**EDUTUTOR AI – PERSONALIZED LEARNING WITH GENERATIVE AI DOCUMENTATION**

**1.Introduction**

**Project Title:** EduTutor AI – Personalized Learning with Generative AI and LMS Integration.

**Team Members:**

Team Member 1 - M.Amreen

Team Member 2 - R.Lalitha

Team Member 3 - M.Lavanya

**2.Project Overview**

**Purpose:**

EduTutor AI is a personalized education platform designed to revolutionize learning and assessment. It leverages IBM Watsonx Granite models, FastAPI, Pinecone, and Streamlit to provide dynamic quiz generation, adaptive learning, real-time feedback, and seamless Google Classroom integration. The system empowers both students and educators by offering individualized learning experiences and data-driven performance insights.

**Features:**

* **Personalized Learning Experience** **Key Point:** Adaptive student learning

**Functionality:** Generates customized quizzes aligned to course topics and difficulty levels, providing instant AI-powered feedback.

* **Educator Dashboard & Insights**

Key Point: Real-time monitoring

Functionality: Displays student performance history, scores, attempted topics, and insights from Pinecone DB.

* **Diagnostic Testing & Adaptive Quizzing**

**Key Point:** Intelligent assessment

**Functionality:** Conducts diagnostic testing upon registration and adapts quiz difficulty accordingly.

* **Google Classroom Integration**

**Key Point:** LMS synchronization

**Functionality:** Syncs courses, subjects, and class data to ensure academic alignment.

* **Feedback & Analytics**

**Key Point:** Continuous improvement

**Functionality:** Provides educators with quiz analytics and improvement insights.

**Use Case Scenarios:**

* **Personalized Learning:** Students sync courses and receive AI-generated quizzes with feedback.
* **Educator Dashboard:** Teachers monitor quiz performance, history, and trends.
* **Diagnostic & Adaptive Testing:** Students take diagnostic quizzes to personalize learning.
* **Classroom Sync:** Seamless integration with Google Classroom for curriculum consistency.

**3.Architecture**

* + **Frontend (Streamlit):** Student and educator dashboards with quiz history, performance analytics, and real-time interaction.

* + **Backend (FastAPI):** APIs for quiz generation, evaluation, authentication, and classroom sync.

* + **AI Integration (IBM Watsonx Granite + LangChain):** Quiz generation, adaptive learning, and feedback via Watsonx models.

* + **Vector Database (Pinecone):** Stores quiz data, scores, and student embeddings for insights.

* + **LMS Integration (Google Classroom):** Syncs class data for quiz alignment.

**4.Setup Instructions**

**Prerequisites:**

* Python 3.10+
* FastAPI, Uvicorn
* LangChain
* Pinecone
* Streamlit
* Google Auth & API Client
* python-dotenv

**Installation Process:**

* Install dependencies: pip install -r requirements.txt
* Configure .env file with Watsonx, Pinecone, and Google API credentials
* Start FastAPI backend: uvicorn app.main:app --reload
* Run Streamlit frontend: streamlit run frontend/Home.py

**5.Folder Structure**

* + **app/** – FastAPI backend services
  + routes/ – Endpoints for quiz, auth, feedback, classroom sync services/ – AI integration, evaluation, and analytics logic **frontend/** – Streamlit UI for students and educators
  + Home.py – Entry dashboard pages/ – Student and Educator panels **quiz\_generator.py** – AI-powered quiz creation **answer\_evaluator.py** – Quiz evaluation logic **classroom\_sync.py** – Google Classroom integration **pinecone\_handler.py** – Pinecone database interactions

**6.Running the Application**

* Start FastAPI backend with Uvicorn
* Run Streamlit dashboard
* Student: Login → Take Quiz → View History → Receive Feedback
* Educator: Login → Access Dashboard → Monitor Analytics
* Sync with Google Classroom for curriculum-based quizzes

**7.API Documentation**

* + **POST /generate-quiz** – Creates quizzes based on selected topic & difficulty

