PROJECT REPORT

EDUTUTAR AI

Personalized Learning with Generative AI and LMS Integration

YEAR : 2025 – 2026 COLLEGE NAME : **K.C.S KASI NADAR COLLEGE OF ARTS & SCIENCE**

CODE :UMN203

DEPARTMENT : **COMPUTER APPLICATION**

PROGRAM : **BCA**

SEMESTER :**V**

PROJECT SUBMITTED TO: UNIVERSITY OF MADRAS / NAAN MUDALVAN

COURSE NAME : GENERATIVE AI WITH IBM

TEAM LEADER: G.SANDHIYA

MEMBERS:

1. N.MONISHA
2. B.THENMOZHI
3. T.THENMOZHI

GUIDED BY: MRS.S.KALAI SELVI

1. Introduction

Education has always played a central role in shaping both individual growth and the overall progress of society. It equips people with the knowledge, skills, and confidence needed to contribute to the world around them. However, despite its importance, the traditional approach to teaching often presents challenges. Classrooms usually follow a fixed pace and standardized curriculum, which may not suit every learner. Some students struggle to keep up when topics are taught too quickly, while others may find the material too basic and desire deeper exploration. This gap in learning styles and speeds highlights the limitations of conventional methods and points to the need for a more flexible and personalized system of education.

In recent years, technological advancements have introduced new opportunities to address these challenges. Artificial Intelligence (AI), in particular, has shown great potential in transforming education by providing tailored learning experiences. Personalized learning allows students to study at their own pace, receive instant feedback, and explore subjects according to their interests. EduTutor AI is one such project developed with this vision. It aims to create an AI-powered tutoring system capable

of simulating the role of a human tutor. By utilizing IBM Granite models hosted on Hugging Face, EduTutor AI can process natural language queries from students and generate meaningful, contextaware responses.

The project is designed to support students in multiple ways. It can explain complex concepts in simpler terms, generate quizzes to test understanding, and encourage interactive learning sessions that make the educational journey more engaging. A major advantage of EduTutor AI is its accessibility. Since it is deployed using Google Colab, students do not require expensive hardware or specialized systems to benefit from it. Even with limited resources, learners can access the platform and make use of advanced AI technology.

The motivation behind EduTutor AI lies in bridging the gap between traditional learning methods and the needs of modern students. Instead of passively reading textbooks or relying solely on lectures, learners can actively engage with the AI by asking questions, clarifying doubts, and testing their knowledge in real time. For teachers and institutions, EduTutor AI can serve as a valuable tool to design practice exercises, track student performance, and enhance overall classroom engagement. Its simplicity, scalability, and intelligence make it a strong step toward reshaping digital education. Ultimately, EduTutor AI represents not only a technological innovation but also a commitment to making education more inclusive, interactive, and adaptive to the diverse needs of students in the 21st century.

Furthermore, the project demonstrates how advanced AI can be integrated into practical learning environments without overwhelming students or teachers. It proves that cutting-edge tools can be simplified for everyday use, removing the barriers that often come with emerging technologies. It proves that cutting-edge tools can be simplified for everyday use, removing the barriers that often come with emerging technologies. By combining innovation with accessibility, EduTutor AI shows that the future of education can be both technologically advanced and human-centered. This balance ensures that while AI provides the intelligence, the human aspect of curiosity and creativity continues to drive meaningful learning outcomes.

2.Project Overview

EduTutor AI is envisioned as an innovative platform that transforms the learning process by integrating Artificial Intelligence into the core of education. The project is built with the primary purpose of making learning more personalized, accessible, and engaging for students, while also equipping teachers with modern tools to enhance their teaching efficiency. Traditional education often lacks the flexibility to address individual learning needs, but EduTutor AI seeks to overcome this challenge by creating a supportive, adaptive learning environment.

