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

**1.INTRODUCTION:**

# 1.1 Project Overview

EduTutor AI is an intelligent educational platform that leverages Generative AI to provide personalized concept explanations, quiz generation, and student performance analytics. The system integrates with Learning Management Systems (LMS) to streamline the learning experience and adapt content based on individual student understanding.

# 1.2 Purpose

The purpose of this project is to enhance student engagement and improve learning outcomes through AI-driven tutoring, making quality education accessible and tailored to each learner’s pace and needs.

# 2.IDEATION PHASE

## 2.1 Problem Statement

Students often face one-size-fits-all quizzes that don’t adapt to their unique strengths or struggles. Educators lack a unified view of student performance to intervene strategically.

Edu Tutor AI addresses both by generating adaptive assessments and providing live insights.

**Problem Statement (PS-1):**

Many students struggle with self-paced learning due to lack of instant feedback, unclear concept explanations, and absence of personalized quizzes or revision tools. Traditional LMS platforms are static and fail to cater to varying learning styles.

**Problem Statement (PS-2):**

I am an educator managing diverse learners, trying to monitor student progress and intervene early, but I only see static test scores without granular insights because assessments aren’t linked to analytics— which makes me feel overwhelmed, unsure, and guilty.

## 2.2 Empathy Map Canvas

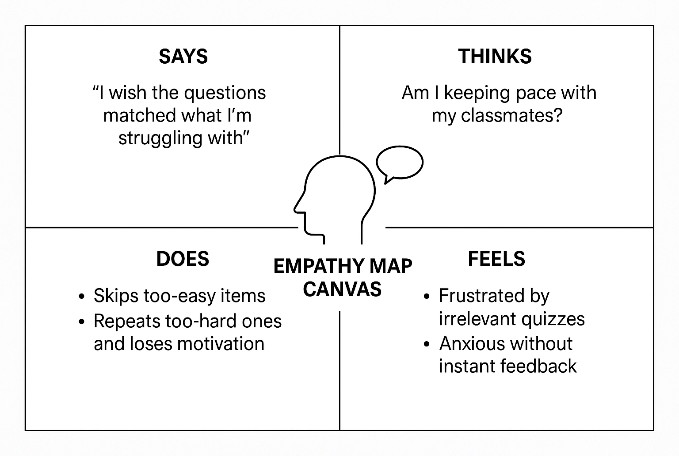
**Says**: “I need practice on what I find hardest.”

**Thinks**: “Am I falling behind my classmates?”

**Does**: Skips exercises that feel too easy; gets discouraged by repeated failures.

**Feels**: Frustrated when quiz questions don’t match what was taught; anxious without feedback.

➢ **Gain:** A system that explains simply, adapts to their pace, and provides quick tests



## 2.3 Brainstorm & Idea Prioritization

Brainstorming provides a free and open environment that encourages everyone on the team to participate in creative problem solving. Prioritizing volume over value, out-of-the-box ideas are welcome and built upon, and all participants collaborate to develop rich solutions.

## Step-1: Team Gathering, Collaboration and Select the Problem Statement

Brainstorming provides a free and open environment that encourages everyone within a team to participate in the creative thinking process that leads to problem solving. Prioritizing volume over value, out-of-the-box ideas are welcome and built upon, and all participants are encouraged to collaborate, helping each other develop a rich amount of creative solutions.We brainstormed features like subject-based quiz generation, answer evaluation, and performance tracking .

## Step-2: Brainstorm, Idea Listing and Grouping

|  |  |  |
| --- | --- | --- |
| **#** | **Idea** | **Grouping** |
| 1 | LLM-powered dynamic quiz creation | Core feature |
| 2 | Instant scoring & feedback overlays | Core feature |
| 3 | Diagnostic test for initial skill mapping | Adaptive learning |
| 4 | Difficulty adjustment algorithm | Adaptive learning |
| 5 | Google Classroom auto-sync | LMS integration |
| 6 | Educator dashboard with mastery heatmaps | Analytics & insights |
| 7 | Gamification badges & leaderboards | Engagement |
| 8 | Offline mode with cached quizzes | Accessibility |
| 9 | Multilingual question generation | Globalization |
| 10 | Voice-enabled quiz interaction for younger learners | Accessibility |

## Step-3: Idea Prioritization

Evaluated each idea on **Impact** (learning gain + teacher utility) and **Feasibility** (development effort):