* + **POST /evaluate-quiz** – Evaluates student answers and returns scores
  + **GET /quiz-history** – Retrieves quiz records from Pinecone
  + **POST /classroom-sync** – Syncs courses and subjects from Google Classroom
  + **POST /feedback** – Stores educator/student feedback
  + **GET /docs** – Swagger API documentation

**8.Authentication**

* + Google OAuth2 login for students and educators
  + Token-based authentication for backend API access
  + Role-based access (student, educator, admin)

**9.User Interface**

* + **Student Panel:** Login, Take Quiz, View History
  + **Educator Panel:** Real-time analytics, performance dashboard
  + **Adaptive Testing Interface:** Personalized quiz difficulty
  + **Feedback Section:** Educators and students provide feedback
  + **Classroom Sync:** Import courses directly from Google Classroom

**10.Testing**

* + **Unit Testing:** Backend quiz generation and evaluation
  + **API Testing:** Swagger UI, Postman

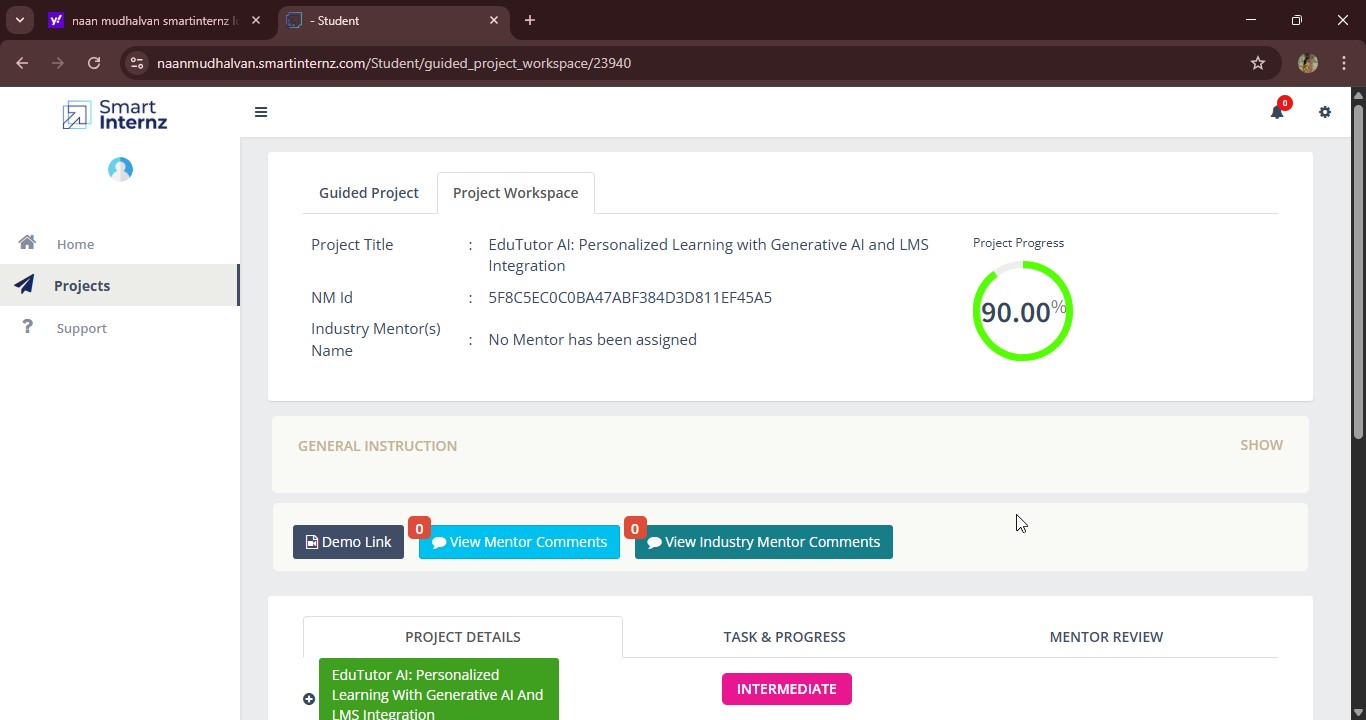
**Manual Testing:** Student quiz attempts, educator analytics

