# EduTutor AI
# Project Documentation

1. Introduction
   • Project Title: EduTutor AI: Personalized Learning with Generative AI and LMS Integration
   • Team Members:

- Team Member 1: K.DEEPIKA
- Team Member 2: A.MONICA
- Team Member 3: R.JAYASRI
- Team Member 4: B.PAVITHRA(26/11)

2. Project Overview
   • Purpose:
   EduTutor AI is designed to provide personalized learning experiences for students and actionable insights for educators. It leverages AI and LMS integration to optimize learning, generate quizzes, evaluate student performance, and provide real-time feedback.

• Features:

- Personalized Quizzes: Auto-generated based on course content.
- Adaptive Learning: Quiz difficulty adjusts based on performance.
- Instant Feedback: Immediate scoring and explanations.
- Student Dashboard: Tracks progress and quiz history.
- Educator Dashboard: Analytics on student performance.
- Google Classroom Integration: Sync courses and topics.
- Diagnostic Testing: Initial assessment to personalize learning.
- Insights & Recommendations: Highlights weak areas for improvement.
- Streamlit Frontend UI: User-friendly web interface.

3. Architecture
   • Frontend (Streamlit): Interactive web UI with dashboards, quiz forms, and

progress charts.
• Backend (FastAPI): REST APIs to handle quiz generation, evaluation, and data management.
• LLM Integration (IBM Watsonx Granite): Generates quizzes, explanations, and adaptive content.
• Vector Search (Pinecone): Stores and retrieves topic insights and explanations.
• ML Modules: Adaptive difficulty algorithms and keyword-based short-answer evaluation.

4. Setup Instructions
   Prerequisites:

- Python 3.9 or later
- pip and virtual environment tools
- Internet access

Installation:

1. Clone the repository
2. Install dependencies from requirements.txt
3. Configure any required API keys (if applicable)
4. Run FastAPI backend server
5. Launch Streamlit frontend
6. Sync courses, upload data, and start interacting with the platform
7. Folder Structure

- app/ – FastAPI backend logic
- core/ – Quiz engine, evaluation, auth, and vector search modules
- data/ – Sample courses and topic JSON files
- app.py – Streamlit frontend entry script

6. Running the Application

- Launch FastAPI server for backend
- Run Streamlit dashboard for frontend
- Navigate via sidebar

- Take quizzes, view history, and access analytics

7. API Documentation

- POST /chat/ask – AI-generated responses
- POST /upload-doc – Upload and embed documents
- GET /search-docs – Retrieve semantically similar content
- GET /get-eco-tips – Personalized learning tips (mock for educational context)
- POST /submit-feedback – Store user feedback

8. Authentication

- Mock login for demonstration
- Can implement JWT, OAuth2, or role-based access for production

9. User Interface

- Sidebar navigation
- Dashboard cards
- Tabbed pages for quizzes, analytics, and feedback
- Real-time form handling

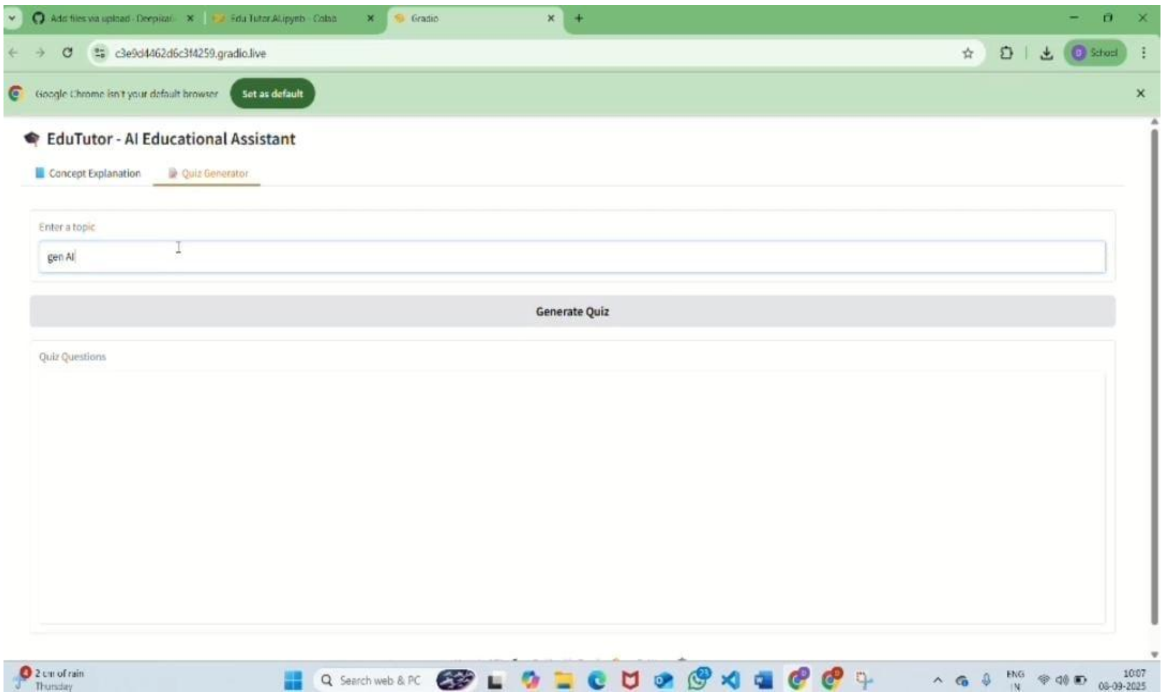
10. Testing

- Unit Testing: Core functions
- API Testing: FastAPI endpoints
- Manual Testing: Frontend interactions
- Edge Case Handling: Invalid inputs, missing data

11. Known Issues

- Google Classroom integration is currently mocked
- LLM responses are simulated; replace with real API for production
- Limited error handling for large inputs

## 12.Screenshot


c3e9d4462d6c3f4259.gradio.live
Google Chrome isn't your default browser
Set as default
EduTutor - AI Educational Assistant
Concept Explanation
Quiz Generator
Generate Quiz
Quiz Questions

Enter a topic

gen AI

## 13.Known Issues

- Google Classroom integration is currently mocked
- LLM responses are simulated; replace with real API for production
- Limited error handling for large inputs

## 14.Future Enhancements

- Integrate real IBM Watsonx / Granite APIs
- Real Google OAuth and Classroom integration
- Enhanced analytics with charts and performance trends
- Mobile-friendly responsive UI