CHITKARA UNIVERSITY

Edemy – E-Learning Platform

1. Project Statement:

Online education has become essential, but existing platforms face challenges that hinder accessibility, engagement, and affordability. Some common issues include:

- **Limited Accessibility**: Many platforms lack support for different learning needs.
- **Lack of Interactive Learning**: Absence of live Q&A, gamification, and peer discussions.
- **High Subscription Costs**: Quality education remains expensive for many users.
- **Inefficient Course Discovery**: Poor recommendation algorithms make it hard for learners to find relevant content.
- **Security Concerns**: Ensuring user data privacy and secure transactions is critical.

Edemy aims to solve these issues by providing an intuitive, AI-powered e-learning platform with affordable access to high-quality courses, interactive learning features, and a seamless user experience.

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

WEEKS	DURATION	WORK TO BE DONE
Week 1	4 hrs	Project Discussion & Requirements Gathering & Planning
Week 2	8 hrs	Initial Setup (Backend, Frontend, Database)
Week 3	8 hrs	User Authentication & Profiles
Week 4	12 hrs	Course Management System Development
Week 5	12 hrs	Course Recommendation Module

Week 6	12 hrs	Community Features
Week 7	12 hrs	Testing & Debugging
Week 8	12 hrs	Deployment & Final Review

Table1: Timeline

3. Proposed Project In-charge:

Dr. N. Bharathiraja

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

Anirudh Jamwal: 2210991267
Ankit Garg: 2210991277
Anmol Singh: 2210991286
Ansh Dabra: 2210991291

5. Check Points:

a. **Does the project statement result in a product? If yes, what type of product?**
Yes, this project results in an e-learning platform that allows users to enroll in, access, and interact with online courses.

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 for the e-learning website. The prototype would demonstrate key features such as:
a. **User Authentication:** Secure account creation and login.
b. **Property Search and Listings**: Search for properties based on criteria and view detailed listings.
c. **Wishlist Management:** Add or remove properties from a Wishlist.
d. **Communication Tools**: Real-time chat for user-agent interaction.
e. **Responsive Design:** Mobile-friendly layout for various devices.
This prototype would effectively showcase the core functionalities of the real estate website for evaluation.

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) technologies.
b. **Database Management:** Using MongoDB for data storage.
c. **Authentication and Authorization:** Implementing secure login and role-based access control.
d. **Marketplace Management:** Creating and managing property listings.
e. **User Experience**: Designing intuitive interfaces and user 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, managing database interactions, and developing APIs.
c. **Authentication:** Implementing secure login and role management.
d. **Marketplace Development:** Creating the logic for property listings and transactions.
e. **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, MongoDB
2. Example projects and source code for similar applications
3. Books and online courses on full-stack web development
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