* + **Edge Cases:** Large classrooms, malformed quiz requests, invalid credentials

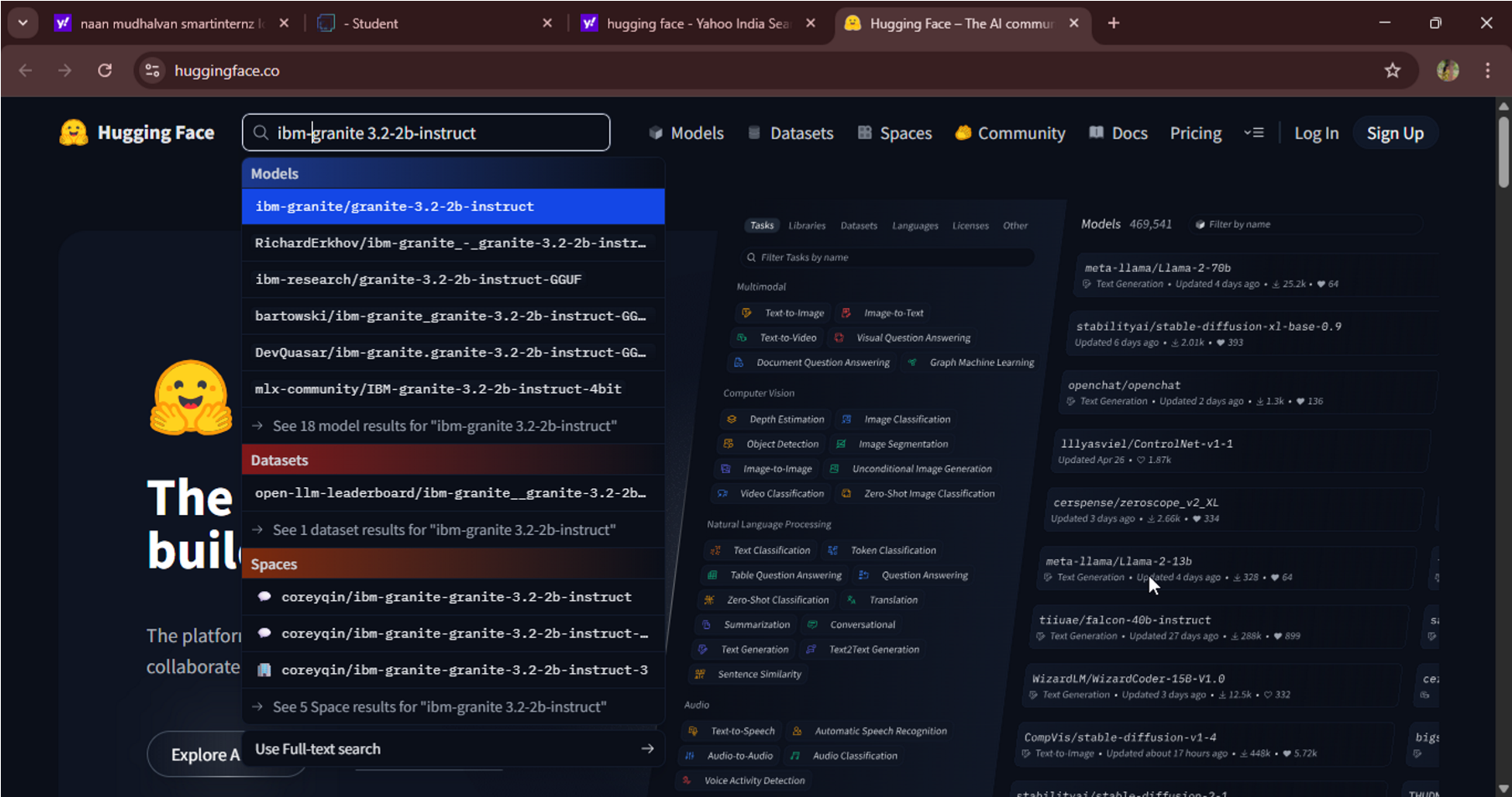
