**Edu Tutor AI :Personalized Learning with Generative AI and LMS Integration**

**1. Introduction**

* Project title: Edu Tutor AI: Personalized Learning with Generative and LMS Integration
* Team member: Monika devi K
* Team member: Anushiah A
* Team member: Kamali S
* Team member: Dhanushree K

**2.Project Overview:**

• Purpose: Edu Tutor AI is designed to provide personalized learning experiences by combining Generative AI with seamless Learning Management System (LMS) integration. It empowers students to understand complex concepts through AI-generated explanations and practice with quizzes tailored to their needs. Educators benefit from real-time analytics, progress tracking, and simplified content delivery.

**Features:**

* Concept Explanation: AI-generated detailed explanations with examples.
* Quiz Generator: Automatically creates quizzes with multiple question types.
* LMS Integration: Connects with existing LMS platforms for tracking student progress.
* Personalized Learning: Adapts responses based on learner profiles.
* Analytics Dashboard: Provides educators insights into student performance.
* User-Friendly Interface: Gradio-based simple and interactive UI

**3.Architecture**

* Frontend (Gradio): Provides an intuitive user interface with tabs for concept explanations, quiz generation, and student dashboards.
* Backend (Python + Transformers): Handles request processing, integrates with Granite LLM for natural language understanding and generation.
* LMS Integration: Connects Edu Tutor AI with existing learning systems for progress tracking.
* Database: Stores user interactions, generated quizzes, and student progress.
* LLM Integration (IBM Watsonx Granite / OpenAI): Generative AI models for Q&A, summarization, and content generation.
* Vector Search (Pinecone / FAISS): For semantic search of course materials, textbooks, and learner notes.
* ML Modules (Analytics & Forecasting): Forecast student performance, detect anomalies in learning behavior, and recommend interventions.

**4. Setup Instructions:**

* Install Python 3.9 or later
* Clone the repository
* Install dependencies from requirements.txt
* Create a .env file with necessary API keys and credentials
* Run the application using Gradio interface

**5. Folder Structure**

* app/ – Core backend logic including concept explanation and quiz generation.
* ui/ – Gradio frontend components.
* models/ – Model configurations and LLM integration scripts.
* data/ – Stores user data, logs, and generated quizzes
* main.py – Entry script for launching Edu Tutor AI

**6. Running the Application:**

* Launch the Gradio interface using the main script.
* Navigate between tabs: Concept Explanation and Quiz Generator.
* Enter topics to generate explanations or quizzes.
* View personalized results and analytics.

**7. API Documentation:**

* POST /explain-concept – Input a topic and receive AI-generated explanations.
* POST /generate-quiz – Input a topic and generate quiz questions with answers.
* GET /analytics – Retrieve student progress and performance analytics.

**8. Authentication:**

* The demo version runs in an open environment.
* Future releases will support:
  + Token-based authentication (JWT)
  + Role-based access (Admin, Teacher, Student)
  + Secure integration with LMS login systems

**9. User Interface:**

* Sidebar navigation with multiple tabs.
* Concept Explanation tab for AI-generated learning materials.
* Quiz Generator tab for practice tests.
* Student Dashboard: Learning path, quizzes, progress tracking
* Simple, clean design for accessibility.

Screenshot: Concept Explanation Tab

Screenshot: Quiz Generator Ta

**10. Testing**

Testing was performed across multiple phases: • Unit Testing: Validated core functions for explanation and quiz generation. • API Testing: Verified endpoints with Postman. • Manual Testing: Checked UI functionality in Gradio. • Edge Case Handling: Tested invalid or empty inputs.

