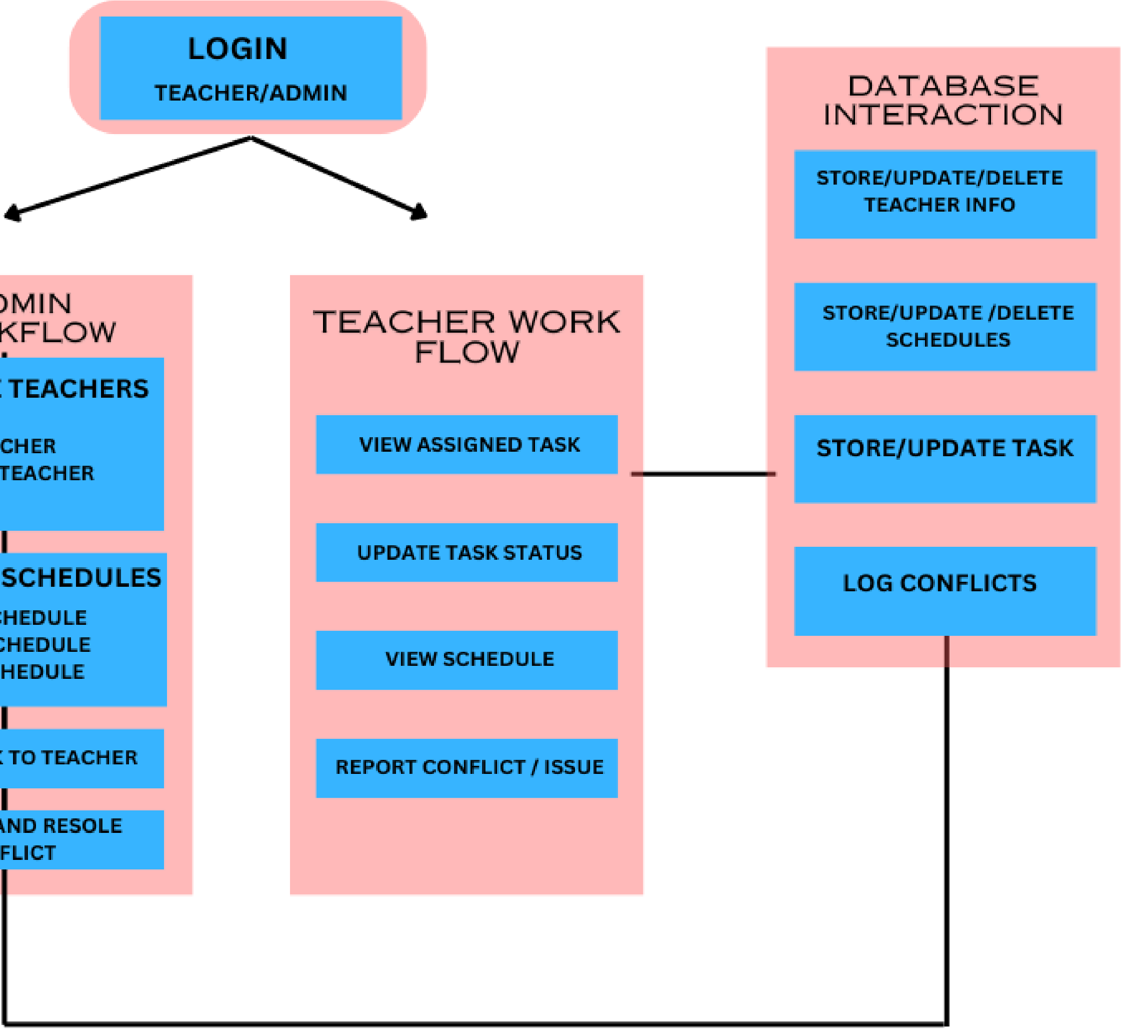
Dependencies:

- Integration with Google OAuth for user authentication.
- Consistent performance and availability of the existing email server.

Functional Requirements:

- User Authentication: Secure login using Google OAuth.
- Mailer Request Form: Users input mail content, scheduling time, category, and recipients.
- Conflict Resolution: Automatic detection of scheduling conflicts with options for adjustment.
- Dynamic Dashboard: Real-time schedule viewing and interaction.
- Priority Algorithm: Automated prioritization of communications based on rules.

WORKFLOW :



Explanation:

1. **User Authentication**:
   - ○ Both admin and teacher log into the system.
2. **Admin Workflow**:
   - ○ **Manage Teachers**: Admin adds or updates teacher information.
   - ○ **Manage Schedules**: Admin creates, updates, or deletes schedules.

   - **Assign Tasks to Teachers**: Admin assigns specific tasks to teachers.
   - **Monitor and Resolve Conflicts**: Admin addresses any schedule conflicts or issues.

3. **Teacher Workflow**:
   - **View Assigned Tasks**: Teachers view their assigned tasks.
   - **Update Task Status**: Teachers update the status of their tasks.
   - **View Schedule**: Teachers check their schedules.
   - **Report Issues/Conflicts**: Teachers report any conflicts or issues encountered.
4. **Database Interaction**:
   - The system interacts with the database to store, update, and manage information related to teachers, schedules, and tasks.