EDUTUTOR AI– PERSONALIZED LEARNING

Project Documentation

1. Introduction

Project Title: Edututor AI – Personalized Learning

Team ID: NM2025TMID03835

Team Leader: MUTHURAJA MOHAMED S

Team Member 1: PRADEEP VARSAN R

Team Member 2: PARTHA SARATHI S

Team Member 3: JEEVANANTHAN D

2. Project Overview

Purpose:
Edututor AI is designed to transform education through personalized and adaptive learning. It leverages AI models to explain concepts in simple terms, generate interactive quizzes, and help learners at their own pace. The project aims to support students, teachers, and institutions by making learning more engaging and efficient.

Existing Features:

Concept Explanation: Learners can input any topic and get a detailed, simplified explanation with examples.

Quiz Generator: Automatically generates multiple types of questions (MCQ, True/False, Short Answer) with answers.

Planned Features:

Learning progress tracking and analytics.

Personalized recommendations for further study.

Multilingual support for diverse learners.

AI-powered doubt resolution chatbot.

3. Architecture

Frontend:

Built using Gradio UI with two tabs:

Concept Explanation – Explains any concept in detail.

Quiz Generator – Creates topic-based quizzes with answers.

Backend:

Model: IBM Granite 3.2-2B Instruct (Hugging Face model).

Frameworks: PyTorch, Transformers library.

Functionality:

Text generation for explanations.

Quiz creation with different question types.

Planned Enhancements:

Integration with FastAPI for scalable backend.

Database support to store student progress and quiz results.

Advanced ML algorithms for adaptive learning paths.

4. Setup Instructions

Run the following command to install required libraries:

!pip install transformers torch gradio PyPDF2 -q

This will install dependencies for model loading and Gradio UI.

5. Folder Structure

Suggested project structure:

```
├── app/                    # Backend logic (future FastAPI integration)
│   ├── granite_llm.py      # Handles model communication
│   ├── document_tools.py   # Future text/PDF helpers
│   ├── forecast.py         # Planned: learning analytics
│   ├── anomaly.py          # Planned: anomaly detection for performance
│
├── ui/                     # Frontend components
│   ├── concept_tab.py      # Concept explanation UI
│   ├── quiz_tab.py         # Quiz generator UI
│
├── edututor_project.py     # Main entry file with Gradio UI
├── requirements.txt        # Dependencies
```

6. Running the Application

1. Install dependencies.

2. Run the edututor_project.py file.

3. Gradio will generate a link – open it in a browser.

4. Use Concept Explanation or Quiz Generator tabs to interact.

5. Planned: Save results, track learning history, and download reports.

7. API Documentation (Planned with FastAPI)

POST /explain-concept – Generate concept explanation.

POST /generate-quiz – Create quiz questions and answers.

POST /track-progress – Save student progress and results.

POST /recommend-topics – Suggest next learning topics.

8. Authentication

Future secure deployment will support:

Token-based authentication (JWT or API keys).

OAuth2 for login (Google, Microsoft).

Role-based access (Student, Teacher, Admin).

9. User Interface

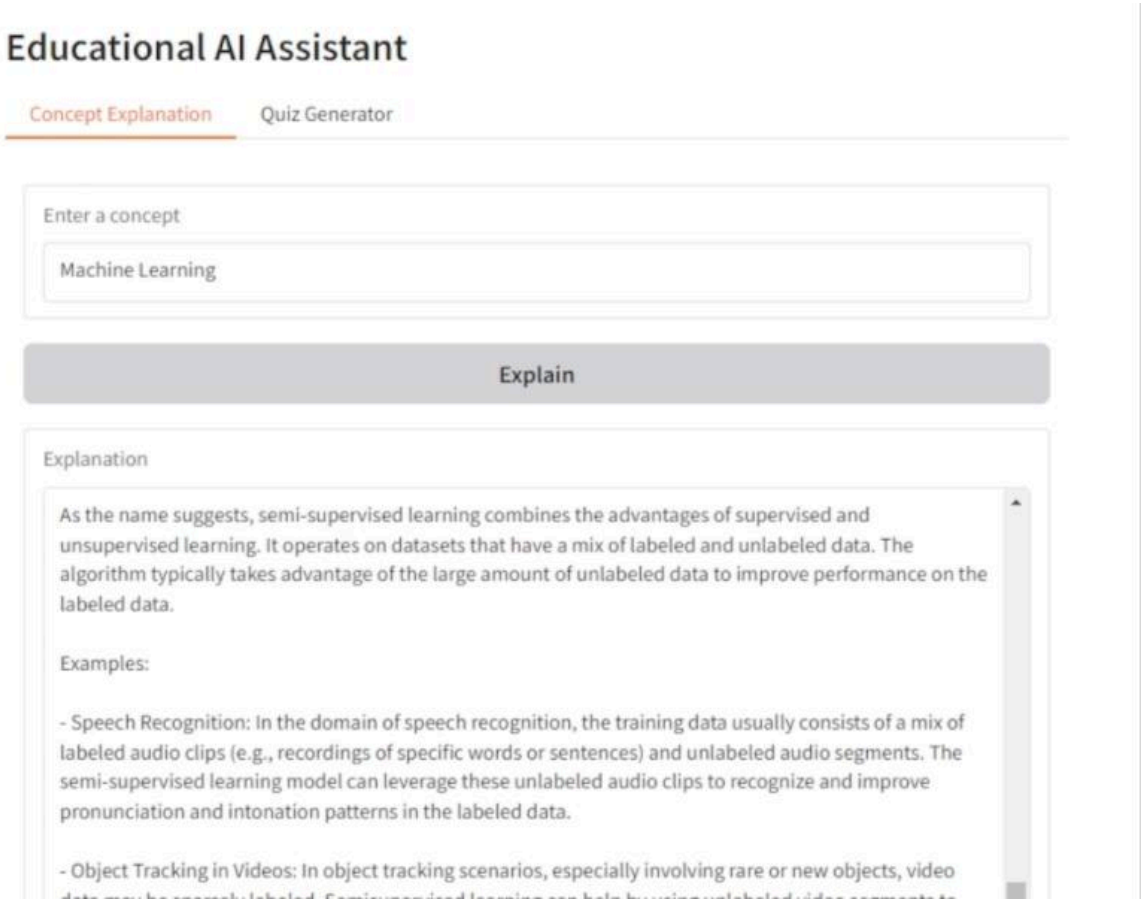
Gradio UI with clean tabs.

