ABSTRACT

EduPulse: A MERN-Based Feedback & Interaction Platform for Students, Faculty, and Administrators

This project presents the design, development, and deployment of **EduPulse**, a full-stack **Software-as-a-Service (SaaS)** platform implemented using the **MERN stack — MongoDB, Express.js, React.js, and Node.js**. EduPulse enhances faculty-student interaction by providing structured feedback mechanisms, real-time communication, and transparent performance insights. It empowers students to share daily classroom feedback, faculty to analyze teaching effectiveness, and administrators to monitor institutional progress through a secure, role-based environment.

The application integrates **modern authentication**, **role-based dashboards**, and **real-time WebSocket-powered chat** for seamless interaction between students, faculty, and admins. Its modular architecture ensures **scalability, security, and responsiveness** while maintaining a simple and intuitive user experience.

Key Features

1. **User Authentication & Role Management**
    - Secure sign-up and login with JWT-based authentication.
    - Role-based dashboards for Students, Faculty, and Admins.
    - Session handling and restricted access to protected routes.
2. **Feedback & Reporting**
    - Daily and monthly student feedback submissions.
    - Attendance-based restrictions to prevent false entries.
    - Faculty dashboards to view, filter, and compare feedback.
    - Admin dashboards for department-wise performance reports.
3. **Real-Time Communication**
    - WebSocket-based chat for Student ↔ Faculty and Student ↔ Admin.
    - Broadcast announcements for urgent updates.
    - Anti-ragging support chat with admin for immediate help.
4. **Timetable & Mapping**
    - Pre-loaded timetable visibility for students.
    - Faculty-subject-student mapping (create, edit, and update).
5. **Reward & Engagement System**
    - Students earn points for consistent feedback.
    - Monthly surveys and polls for academic improvements.

6. **Admin Tools**
    - Manage users and roles (add, update, delete).
    - Track feedback statistics and submission rates.
    - Export reports in tabular formats (PDF/Excel).

Architecture

- **Frontend (React.js + Bootstrap/Tailwind CSS)**: Responsive Single Page Application (SPA) with protected routes and dynamic dashboards.
- **Backend (Express.js + Node.js)**: REST APIs for authentication, feedback, reporting, and chat integration.
- **Database (MongoDB)**: Stores user profiles, feedback, timetable, and reports with role-based access.
- **Authentication (JWT)**: Ensures secure access and role-based restrictions.
- **Real-Time Communication (Socket.IO)**: Enables chat and notifications for instant interaction.

Security Measures

- **JWT-based Authentication** with role-encoded claims.
- **Access Control** for Students, Faculty, and Admins.
- **Input Validation & Sanitization** to prevent malicious entries.

Performance & Scalability

- **Modular MERN architecture** for easy scaling of services.
- **Horizontal Scaling** possible for backend services.
- **Optimized MongoDB queries** for fast feedback retrieval.
- **WebSocket rooms** for scalable chat handling.

Development Process

1. **MVP Phase** – Authentication, role-based dashboards, and feedback system.
2. **Engagement Phase** – Timetable, rewards, and surveys.
3. **Interaction Phase** – WebSocket-based chat and notifications.
4. **Reporting Phase** – Admin analytics and export features.

Technologies Used

- **Frontend**: React.js, React Router, Bootstrap/Tailwind CSS
- **Backend**: Node.js, Express.js, Socket.IO
- **Database**: MongoDB
- **Authentication**: JWT (JSON Web Token)
- **Deployment**: Vercel/Netlify (frontend), Render/Heroku (backend), MongoDB Atlas (database)

Impact & Benefits

- **For Students**: Transparent feedback system, reward points, and real-time support.
- **For Faculty**: Performance insights, comparative analysis, and mentorship support.
- **For Admins**: Efficient user management, department-wide reporting, and anti-ragging measures.
- **For Institutions**: Data-driven decisions, improved teaching quality, and stronger student engagement.

EduPulse demonstrates the integration of structured feedback systems, real-time communication, and secure user management into a **scalable MERN application**, showcasing strong full-stack engineering and practical problem-solving for the educational domain.