**11. Sample Outputs**

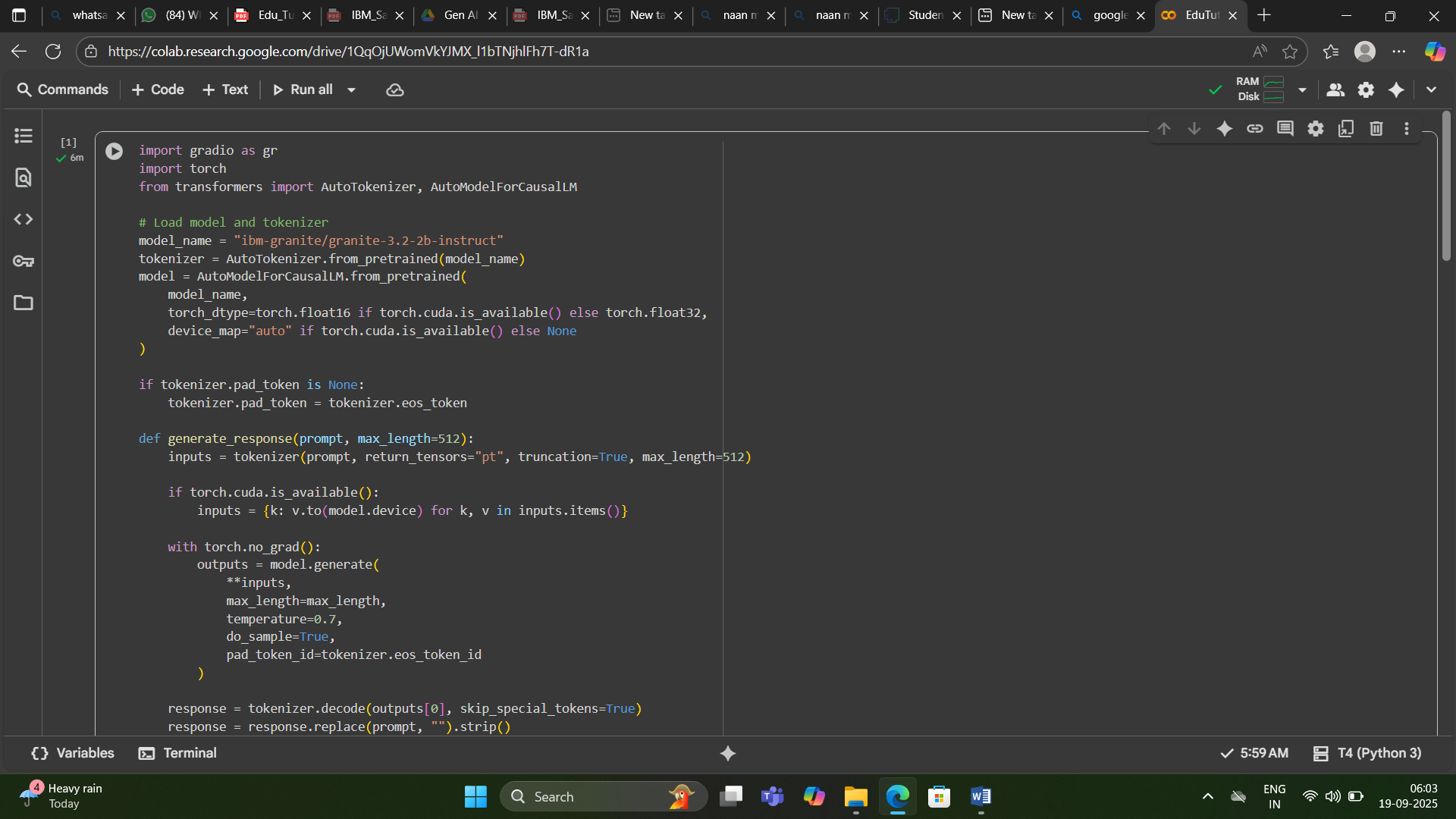
Example Concept Explanation (Machine Learning):