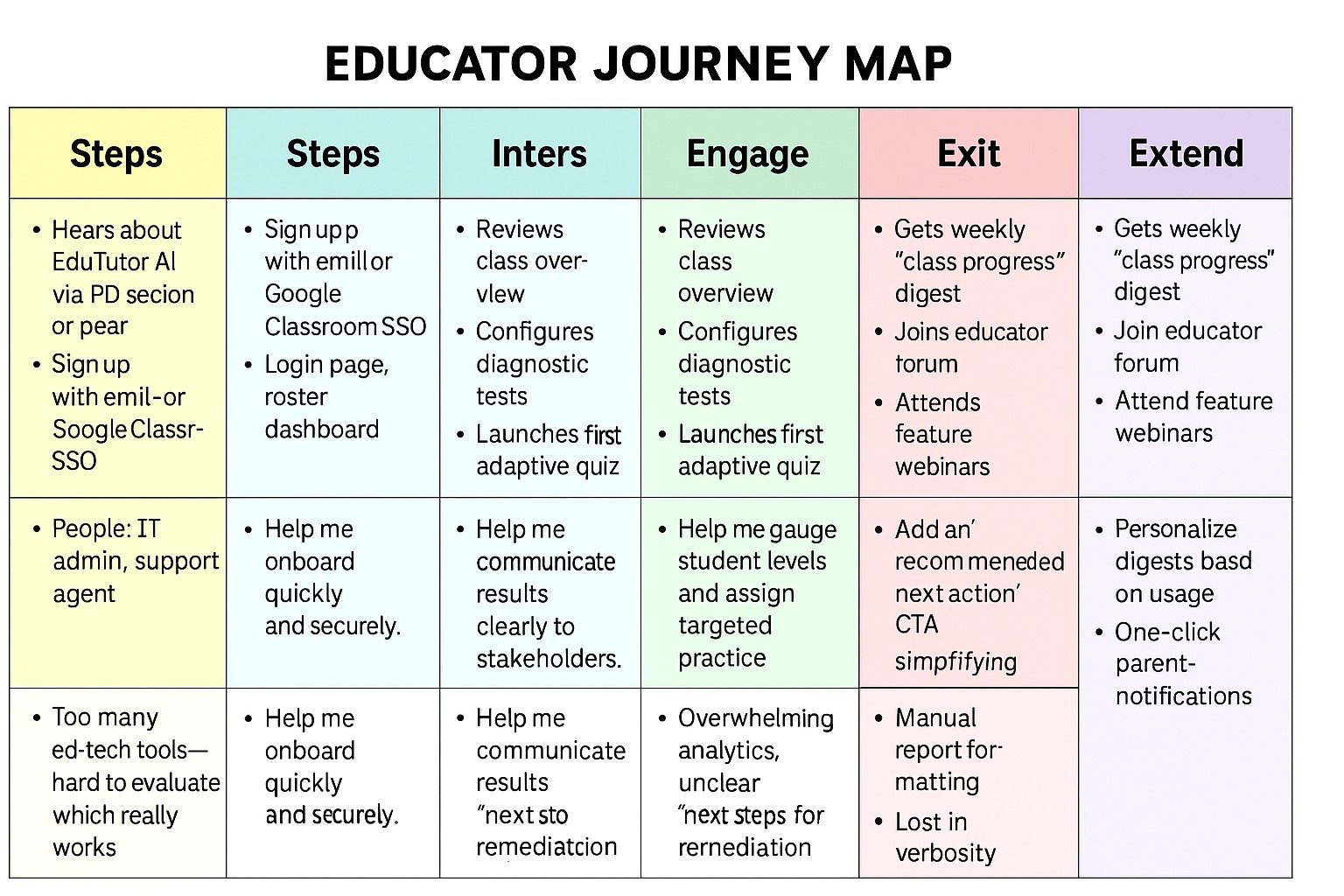
|  |  |  |  |
| --- | --- | --- | --- |
| **Idea** | **Impact** | **Feasibility** | **Priority** |
| Dynamic quiz creation | High | High | ✓✓✓ |
| Instant scoring & feedback | High | High | ✓✓✓ |
| **Idea** | **Impact** | **Feasibility** | **Priority** |
| Diagnostic test for skill mapping | High | Medium | ✓✓ |
| Google Classroom auto-sync | High | Medium | ✓✓ |
| Educator dashboard with mastery heatmaps | High | Medium | ✓✓ |
| Difficulty adjustment algorithm | Medium | Medium | ✓ |
| Gamification badges & leaderboards | Medium | Low | ✓ |
| Offline mode | Low | Low | – |
| Multilingual question generation | Low | Low | – |
| Voice-enabled quiz interaction | Low | Low | – |

**Top Priorities:**

* Dynamic quiz generation
* Instant scoring & feedback
* Diagnostic skill mapping
* Google Classroom synchronization
* Educator performance dashboard

