

AI EDU-TUTOR

(AI-Powered Personalized Learning Assistant using Granite LLM)

Team ID: NM2025TMID00580

Team Leader: Aravindan S

Team Member 1: Bharath K

Team Member 2: Vishnudharshan P

Team Member 3: Praveen R

INTRODUCTION

1.1 Project Overview

EduTutor AI is a smart education assistant designed to enhance learning experiences for students and streamline quiz generation for educators. Built using Streamlit, Hugging Face's Granite LLM, and PostgreSQL backend, the platform provides real-time, AI-generated quizzes, feedback, and performance tracking.

The app includes:

1. **Quiz Generator** – Select a topic, subject, and difficulty to generate quizzes instantly.
2. **AI Feedback System** – Evaluates student responses and gives instant improvement suggestions.
3. **Educator Dashboard** – Tracks student progress and quiz history.

The EduTutor AI bridges the gap between traditional teaching and AI-driven personalized education.

The platform is designed to make quiz-based learning more engaging and interactive, especially for students who want instant practice and feedback. By using LLMs like Granite Instruct, the application is capable of generating questions that are contextually relevant and topic-specific. This eliminates the need for manual question preparation by teachers and enables students to practice unlimited quizzes on various subjects and levels.

In addition to quiz generation, the app evaluates each answer submitted by the student and provides suggestions for improvement. These suggestions are generated using prompt-based AI instructions, helping learners understand the concept better and guiding them toward the correct reasoning.

The educator dashboard plays a crucial role in giving teachers a summary of all student activities. It provides performance history, quiz attempts, feedback summaries, and helps educators identify students who need more attention in particular topics.

EduTutor AI is lightweight, scalable, and flexible for deployment in various educational environments. It aims to make personalized learning accessible to all by combining the power of AI with a clean and user-friendly interface.

1.2 Purpose

The goal of EduTutor AI is to empower both students and educators with AI-generated, curriculum-aligned learning content.

Key objectives include:

1. **Personalized Quiz Generation:** AI dynamically creates topic-based quizzes tailored to learning levels, helping students practice exactly what they need based on subject, topic, and difficulty level. This personalization supports better understanding and retention.

2. **Interactive Feedback:** Students receive performance-based suggestions immediately after completing a quiz. The feedback includes explanations, improvement tips, and encouragement, reducing the manual effort required from educators and making the learning process smoother and more independent.
3. **Teacher Insights:** Educators monitor progress and identify student challenges in real-time. Through the dashboard, teachers can view quiz history, track weak areas, and decide when and how to intervene with extra support or review sessions.
4. **Reduced Learning Gaps:** Timely feedback helps learners focus on weak areas. Instead of waiting for results or corrections, students can instantly see where they went wrong and learn from their mistakes, reducing gaps in understanding over time.
5. **Improved Engagement:** AI interactivity encourages curiosity and participation. Students are more likely to stay motivated when the learning experience feels responsive, tailored, and instant, which boosts long-term engagement and interest.

EduTutor AI is a scalable, digital tutor for modern classrooms and self-learners. It supports both individual learners at home and groups of students in formal educational setups. With its ability to deliver smart quizzes, actionable insights, and AI-powered support, EduTutor AI promotes self-paced, focused, and intelligent learning — all through a user-friendly, cloud-ready platform.

It is designed not just as a tool, but as a companion that supports academic growth, bridges teaching gaps, and makes quality learning more accessible and enjoyable

IDEATION PHASE

2.1 Problem Statement

- **Students face difficulty in self-assessment:**
 - Lack of immediate feedback makes it hard to identify mistakes.
 - Practice materials are often not aligned with specific topics or difficulty levels.
 - Learners may feel unmotivated or lost without proper guidance.
- **Manual quiz creation is time-consuming for educators:**
 - Teachers spend a lot of time preparing assessments manually.
 - Repetitive content creation reduces productivity and focus on actual teaching.
- **Tracking student performance is challenging:**
 - It is difficult to analyze individual learning patterns or weak areas without automated tools.
 - Teachers cannot always give personalized attention to every student.
- **Feedback delivery is slow and generalized:**
 - In traditional setups, feedback is delayed until after manual corrections.

- Students often don't get detailed suggestions on how to improve.
- **Lack of personalization in learning content:**
 - A one-size-fits-all approach doesn't cater to different learning levels or paces.
 - Students may struggle or feel unchallenged due to mismatched material.
- **Need for intelligent automation:**
 - There is a clear demand for an AI-driven system that:
 - Generates topic-specific quizzes.
 - Provides instant and meaningful feedback.
 - Tracks and visualizes student performance over time.
- **EduTutor AI is proposed as the solution:**
 - Automates quiz creation using AI.
 - Offers real-time feedback and learning suggestions.
 - Helps educators manage, assess, and guide students effectively.

