Campus Connect

1. Project Statement:

In a university setting, students, faculty, and clubs often face challenges in communication, event management and academic tracking. The lack of unified platform leads to fragmented information, inefficient processes and reduced engagement. The University Engagement and Management System (UEMS) aims to address these issues by providing an integrated platform that enhances interaction, event management, academic tracking for all university stakeholders.

2. Approximate duration (in hours) to complete the project:

Weeks	Time	Work to be done
Week 1	4 hrs	Project Discussion
Week 2	10 hrs	Initial UI Design & Setup
Week 3	2 hrs	Login Page & Role Bases UI
Week 4	6 hrs	Student Pages UI
Week 5	5 hrs	Club Pages UI
Week 6	6 hrs	Teacher Pages UI
Week 7	3 hrs	Admin Pages UI
Week 8	1 hrs	Final Touch

Table1: Timeline

3. Proposed Project In-charge:

Ms. Preenu Mittan

4. Team Members, along with roll no's:

Bhuvesh Mittal: 2210991450
Daanushi Sharma: 2210991469

5. Check Points:

a. **Does the project statement result in a product? If yes, what type of product?**
Yes, the project statement results in a product. It is a web-based university engagement and management platform. This product facilitates various functionalities for students, faculty, and clubs within a university setting.

b. If it is a product, can a prototype be made, if not, what is it, which we can produce that our teachers can evaluate.

Yes, a prototype can be made. The prototype would be a working version of the web application demonstrating the key features such as:

a. User authentication (students, faculty, clubs)
b. Event management (creating, joining, viewing events)
c. Quiz management (uploading, taking quizzes)
d. Attendance tracking
e. Profile management
f. Club activities and management

c. Does the project statement use multiple concepts to achieve the outcome?

Yes, the project statement uses multiple concepts to achieve the outcome. These include:

a. Web Development: Frontend (React.js) and Backend (Node.js, Express.js)
b. Database Management: PostgreSQL or MongoDB
c. Authentication and Authorization: Implementing secure login and role-based access control
d. Event Management: Creating and managing events
e. Quiz and Attendance Management: Uploading quizzes and tracking attendance
f. User Experience: Designing user interfaces and user experience flows

d. Does it have enough for our team members to do enough work?

Yes, the project has enough scope for team members to engage in a sufficient amount of work. The tasks can be divided into different modules such as:

a. Frontend Development: Designing and implementing the user interface
b. Backend Development: Setting up the server, database interactions, and API development
c. Authentication: Implementing secure login and role management
d. Real-Time Features: Developing the real-time chat functionality
e. Event and Quiz Management: Creating the logic for event creation, joining, and quiz management
f. Database Design: Designing and managing the database schema

6. Technical Nodes

Subject / Area / Topic	Technical Nodes
Frontend Development	EJS
Backend Development	Node.js, Express.js
Database Management	MongoDB
Authorization and Authentication	JWT

Table 2: Technical Nodes

7. Prerequisites:

a. Knowledge

1. Familiarity with frontend frameworks (React.js)
2. Knowledge of backend development (Node.js, Express.js)
3. Understanding of relational and non-relational databases (PostgreSQL, MongoDB)
4. Basic concepts of authentication and authorization
5. Familiarity with RESTful API design

b. Concepts

1. MVC (Model-View-Controller) architecture
2. RESTful APIs
3. CRUD operations
4. Asynchronous programming and Promises
5. State management in frontend frameworks
6. Responsive design principles

c. Materials

1. Tutorials and documentation for React.js, Node.js, Express.js, PostgreSQL, and MongoDB
2. Example projects and source code for similar applications
3. Books and online courses on full-stack web
4. Design resources for creating user interfaces (wireframes, mockups)

8. Material that may be required to make the project and where it might be available

a. Development Tools

1. IDE/Text Editor: Visual Studio Code
2. Version Control: Git, GitHub (available at Git and GitHub)
3. Package Manager: npm (included with Node.js installation from Node.js)

b. FrontEnd

1. EJS: Express.js

c. BackEnd

1. Node.js & Express.js: Node.js, Express.js
2. JWT: jwt

d. DataBase

1. MongoDB: MongoDB

9. What could the total cost of the project?

N/A

10. Resources available to us:

a. Computer/Laptops
b. Development Tools
c. Internet Access
d. Guidance from faculty
e. Access to online tutorials & documentation