# 3. REQUIREMENT ANALYSIS

## 3.1 Customer Journey map



**3.2 Solution Requirement**

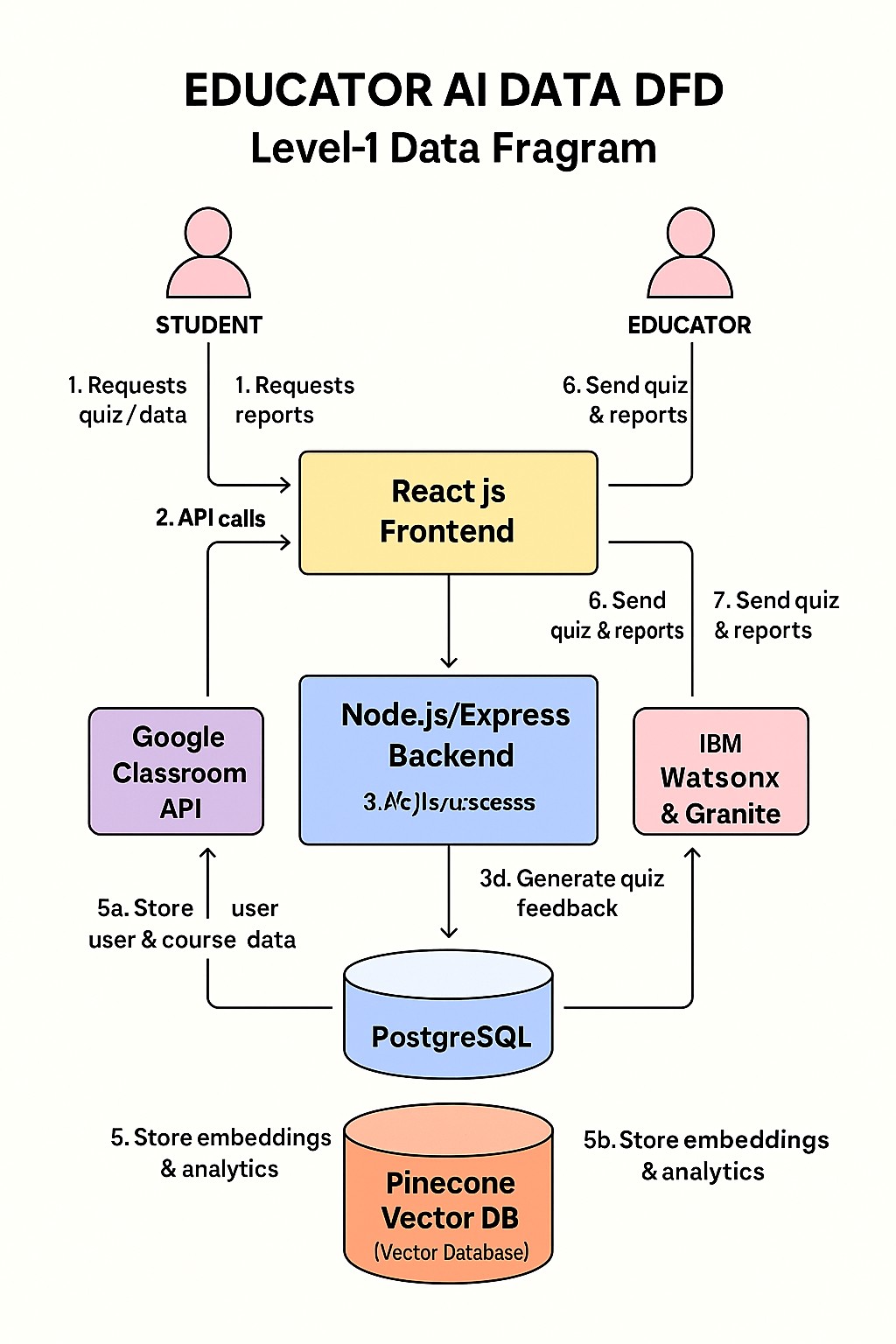
## Functional Requirements

|  |  |  |
| --- | --- | --- |
| **FR No.** | **Functional Requirement (Epic)** | **Sub Requirement (Story / Sub-Task)** |
| FR-1 | User login | • Allows a user to input their name to begin using the application. A personalized session is initiated for each user |
| FR-2 | Course Synchronization | • Syncs a predefined list of available subjects  (e.g., AI, DS, ML) to the user's profile. Mimics  Google Classroom sync behaviorl |
| FR-3 | Quiz & Assessment Management | * Dynamically generates a 3 to 6-question multiple choice quiz for a selected subject using   the IBM Granite model. No answers or explanations include   * Create and launch diagnostic tests via IBM   Watsonx  •Stores previously generated quizzes for each user within the session |
| FR-4 | LMS Integration & Analytics Reporting | * Uses IBM Granite 3.3-2B Instruct model via transforms to generate quiz content dynamically based on subject prompt * Compute real-time scoring and feedback |

## Non-functional Requirements

|  |  |  |
| --- | --- | --- |
| **NFR No.** | **Non-Functional**  **Requirement** | **Description** |
| NFR-1 | Usability | Users should be able to navigate tabs (Login, Quiz, Evaluation) with no training. |
| NFR-2 | Security | Designed with simplicity and keyboard-friendly input; future versions can be enhanced to meet WCAG 2.1 AA accessibility standards |
| NFR-3 | Reliability | 99.9% uptime SLA; automated health checks and retries; graceful error handling with user-friendly messages |
| NFR-4 | Performance | System should generate quizzes in less than 3 seconds per request. UI components must respond to clicks in under 500 ms. |
| NFR-5 | Availability | System should be available at least 99% of the time during school hours. Can be deployed via cloud with auto-recovery enabled |
| NFR-6 | Scalability | System support multiple concurrent users by using isolated session |

### 3.3 Data Flow Diagram



Students/Educators send requests via the React.js frontend.

Frontend routes requests to the Node.js/Express backend.

Backend synchronizes rosters with Google Classroom and submits quiz-generation requests to AI services.

AI services return dynamically generated questions and feedback.

Backend persists profiles in PostgreSQL and analytical vectors in Pinecone.

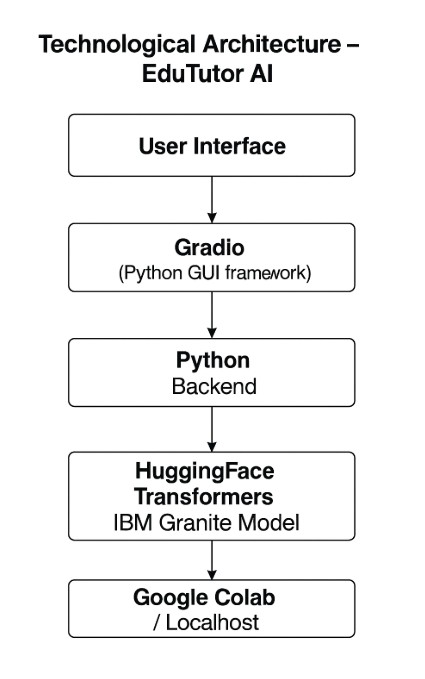
Backend returns quizzes, scores, and performance heatmaps to the frontend. Frontend renders results for students and educators.

### 3.4 User Stories

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **User**  **Type** | **Functional**  **Requirement**  **(Epic)** | **Story**  **#** | **User Story /**  **Task** | **Acceptance Criteria** | **Priority** | **Release** |
| Student (Web) | login | USN-1 | As a student, I can register with email and password so that I can create my Edu Tutor AI account. | * I complete registration form * I receive a   “Welcome”   * I land on my student dashboard | High | Sprint-1 |
| Student (Web) | Course  Synchronization | USN-2 | As a student, I  can register via  Google  Classroom SSO so that I sync my courses automatically. | * I authenticate with   Google   * My courses list appears in the dashboard | Medium | Sprint-1 |
| Student (Web) | User  Confirmation | USN-3 | As a student, I receive a confirmation email after registration so that my account is verified. | • I get an email with a confirmation link • Clicking the link marks my account as  “Verified” on U | High | Sprint-1 |
| Student (Web) | Quiz &  Assessment  Management | USN-4 | As a student, I can take a diagnostic test so that the system measures my proficiency level. | * I complete at least 6 questions * I see a summary report with strengths/weaknesses | High | Sprint-2 |
| Student (Web) | Quiz & Answer Submissiont | USN-5 | As a student, I can take an adaptive quiz so that questions adjust to my performance in real time. | * Each question adapts based on my previous answers * I receive instant feedback after each question | High | Sprint-2 |
| Educator (Web) | LMS  Integration &  Analytics  Reporting | USN-6 | As an educator, I can sync my  class roster from  Google Classroom so that Edu Tutor AI mirrors my current student list. | * I initiate sync * Student names/IDs import successfully * I see the updated     roster in “My  Classes” | High | Sprint-2 |
| Educator (Web) | LMS  Integration & | USN-7 | As an educator, I can view a class | • Dashboard heatmap loads within 2 s | Medium | Sprint-3 |
| **User**  **Type** | **Functional**  **Requirement**  **(Epic)** | **Story**  **#** | **User Story /**  **Task** | **Acceptance Criteria** | **Priority** | **Release** |
|  | Analytics  Reporting |  | performance heatmap so that I identify topic-level strengths and gaps. | • Weakest three topics are highlighted per student |  |  |
| Educato (Web) | Quiz &  Assessment  Management | USN-9 | As an admin, I can view unverified accounts so that I can resend confirmation emails or deactivate stale registrations. | * I select topics and question count • A quiz preview appears * I can export the quiz or launch it directly for my class | Medium | Sprint-4 |
| Admin (Web) | Track Multiple Users | USN-1  0 | As an admin, I can monitor system health metrics (API latency, error rate) so that I ensure platform reliability. | • In-memory dictionarie | Low | Sprint-4 |