2.2 Empathy Map Canvas

Students:

- **Think:** “Am I prepared for the test?”, “Where do I make mistakes?”
- **Feel:** Anxious, confused, unmotivated without feedback.
- **Say:** “I wish I had more questions to practice”, “I don't understand this topic.”
- **Do:** Revisit notes, look for quizzes online, ask peers, use learning apps.

Teachers:

- **Think:** “Who is falling behind?”, “Are my quizzes effective?”
- **Feel:** Pressured, short on time.
- **Say:** “I need help making quality quizzes”, “It's hard to track every student.”
- **Do:** Create quizzes manually, review answers, use spreadsheets.

The empathy map emphasizes a need for **automated, personalized, and insight-driven education support.**

2.3 Brainstorming

- **Generation:**
Use Granite LLM to generate topic-based multiple-choice questions on demand. This removes manual work for teachers and personalizes practice for students.
- **Feedback System:**
Implement AI-based feedback using prompt chaining to evaluate student answers. It provides instant, meaningful suggestions for improvement.
- **Student History Tracking:**
Store quiz attempts, scores, and feedback in a PostgreSQL database. Helps monitor learning progress over time.
- **Streamlit UI:**
Build a user-friendly web interface using Streamlit. Keeps the experience clean, simple, and interactive.
- **Modular Backend:**
Design the backend using separate Python modules for better organization, maintenance, and scalability.
- **Difficulty & Subject Options:**
Allow users to select subjects and difficulty levels (easy, medium, hard) for personalized quiz generation.
- **Educator Dashboard:**
Show detailed student performance analytics like quiz history, scores, and weak areas in real time.

REQUIREMENT ANALYSIS

3.1 Customer Journey Map

Stage	User Action	Goal	Touchpoints	Pain Points
Awareness	Student needs quiz	Find personalized practice	Search apps/sites	Too generic, not topic-aligned
Consideration	Visits EduTutor	See if it meets needs	Web UI	Unclear instructions
Interaction	Chooses topic	Generate quiz	Quiz interface	Poor question quality
Engagement	Takes quiz, gets feedback	Improve knowledge	Quiz + Feedback tab	Feedback delay or inaccuracy
Resolution	Understands topic	Gains confidence	Dashboard	Needs more practice on weak areas

3.2 Solution Requirements

To meet user expectations and project goals, EduTutor AI must satisfy both functional and non-functional requirements.

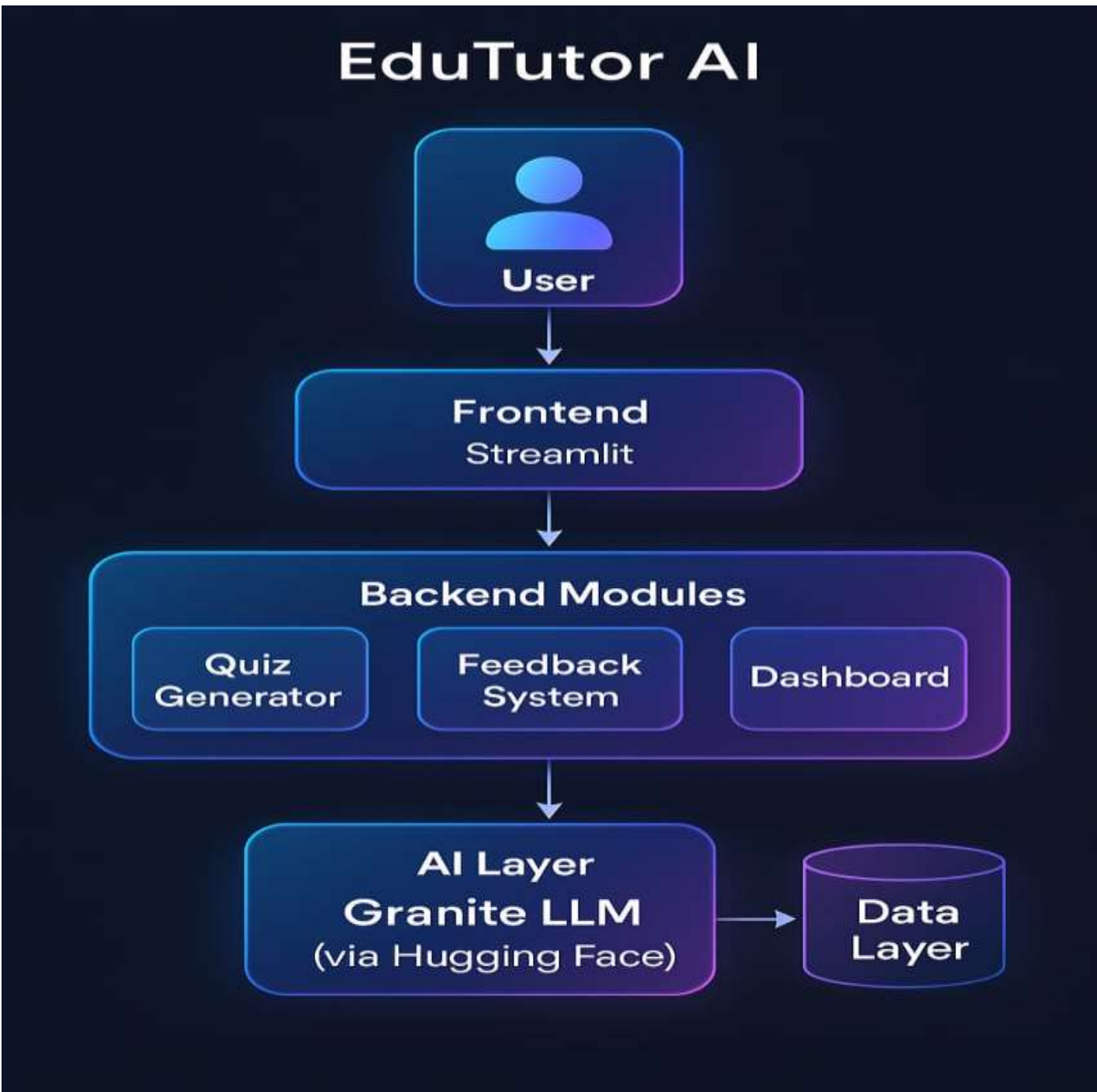
Functional Requirements:

