PROJECT DOCUMENTATION

Education Platform with Courses Website Using Mean Stack

Team Members:

- **Parth Patel (2203051050759)**
- **Mit Parmar (2203051050347)**
- **Prerna Solanki (2203051050014)**
- **Harsh Bosamiya (2203051050211)**
- **Mohit Jobanputra (2203051050740)**

Under the Guidance of:

Maksud Vohra Sir

Table of Contents

10. Evaluation Criteria
11. Submission Guidelines
12. References

Introduction

Educational platform with courses project using MEAN(MongoDB,Express.js,angular,Node.js) Stack

1. Purpose

The purpose of this project is to develop a comprehensive **Education Platform with Courses** that offers flexible, personalized, and engaging learning experiences. It aims to make quality education accessible and adaptable for learners of all backgrounds, levels, and goals.

---

2. Background

Traditional learning models often lack flexibility, personalization, and the interactive tools needed to keep learners motivated. Our platform addresses these gaps by leveraging technology to provide dynamic, self-paced learning paths. Inspired by popular platforms like GeeksforGeeks, this project integrates various learning styles, community engagement, and credentialed instruction into a unified system.

---

3. Scope

This platform will focus on providing:

- Courses across diverse subjects, including technology, business, arts, and humanities.
- Multi-level learning options (beginner, intermediate, and advanced).
- Personalized learning paths tailored to user preferences.
- Features for self-paced and real-time learning.
- Interactive tools like quizzes, hands-on projects, and community support.
- Access across mobile and desktop platforms.

---

4. Problem Statement

Access to quality, affordable, and tailored education remains a significant barrier for many learners. Existing platforms may not fully cater to different learning styles or provide sufficient personalization. Our project addresses these challenges by offering a customizable, user-centric, and engaging learning experience.

---

5. Solution

Our solution is an **Education Platform with Courses** that:

- Offers personalized course recommendations and learning paths.
- Provides interactive tools, including quizzes, projects, and forums.
- Supports flexible learning options: self-paced and live sessions.
- Includes credentialed instructors with proven expertise.

---

6. Objectives

1. Develop a user-friendly interface for browsing courses and managing learning progress.
2. Implement personalized course recommendations and adaptive learning paths.
3. Incorporate interactive content, such as quizzes and projects, for hands-on practice.
4. Provide comprehensive support through live chat, email, and a knowledge base.
5. Ensure platform scalability and continuous improvement.

---

7. Methodology

7.1 Data Collection

- **User Feedback**: Gather insights from learners to understand preferences and challenges.
- **Instructor Input**: Collect feedback on course management needs.
- **Market Research**: Analyze leading platforms to identify best practices and innovations.

7.2 Tools and Technologies

- **Frontend**: HTML, CSS, JavaScript for responsive user interfaces.
- **Backend**: Python or Node.js for server-side logic and APIs.
- **Database**: MongoDB or SQL for managing course data and user profiles.
- **Authentication**: JWT (JSON Web Tokens) for secure login.
- **Mobile Access**: Responsive design or mobile app support.

7.3 Implementation Process

1. **Requirement Analysis**: Define system requirements and user stories.
2. **System Design**: Develop architecture, database schema, and wireframes.
3. **Frontend Development**: Build interfaces for course browsing, quizzes, and learning paths.
4. **Backend Development**: Implement APIs for user data, progress tracking, and course management.
5. **Testing**: Conduct unit, integration, and user acceptance testing.
6. **Deployment**: Host the platform on a cloud service for public access.

8. Project Plan

8.1 Timeline

- **Week 1-2**: Requirement gathering, design, and system architecture.
- **Week 3-4**: Frontend development for key features.
- **Week 5-6**: Backend implementation and database integration.
- **Week 7**: User interface refinement and interactive features.
- **Week 8**: Testing and quality assurance.
- **Week 9**: Deployment and documentation.

8.2 Resources Needed

- **Frontend Developer**: For user interface and responsive design.
- **Backend Developer**: For API development and database management.
- **UI/UX Designer**: For wireframes and user experience.
- **Testing Tools**: Mocha, Jest, or similar frameworks.

---

9. Expected Deliverables

1. Fully functional education platform.
2. Personalized learning path feature.
3. Interactive quizzes and community forum.
4. Mobile-responsive design.
5. Complete documentation and user manuals.

---

10. Evaluation Criteria

- **Functionality**: Course browsing, quizzes, and progress tracking.
- **Usability**: Intuitive navigation and user engagement.
- **Performance**: Efficient handling of user data and interactions.
- **Security**: Proper authentication and data privacy.
- **Scalability**: Capacity for future feature expansions.

---

11. Submission Guidelines

1. Source code with comments and documentation.
2. Project report detailing objectives, design, and implementation.

3. Live demo or deployment link.

---

12. References

1. Books:

- “Learning Angular” by Brad Green and Shyam Seshadri.
- “Node.js Design Patterns” by Mario Casciaro.
- “MongoDB: The Definitive Guide” by Kristina Chodorow

2. Websites:

- MongoDB Documentation
- Angular Official Documentation
- Node.js Official Documentation
- Stripe API Documentation o [Express.js