**11.Screenshots**



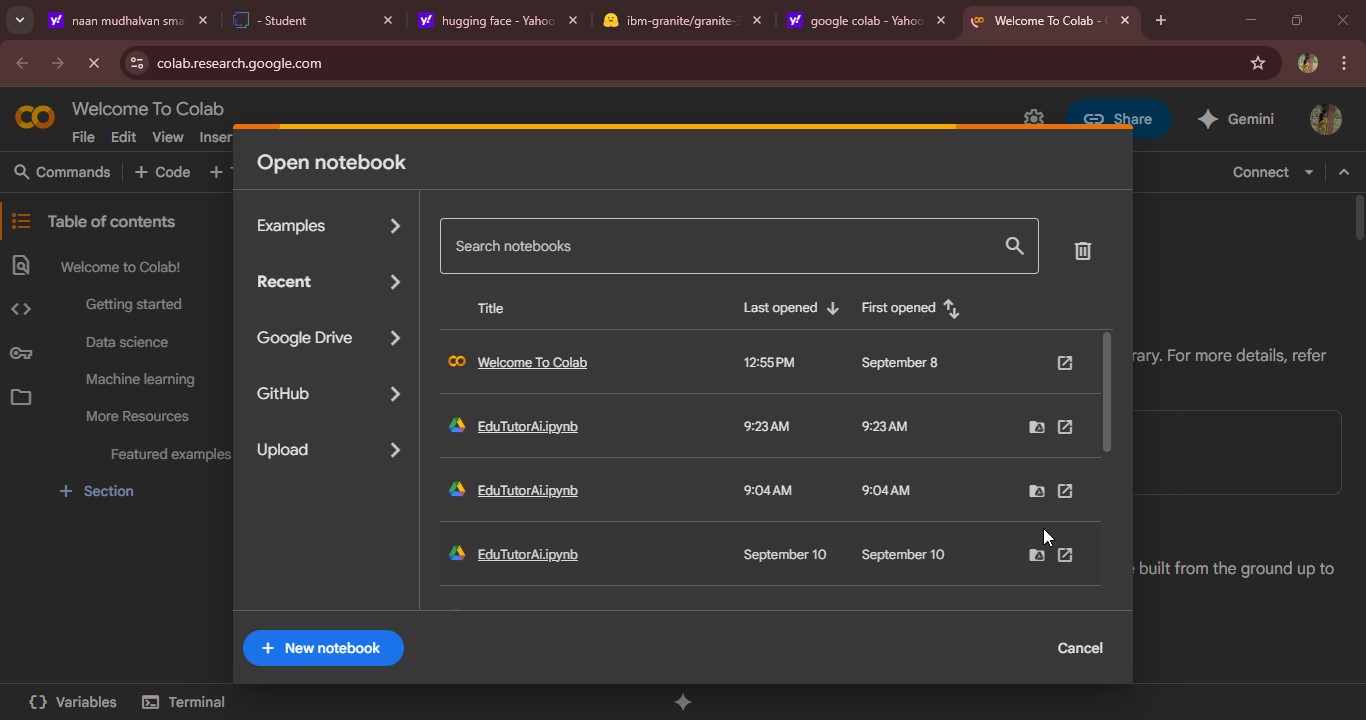
**Smartinternz Login Page**



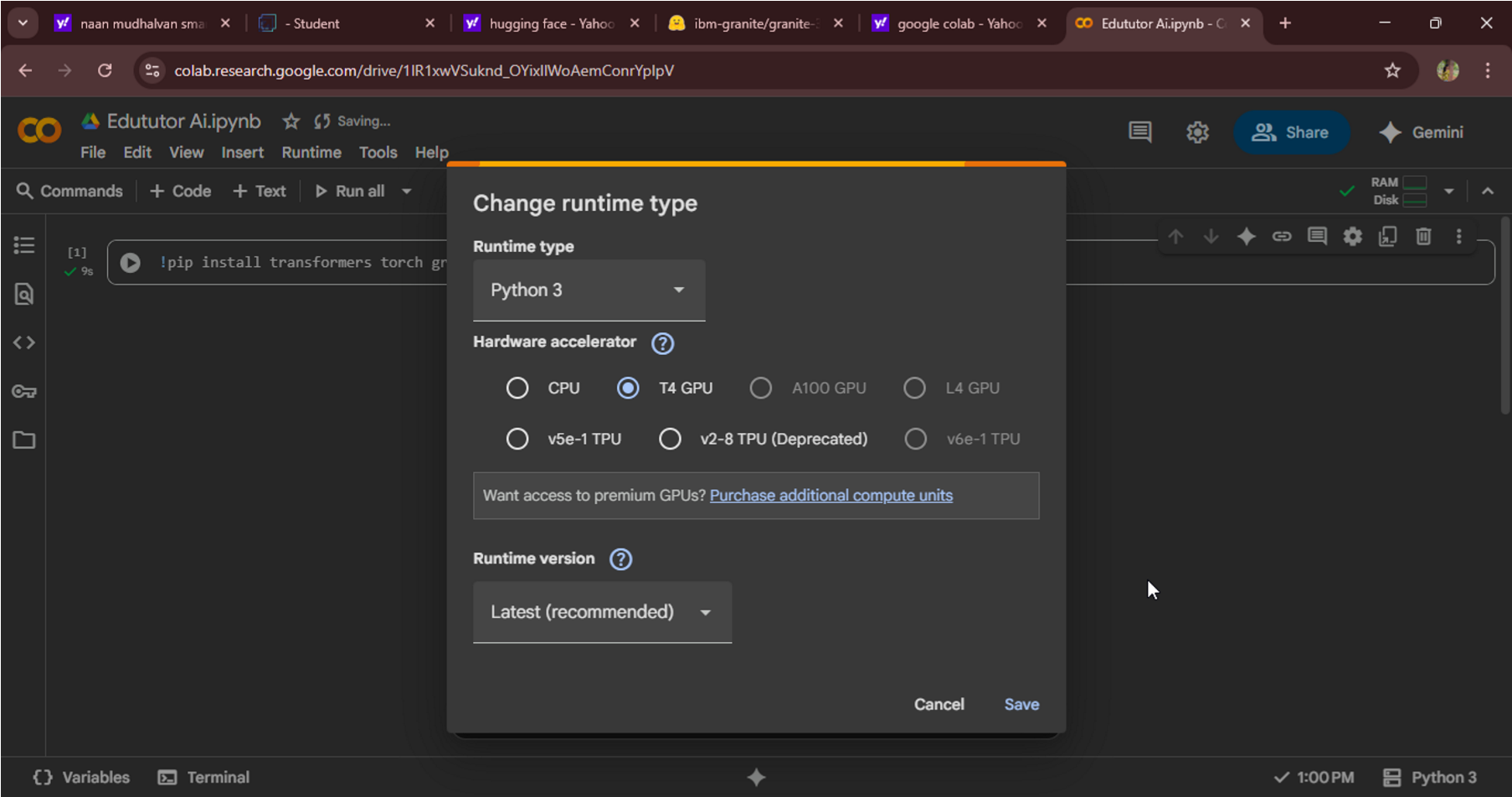