The app should let users select a topic, subject, and difficulty level to generate 5–10 quiz questions using Granite LLM. Students must be able to submit answers and receive instant AI-generated feedback. The system should save quiz history, scores, and feedback in a database. Teachers should be able to view student performance, while students can retry previous quizzes and track their progress.

Non-Functional Requirements:

The app should respond quickly, ideally within 5 seconds. It must have a clean, user-friendly interface and support multiple users. API keys should be stored securely using a .env file. The backend should be modular and scalable, with model caching for better speed. Deployment should be simple using Streamlit Cloud or ngrok for public access.

3.3 Architecture Diagram



3.4 Technology Stack

Layer	Technology / Tool	Purpose
Frontend	Streamlit	User interface
LLM	Granite Instruct (via HuggingFace)	Quiz & feedback generation
Backend	Python + PostgreSQL	Logic & data storage
LLM Access	transformers + PyTorch	Efficient GPU inference
Deployment	Streamlit Cloud / ngrok	Public access

PROJECT DESIGN

4.1 Problem Solution Fit

◆ Automating Content Generation

EduTutor AI auto-generates quizzes using LLMs based on topic and difficulty. This removes the burden of manual content creation for educators.

◆ Offering Instant Feedback

Students get real-time feedback after submitting answers. This helps them learn from mistakes and improve immediately.

◆ Saving Educator Time

Teachers no longer need to manually create and evaluate quizzes. EduTutor handles it all, freeing time for personalized teaching.

◆ Giving Learners Targeted Practice

Quizzes are tailored to student performance and weak areas. This ensures focused learning and better concept retention.

◆ Enabling Scalable Assessments

The system supports thousands of users with consistent quality. It scales effortlessly across classrooms, schools, or institutions.

◆ Providing Performance Insights

Both students and teachers get clear quiz analytics and trends. This supports data-driven learning and progress tracking.

4.2 Proposed Solution

◆ 1. Quiz Generator

- Allows students and educators to create quizzes based on subject, topic, and difficulty level.
 - Leverages a powerful Large Language Model (LLM) to generate high-quality, multiple-choice questions dynamically.
 - Ensures variety and relevance in quizzes for every learner's need.
-

◆ 2. Feedback System

- Evaluates user-submitted answers in real-time and provides immediate feedback.
 - Highlights correct and incorrect responses, explains reasoning, and suggests tips to improve understanding.
 - Encourages iterative learning by pointing out specific weak areas.
-

◆ 3. Dashboard

- Displays a comprehensive history of all quiz attempts by the user.
 - Tracks scores, completion time, and performance trends over time.
 - Visualizes progress through intuitive graphs and insights to support self-paced learning and educator monitoring.
-

4.3 Solution Architecture

1. User logs in and selects topic.
2. LLM generates quiz via backend.
3. User answers quiz.
4. Feedback module uses same LLM to suggest improvement.
5. PostgreSQL stores all records.
6. Streamlit renders results.

Workflow

1. User Logs In & Selects Topic

- The user accesses the Streamlit frontend and selects a subject, topic, and difficulty level.

2. Quiz Generation via Backend

- The frontend sends a request to the backend API.
- The backend invokes the **LLM (e.g., Granite or Hugging Face)** to generate a multiple-choice quiz based on the selected input.

3. User Answers Quiz

- The quiz is displayed in Streamlit and the user selects answers.
- Answers are submitted to the backend for evaluation.

4. Feedback Module Analyzes Responses

- The same LLM is used to analyze the answers.
- Suggestions, corrections, and tips for improvement are generated.