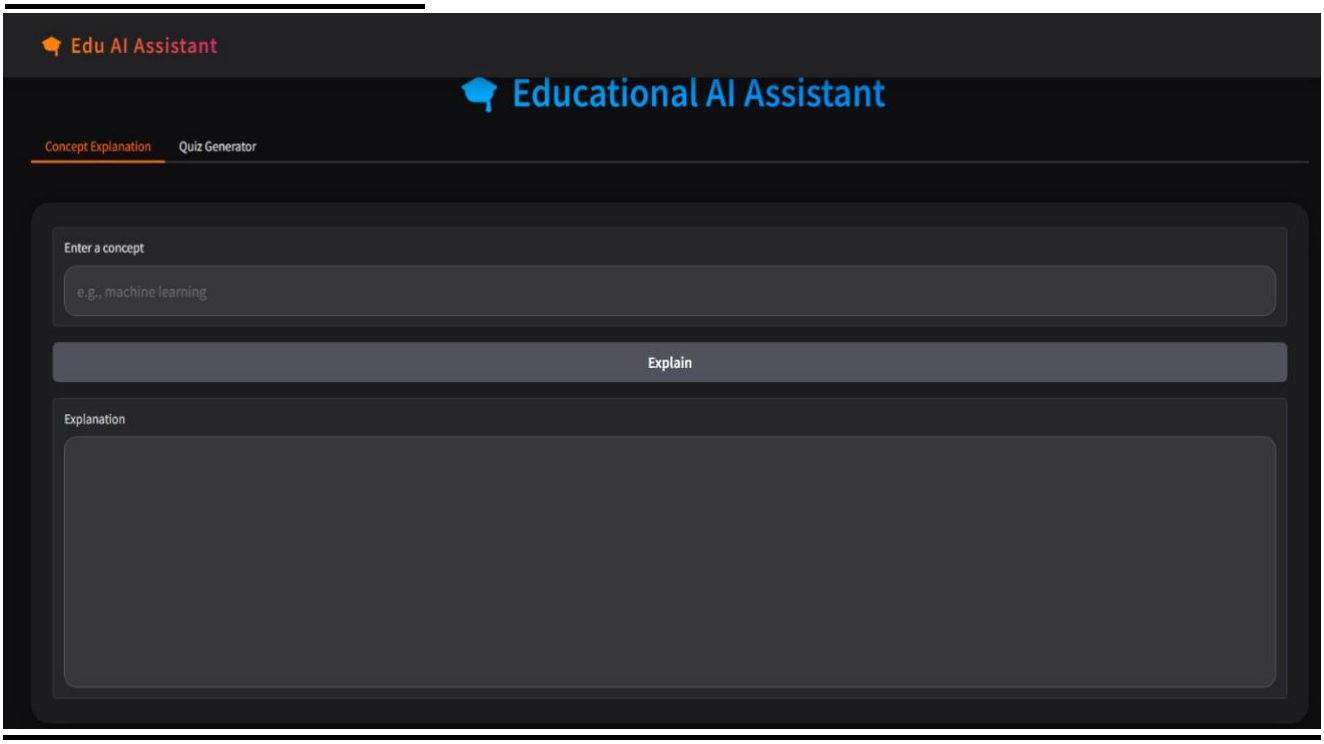
At its core, EduTutor AI is designed to assist learners in three significant ways. The first is Concept Explanation. Students can ask questions in natural language, and the AI responds with structured, simplified, and easy-tounderstand explanations. For instance, if a learner asks, *“Explain objectoriented programming in Python”*, EduTutor AI would provide a clear breakdown of the concept, complete with examples, ensuring that students not only memorize but also truly understand the subject matter.

The second feature is Quiz Generation, which allows EduTutor AI to create practice quizzes on almost any topic. This functionality enables learners to test their knowledge, identify areas where they need improvement, and prepare more effectively for assessments. Because the quizzes can be generated instantly, students benefit from real-time practice, promoting self-paced learning without depending solely on classroom schedules.

The third and equally important aspect is Interactive Engagement. EduTutor AI integrates a conversational interface built using Gradio, enabling students to interact with the AI in a natural, dialogue-based manner. This experience closely mirrors communication with a teacher or tutor, making the learning process more interactive and less mechanical. Through this dynamic exchange, learners stay motivated, curious, and actively engaged in their educational journey. From a technical standpoint, EduTutor AI is deployed on Google Colab, ensuring that it remains accessible and cost-effective. Students do not need to install heavy software or rely on expensive hardware, making the platform available to anyone with an internet connection. The use of IBM Granite models guarantees reliable and high-quality performance, while GitHub integration provides version control, collaboration, and smooth project management for the development team.