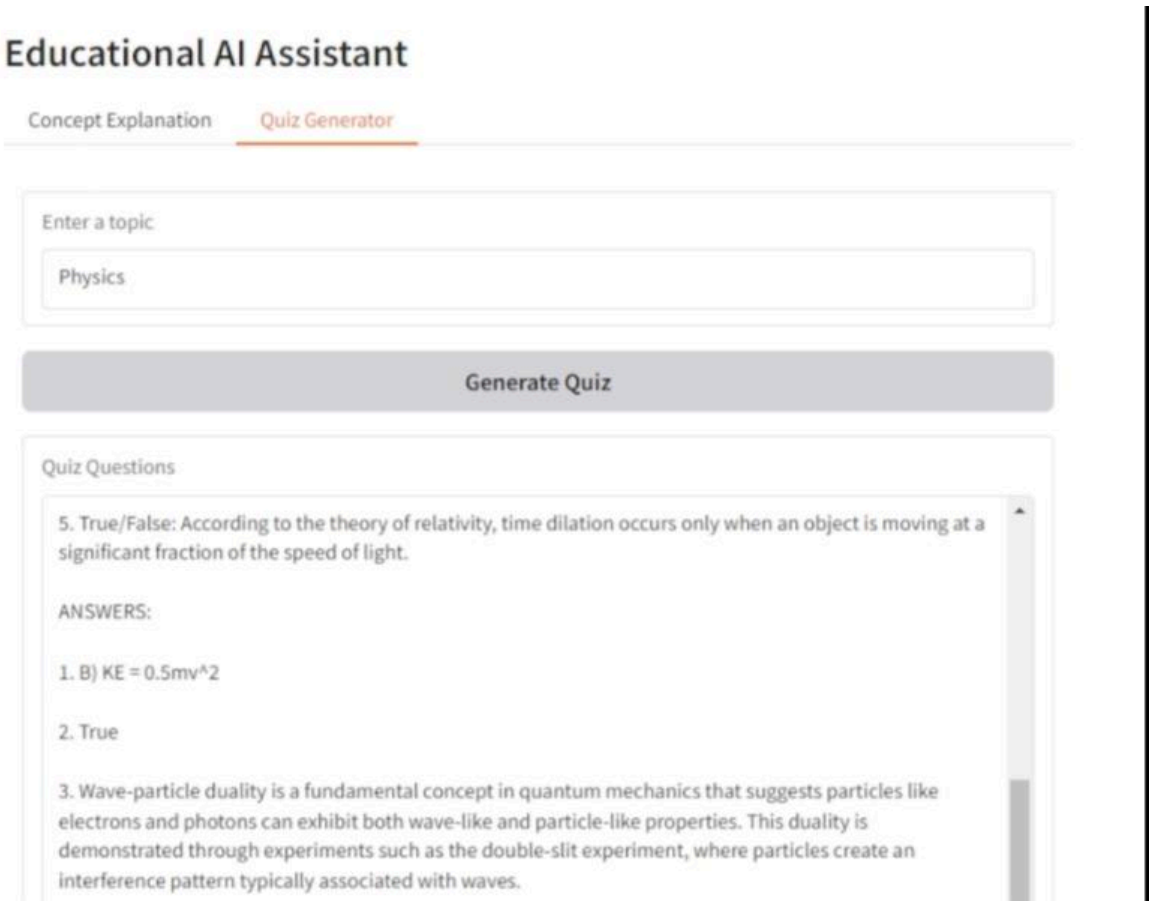
Concept Explanation: Input topic → get detailed explanation.

Quiz Generator: Input topic → get multiple question types + answers.

Planned tabs: Progress tracking, recommendations, and reports.

10.Screen Shot





11. Testing

Unit Testing: For model prompt generation and response formatting.

Manual Testing: Verifying concept explanations and quiz quality.

Planned: Automated testing with FastAPI and Pytest.

12. Known Issues

Limited to text input currently.

No progress tracking or multilingual support yet.

No authentication in the current version.

13. Future Enhancement

Add learning analytics and progress tracking.

Support CSV/Excel input for bulk quiz generation.

Implement AI-based doubt resolution.

Generate PDF progress reports for students.

Add voice interaction for accessibility.

PROJECT DEMO VIDEO LINK:

https://drive.google.com/file/d/1gIbtffRutZlMcSXYIggjxsUFCkN7T6rc/view?usp=drivesdk