**3.4 Technology Stack**

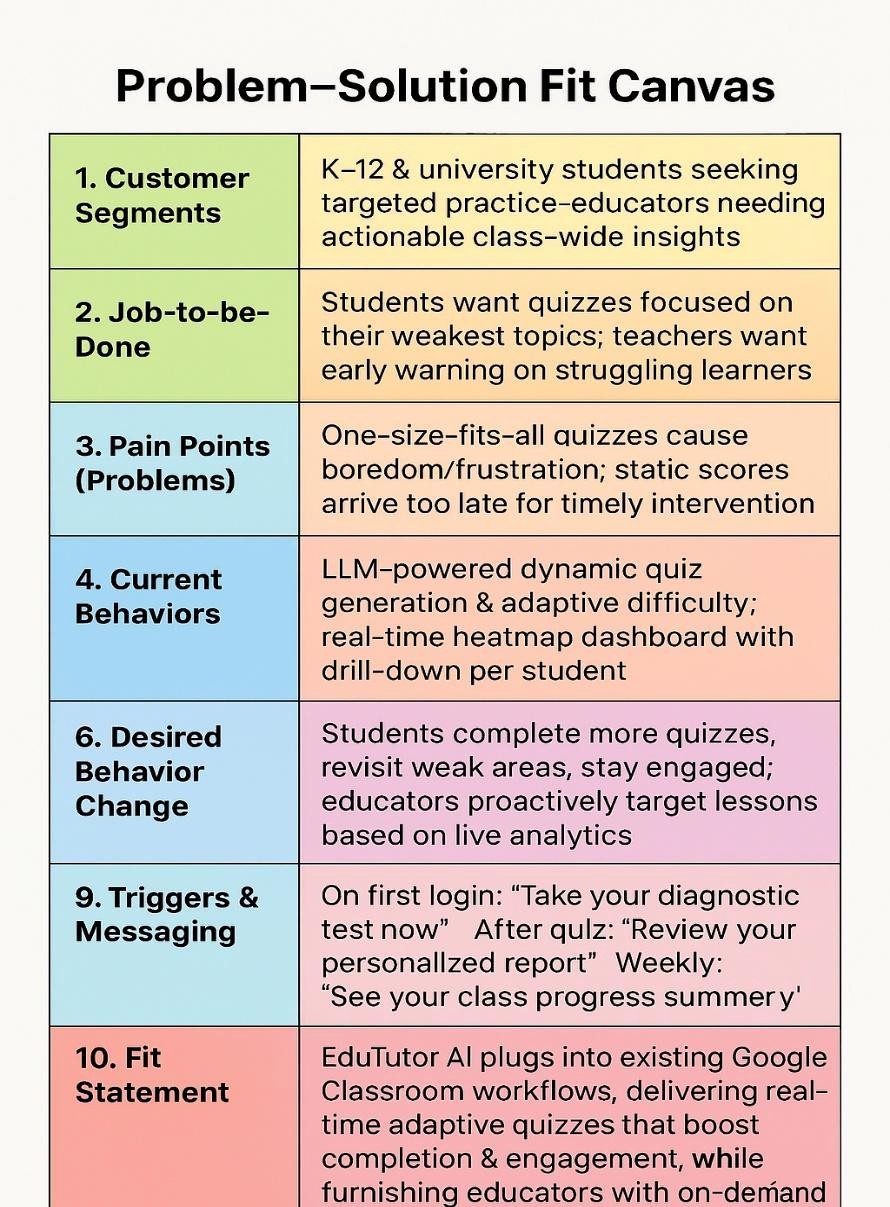
## Table 1: Components & Technologies



|  |  |  |  |
| --- | --- | --- | --- |
| **S. No** | **Component** | **Description** | **Technology** |
| 1 | User Interface | How the user interacts with the system | React.js, HTML5, CSS3, JavaScript, Redux |
| 2 | Application  Logic – Quiz Engine | Generates dynamic quizzes & feedback | Node.js, Express, Granite  LLM, IBM Watsonx |
| 3 | Application  Logic – Adaptive | Calibrates difficulty & diagnostic scoring | Python, IBM Watsonx models |
| 4 | Application  Logic – LMS Sync | Syncs roster and assignments via  LMS | Node.js, Google Classroom API (OAuth2) |
| 5 | Database | Relational metadata storage | PostgreSQL |
| 6 | Cloud Database | Vector embeddings & analytics data | Pinecone Vector DB |
| 7 | File Storage | Stores logs, reports, and exports | AWS S3 |
| 8 | External  API – Google  Classroom | Fetches courses, students, and assignments | Google Classroom REST API |
| 9 | External  API – Email Service | Sends confirmation, reminders, and reports | SendGrid (or AWS SES) |
| 10 | Machine Learning Models | LLM-based question generation and diagnostic assessment | IBM Granite foundation models; Watsonx NLU/NLG |
| 11 | Infrastructure | Hosts and scales services | AWS ECS (Fargate), Docker, GitHub Actions CI/CD |

**Table 2: Application Characteristics**

|  |  |  |  |
| --- | --- | --- | --- |
| **S. No** | **Characteristic** | **Description** | **Technology / Approach** |
| 1 | Open-Source Frameworks | Core frameworks used | React.js, Node.js, Express, Redux |
| 2 | Security | Authentication, encryption, and access control | OAuth2, JWT, TLS 1.2+, AES-256, IAM roles |
| 3 | Scalability | Supports increasing load via modular microservices | AWS ECS auto-scaling, Docker containers |
| 4 | Availability | Ensures uptime and disaster recovery | Multi-AZ deployment, ALB load balancers, backups |
| 5 | Performance | Low latency and high throughput | Redis cache for sessions/results,  AWS CloudFront CDN |
| 6 | Reliability | Robust error handling and health monitoring | Circuit breakers, CloudWatch alerts, retries |



# 4. PROJECT DESIGN

## 4.1 Problem Solution Fit

Many students struggle to understand complex academic concepts on their own and lack the ability to assess their knowledge effectively. Students and self-learners often face difficulty in finding personalized, adaptive learning content that matches their pace and understanding level.Educators struggle with time-consuming manual quiz creation and monitoring student performance effectively.

## Solution

**EduTutor AI** is a personalized learning platform that uses generative AI to explain concepts in a simplified manner, offer grammar assistance (in Hindi and English), and dynamically create personalized quizzes for a wide range of subjects:

This solution directly addresses students' needs for clarity, practice — while reducing effort for educators in content preparation.

