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

# Introduction

Team ID : NM2025TMID2225

Team Leader: GOKUL PRASATH T

Team Member: ANANDHAKUMAR S

Team Member: BALAKUMAR K

Team Member: GOWTHAM M

# 2. Project Overview

EduTutor AI is an educational assistant built with the Granite LLM from Hugging Face.  
It provides personalized learning experiences through:

* 📘 Concept Explanation (detailed breakdowns with examples)
* 📝 Quiz Generator (MCQ, True/False, and short answers with solutions)

The project runs seamlessly in Google Colab for low setup effort and reliable performance, making it easy to deploy and extend with new features.

# 3. Architecture

* Frontend: Gradio interface with tabs for concept explanation and quiz generation.
* Backend: Hugging Face Transformers + IBM Granite Model for inference.
* Deployment: Google Colab for runtime environment and live public links.
* Version Control: GitHub for hosting code and documentation.

Flow:

1. User inputs a topic or concept.
2. Hugging Face Granite LLM processes the input.
3. Response is generated (explanation or quiz).
4. Output displayed in Gradio UI.

# 4. Technologies Used:

- IBM Granite Model (via Hugging Face Transformers)

- Hugging Face transformers library

- PyTorch (torch) for model execution

- Gradio for interactive UI

- Google Colab for deployment

- GitHub for project hosting

# 5. Setup Instructions & Installation

1. Clone the repository:  
 git clone https://github.com/yourusername/edututor-ai-granite.git  
 cd edututor-ai-granite  
  
2. Install dependencies:  
 pip install torch transformers gradio  
  
3. Run the application:  
 python app.py  
  
4. Launch in browser:  
 A local URL will be provided or use share=True for public Colab link.

# 6. Folder Structure & Process

• app.py – Main application script.  
• requirements.txt – Dependencies list.  
• notebook.ipynb – Google Colab notebook.  
• docs/ – Project documentation.  
• screenshots/ – UI screenshots.  
• .gitignore – Git ignore file.

# 7. Running the Application

- \*\*Option 1:\*\* Run locally with Python → http://localhost:7860

- \*\*Option 2:\*\* Run in Google Colab → Generates a `gradio.live` public link.

- Select the desired tab (Concept Explanation or Quiz Generator).

- Input a topic and view the generated outputs.

# 8. API Documentation

EduTutor AI is packaged as a Gradio app but can be extended with APIs:

- \*\*POST /concept-explanation\*\* → Input: {"concept": "Machine Learning"}, Output: Explanation.

- \*\*POST /quiz-generator\*\* → Input: {"topic": "LLM"}, Output: Quiz questions with answers.

# 9. Authentication

- Current demo: No authentication.

- Future options:

- API Key for private access.

- OAuth2 for role-based learning platforms.

- User sessions for history tracking.

# 10. User Interface

- Two main tabs in Gradio:

1. \*\*Concept Explanation\*\*: Input → AI-generated explanation.

2. \*\*Quiz Generator\*\*: Input → AI-generated quiz + answers.

- Clean layout with Markdown headers.

- Shareable link for remote learners.

