Great! Based on your request, here&#39;s a professional and structured project document for your new

system:

�� EduTutor AI: Personalized Learning with Generative AI and LMS Integration

1. Introduction

Project Title: EduTutor AI – Personalized Learning with Generative AI and LMS Integration

Team Members:

1. Angel Christy .J

2. Divyasry. R

3. Dhivyadharshini .K

4. Atchaya .M

2. Project Overview

�� Purpose

EduTutor AI is an intelligent, adaptive e-learning platform that integrates Generative AI with Learning

Management Systems (LMS) to deliver personalized educational experiences.

The system dynamically adjusts learning content based on individual performance, preferences, and

learning pace. It includes features like AI-powered tutoring, smart content generation, auto-grading,

progress forecasting, and real-time learning analytics.

EduTutor AI empowers students and educators by automating manual tasks, offering on-demand

assistance, and enhancing learning outcomes through continuous feedback loops.

�� Key Features

Feature Key Point Description

AI-Personalized

Learning Paths Adaptive content delivery Learners receive custom modules based on quiz

scores, interaction history, and learning preferences

Generative AI Tutor On-demand

explanations/Q&amp;A

AI chatbot answers subject-related queries, explains

concepts, and recommends study tips

Auto Content Dynamic lesson and quiz Teachers can auto-generate topic-based study

Generator creation materials, quizzes, and flashcards using LLMs

LMS Integration Compatibility with Moodle,

Canvas, etc.

Seamless integration with existing learning platforms

via APIs and LTI

Progress Tracker &amp;

Predictor

Forecast academic

performance

Uses time-series and regression models to predict

student success and drop-out risk

Feedback Engine AI-generated personalized

feedback

Real-time, constructive insights on assignments,

quizzes, and engagement

Multilingual Support Language translation &amp;

localization

Supports learning in multiple languages using

translation models

3. System Architecture

��️ Frontend (Streamlit / React)

 Interactive web app with student and teacher views

 Tabs for:

o Dashboard

o Tutor Chat

o Study Materials

o Assignments/Quizzes

o Analytics &amp; Reports

�� Backend (FastAPI)

 RESTful APIs for:

o Content generation

o Chatbot interaction

o Auto-grading

o LMS communication

 Asynchronous endpoints with integrated Swagger documentation

�� LMS Integration

 Supports standards like LTI (Learning Tools Interoperability)

 Compatible with platforms: Moodle, Canvas, Google Classroom

 OAuth2 and API key authentication

�� AI Models Used

Use Case Model(s) / Tool(s) Used

Chat Tutor OpenAI GPT-4o, Claude, or LLaMA

Quiz &amp; Lesson Generator Fine-tuned GPT / T5 / LLaMA 2

Language Translation MarianMT / NLLB / Google Translate API

Auto-Grading BERT / GPT-based classifiers

Progress Prediction LSTM / XGBoost / Linear Regression

Feedback Generation GPT-3.5 / Instruction-tuned LLMs

4. Setup Instructions

✅ Prerequisites

 Python 3.9+

 Node.js (if using React frontend)

 API keys (OpenAI, Google, Hugging Face)

 LMS admin access (for integration testing)

 Docker (optional)

⚙️ Installation Steps

git clone https://github.com/your-org/edututor-ai.git

cd edututor-ai

python -m venv venv

source venv/bin/activate

pip install -r requirements.txt

 Create .env for storing API keys and LMS credentials.

 Start backend:

uvicorn app.main:app --reload

 Launch frontend:

streamlit run ui/dashboard.py # OR npm start for React frontend

5. Folder Structure

edututor\_ai/

├── app/

│ ├── api/

│ │ ├── tutor\_chat.py

│ │ ├── content\_gen.py

│ │ ├── grading.py

│ │ ├── analytics.py

│ │ └── lms\_integration.py

│ ├── models/

│ │ └── ai\_models.py

│ ├── utils/

│ │ └── helper\_functions.py

│ └── main.py

├── ui/

│ ├── dashboard.py

│ └── pages/

│ ├── TutorChat.py

│ ├── Lessons.py

│ ├── Quizzes.py

│ ├── Analytics.py

│ └── Admin.py

├── README.md

├── .env

└── requirements.txt

6. Running the Application

1. Start the FastAPI backend

2. Run the Streamlit (or React) frontend

3. Navigate through modules:

o Ask the AI Tutor questions

o Generate study content

o Assign quizzes and auto-grade them

o View student analytics and predictions

7. API Documentation

Endpoint Method Description

/generate-lesson POST Creates lesson content from topic name

/generate-quiz POST Generates quiz questions from topic

/chat-tutor POST AI answers educational queries

/auto-grade POST Grades submitted answers

/predict-performance GET Predicts learner’s progress

/lms-sync POST Syncs with LMS for data import/export

�� Swagger UI: http://localhost:8000/docs

8. Authentication &amp; Security

 JWT-based login for students, teachers, and admins

 LMS OAuth2 for platform integrations

 Role-based access control (RBAC)

 API throttling and rate limiting

 Optional SSO via Google or Microsoft

9. User Interface Overview

 �� Dashboard: Personalized widgets for students and teachers

 � AI Tutor: Chat interface for doubts and recommendations

 � Learning Content: Auto-generated lessons and study plans

 � Quizzes &amp; Assignments: Upload and grading interface

 � Analytics &amp; Reports: Performance charts, risk prediction

 � Language Selector: Multilingual support UI

10. Testing Strategy

 ✅ Unit Testing for APIs and model logic

 ✅ Mock LMS for integration tests

 ✅ CI Pipeline with GitHub Actions

 ✅ Load Testing for quiz generation and grading

 ✅ User Acceptance Testing (UAT) with student personas

11. Screenshots (To Be Added)

�� Add visuals like:

 Main dashboard

 Tutor chatbot UI

 Quiz generation form

 Analytics reports

 LMS sync panel

12. Known Issues

 Contextual accuracy of AI tutor may drop with vague questions

 Multilingual content quality depends on translation model

 LMS sync may vary slightly by platform (Canvas vs Moodle)

 Long quizzes (&gt;15 questions) take longer to generate

13. Future Enhancements

 �� AI-powered study schedule planner

 �� Integration with Google Classroom and MS Teams

 �� Automatic summarization of textbooks or lecture notes

 �� Adaptive testing algorithms (difficulty scaling)

 �� Certificate generator upon course completion

 �� Peer-to-peer AI-facilitated group discussions

 �� Plagiarism detection and originality scoring

✅ Summary

EduTutor AI is a cutting-edge educational platform built to empower teachers, students, and

institutions by using Generative AI and real-time analytics to optimize learning.

It transforms the traditional LMS into a smart, adaptive, and interactive environment—designed for the

future of education.