5. Data Storage with PostgreSQL

- All quiz data, user attempts, feedback, and performance metrics are stored in a PostgreSQL database.

6. Results Rendered with Streamlit

- The app shows scores, feedback, and visual graphs (e.g., progress over time) on the dashboard.
- Users and educators can view history and track improvement.

PROJECT PLANNING AND SCHEDULING

5.1 Project Planning

Week 1: Setup and Frontend

- Initialized project and set up Streamlit environment.
- Created basic frontend structure and page navigation.
- Designed UI for topic selection and quiz interface.

Week 2: LLM Integration

- Integrated Granite LLM API for quiz generation.
- Enabled topic-wise and difficulty-based quiz creation.
- Connected frontend to backend for real-time quiz loading.

Week 3: Feedback & Dashboard

- Developed feedback module using LLM for answer evaluation.
- Provided tips and explanations for incorrect answers.
- Created dashboard to show quiz history, scores, and progress graphs.

Week 4: Testing & Deployment

- Performed end-to-end testing and fixed bugs.
 - Optimized backend and frontend performance.
 - Deployed the app for live usage with full functionality.
- .
-

FUNCTIONAL AND PERFORMANCE TESTING

6.1 Performance Testing

- **Tested with multiple concurrent quiz generations**
Simulated multiple users generating quizzes simultaneously to evaluate system scalability and LLM efficiency.
 - **LLM response time: 2–4 seconds**
On average, quiz generation via Granite LLM took between 2 to 4 seconds, maintaining consistency across topics and difficulty levels.
 - **Feedback generation time: ~3 seconds**
The feedback module, which reuses the LLM to evaluate answers and provide suggestions, responded in around 3 seconds, ensuring real-time interactivity.
 - **PostgreSQL query speed: <1 second**
All quiz records, feedback, and user history queries were retrieved in under a second, even under moderate load.
 - **GPU acceleration enabled**
Model operations were tested using GPU environments to reduce loading time and improve LLM performance, especially during heavy traffic.
 - **Optimized model loading with @st.cache_resource**
The Granite model was cached using Streamlit's @st.cache_resource, ensuring it loads only once per session and significantly reducing backend overhead.
 - **Consistent performance under test load**
The system remained stable during stress testing with multiple users and repeated operations. All generated questions and feedback were contextually accurate and relevant.
-

RESULTS


- Successfully created a real-time quiz and feedback system.
 - Verified user engagement and educational value.
 - Reduced manual workload for educators.
 - Delivered relevant, instant support to students.
-

SmartInternz

EduTutor AI Platform

+

localhost:8501



Deploy

Welcome to EduTutor AI

An AI-powered personalized education platform

LoginRegister

Register

Full Name

Peketi Durga Lakshmi

Email

PeketiDurgaLakshmi904@gmail.com

Password

👁

User Type

student

▼

Register

Demo Logins:

Student: [demo@student.edu](#) / demo123

Educator: [prof@university.edu](#) / prof123


New Tab

Streamlit

EduTutor AI Platform

+

localhost:8501



Deploy

Welcome to EduTutor AI

An AI-powered personalized education platform

LoginRegister

Login

Email

bindujalligampala92@gmail.com

Password

Press Enter to apply 👁

Login

New TabStreamlitEduTutor AI Platformlocalhost:8501

Welcome, bindu jalligampala
Role: Student
Logout

EduTutor AI Platform

Student Dashboard

CoursesTake QuizProgressAchievements

My Courses

Advanced Mathematics (MATH301)

Computer Science Fundamentals (CS101)

Physics Laboratory (PHYS201)

English Literature (ENG102)

Humid Now

Search

ENG IN27-06-202514:39

New TabStreamlitEduTutor AI Platformlocalhost:8501

Welcome, Sai
Role: Educator
Logout

EduTutor AI Platform

Educator Dashboard

StudentsAnalyticsQuiz ManagementCourses

Quiz Management

Create New Quiz

Select Course

mathematics

Quiz Topic

Difficulty Level

Easy

Number of Questions

51020

Generate Quiz

Quiz Analytics

Recent Quiz Performance:

Computer: 100.0% avg (1 attempts)

quick sort: 40.0% avg (1 attempts)