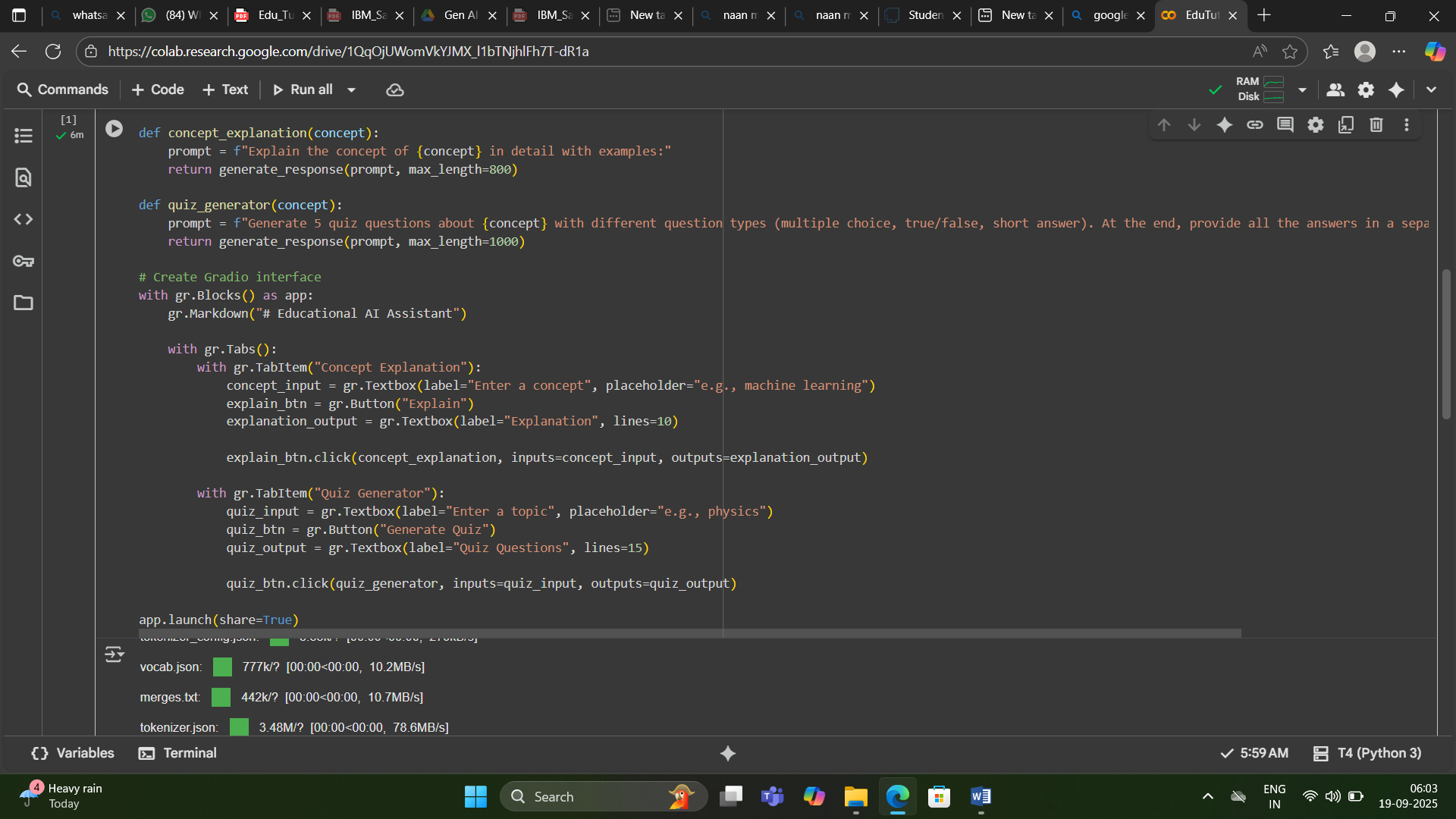
Machine learning is a subset of AI that allows systems to learn and improve from experience without being explicitly programmed. Example: A spam filter that improves its accuracy over time by learning from user inputs.