Overall, EduTutor AI aims to empower students to take control of their own learning journeys. By providing instant feedback, adaptive practice opportunities, and resources tailored to individual needs, it ensures that education becomes more inclusive and student-centered. Teachers and institutions can also benefit by generating learning materials, tracking student progress, and enhancing classroom engagement.

USER INTERFACE



3. Features

EduTutor AI provides a comprehensive suite of features designed to significantly enhance the overall learning experience for students while simultaneously supporting educators in their teaching process. In today's fastpaced educational landscape, where personalized attention and timely guidance are often limited, EduTutor AI steps in as a reliable and intelligent virtual assistant. Each feature of the platform has been meticulously developed to ensure that learning becomes more interactive, accessible, and effective, while also streamlining teaching responsibilities for educators.

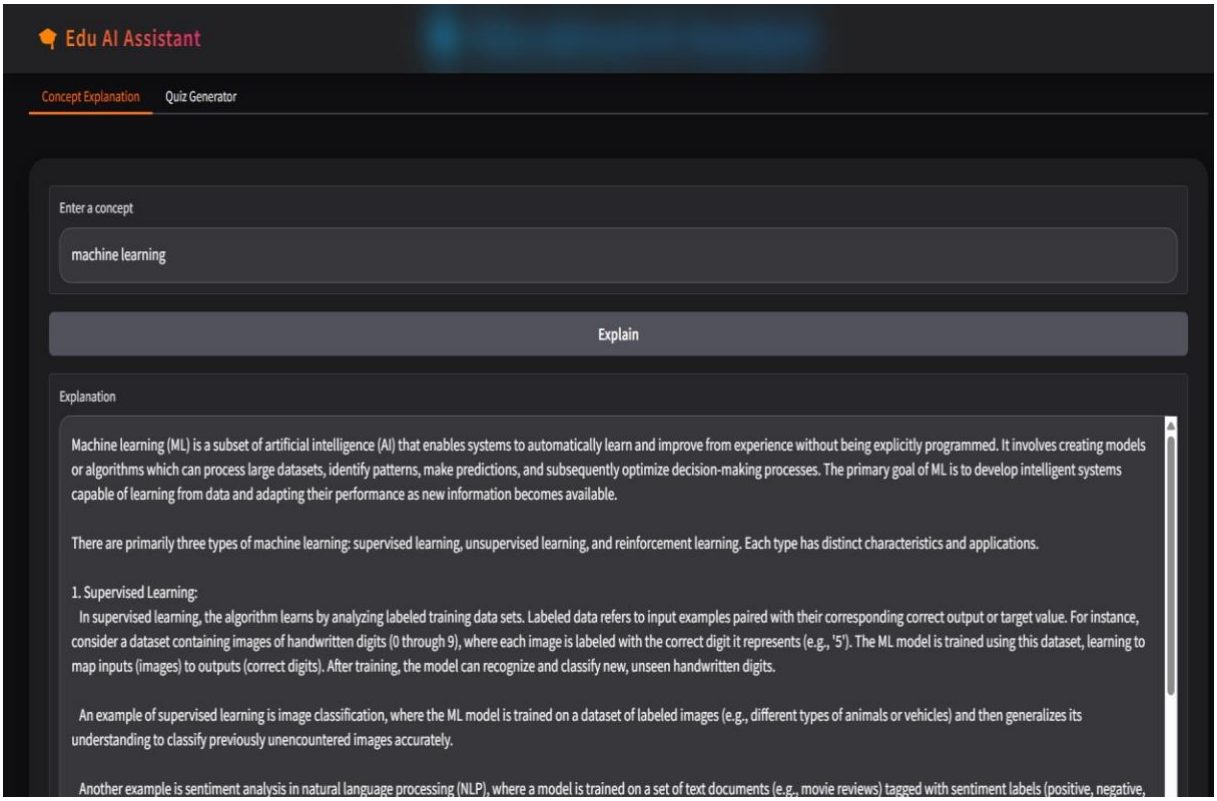
At the heart of EduTutor AI lies the **Concept Explanation** feature, arguably the most crucial component of the platform. This feature transforms the system into a virtual tutor that can respond to student queries in natural, conversational language. Students are no longer limited to rigid textbook explanations; instead, they can ask questions in the same way they would speak to a human teacher. For instance, if a student asks, "What is recursion in programming?" the AI delivers a detailed explanation, breaking down the concept step by step. The system uses simple, understandable language while providing practical examples that illustrate the topic in real-world terms. This ensures that even learners who are new to the subject can grasp complex ideas without feeling intimidated or overwhelmed. By making abstract or difficult topics clear and accessible, EduTutor AI empowers students to build a solid foundational understanding that can support further exploration and learning.

