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

## 1. Introduction

| | |
|---|---|
| Team Id : | NM2025Tmid12252 |
| Team Leader: | Srinivasan M |
| Team Member: | Shankar R |
| Team Member: | Sanjeevkumar 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

- Simple token-based access control.
- Best for **server-to-server** or **internal API** use.
- Easily revocable or regeneratable if compromised.
- Can be IP-restricted or scoped to certain endpoints.
- Enables **secure third-party access** without exposing user credentials.
- Suitable for **role-based systems** (e.g., admin vs student access in LMS).

## 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.

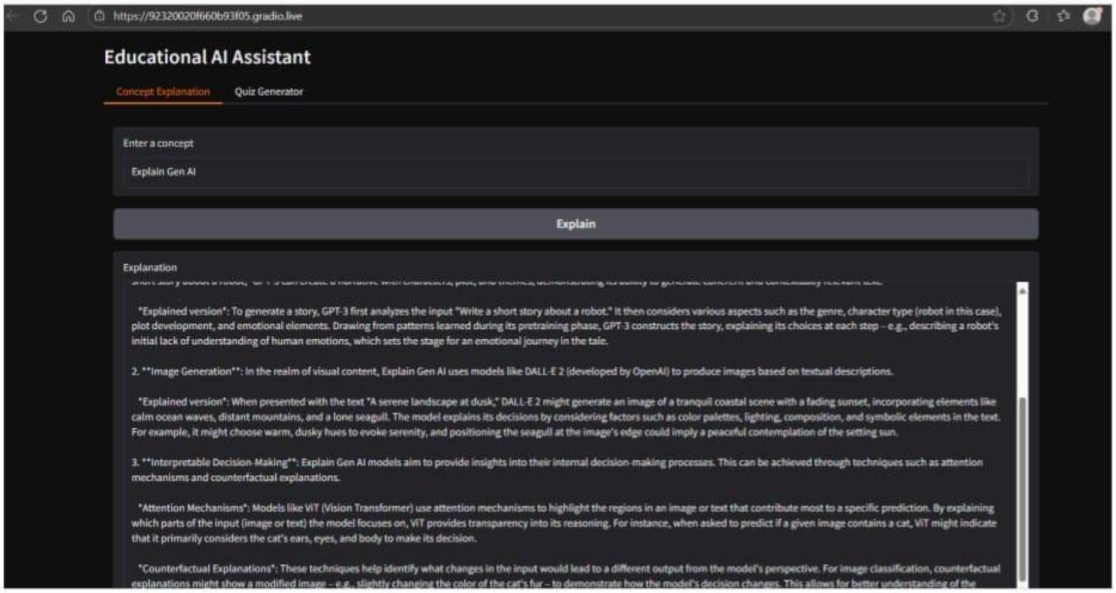
## 11. Testing

- **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

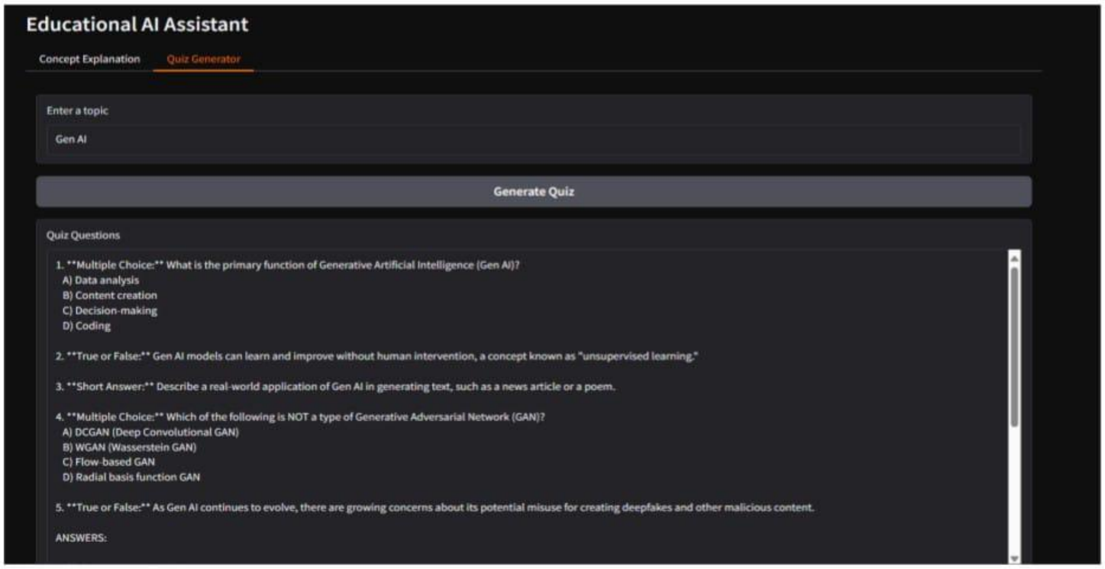


Enter a concept

Explain Gen AI

Explanation

```
  *Explained version*: To generate a story, GPT-3 first analyzes the input "Write a short story about a robot." It then considers various aspects such as the genre, character type (robot in this case), plot development, and emotional elements. Drawing from patterns learned during its pretraining phase, GPT-3 constructs the story, explaining its choices at each step – e.g., describing a robot's initial lack of understanding of human emotions, which sets the stage for an emotional journey in the tale.

2. **Image Generation**: In the realm of visual content, Explain Gen AI uses models like DALL-E 2 (developed by OpenAI) to produce images based on textual descriptions.

  *Explained version*: When presented with the text "A serene landscape at dusk," DALL-E 2 might generate an image of a tranquil coastal scene with a fading sunset, incorporating elements like calm ocean waves, distant mountains, and a lone seagull. The model explains its decisions by considering factors such as color palettes, lighting, composition, and symbolic elements in the text. For example, it might choose warm, dusky hues to evoke serenity, and positioning the seagull at the image's edge could imply a peaceful contemplation of the setting sun.

3. **Interpretable Decision-Making**: Explain Gen AI models aim to provide insights into their internal decision-making processes. This can be achieved through techniques such as attention mechanisms and counterfactual explanations.

  *Attention Mechanisms*: Models like ViT (Vision Transformer) use attention mechanisms to highlight the regions in an image or text that contribute most to a specific prediction. By explaining which parts of the input (image or text) the model focuses on, ViT provides transparency into its reasoning. For instance, when asked to predict if a given image contains a cat, ViT might indicate that it primarily considers the cat's ears, eyes, and body to make its decision.

  *Counterfactual Explanations*: These techniques help identify what changes in the input would lead to a different output from the model's perspective. For image classification, counterfactual explanations might show a modified image – e.g., slightly changing the color of the cat's fur – to demonstrate how the model's decision changes. This allows for better understanding of the
```



Enter a topic

Gen AI

Quiz Questions

```
1. **Multiple Choice:** What is the primary function of Generative Artificial Intelligence (Gen AI)?
 A) Data analysis
 B) Content creation
 C) Decision-making
 D) Coding

2. **True or False:** Gen AI models can learn and improve without human intervention, a concept known as "unsupervised learning."

3. **Short Answer:** Describe a real-world application of Gen AI in generating text, such as a news article or a poem.

4. **Multiple Choice:** Which of the following is NOT a type of Generative Adversarial Network (GAN)?
 A) DCGAN (Deep Convolutional GAN)
 B) WGAN (Wasserstein GAN)
 C) Flow-based GAN
 D) Radial basis function GAN

5. **True or False:** As Gen AI continues to evolve, there are growing concerns about its potential misuse for creating deepfakes and other malicious content.

ANSWERS:
```

## 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.