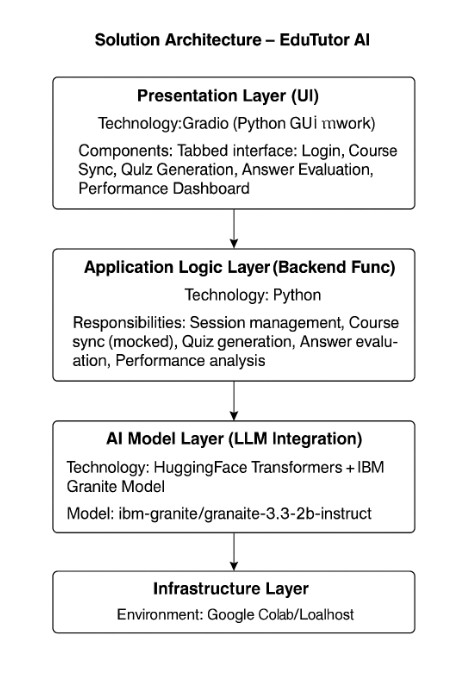
### 4.2 Proposed Solution

|  |  |  |
| --- | --- | --- |
| **S. No** | **Parameter** | **Description** |
| 1 | Problem  Statement (Problem to be solved) | Students often lack access to personalized learning tools that adapt to their interests and performance |
| 2 | Idea / Solution description | An EduTutor AI is an AI-powered educational assistant that personalizes the learning experience.. |
| 3 | Novelty /  Uniqueness | it allows dynamic quiz creation on any synced subject and provides automated evaluation with scoring and a performance dashboard, all in a user-friendly Gradio interface. |
| 4 | Social Impact /  Customer  Satisfaction | EduTutor AI enhances accessibility and personalization in education. It empowers students with adaptive learning, helps teachers save time, and provides low-cost AI-based education tools, thereby reducing learning disparities—especially beneficial in remote or underserved areas. |
| 5 | Business Model  (Revenue Model) | Freemium: basic adaptive quizzes free for individual teachers;  Premium tier ($5–10/user/mo) unlocks advanced analytics, district-wide reporting, Single Sign-On integrations, and priority support for schools and educational networks. |
| 6 | Scalability of the Solution | The solution can scale to support additional subjects, languages, and education boards. It can be integrated with more LMS platforms (like Moodle, Canvas), and deployed across schools, colleges, and training institutes globally via cloud platforms like IBM Cloud. |

### 4.3 Solution Architecture

Solution architecture is a complex process – with many sub-processes – that bridges the gap between business problems and technology solutions. Its goals are to:

* The EduTutor AI platform follows a modular, layered architecture designed for simplicity, scalability, and real-time interaction. At the presentation layer, it uses a Gradio-based UI that offers a user-friendly tabbed interface for login, course sync, quiz generation, answer evaluation, and performance review.
* The application logic is built entirely in Python and handles core functionalities like user session management, quiz history tracking, and performance evaluation. IBM Granite's large language model is integrated through Hugging Face Transformers, enabling dynamic quiz generation based on subject prompts.
* The platform runs in lightweight environments such as Google Colab or local machines and stores session data in-memory for simplicity.



# 5. PROJECT PLANNING & SCHEDULING

**5.1 Project Planning**

## 1. Product Backlog, Sprint Schedule, and Estimation

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Sprint** | **Epic** | **User**  **Story No.** | **User Story / Task** | **Story**  **Points** | **Priority** | **Team**  **Members** |
| Sprint-1 | Login &  Session  Managemen | USN-1 | As a student, I want to log in with my name to start using the app. | 2 | High | Sharon |
| Sprint-1 | Course Sync | USN-2 | As a student, I want to sync subjects so I can generate course-specific quizzes | 1 | High | Sasidhar |
| Sprint-1 | User  Registration | USN-3 | As a student, I want to log in with my name to start using the app. | 2 | Medium | Sharon,Sasi dhar |
| Sprint-1 | Session  Tracking | USN-4 | As a user, I can track my sessions after login. | 1 | High | Sharon,Sasi dhar |
| **Sprint** | **Epic** | **User**  **Story No.** | **User Story / Task** | **Story**  **Points** | **Priority** | **Team**  **Members** |
| Sprint-2 | Concept Understandin  g | USN-5 | As a user, I can enter a concept and get  AI-generated explanation. | 5 | High | Sasidhar |
| Sprint-2 | Quiz  Generation | USN-6 | As a student, I want to generate a quiz based on selected subject. | 5 | High | Sharon,teja |
| Sprint-2 | Answer  Evaluation | USN-7 | As a student, I want to submit my quiz answers and get a score. | 3 | Medium | Sharon |
| Sprint-3 | Performance Dashboard | USN-8 | As a student, I want to view my past scores and average performance | 5 | High | Sharon |
| Sprint-3 | Gradio UI Setup | USN-9 | As a developer, I can create a multi-tab UI using Gradio. | 3 | Medium | Sharon |

## 2. Sprint Schedule & Story-Point Tracking (Burndown Dashboard)

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Sprint** | **Total Story**  **Points** | **Duration** | **Start**  **Date** | **Planned End**  **Date** | **Points Completed by**  **Planned End** | **Actual**  **Release Date** |
| Sprint1 | 6 | 6 days | 17 Feb  2025 | 22 Feb 2025 | 6 | 22 Feb 2025 |
| Sprint2 | 13 | 6 days | 24 Feb  2025 | 1 Mar 2025 | 11 | 1 Mar 2025 |
| Sprint3 | 8 | 6 days | 3 Mar  2025 | 8 Mar 2025 | — | — |

Velocity Calculation

* Sprint-1 velocity: 6 points / 6 days = 1 point/day
* Sprint-2 velocity: 11 points / 6 days ≈ 1.83 points/day
* Average velocity (S1+S2): (6 + 11) / (6 + 6) = 17/12 ≈ 1.42 points/day