**Smartinternz dashboard**



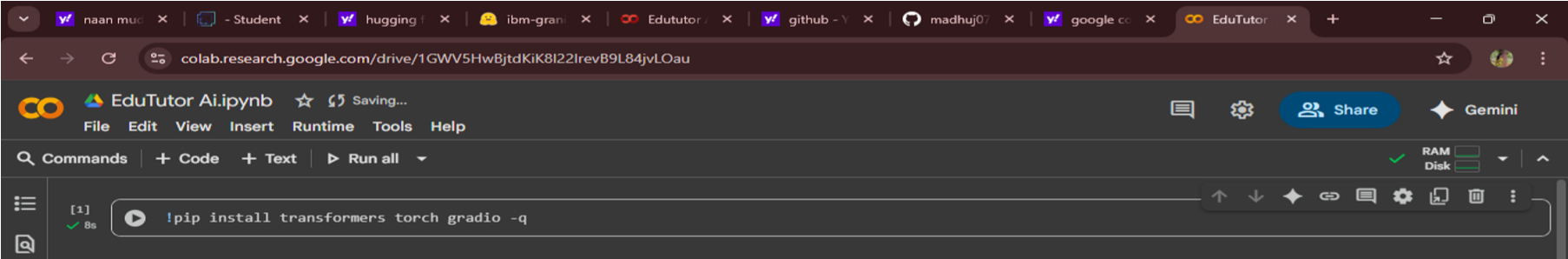
**granite-3.2-2b-instruct in hugging face**

