

# ■ EduTutorAI Project Documentation (with Editing Option)

## 1. Introduction

**Project Title:** EduTutorAI:Personalized Learning Generative AI with IBM

**Team Members:**

- Ishwarya K
- Gopika M
- Gopika V
- Deepika T

## 2. Project Overview

**Purpose:**

EduTutorAI is an AI-powered learning assistant designed to simplify complex concepts, generate quizzes, create summaries, provide flashcards, solve doubts, and offer personalized study tips. It leverages LLMs (Granite 3.2 Instruct model) for generating human-like responses, helping students learn faster and more effectively.

**Features:**

- Concept Explanation – Explains any concept in detail with examples.
- Quiz Generator – Creates 5 diverse quiz questions with an answers section.
- Summary Generator – Summarizes topics into 5–7 bullet points for quick learning.
- Flashcards – Generates 5 Q&A; flashcards for revision.
- Doubt Solver – Provides simple, clear answers to student questions.
- Learning Tips – Suggests 7 study tips for any subject.
- Editing Option: Users can edit generated content inside the UI before saving or exporting.

## 3. Architecture

**Frontend (Gradio):** Interactive UI with tab-based navigation.

**Backend (Transformers + PyTorch):** Uses IBM Granite LLM for natural language processing.

**LLM Integration:** Tokenization and inference managed via Hugging Face Transformers.

## 4. Setup Instructions

**Prerequisites:**

- Python 3.9+
- pip + virtual environment
- GPU (Optional, for faster inference)

**Installation Steps:**

```
git clone  
cd EduTutorAI  
pip install -r requirements.txt
```

**Run Application:**

```
python EduTutorAI.ipynb
```

## 5. Folder Structure

EduTutorAI/

- app.py / EduTutorAI.ipynb # Main app file
- requirements.txt # Dependencies
- models/ # Model configs (optional)
- docs/ # Documentation files

## 6. Running the Application

1. Launch Jupyter Notebook or run the Python file.
2. Install dependencies (transformers, torch, gradio).
3. Start the app → Gradio UI will open in the browser.
4. Use the tabs to explore features.

## 7. API Documentation

Planned API Endpoints:

- POST /explain → Returns concept explanation
- POST /quiz → Generates quiz questions
- POST /summary → Provides summary
- POST /flashcards → Creates flashcards
- POST /doubt → Solves a doubt
- POST /tips → Returns learning tips

## 8. Authentication

Currently, EduTutorAI runs in an open environment for demonstration purposes.

Planned Authentication Options:

- JWT or API Key-based login
- Role-based access (student/teacher)
- User sessions and progress history

## 9. User Interface

**Tab-based layout:**

- Concept Explanation
- Quiz Generator
- Summary Generator
- Flashcards
- Doubt Solver
- Learning Tips

**Editing Option:**

- Editable textboxes
- Export to PDF/DOCX

## 10. Testing

**Unit Testing:** Functions tested with sample prompts.

**Manual Testing:** Verified multiple subjects and topics.

**Edge Case Handling:** Handled missing inputs, long prompts, GPU/CPU fallback.

## 11. Future Enhancements

- Export to PDF/DOCX for study material.
  - Voice input/output for accessibility.
- Multi-language support (Tamil, Hindi, English).
- Integration with student progress tracking system.
- Cloud deployment for wider access.

## 12. Screenshot