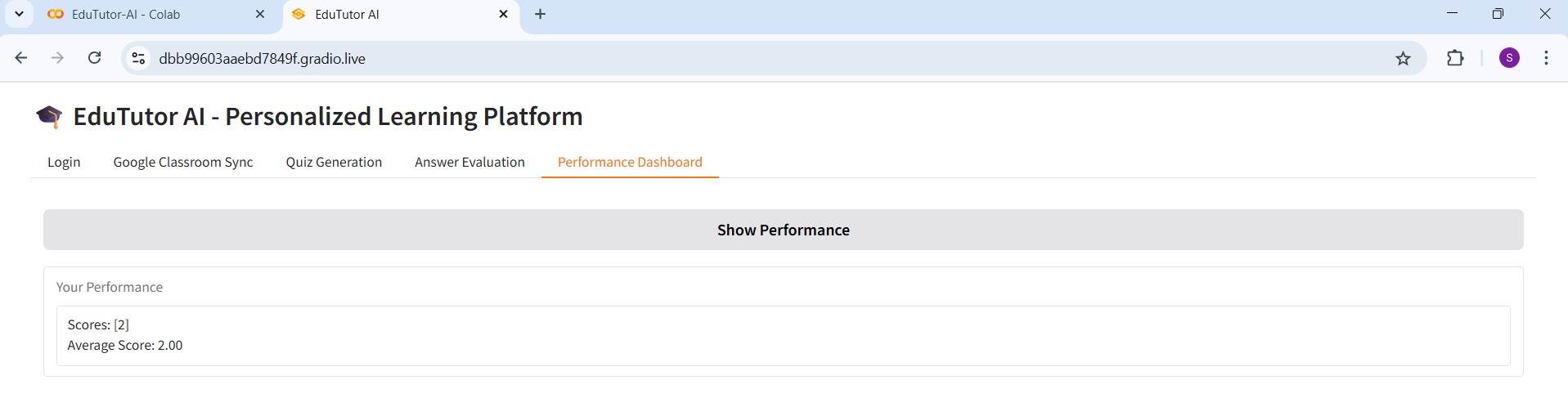
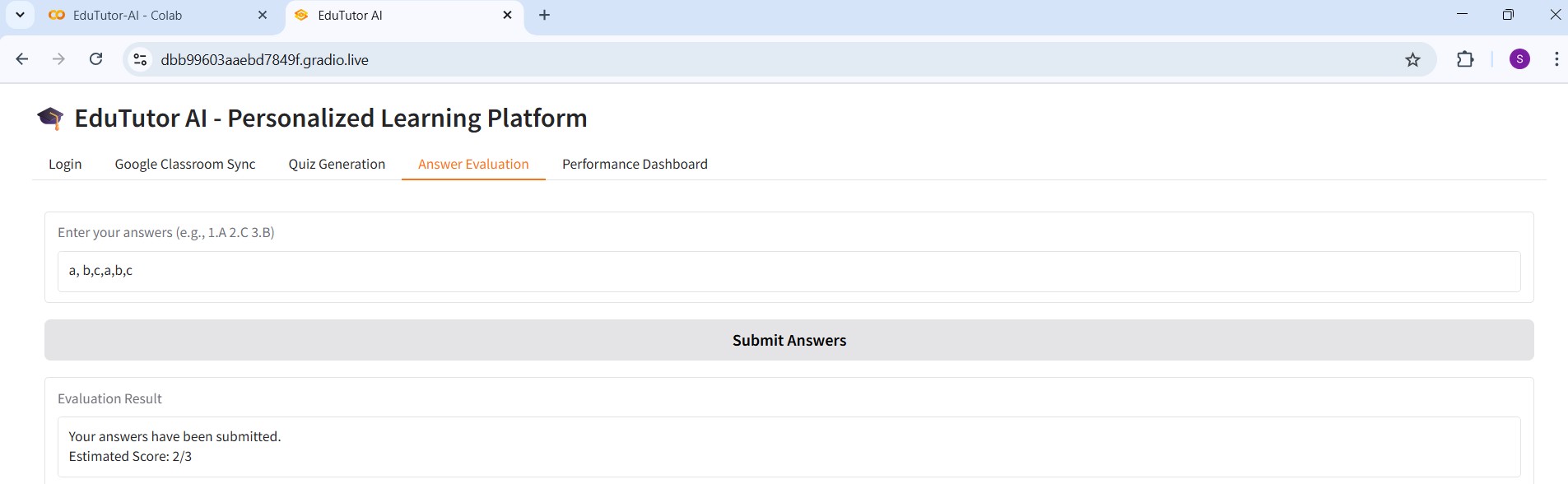
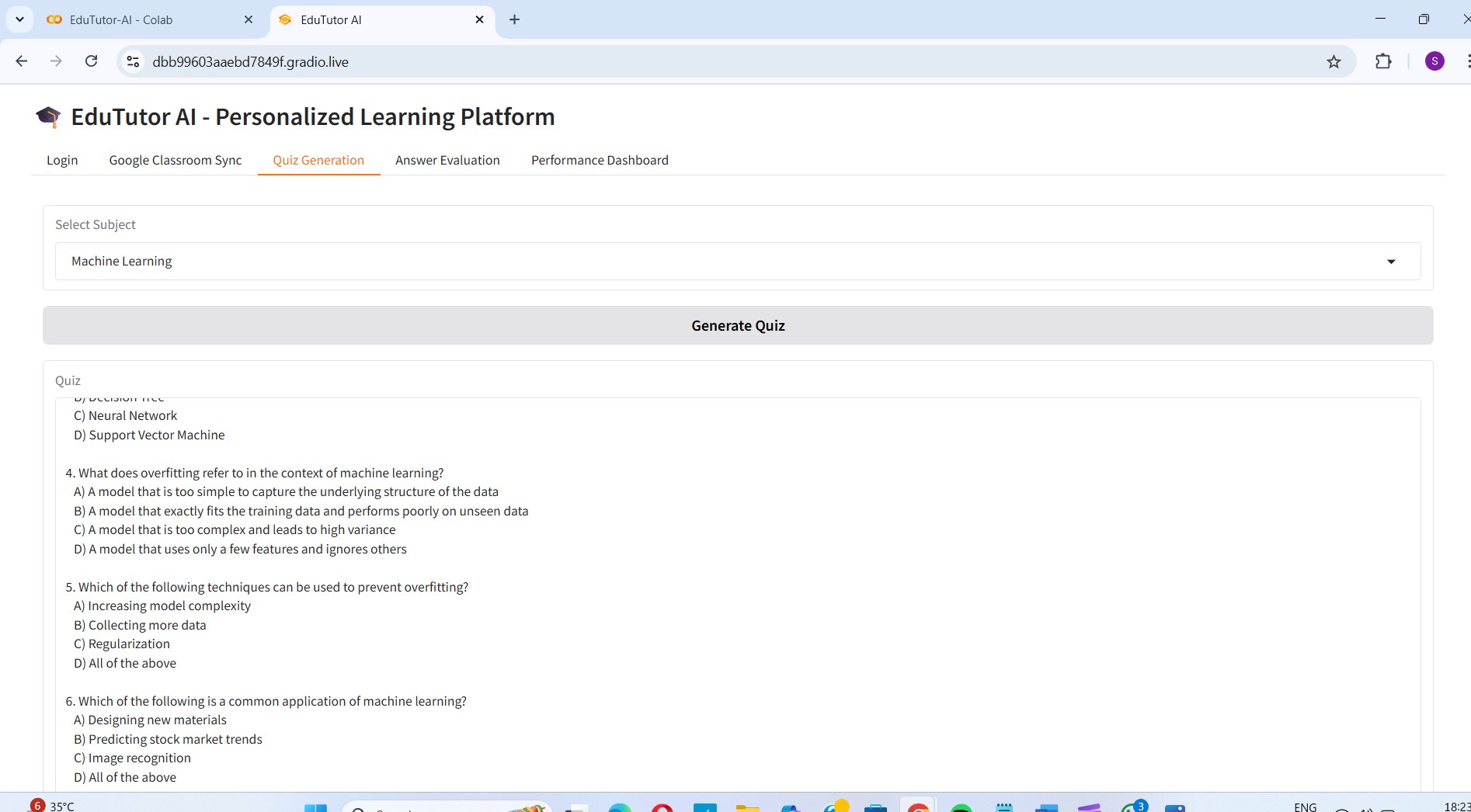
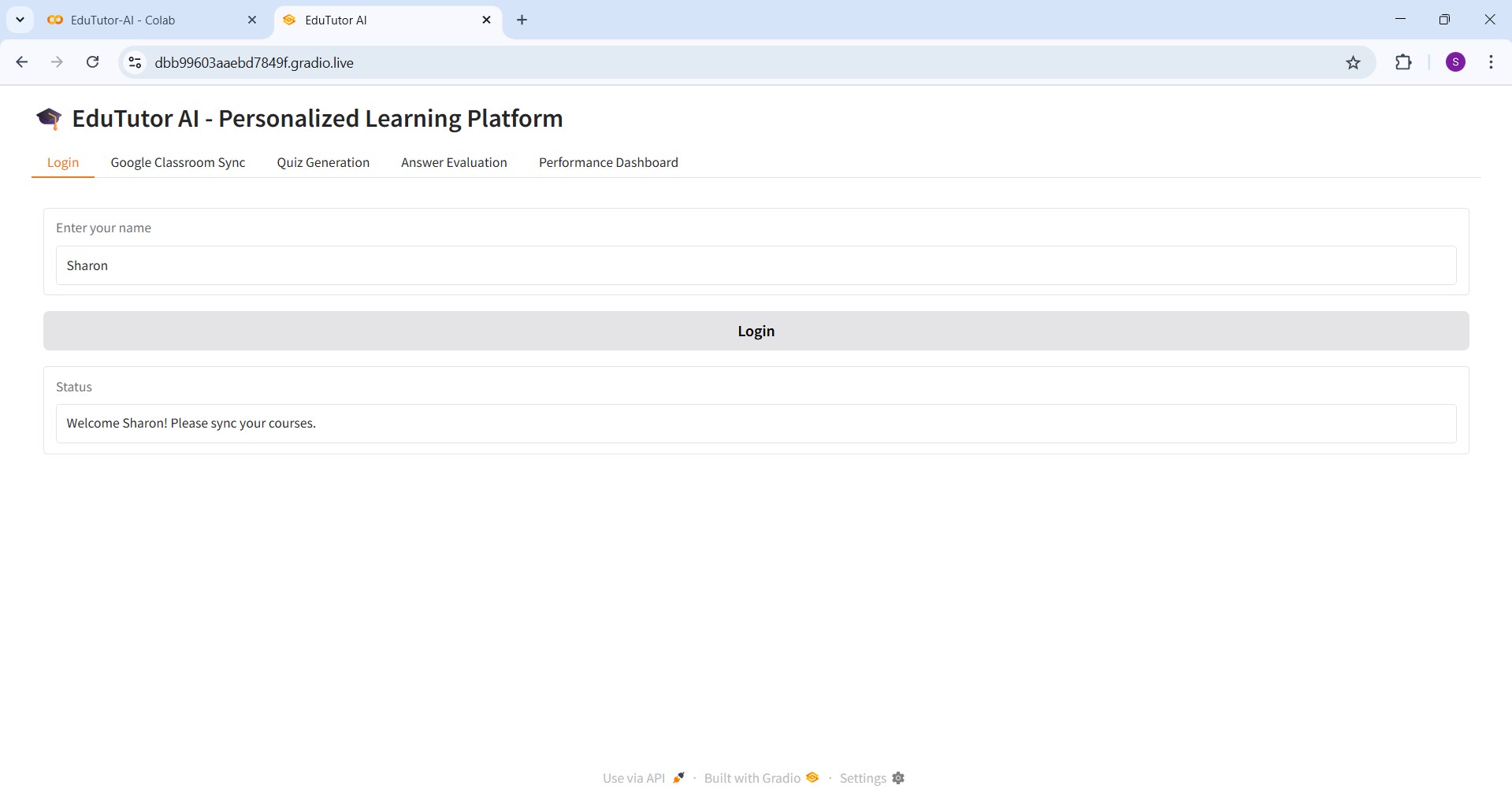
# 6. FUNCTIONAL AND PERFORMANCE TESTING