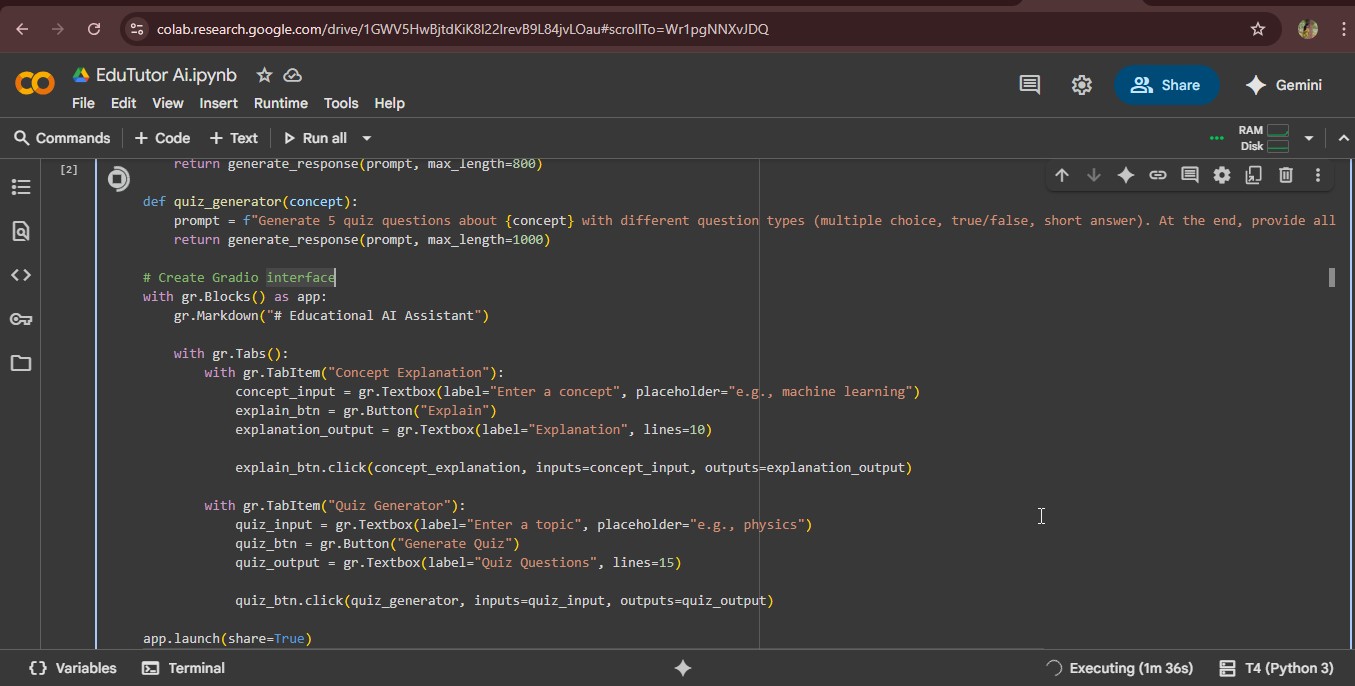


**New Notebook notebook page in Google colab**

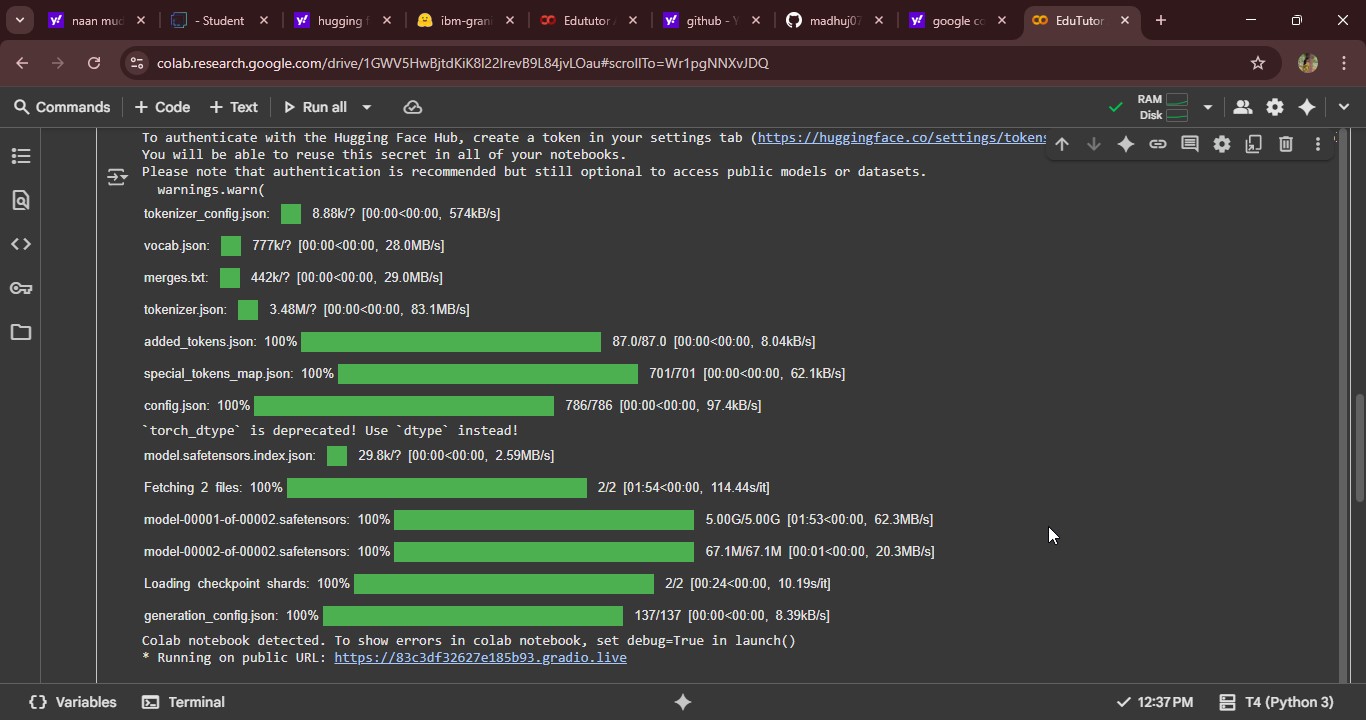


**Choose “T4 GPU” and click on “Save”**

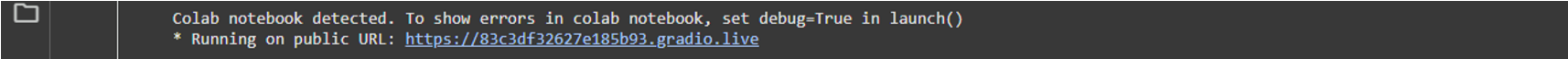


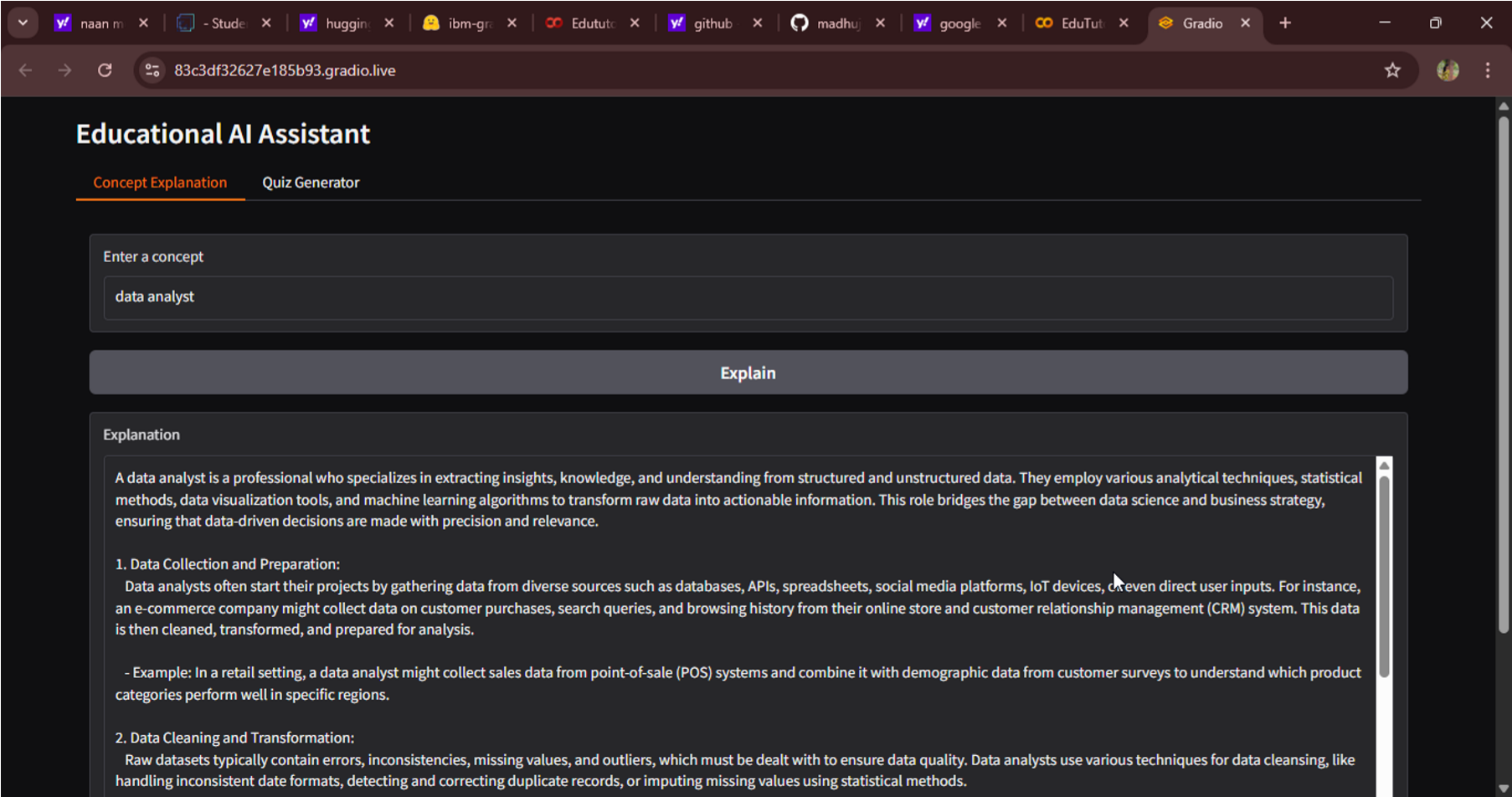


**EduTutorAI Coding page**



● Click on the URl to open the Gradio Application click on the link.





**Edututor AI Home Page**



**12.Known Issues**

* Frontend quiz UI incomplete in current version
* Dependency on Watsonx and Pinecone API quota
* Requires stable internet connectivity
* Limited offline support

**13.Future Enhancements**

* Full-featured student quiz frontend
* Mobile-friendly dashboard
* Multi-language quiz support
* Deeper Google Classroom integration
* AI-driven personalized study plans
* Advanced analytics with predictive performance insights