Complementing the Concept Explanation feature is **Quiz Generation**, a powerful tool that turns learning into an interactive experience. The AI can

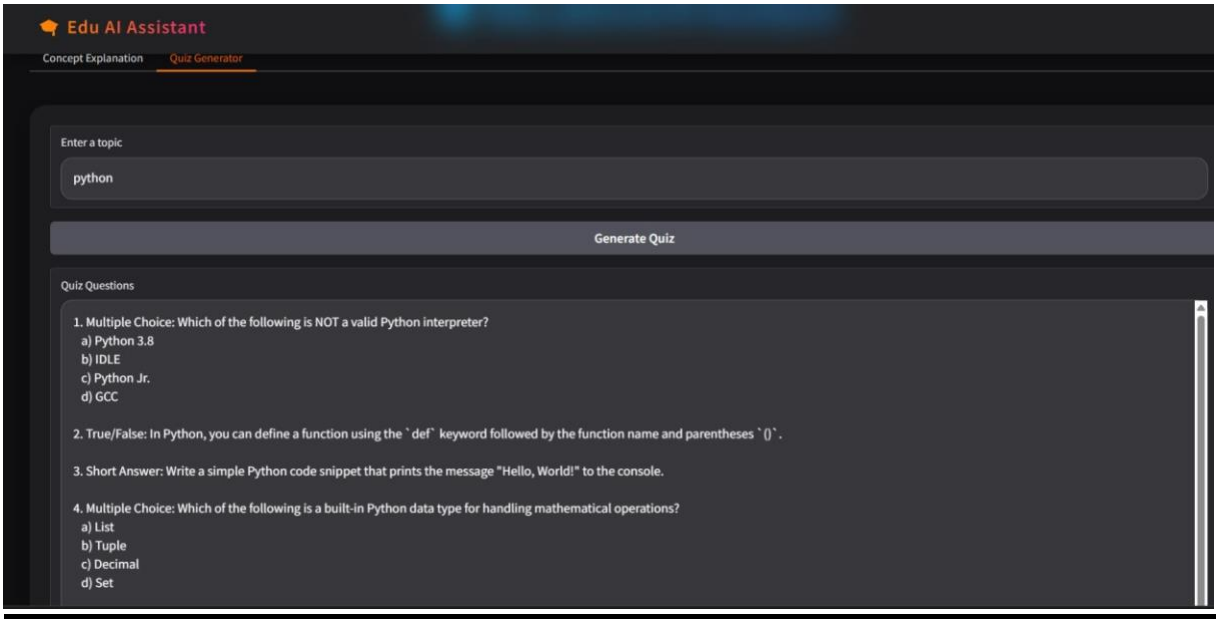
automatically create assessment materials across a variety of formats, including multiple-choice questions, fill-in-the-blank exercises, and short-answer questions. Whether the input comes from the student's area of interest or the teacher's curriculum, the system can generate quizzes dynamically, ensuring content remains relevant and personalized. This immediate assessment capability allows learners to test their understanding as they progress through a topic, helping to reinforce knowledge and identify areas requiring further study. For teachers, Quiz Generation offers an invaluable time-saving tool. Educators can rapidly prepare practice exercises and evaluation materials without spending hours manually creating questions. By promoting active engagement through interactive assessments, this feature supports knowledge retention, boosts confidence, and encourages consistent, self-paced revision.

Another cornerstone of EduTutor AI is **Personalized Learning Support**, which differentiates the platform from traditional learning methods and static digital resources. Unlike conventional textbooks or pre-recorded e-learning modules that provide the same content to every learner, EduTutor AI adapts to individual needs.