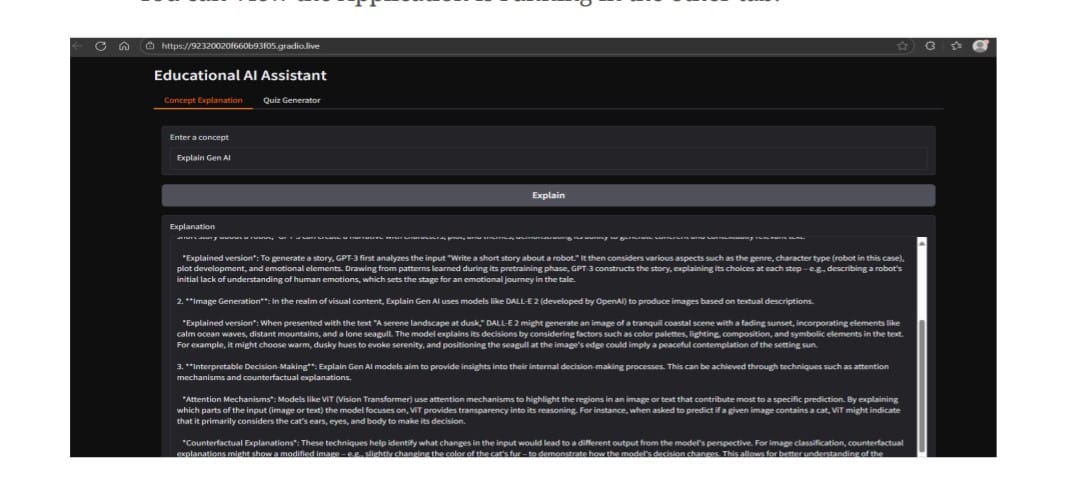
# 11. Testing

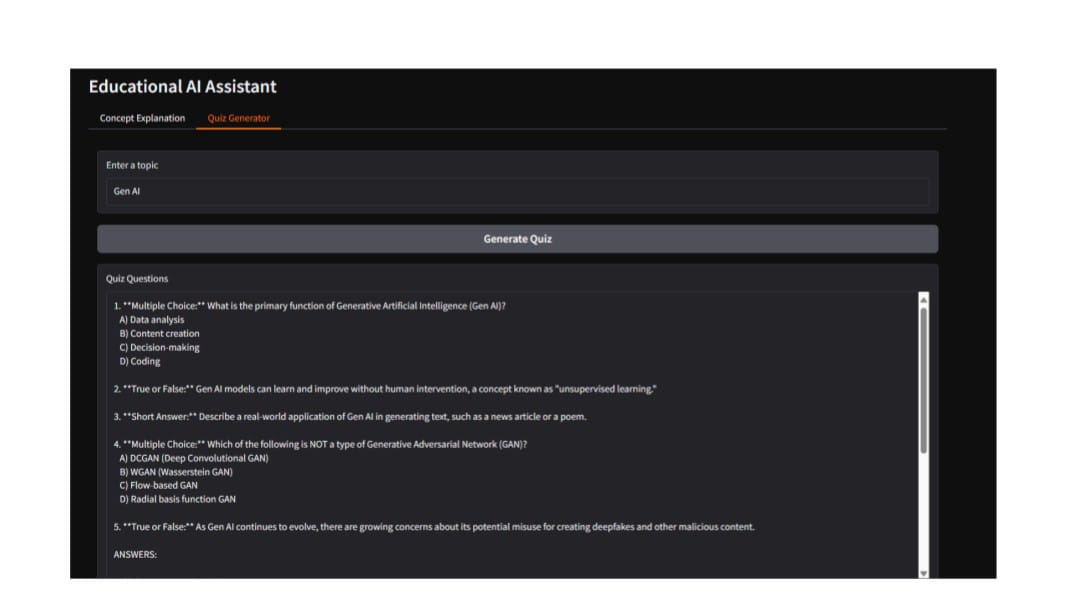
- \*\*Unit Testing:\*\* Functions like `generate\_response`, `concept\_explanation`, and `quiz\_generator`.

- \*\*Integration Testing:\*\* Full Gradio UI workflow.

- \*\*Edge Cases:\*\* Long input prompts, empty fields, unsupported topics.

# 12. Screenshots





# 13. Known Issues

• Response length may sometimes exceed expectations.  
• Limited control over question variety.  
• Requires internet connection for Hugging Face model loading.

# 14. Future Enhancements

- \*LMS Integration:\*\* Connect with Moodle/Canvas for personalized student progress.

- \*\*Voice Support:\*\* Speech-to-text for easier accessibility.

- \*\*Teacher Dashboard:\*\* Track student quiz performance.

- \*\*Expanded Subjects:\*\* Add more structured datasets for domain-specific quizzes.