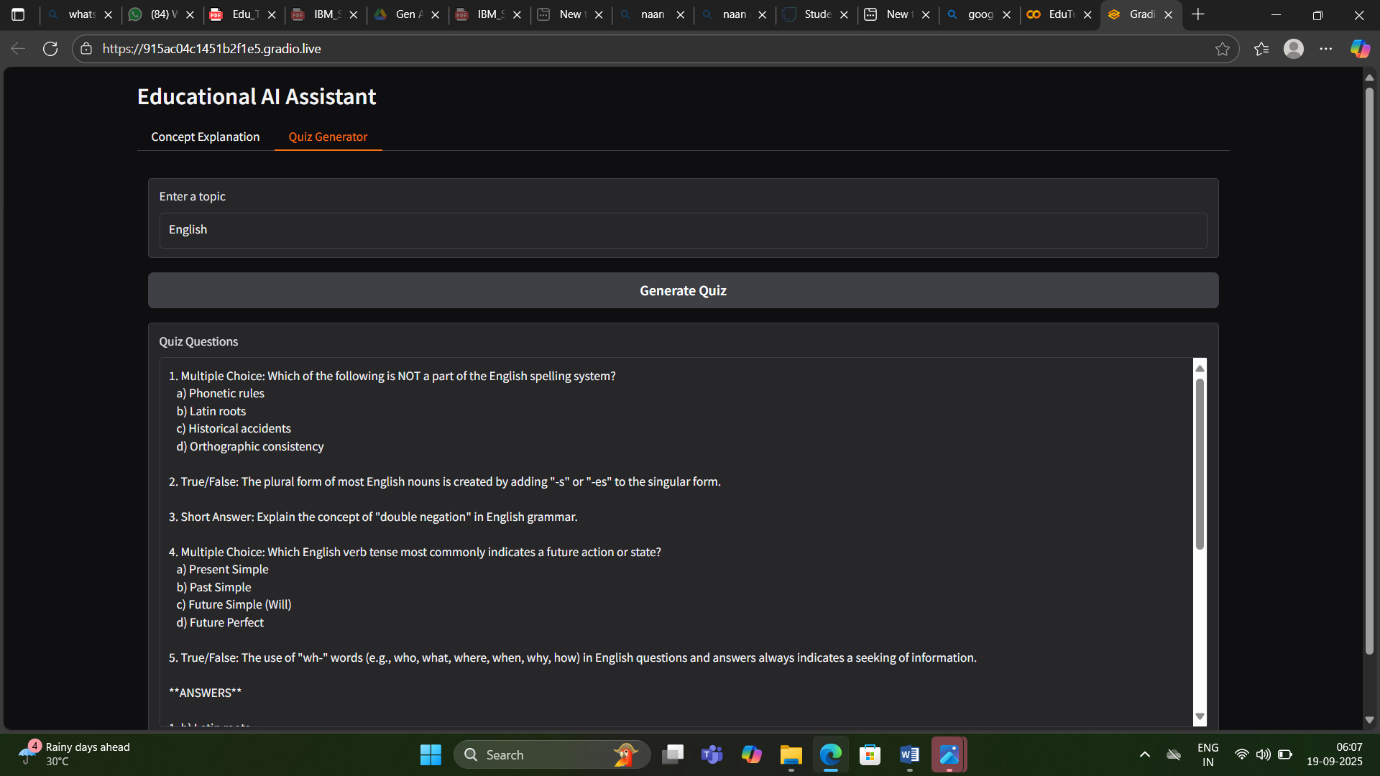
Example Quiz (Topic: Physics):