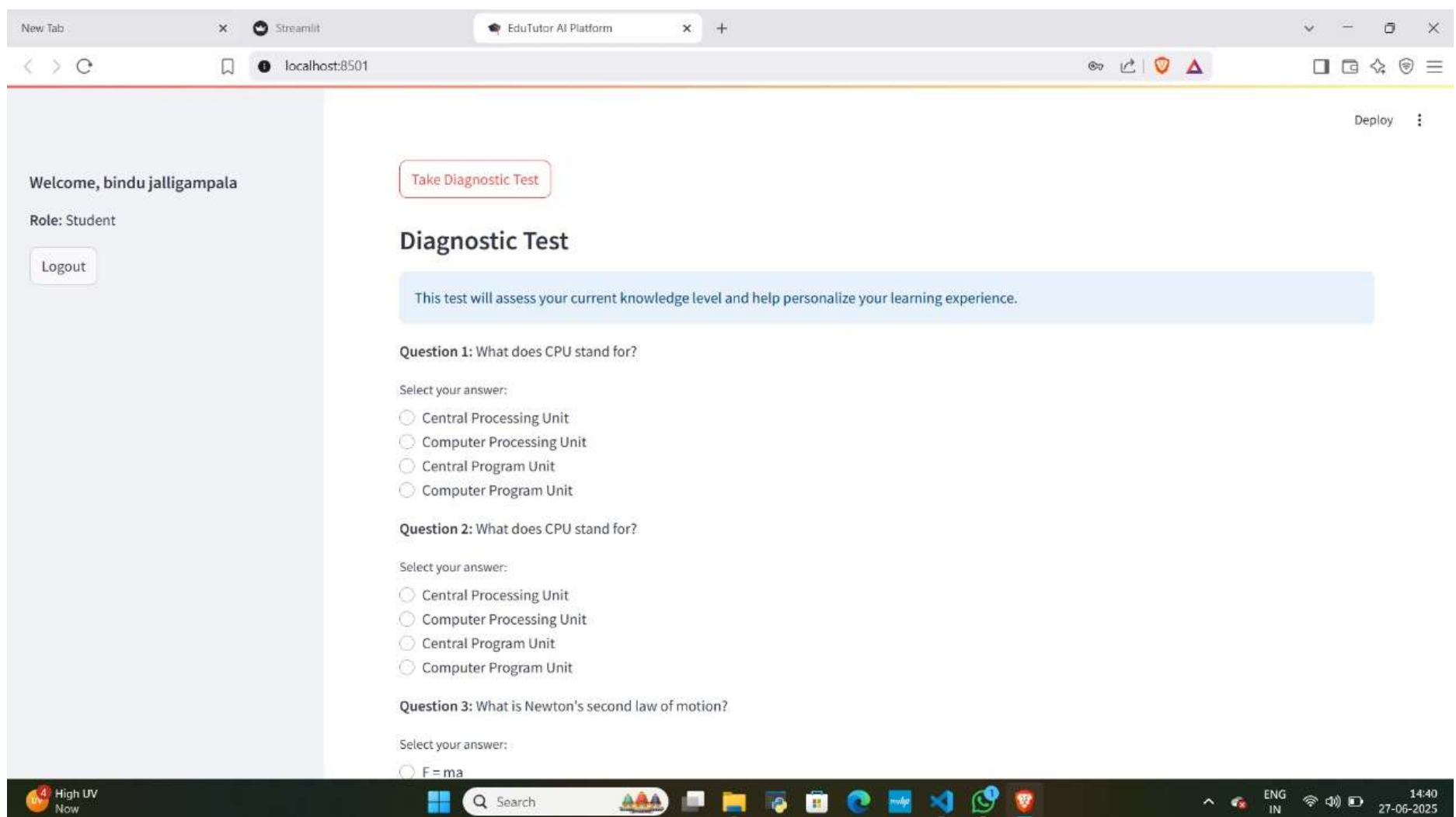
