EduTutor AI - Project Documentation

1. INTRODUCTION

1.1 Project Overview

Edu Tutor AI is a web-based personalized learning platform designed to assist students in mastering advanced mathematics through a dynamic and interactive dashboard. It integrates AI tutoring, a learning progress tracker, personalized learning paths, and Google Classroom integration.

1.2 Purpose

The goal is to provide an engaging digital companion for learners, empowering them to achieve subject mastery at their own pace while integrating seamlessly with existing education platforms.

---

2. IDEATION PHASE

2.1 Problem Statement

The goal is to provide an engaging digital companion for learners, empowering them to achieve subject mastery at their own pace while integrating seamlessly with existing education platforms.Educator:

2.2 Empathy Map

- **Think & Feel:** "I want to understand better, but I get stuck."
- **See:** Class dashboards with no personal feedback or adaptive help
- **Hear:** Encouragement from teachers, peer pressure
- **Say & Do:** Ask for help, re-watch tutorials, occasionally give up
- **Pain:** Gets lost in topics, doesn't know where to begin or how far they've come
- **Gain:** Wants a companion that adapts, encourages, and informs them

- **2.3 Brainstorming**
  - AI-powered question answering
  - Dynamic learning path adjustment
  - Google Classroom sync
  - Chat interface for a friendly tutor
  - Visual dashboard showing streaks and completed topics
  - Progress Tracking
  - Analytics Dashboard

---

3. REQUIREMENT ANALYSIS

3.1 Customer Journey

From registration to personalized AI sessions, students experience a guided flow: Landing → Signup/Login → Dashboard → Complete Lessons → Chat with AI → Track Progress

3.2 Solution Requirements

- Frontend: React (with routes for login, dashboard)
- Backend: Node.js + Express with Google OAuth
- Google Classroom API for fetching course data
- MongoDB for storing user and progress records
- Session-based authentication

3.3 Data Flow Diagram

- NFR-1: Usability: Simple and intuitive Streamlit interface
- NFR-2: Security: Hashed passwords using Werkzeug
- NFR-3: Performance: Quiz generation under 2 seconds
- NFR-4: Reliability: Handles DB reconnects and reruns
- NFR-5: Availability: Deployed via Streamlit cloud with minimal downtime

3.4 Data Flow Diagram

- User → Login → Backend → Authentication via Google → Session Created User → Dashboard → Request Progress & AI Tutor → Backend → Response
- Admin → UI → Dashboard analytics → View student KPIs

Entities: Student, Educator, Dashboard, Coures Quizzes.

Level 0: Generate quiz → Submit answers → Evaluate → Save → Display progress

3.6 Technology Stack

- **Frontend:** React, CSS
- **Backend:** Node.js, Express.js
- **Database:** MongoDB
- **Authentication:** Google OAuth via Passport.js
- **External APIs:** Google Classroom API

---

4. PROJEC DESIGN

4.1 Problem Solution Fit

A guided, gamified platform with AI guidance offers deeper learning engagement and visibility over progress.

4.2 Proposed Solution

Edu Tutor AI customizes a student’s dashboard with real-time streaks, subject focus, and personalized lessons supported by AI-generated insights and conversation.

4.3 Solution Architecture

- React frontend interfaces with backend APIs
- Passport manages Google login session
- Backend routes handle authentication, user session, and Google API calls
- MongoDB stores persistent user data
- AI chatbot logic embedded in dashboard interface

---

5. PROJECT PLANNING & SCHEDULING

5.1 Project Planning

- **Week 1–2:** Ideation, backend setup with Google OAuth ,Frontend UI buildout and dashboard design
- **Week 3–4**：Frontend UI buildout and dashboard design,Google Classroom integration & AI tutor prototype
- **Week 5**： Testing and deployment to Render/Vercel

---

6. FUNCTIONAL AND PERFORMANCE TESTING

6.1 Performance Testing

- Tested Google OAuth latency and redirect flow
- Verified dashboard loads with minimal delay across devices
- Tested concurrency with multiple students logging in simultaneously
- AI chat response time remained under 300ms (local)

---

7. RESULTS

7.1 Output Screenshots

- Login Page with Google sign-in
- Dashboard with progress bars and AI tutor
- Personalized Learning Path flow
- Classroom course list rendered from Google Classroom API

---

8. ADVANTAGES & DISADVANTAGES

Advantages

- Personalized learning path visualizations
- AI tutor for targeted practice
- Seamless Google Classroom integration
- Modern UI for engagement
- Session persistence

Disadvantages

- Requires internet connection
- Dependent on Google OAuth session
- AI chatbot currently uses rule-based or placeholder logic

---

9. CONCLUSION

Edu Tutor AI serves as an intelligent, supportive learning companion that blends AI-driven help with real user data and learning structure. Its goal is to make education personal, effective, and motivating through interactive design and smart integration.

---

10. FUTURE SCOPE

- Upgrade AI tutor to use real-time LLMs via an API
- Add quiz and worksheet modules
- Gamify dashboard rewards and add leaderboards
- Provide analytics to educators
- Mobile app version for wider access

---

11. APPENDIX

- **Source Code:** Available in /frontend(https://edu-tutor-ai-ten.vercel.app)/ and /backend(https://edu-tutor-ai.onrender.com)/ directories
- **Dataset:** N/A (API-based application)
- **GitHub:** https://github.com/Reita-123/edu-tutor-ai.git
- **Live Demo:** https://edu-tutor-ai-reitas-projects-0d108c7a.vercel.app

---

End of Document