Concept explanation:



QUIZ GENERATION:



4.System Architecture:

The system architecture of EduTutor AI has been carefully designed to prioritize simplicity, scalability, and accessibility. The architecture is organized into four main layers: the User Layer, Frontend Layer, Backend Layer, and Deployment Layer, each serving a critical role in ensuring seamless interaction between the user and the AI system. This layered design not only improves performance but also makes maintenance and future enhancements easier to implement. The **User Layer** represents the primary stakeholders of EduTutor AI, including students and teachers. This layer is responsible for capturing user input and delivering output in a manner that is intuitive and effective. Students can submit queries in natural language, such as "Explain recursion in programming" or "Generate a quiz on binary trees," while teachers can request educational content or assessment materials for classroom use. By supporting natural language interactions, this layer ensures that users do not require technical expertise to operate the system, making it accessible to learners of varying ages and backgrounds.

The **Frontend Layer** is implemented using **Gradio**, a lightweight and userfriendly framework for building web-based interfaces. Gradio allows users to interact with EduTutor AI through a clean and interactive interface where they can type questions, receive explanations, and attempt quizzes. One of the main advantages of Gradio is its rapid prototyping capability, enabling developers to create functional interfaces without extensive knowledge of UI/UX design. The intuitive design of the frontend ensures that students can focus on learning rather than navigating complex software, while teachers can quickly access content and monitor student progress.

The **Backend Layer** is the core computational component of EduTutor AI and is powered by **IBM Granite models** hosted on **Hugging Face**. Specifically, the Granite-3.2-2B-Instruct model has been chosen for its balance of efficiency, lightweight architecture, and suitability for educational tasks. This layer is responsible for processing natural language queries, understanding context, and generating accurate and meaningful responses. By leveraging pre-trained AI models, the backend can handle complex educational queries, generate quizzes dynamically, and provide step-by-step explanations, ensuring high-quality tutoring support.

The **Deployment Layer** utilizes **Google Colab**, providing a GPU-enabled environment that allows AI models to run efficiently without requiring users to install software locally. This layer ensures that students with limited hardware resources can still benefit from AI-driven learning. Additionally, **GitHub integration** is incorporated for version control, collaboration, and submission. This enables team members to work together seamlessly, track changes, and maintain a structured project repository.

Overall, the architecture of EduTutor AI is designed to be reliable, costeffective, and scalable.

5. Setup Instructions

Setting up EduTutor AI is a straightforward process that can be completed by anyone with basic programming knowledge. The project leverages **Google Colab**, a free, cloud-based platform that provides GPU support, ensuring that even students or educators using low-end devices can run the AI system efficiently. By following a few simple steps, users can have EduTutor AI up and running, enabling interactive learning and quiz generation without needing advanced technical skills or complex local installations.

Step 1: Open Google Colab and Create a New Notebook

Begin by opening **Google Colab** in a web browser and creating a new notebook. Once the notebook is created, rename it to something identifiable, such as **EduTutor AI**, to keep your project organized. This notebook will serve as the main workspace for setting up and running the AI system.

Step 2: Enable GPU for Faster Performance

To enhance processing speed, it is recommended to use a GPU-enabled runtime. Navigate to the menu and select **Runtime → Change Runtime Type → Hardware Accelerator → GPU**, then save the settings. This step ensures that the IBM Granite model can process natural language queries efficiently and provide quick responses to user inputs.

Step 3: Install Required Libraries

Next, install the necessary Python libraries by running the command:

```
!pip install transformers torch gradio -q
```

This command installs **Hugging Face Transformers**, **PyTorch**, and **Gradio**, which are essential for loading AI models, handling computations, and creating an interactive web interface. The -q flag ensures a clean installation with minimal output.

Step 4: Load the IBM Granite Model

Import the **granite-3.2-2b-instruct** model from Hugging Face. Both the **tokenizer** and **model** need to be loaded in Python to enable the AI to understand natural language queries and generate meaningful responses. This lightweight model is optimized for educational tasks and works effectively in a cloud-based environment.

Step 5: Define Processing Functions and Interface

Create functions to handle user input and generate AI responses. Use **Gradio** to build an interactive interface where learners can type questions, request explanations, or generate quizzes. This interface provides real-time results, making the learning experience immediate and engaging.

Step 6: Run and Test the Application

Run the notebook in Colab. A **Gradio app link** will be generated, which opens the EduTutor AI interface in a new browser tab. Test the system by entering queries and ensuring responses are accurate and timely.