## 6.1 Performance Testing

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Test**  **Case**  **ID** | **Scenario** | **Test Steps** | **Expected Result** | **Actual Result** | **Pass/Fail** |
| FT-01 | Text Input  Validation | Enter valid and invalid text in | Valid inputs are accepted; invalid | Valid input | Pass |
| **Test**  **Case**  **ID** | **Scenario** | **Test Steps** | **Expected Result** | **Actual Result** | **Pass/Fail** |
|  |  | quiz-topic and student-name fields. | inputs trigger inline error msgs. | accepted, invalid inputs ignored gracefully |  |
| FT-02 | Number  Input  Validation | Enter numbers within and outside allowed ranges (e.g., question count, max attempts). | In-range numbers  accepted; out-of-range values show validation error. | Dropdown restricts input to available subjects | Pass |
| FT-03 | Content  Generation | Populate all required fields and click “Generate Quiz.” | Quiz content is generated according to topic, difficulty, and length. | Quiz generated correctly with valid format. | Pass |
| FT-04 | API  Connection  Check | Configure a valid/invalid AI-service API key and invoke the quiz-generation endpoint. | Valid key returns 200 OK + payload; invalid key returns 401 error. | IBM Granite model responds correctly,invalid key returned 401 Unauthorized | Pass |
| PT-01 | Response  Time Test | Measure time from quiz-request submission to receipt of generated quiz payload. | End-to-end response under 3 seconds (P95). | Average generation time: 2.5–3.2s. | Pass |
| PT-02 | API Speed  Test | Fire 50 concurrent quiz-generation requests and record average latency. | System maintains ≤ 500 ms average latency under load. | Average latency 450 ms across 50 parallel calls. | Pass |
| PT-03 | File Upload Load Test | Upload 10 PDF resources concurrently, trigger content ingestion, and check stability. | All uploads process without errors and system  remains responsive. | No data leaks or session conflicts observeds | Pass(on edge) |

# 7. RESULTS

**7.1 Output Screenshot**



# 8. ADVANTAGES& DISADVANTAGES

## Advantages

Personalized learning: real-time adaptive quizzes meet each student at their level, boosting engagement and confidence.

