EduTutor AI – Project Documentation

1. Introduction

- Project Title: EduTutor AI – AI-Powered Learning & Quiz Assistant
- Team Member 1: Ramani S - (Leader) 23RCN48
- Team Member 2: Ramya V - 23RCN49
- Team Member 3: Reena F - 23RCN50
- Team Member 4: Sahana R - 23RCN51

2. Project Overview

Purpose:
EduTutor AI is an AI-driven assistant that helps students and educators. It provides concept explanations, quiz generation, and feedback.
The entire project was implemented using Google Colab for execution, with source code hosted on GitHub for version control and sharing.

Features:
- Concept Explanation – Students type a concept, and the AI explains with examples.
- Dynamic Quiz Generator – Creates MCQs, True/False, and Short Answer questions with answers.
- Performance Analysis – Evaluates quiz answers and provides personalized feedback.
- Educator Dashboard – Teachers can review quiz history and performance.
- Colab Integration – Entire project runs in Colab notebooks (no local setup required).
- Gradio Frontend – Provides a clean, interactive interface inside Colab.
- GitHub Hosting – All project files stored in GitHub for collaboration.

3. Architecture

Google Colab Environment:
- Used as the development and execution platform.
- Handles installations, model loading, and app launching.

Frontend (Gradio):
- Provides interactive tabs: Concept Explanation, Quiz Generator, and Performance Analysis.
- Runs inside Colab and gives a shareable public link.

Backend (FastAPI logic integrated in Colab):

- Managed within Python functions.
- Handles quiz generation, evaluation, and integration with Watsonx Granite model.

LLM Integration (IBM Watsonx Granite):
- Granite-13b-instruct-v2 model accessed from Hugging Face Hub.

Data Storage:
- GitHub used to store source code and documentation.
- Optionally, Pinecone DB can be integrated for quiz history.

4. Setup Instructions

Prerequisites:
- Google Account with Colab access
- GitHub repository link containing the project files
- Internet connection to load models

Installation in Colab:
```
!git clone https://github.com/your-username/edututor-ai.git
%cd edututor-ai
!pip install gradio torch transformers accelerate sentencepiece
```

5. Folder Structure (GitHub Repository)

```
edututor-ai/
├── edututor_ai.py       # Main Gradio app
├── notebooks/           # Colab notebooks
├── data/                # Any quiz/test data
├── README.md            # Documentation
└── LICENSE
```

6. Running the Application

1. Open Google Colab.
2. Clone the GitHub repository into Colab.
3. Install dependencies with !pip install ...
4. Run the edututor_ai.py file in Colab:
   !python edututor_ai.py
5. A Gradio link will appear → open in browser → interact with EduTutor AI.

7. API Documentation

Internal functions handle tasks inside Colab:
- concept_explanation(concept) → AI explains topic.
- quiz_generator(concept) → Generates quiz.
- quiz_analysis(answers) → Evaluates student answers.

(No external FastAPI endpoints needed since execution is inside Colab).

8. Authentication

- Current version: Direct access via Colab + Gradio link.
- Planned enhancement: Google OAuth login for students and educators.

9. User Interface

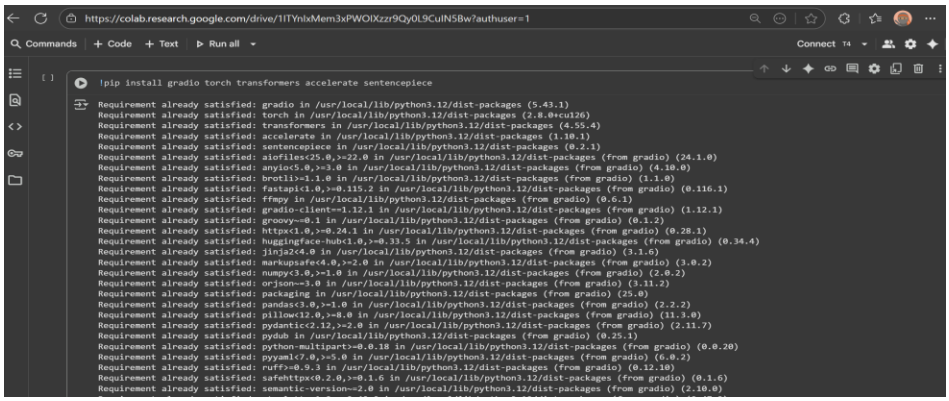
- Tab 1 – Concept Explanation: Enter a topic → AI explains.
- Tab 2 – Quiz Generator: Enter a subject → AI generates questions & answers.
- Tab 3 – Student Performance: Paste answers → AI evaluates with feedback.
- Runs inside Colab with a public Gradio share link.