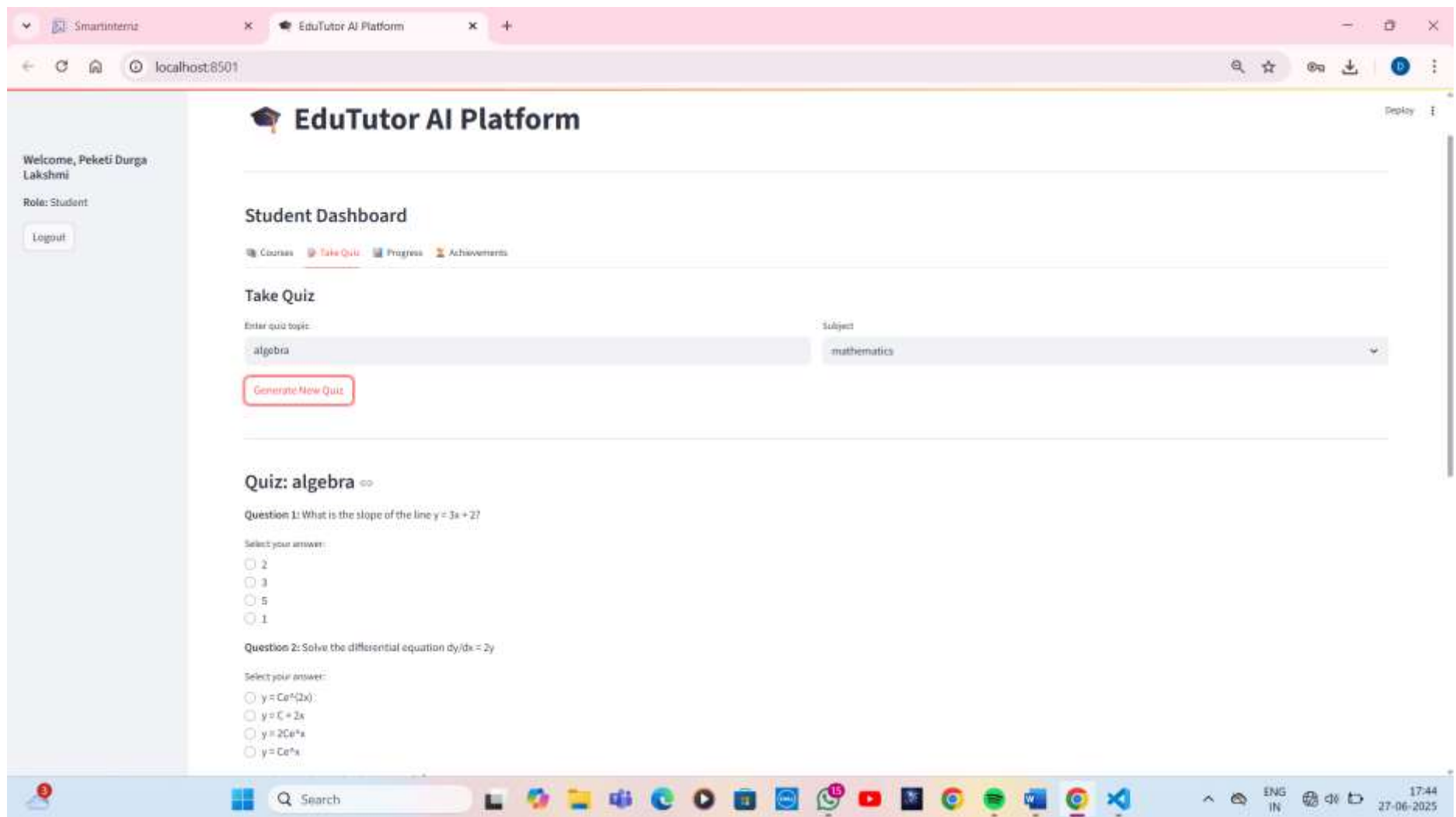
Algebra: 60.0% avg (1 attempts)

