EduTutor AI Project Documentation

# 1. Introduction

• Project Title: EduTutor AI with IBM

Team ID : NM2025TMID08117

Team Leader : EBSEEBA LOURDEMARY A

Team member : ARULSELVI A

Team member: ATCHAYA D

Team member : GOPIKA G

# 2. Project Overview

EduTutor AI uses the Granite model from Hugging Face to create simple, personalized learning tools like concept explainers, quizzes generator, and more functionalities. This project is deployed in Google Colab using Granite for low setup effort and reliable performance.

# 3. Pre-requisites

• Gradio Framework Knowledge

• IBM Granite Models (Hugging Face)

• Python Programming Proficiency

• Version Control with Git

• Google Colab’s T4 GPU Knowledge

# 4. Project Workflow

• Activity-1: Exploring Naan Mudhalavan Smart Interz Portal.

• Activity-2: Choosing an IBM Granite Model from Hugging Face.

• Activity-3: Running Application in Google Colab.

• Activity-4: Uploading your Project in GitHub.

# 5. Architecture

Frontend (Gradio): Provides a user-friendly interface for interacting with the AI models.

Backend (Google Colab + Python): Runs the model, handles data processing, and serves responses.

Model (IBM Granite - Hugging Face): Provides natural language understanding and generation.

# 6. Setup Instructions

• Search for 'Google Colab' and open a new notebook.

• Change runtime type to T4 GPU.

• Run the command: !pip install transformers torch gradio -q

• Run the provided EduTutor AI code.

• Click the URL to launch the Gradio Application.

# 7. Folder Structure

Since the project runs on Google Colab and GitHub, folder structure may include:

• main.ipynb – Main Google Colab notebook

• requirements.txt – Dependencies list

• app.py – Application script for Gradio interface

• README.md – Documentation file in GitHub repository

# 8. Running the Application

➢ Launch the Colab Notebook.  
➢ Install dependencies.  
➢ Run the model cells.  
➢ Click on the Gradio link generated.  
➢ Interact with EduTutor AI through the web interface.

# 9. API Documentation

The project does not provide standalone APIs but relies on Hugging Face models and Gradio framework.

Key functionalities include:

• Concept Explainers

• Quiz Generator

• Personalized Learning Guidance

# 10. Authentication

Currently runs in an open environment for demonstration.

Secure deployments can integrate authentication such as:

• API Keys

• OAuth2

• GitHub private repositories for code access

# 11. User Interface

The interface is built using Gradio with a simple layout.

Features include:

• Input text box for queries

• Output display for explanations and quizzes

• Easy deployment through Google Colab

# 12. Testing

Testing involves:

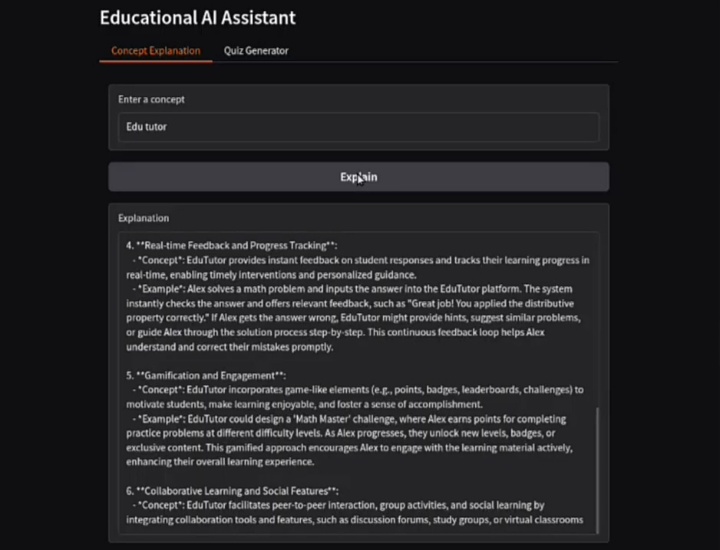
• Unit testing Python functions

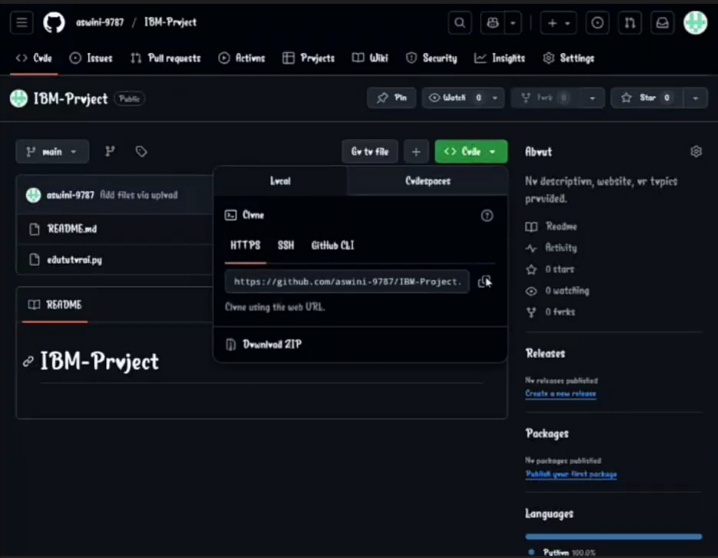
• Manual testing of Gradio interface

• Checking outputs for correctness

• Handling edge cases like invalid inputs

# 13. Screenshots





# 14. Known Issues

• Dependency installation errors in Colab

• GPU availability limitations in Google Colab

• Limited customization in free Hugging Face and Colab versions

# 15. Future Enhancements

• Adding more AI-powered learning tools

• Deploying as a standalone web application

• Integrating database for storing user progress

• Enhanced personalization features