Step 7: Save and Upload to GitHub

After successful testing, download the notebook as a **.py file**. Upload this file to **GitHub** for version control, collaboration, and easy submission.

6. PROJECT WORKFLOW:

Setting up EduTutor AI is a straightforward process that can be completed by anyone with basic programming knowledge. The project leverages **Google Colab**, a free, cloud-based platform that provides GPU support, ensuring that even students or educators using low-end devices can run the AI system efficiently. By following a few simple steps, users can have EduTutor AI up and running, enabling interactive learning and quiz generation without needing advanced technical skills or complex local installations.

Step 1: Open Google Colab and Create a New Notebook

Begin by opening **Google Colab** in a web browser and creating a new notebook. Once the notebook is created, rename it to something identifiable, such as **EduTutor AI**, to keep your project organized. This notebook will serve as the main workspace for setting up and running the AI system.

Step 2: Enable GPU for Faster Performance

To enhance processing speed, it is recommended to use a GPU-enabled runtime. Navigate to the menu and select **Runtime → Change Runtime Type → Hardware Accelerator → GPU**, then save the settings. This step ensures that the IBM Granite model can process natural language queries efficiently and provide quick responses to user inputs.

Step 3: Install Required Libraries

Next, install the necessary Python libraries by running the command:

```
!pip install transformers torch gradio -q
```

This command installs **Hugging Face Transformers**, **PyTorch**, and **Gradio**, which are essential for loading AI models, handling computations, and creating an interactive web interface. The -q flag ensures a clean installation with minimal output.

Step 4: Load the IBM Granite Model

Import the **granite-3.2-2b-instruct** model from Hugging Face. Both the **tokenizer** and **model** need to be loaded in Python to enable the AI to understand natural language queries and generate meaningful responses. This lightweight model is optimized for educational tasks and works effectively in a cloud-based environment.

Step 5: Define Processing Functions and Interface

Create functions to handle user input and generate AI responses. Use **Gradio** to build an interactive interface where learners can type questions, request explanations, or generate quizzes. This interface provides real-time results, making the learning experience immediate and engaging.

Step 6: Run and Test the Application

Run the notebook in Colab. A **Gradio app link** will be generated, which opens the EduTutor AI interface in a new browser tab. Test the system by entering queries and ensuring responses are accurate and timely.

Step 7: Save and Upload to GitHub.

7. Implementation

The implementation of EduTutor AI focuses on integrating **IBM Granite models** with an interactive interface using **Gradio**, creating a seamless and userfriendly experience for students and educators. The process begins with importing essential Python libraries, including **Transformers** from Hugging Face, **PyTorch** for model computations, and **Gradio** for building the web interface. These libraries form the backbone of the system, enabling efficient handling of natural language processing tasks and interactive user engagement.
The first critical step in the implementation is **loading the AI model and tokenizer** from Hugging Face. The Granite-3.2-2B-Instruct model is selected for its lightweight architecture and suitability for educational purposes. In Python, this is done as follows:

```
from transformers import AutoTokenizer, AutoModelForCausalLM

tokenizer = AutoTokenizer.from_pretrained("ibm-granite/granite-3.2-
2binstruct") model = AutoModelForCausalLM.from_pretrained("ibm-
granite/granite-3.2-2binstruct")
```

The tokenizer converts the user's natural language input into a format the model can process, while the model generates intelligent responses based on the input query. This combination allows the system to understand complex questions and provide meaningful answers efficiently.
Next, a **processing function** is created to handle user input. This function takes a student's question, tokenizes it using the Granite tokenizer, feeds it to the model, and decodes the output into human-readable text. For instance, if a student asks, "Explain binary trees," the AI generates a structured, step-by-step explanation that is clear and easy to understand. This function can also be adapted to generate quizzes by providing prompts such as, "Generate five multiple-choice questions on Python functions." The AI then dynamically produces assessment content, making the learning experience interactive and engaging.
After the processing function is defined, a **Gradio interface** is built to enable real-time interaction. Students can type their queries into a simple text box and receive immediate answers from the AI tutor. The interface supports both concept explanations and quiz generation, creating a versatile educational tool. Once the Colab notebook is executed, Gradio provides a shareable app link, which opens in a browser and allows seamless access to the AI tutor without requiring local installations.