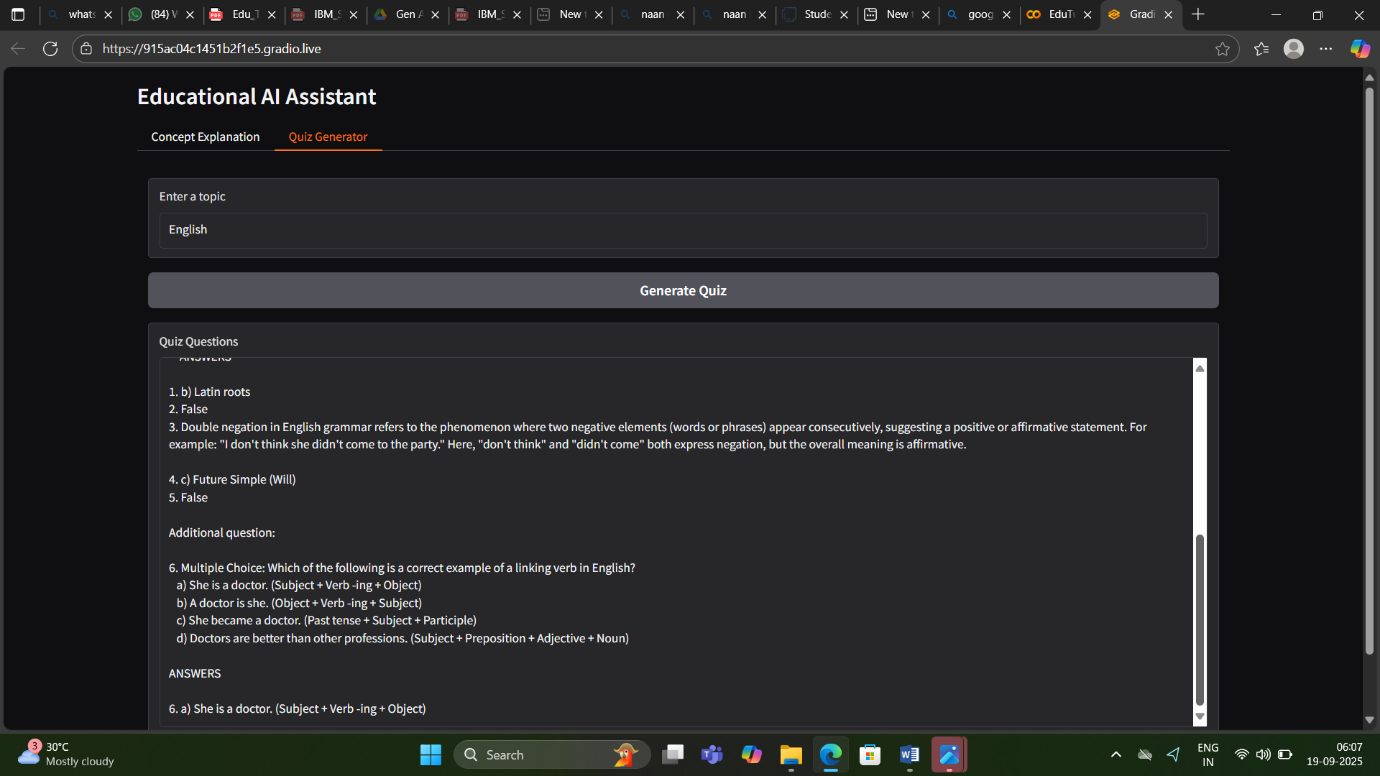
* Q1: What is Newton's Second Law? (Short Answer)
* Q2: True or False: Energy can be created or destroyed.
* Q3: Which of the following is a scalar quantity? (a) Velocity (b) Force (c) Speed (d) Acceleration
* Q4: Explain the concept of inertia with an example.
* Q5: Match the following: Kinetic Energy – Motion, Potential Energy – Position.
* Answers: Q1: F=ma, Q2: False, Q3: (c), Q4: Example-based,
* Q5: Kinetic–Motion, Potential–Position

**12.Screenshots:**

****

****





**13. Known Issues:**

* Limited LMS integrations currently supported
* Requires stable internet for model inference
* No offline mode available yet.

**14. Future Enhancements:**

* Adaptive learning paths based on student history.
* Support for multiple languages.
* Gamification features to increase engagement.
* Advanced analytics with predictive insights.
* Deep LMS integration with gradebook sync