Student Name : Mohankumar S

Roll Number : 7376221cs227

Seat Number : 265

Project ID : 35

Project Title : Discipline committee

Full Stack Component: MEAN STACK

Component	Tech Stack
Front End	Angular (Js Framework)
Back End	Express.js (Web framework for Node.js) Node.js (JavaScript runtime environment)
Database	MongoDB (NOSQL Database)
API	RESTful API / GraphQL APIs

PROBLEM STATEMENT

Background:

The Discipline Committee (DC) at our college is responsible for managing student conduct issues, ensuring adherence to the institutional code of conduct, and handling disciplinary actions. Effective management of inquiries and follow-ups related to

disciplinary matters is essential for maintaining a fair and transparent process, but several challenges are currently affecting the committee's performance in these areas.

Problem: The current process for managing inquiries and follow-ups related to disciplinary issues is inefficient and lacks clarity, leading to several problems:

1.Delayed Responses to Inquiries: The committee often faces delays in responding to inquiries related to disciplinary matters, which can create frustration and a perception of unfairness among students and staff.

2.Inconsistent Follow-Up Procedures: There is a lack of standardized procedures for following up on ongoing disciplinary cases, resulting in inconsistent handling and potential oversights.

3.Fragmented Communication Channels: The communication process for handling inquiries and follow-ups is fragmented, with information being spread across multiple platforms, leading to confusion and inefficiencies.

4.Limited Tracking and Documentation: Inadequate tracking and documentation of inquiries and follow-ups make it difficult to monitor progress, identify patterns, and ensure that all issues are addressed in a timely manner.

5.Insufficient Transparency: The current process does not provide clear visibility into the status and resolution.

USER PERSONAS

• **Admin**

Requires the system to easily fetch the details of the students particularly with ease .

USER STORIES

1.I would like a detailed description of the inquiry procedure and schedule as a teacher taking part in a discipline committee inquiry so that I can adequately prepare and contribute.

2.As a teacher under investigation, I want regular updates on the status of the investigation and any decisions made, to reduce uncertainty and anxiety during the process. As a teacher facing a discipline committee inquiry, I need access to support resources and guidance on how to navigate the process, ensuring I can present my case adequately.

3.As a teacher facing a discipline committee investigation, I need a way to safely upload and distribute pertinent files and proof to back up my claims.

4.As a teacher waiting on the results of a discipline committee investigation, I want to be informed in a clear and concise manner about

1. PROJECT FLOW

User Registration and Login:

➔ Users (committee members, administrators, and individuals involved in disciplinary cases) register with the website.
➔ Users log in to access the system using secure credentials.

Dashboard:

➔ Upon logging in, users are greeted with a dashboard displaying relevant information such as pending cases, recent updates, and notifications.

Disciplinary Case Submission:

➔ Users can submit new disciplinary cases through a structured form, including details of the incident, involved parties, evidence, and nature of the violation.
➔ The system generates a unique case ID for tracking purposes.

Case Assignment and Review:

➔ The system automatically assigns cases to committee members based on availability and expertise.
➔ Committee members review case details, evidence, and statements submitted.

Enquiry Process:

➔ Committee members conduct an enquiry by interviewing involved parties, reviewing evidence, and gathering additional information.
➔ Progress updates and notes are recorded within the system for reference.

Decision Making:

➔ After the enquiry, committee members deliberate on the case and make a decision based on the information gathered.
➔ Decisions can include warnings, sanctions, dismissals, or recommendations for further action.

Notification and Communication:

➔ The system automatically notifies involved parties of the committee's decision.
➔ Users can communicate within the system regarding specific cases, seeking clarifications or providing additional information.

1.2 Scope

The project scope includes developing a secure student database management system for educational institutions. Key features are user authentication, student profile management, department dashboard for department details and batch-wise dashboard for specific details. The system aims to efficiently manage student data, enhance administrative processes and ensure data security.

2. System Overview

2.1 Users

• **Admins**

Only admins have access to the student database management system. They have access to view the student details in the college, retrieve the details related to batch details, department wise details and add the new student details to the database.

3.FUNCTIONAL REQUIREMENTS

- ● **User Roles:** Define different user roles (committee members, administrators, parties involved) with specific permissions.

- ● **User Registration:** Allow users to register with the system and manage their profiles.

- ● **Authentication:** Secure login functionality with password protection.

- ● **Case Submission:** Enable users to submit new disciplinary cases with all necessary details.

- **Case Assignment:** Automatically assign cases to committee members based on workload or expertise.

- **Case Tracking:** Provide a unique case ID for each case and allow users to track case progress.

- **Decision Recording:** Capture and store decisions made by the committee for each case.

- **Appeal Submission:** Enable parties to submit appeals against committee decisions.

4.NON-FUNCTIONAL REQUIREMENTS
1.Performance:

- The system should be responsive and provide quick response times to ensure efficient case management.
- It should be able to handle a large number of concurrent users without significant performance degradation.

2.Security:

- Data should be securely stored and transmitted using encryption protocols to protect sensitive information.
- User authentication mechanisms should be robust to prevent unauthorized access.

3.Scalability:

- The system should be designed to handle an increasing number of users, cases, and data over time without significant performance issues.
- It should be scalable both vertically and horizontally to accommodate growth.

4.Reliability:

- The system should have high availability, with minimal downtime for maintenance or upgrades.
- It should have backup and recovery mechanisms in place to prevent data loss.

5.Usability:

- The user interface should be intuitive and easy to use, with clear navigation and minimal training required for users.
- Accessibility features should be implemented to cater to users with disabilities.

6.Compatibility:

- The website should be compatible with a range of browsers and devices to ensure a consistent user experience.
- It should also be compatible with different operating systems and screen sizes.

7.Maintainability:

- The codebase should be well-structured and documented for ease of maintenance and future enhancements.
- Regular updates and bug fixes should be implemented to ensure system stability.

DATABASE DESIGN

1.Batch 2022 entity [Same table design for 2020,2021,2023 batch]

Name	String
Roll number	Data data type
Department	String
Year	Data data type
E-mail ID	String
Attendance percentage	Data data type

FLOWCHART

start

Faculty Access

User Authentication — invalid → logout

student name and roll number — student details → DB

place and incident time

received by admin team

proceed with enqiury — no → stop

Determine Action ← yes

strictly warning

stop

Fine amount if eligible

Fine amount show in dashboard

susipension ← yes

yes — Attendence percentage → DB

start date and end date with timing

suspension details added to dashboard

mentor will receive message

Report details

message receive before 2 days suspension date → End