The final implementation step involves **saving the notebook as a .py file** and uploading it to **GitHub**. This ensures that the project is accessible, shareable, and reusable, while also enabling version control and collaboration among team members.

8. Future Enhancements

While EduTutor AI already offers a robust and interactive learning experience, there are several enhancements that could make the platform even more powerful, user-friendly, and accessible in the future. These improvements aim to expand the system's capabilities, reach, and adaptability, transforming EduTutor AI into a truly comprehensive digital learning companion.

One significant enhancement is Voice Interaction. By allowing students to ask questions and receive responses through speech, EduTutor AI can become more natural and engaging, particularly for younger learners or users who may have difficulty typing. Voice-enabled interaction would make the platform more inclusive, accommodating students with disabilities and enhancing the overall ease of use. Implementing speech recognition and text-to-speech capabilities could further enrich the learning experience, allowing students to listen to explanations and participate in quizzes orally.

Another potential improvement is the development of a Mobile Application. Smartphones are ubiquitous, and a dedicated EduTutor AI app would enable students to access their AI tutor anytime and anywhere. A mobile app could offer offline access, push notifications for learning reminders, and a more personalized interface optimized for small screens, increasing engagement and convenience for learners on the go.

Adaptive Learning is another critical area for enhancement. By tracking a student's performance over time, EduTutor AI can adjust the difficulty of explanations and quizzes dynamically. For instance, if a student demonstrates strong understanding of a topic, the system can generate more challenging questions to encourage growth. Conversely, for topics where a student struggles, the AI can provide simpler explanations, additional examples, or extra practice exercises, ensuring personalized learning at every step.

Integration with Learning Management Systems (LMS) such as Google Classroom, Moodle, or Canvas would further enhance the platform's utility for educators. This integration would allow teachers to assign AI-generated quizzes directly, track student progress, and monitor class performance efficiently. Coupled with an Analytics Dashboard, teachers could gain deeper insights into individual and group performance, identifying strengths and weaknesses and providing targeted support where needed.

Finally, expanding multilingual support would enable EduTutor AI to reach students from diverse linguistic backgrounds, breaking down language barriers in education and promoting inclusive learning worldwide.
By implementing these enhancements—voice interaction, mobile access, adaptive learning, LMS integration, analytics, and multilingual support—EduTutor AI can evolve into a fully integrated, future-ready educational platform. These improvements would bridge the gap between traditional teaching methods and modern AI-driven learning, creating a more interactive, personalized, and globally accessible educational experience for students everywhere.

9. Conclusion

EduTutor AI represents the integration of Artificial Intelligence and education, bringing personalized learning to every student's fingertips. By leveraging IBM Granite models, Hugging Face, Google Colab, and Gradio, this project demonstrates how modern tools can simplify education and make it more engaging.

The project successfully provides concept explanations, quiz generation, and interactive engagement. Unlike traditional classroom teaching, EduTutor AI adapts to the learner's pace, providing instant feedback and endless practice opportunities. Teachers can also benefit by using the system to generate study materials quickly.

One of the project's biggest strengths is accessibility. Running on Google Colab removes the need for powerful hardware, making AI-driven learning tools available to anyone with internet access. Furthermore, GitHub integration ensures collaboration, version control, and future scalability.

EduTutor AI is not just a project but a vision for the future of learning. It shows how AI can act as a personal tutor, guiding students with clarity and patience. As the system evolves with features like voice interaction, adaptive difficulty, and mobile applications, it has the potential to become a mainstream educational tool.

EduTutor AI exemplifies the transformative potential of integrating **Artificial Intelligence** with modern educational practices, bringing personalized, interactive learning directly to every student's fingertips. By leveraging advanced tools such as **IBM Granite models**, **Hugging Face**, **Google Colab**, and **Gradio**, this project demonstrates how contemporary technologies can simplify complex educational concepts, make learning more engaging, and provide tailored support for students of all levels.

In conclusion, EduTutor AI bridges the gap between traditional teaching and AIdriven education. It makes learning smarter, faster, and more personalized, empowering students and teachers alike. With continued development, EduTutor AI can transform the way education is delivered, ensuring no student is left behind.