FEASIBILITY REPORT

# Centralized College Society Management Portal

Course: Software Engineering

College Name: [Your College Name Here]

Team Name: [Add Team Name]

Team Members:

Member 1 (Roll No)

Member 2 (Roll No)

Member 3 (Roll No)

Date: 20-08-2025

# 1. Introduction

Project Title: Centralized College Society Management Portal  
  
Problem Statement:  
Currently, most college societies and clubs rely on emails or social media platforms like Instagram to share event updates, recruitment forms, and reminders. These channels are not ideal because emails often get buried under academic communications, and Instagram feeds are cluttered with unrelated content, causing important updates to go unnoticed. This leads to missed opportunities for students and low participation rates in college activities.  
  
Proposed Solution:  
To address this issue, we propose developing a web-based centralized platform where all college societies can register and share their announcements, event updates, and recruitment forms. Students can log in using their college email IDs (or application number for freshers) to access a personalized dashboard showing relevant events. Additionally, a calendar feature will allow students to view upcoming events and set reminders for the ones they plan to attend.  
  
Approach:  
We will build the platform using the MERN stack (MongoDB, Express.js, React.js, Node.js). The portal will feature secure authentication (college ID verification), event management modules, and an interactive calendar to enhance student engagement.

# 2. Project Objectives

Goals:  
- Create a single platform for all college societies to post announcements, event details, and recruitment forms.  
- Allow students to view relevant events and set personal reminders.  
- Provide a transparent registration system, showing available seats in real time for events with limited capacity.  
  
Expected Benefits:  
- Reduce missed event notifications by providing a centralized platform.  
- Increase participation in college activities.  
- Ensure transparency in event registration with real-time availability status.

# 3. Feasibility Analysis

3.1 Technical Feasibility  
- Technology Stack:  
 - Frontend: React.js  
 - Backend: Node.js with Express.js  
 - Database: MongoDB  
 - Authentication: Email-based verification using college IDs or application numbers  
 - Additional Features: Event calendar, reminders, seat availability tracker  
 - Hosting: Options include Heroku, Vercel, or college servers  
  
- Development Skills:  
 Our team has basic knowledge of frontend technologies and databases. We will learn backend integration, authentication methods, and hosting during development.  
  
- Infrastructure:  
 Initial deployment on free cloud hosting (e.g., Vercel, MongoDB Atlas); scalable architecture for future expansion.

3.2 Operational Feasibility  
- Users: Students, society heads, and college administrators  
- Interaction:  
 - Students can register using college credentials, browse events, and set reminders.  
 - Society heads can post announcements, upload recruitment forms, and manage events.  
 - Admins can monitor overall activity and verify societies.  
- User Adoption: The platform will be simple and intuitive, requiring no special training.  
- Integration: Standalone solution since no similar system exists in our college.

3.3 Legal & Ethical Feasibility  
- Data Privacy: We will store only necessary details (gender, year of study, etc.) and protect them with authentication and secure database handling.  
- Fake Accounts: Restricted registration via verified college email IDs and application numbers to prevent misuse.

# 4. Risk Assessment

Potential Risks:  
- Technical Challenges: Backend and authentication implementation may be new for us.  
- Time Constraints: Limited to 3 months before exams.  
- Data Security: Unauthorized access or leaks if not properly implemented.  
- Adoption Risk: Societies may initially hesitate to switch from Instagram/mail.  
  
Mitigation Strategies:  
- Learning plan for backend and authentication in the first few weeks.  
- Use open-source libraries for secure authentication.  
- Regular testing to identify bugs early.  
- Early engagement with society heads for feedback and support.

# 5. Implementation Timeline

- Week 1-2: Research and design UI/UX, finalize feature list.  
- Week 3-4: Build frontend components and authentication system.  
- Week 5-6: Implement backend and database integration.  
- Week 7: Add event calendar, reminder system, and registration module.  
- Week 8: Testing and bug fixing.  
- Week 9-10: Deployment, final presentation, and documentation.

# 6. Conclusion & Recommendations

The proposed Society Management Portal is technically and operationally feasible within the given timeframe. With proper planning and phased implementation, it can significantly improve event communication and participation in college activities. We recommend proceeding with the project and considering future upgrades like a mobile app and push notifications for better engagement.