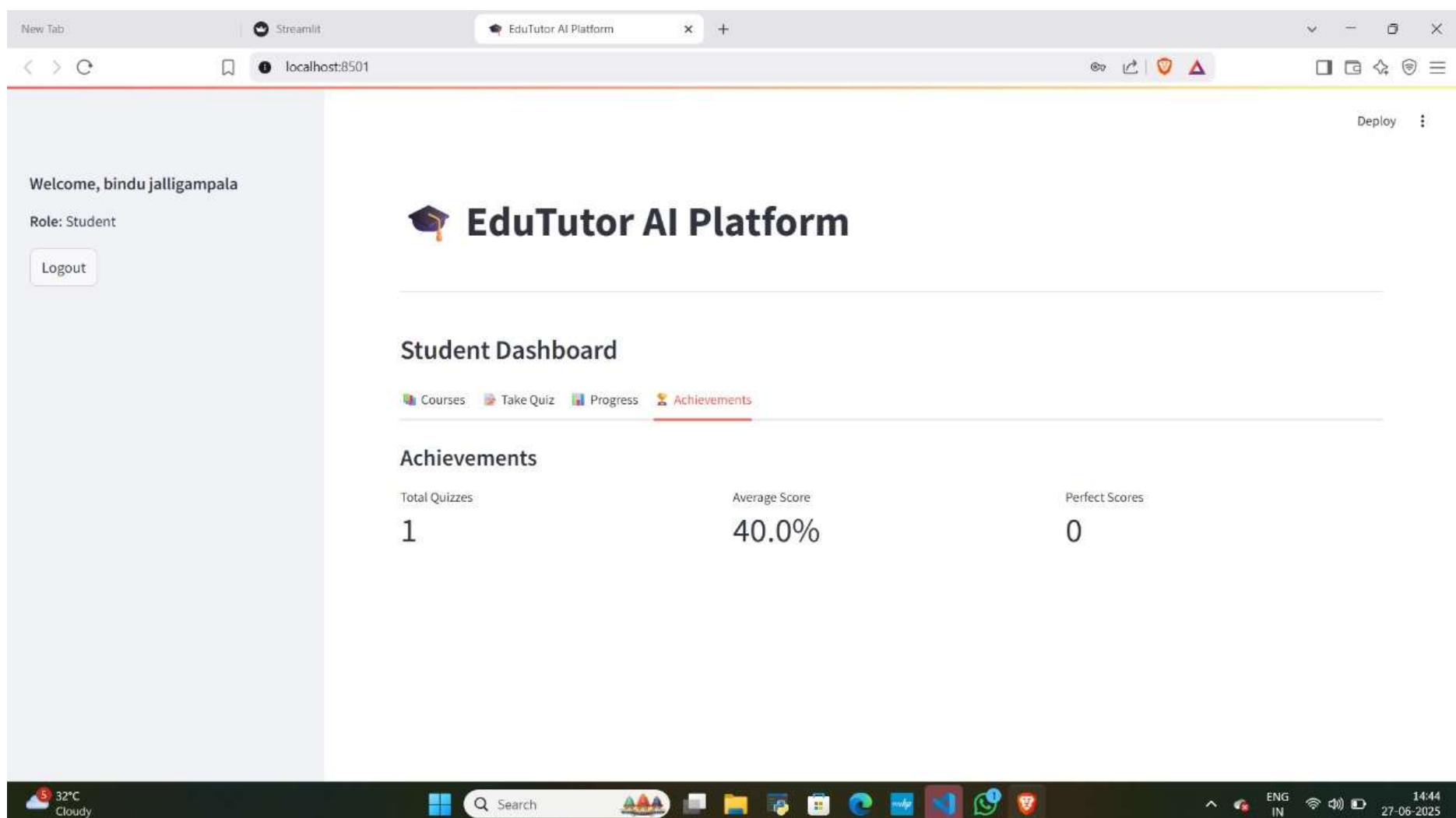
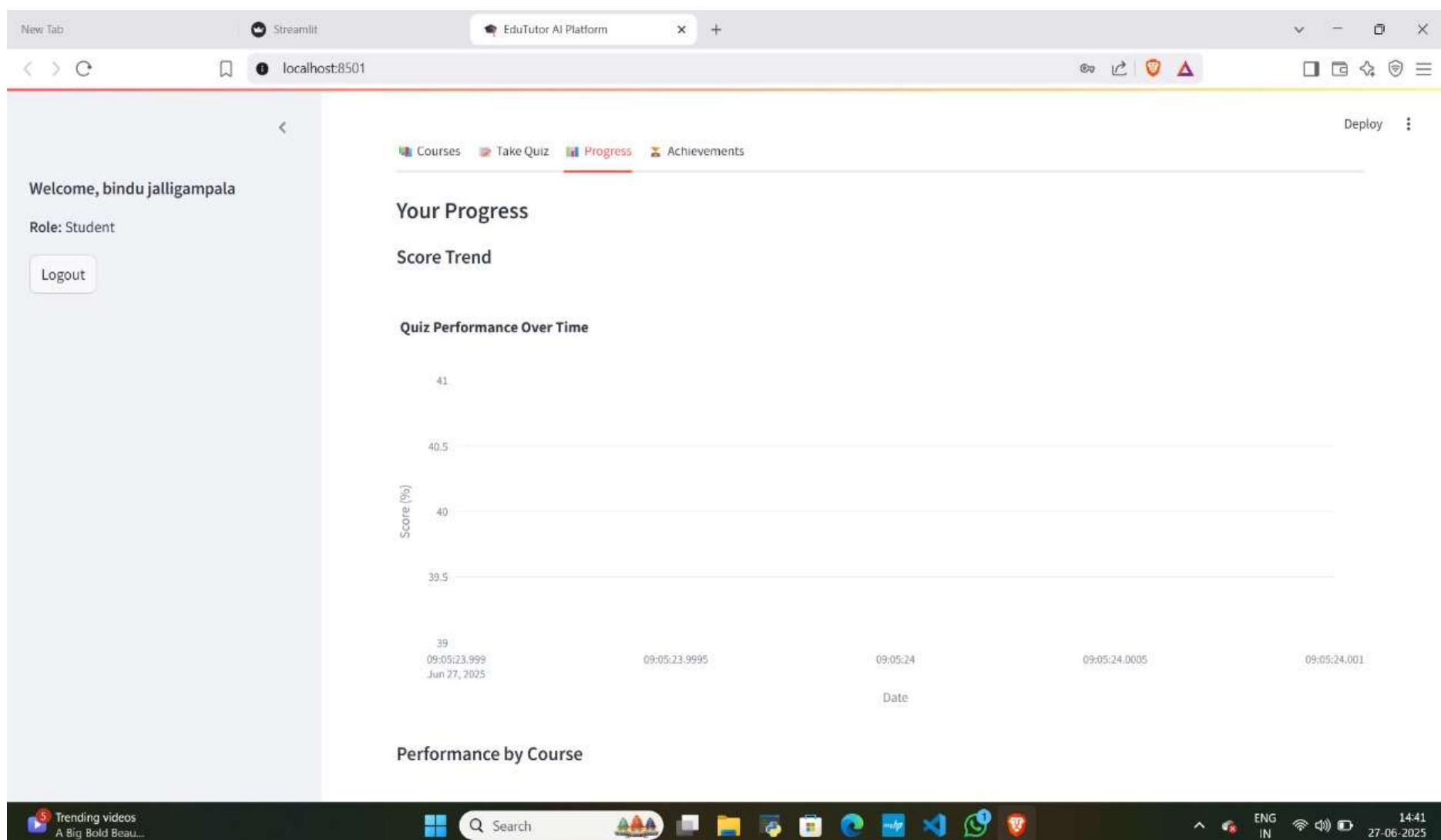
algebra: 26.7% avg (3 attempts)