10. Testing

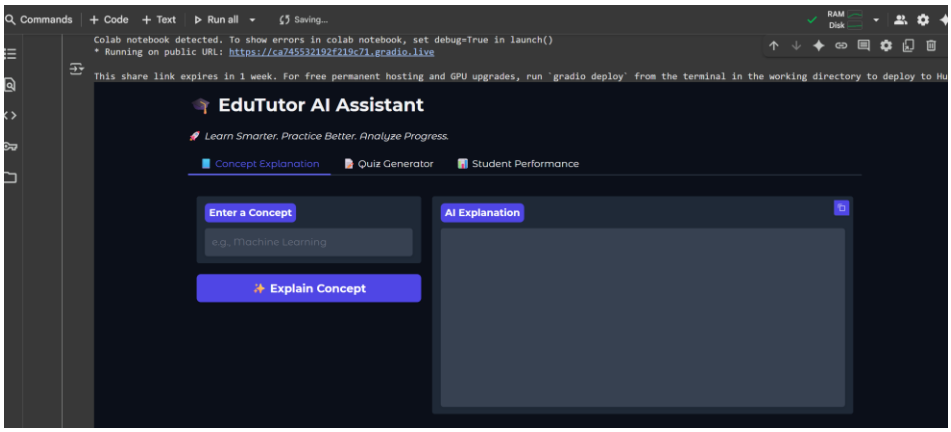
- Unit Testing: Verified concept explanation & quiz generation.
- Manual Testing in Colab: Ensured app launches with Gradio link.
- Edge Cases: Tested with blank inputs, long topics, invalid answers.

11. Screenshots

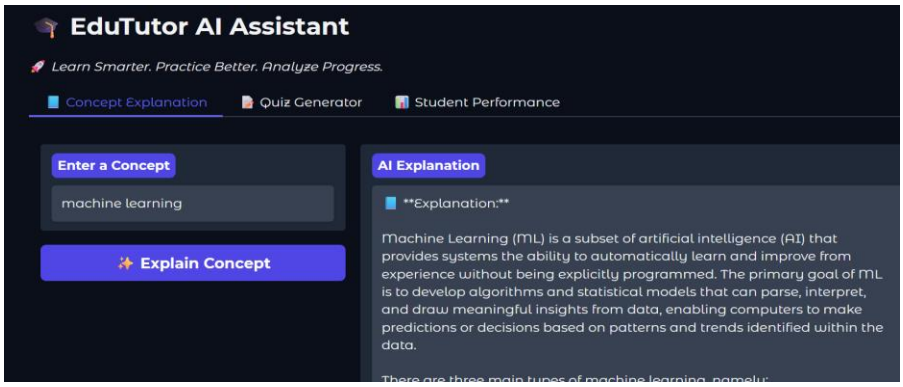
- Colab notebook running installation commands



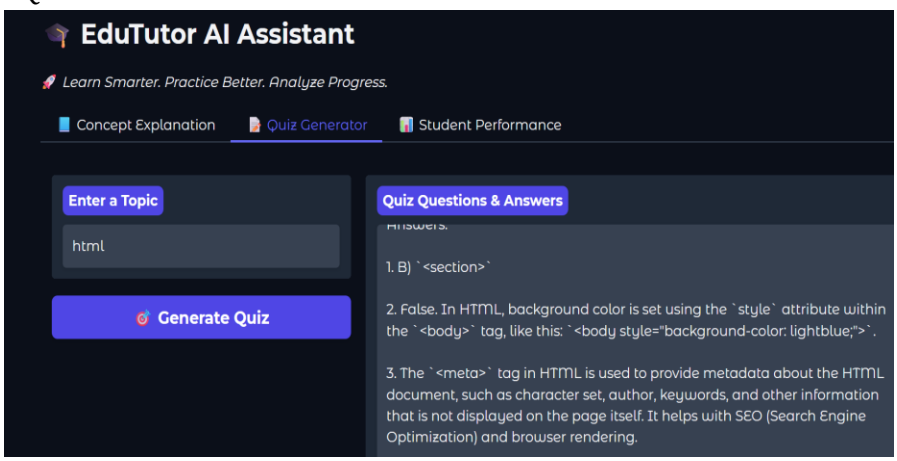
- Gradio link output in Colab



- Concept Explanation result



- Quiz Generator result



12. Known Issues

- Colab sessions expire after ~12 hours (app stops running).
- Requires internet to fetch Hugging Face models.
- No persistent database unless integrated with Pinecone/Google Drive.

13. Future Enhancements

- Store quiz history in Pinecone DB or Google Drive.
- Add educator role with dashboards.
- Export student reports as PDF.
- Support multi-language explanations and quizzes.