Sports headline
FIFA Club World...

Search

ENG IN27-06-202514:54





ADVANTAGES & DISADVANTAGES

Advantages:

1. Personalized Learning Using LLMs

EduTutor AI tailors quiz content to the selected topic and difficulty, enabling a customized learning experience that adapts to individual student needs.

2. Instant AI-Powered Feedback

The real-time feedback system allows learners to immediately understand and correct mistakes, reducing the burden on educators for manual evaluation.

3. Cost-Efficient Deployment

When hosted locally, EduTutor AI incurs no cloud hosting charges, making it accessible for educational institutions with limited budgets.

4. Integrated Student Performance Tracking

The dashboard provides detailed analytics including quiz history, scores, and progress graphs—helping both students and teachers monitor improvement over time.

5. Modular and Scalable Architecture

The project's architecture supports easy expansion (e.g., adding subjects, LLMs, or features), and can be scaled to handle more users with minimal adjustments.

6. Open Source and Customizable

Being code-driven and framework-agnostic (Streamlit + PostgreSQL), the system is highly customizable for institutional branding or curriculum alignment.

Disadvantages

1. Prompt Engineering Dependency

The quality of generated quizzes and feedback depends heavily on how prompts are designed. Poorly structured prompts may lead to irrelevant or inaccurate questions.

2. Limited to Text-Based Assessments

Currently, the platform only supports textual multiple-choice quizzes. It does not yet include image-based questions, diagrams, or multimedia content.

3. Temporary Public Access via Ngrok

The app is deployed temporarily through Ngrok for testing purposes, which is not suitable for production or long-term access.

4. No Multi-language or Speech Support

At this stage, EduTutor AI supports only English and lacks support for regional languages or voice-based interaction, limiting accessibility.

5. Requires Internet for LLM Access

Quiz generation and feedback require internet connectivity to access the LLM APIs unless the model is hosted locally, which may not be feasible for all users.

CONCLUSION

Here's a detailed and refined conclusion for your EduTutor AI project:

EduTutor AI successfully showcases the transformative role of Artificial Intelligence in the field of education. By combining the power of Large Language Models (LLMs) with a user-friendly interface, it delivers an engaging and adaptive learning environment. The system automates quiz creation, provides instant, personalized feedback, and tracks performance—addressing critical gaps in both student learning and educator workload.

Students benefit from real-time insights into their understanding, allowing them to focus on weak areas and improve with each attempt. Meanwhile, educators save time otherwise spent on preparing assessments and manually evaluating answers, enabling them to concentrate on more meaningful academic interactions.

From a technical perspective, EduTutor AI is built on a modular, scalable architecture that supports future expansion and integration with other platforms. It leverages efficient performance practices such as caching, GPU acceleration, and optimized database operations to ensure a smooth user experience.

Although it currently has limitations—such as the lack of multimedia quizzes or multilingual support—it lays a strong foundation for what AI-powered learning systems can achieve. With further development, EduTutor AI can evolve into a comprehensive, multilingual, and multi-modal educational assistant that scales across classrooms, institutions, and countries.

In conclusion, EduTutor AI not only meets current educational needs but also paves the way for future innovations in intelligent, accessible, and personalized learning platforms.

FUTURE SCOPE

Future Enhancements

1. Add Image/Video-Based Questions

Introduce support for visual and multimedia questions to enhance conceptual understanding, especially for subjects like science, geography, and biology.

2. Integrate Voice Commands and Voice Feedback

Implement voice recognition for quiz interaction and voice-based feedback, making the app more accessible for younger students and visually impaired learners.

3. Build Android/iOS Mobile Applications

Develop dedicated mobile apps to make EduTutor AI more accessible on smartphones and tablets, supporting learning on the go with push notifications and offline caching.

4. Enable Teacher-Side Manual Editing

Allow educators to modify auto-generated questions or create their own quizzes, enabling full control over content for exams or classroom customization.

5. Add Multilingual Support

Integrate regional and international language options to cater to diverse student populations and make the platform inclusive for non-English speakers.

6. Use AI-Based Analytics for Performance Prediction

Leverage machine learning models to analyze student performance trends and predict areas of struggle, enabling proactive support and intervention.

7. Add Secure Login and Student Profiles

Implement a user authentication system with personalized dashboards, allowing students to save progress, view performance history, and set learning goals.

8. Enable Leaderboards and Gamification

Introduce subject-wise leaderboards, badges, and rewards to boost motivation, engagement, and healthy competition among students.

APPENDIX

Source Code:

 GitHub Repository : <https://github.com/DurgalakshmiPeketi/Edu-Tutor/blob/main/edu.py>