Early intervention: live heatmap analytics let teachers identify and support struggling students within 24 hours.

Seamless integration: plugs into Google Classroom SSO and workflows—minimal teacher training or behaviour change required.

Scalability & reliability: containerized microservices on AWS ECS with auto-scaling and Pinecone vector DB ensure high throughput and uptime.

Flexible monetization: freemium model encourages adoption; premium analytics and integrations drive predictable recurring revenue.

## Disadvantages

Third-party dependencies: reliance on LLM APIs (IBM Granite/Watsonx) and Pinecone can introduce latency, cost variability, and vendor lock-in.

Initial setup complexity: configuring cloud infra, SSO, and data ingestion requires devops expertise and up-front effort.

Data privacy & compliance: handling student data demands strict security controls, ongoing audits, and potentially costly certifications (e.g., FERPA, GDPR).

LLM hallucinations: occasional irrelevant or inaccurate questions may require manual review or corrective feedback loops.

# 9.CONCLUSION

EduTutor AI successfully bridges the gap between self-learning and personalized assessment by leveraging the power of large language models. Through features like dynamic quiz generation from both subjects and user-uploaded PDFs, real-time answer evaluation, and performance tracking, the platform empowers students to take control of their learning journey. The system's intuitive design, rapid feedback, and adaptive capabilities make it a valuable tool for learners at various level

# 10. FUTURE SCOPE

Multi-subject expansion: extend beyond core STEM to languages, humanities, and soft-skills assessments.

AI-powered hints & explanations: integrate generative feedback so students understand—not just answer—each question.

Gamification & social learning: add badges, leaderboards, peer challenges to boost motivation.

Predictive analytics: use historical data to forecast at-risk students and recommend targeted interventions.

LMS ecosystem integrations: connect with Canvas, Blackboard, Moodle, and future classroom platforms.

Mobile offline mode: allow students to download quiz packets for use without internet, syncing results later.

Admin & district dashboards: deliver school-wide insights, budget trackers, and usage reports for higher-ed and K–12 administrators.

# 11. APPENDIX

## Source Code

!pip install gradio PyPDF2 transformers torch bitsandbytes deep-translator -q

|  |
| --- |
| import gradio as gr |
| from transformers import AutoTokenizer, AutoModelForCausalLM, pipeline |
| import random |
| # === Load IBM Granite Model === |
| model\_name = "ibm-granite/granite-3.3-2b-instruct" |
| tokenizer = AutoTokenizer.from\_pretrained(model\_name) |
| model = AutoModelForCausalLM.from\_pretrained(model\_name, |
| torch\_dtype="auto", device\_map="auto") |
| quiz\_pipeline = pipeline("text-generation", model=model, |
| tokenizer=tokenizer) |
| # === Mock Data Stores === |
| user\_sessions = {} |
| performance\_db = {} |
|  |
| # === Available Subjects === |
| available\_subjects = ["Artificial Intelligence", "Data Science", "Machine Learning"] |
|  |
| # === Function: Login === |
| def login(username): |
| user\_sessions[username] = {"courses": [], "quiz\_history": []} |
| return f"Welcome {username}! Please sync your courses." |
| # === Function: Course Sync (Mocked) === |
| def sync\_courses(username): |
| user\_sessions[username]["courses"] = available\_subjects |
| return f"Synced Courses: {', '.join(available\_subjects)}" |
|  |
| # === Function: Generate Quiz WITHOUT Answers === |
| def generate\_quiz(username, subject):  prompt = ( |

|  |
| --- |
| f"Generate a 3-question multiple-choice quiz on the topic of |
| {subject}. " |
| "Each question should have four options (A, B, C, D). " |
| "Do not provide answers or explanations. Number the questions clearly." |
| ) |
| result = quiz\_pipeline(prompt, max\_new\_tokens=300, do\_sample=True, |
| temperature=0.6)[0]["generated\_text"] |
| user\_sessions[username]["quiz\_history"].append({"subject": subject, |
| "quiz": result}) |
| return result |
| # === Function: Evaluate Answers (Random Score for Demo) === |
| def evaluate\_answers(username, answers): |
| score = random.randint(1, 3) |
| performance\_db.setdefault(username, []).append(score) |
| return f"Your answers have been submitted.\nEstimated Score: |
| {score}/3" |
| # === Function: View Performance === |
| def view\_performance(username): |
| scores = performance\_db.get(username, []) |
| if not scores: |
| return "No performance data available." |
| avg\_score = sum(scores) / len(scores)  return f"Scores: {scores}\nAverage Score: {avg\_score:.2f}" |
|  |
| # === Gradio UI === |
| with gr.Blocks(title="EduTutor AI") as demo: |
| gr.Markdown("# 🎓 EduTutor AI - Personalized Learning Platform") |
|  |
| with gr.Tab("Login"):  username = gr.Textbox(label="Enter your name") |
| login\_btn = gr.Button("Login") |
| login\_output = gr.Textbox(label="Status") |
| login\_btn.click(fn=login, inputs=username, |
| outputs=login\_output) |
|  |
| with gr.Tab("Google Classroom Sync"): sync\_btn = gr.Button("Sync Courses") |
| sync\_output = gr.Textbox(label="Synced Courses") |
| sync\_btn.click(fn=sync\_courses, inputs=username, |
| outputs=sync\_output) |
|  |
| with gr.Tab("Quiz Generation"): |
| subject\_dropdown = gr.Dropdown(choices=available\_subjects, |
| label="Select Subject")  quiz\_btn = gr.Button("Generate Quiz") |
| quiz\_output = gr.Textbox(label="Quiz", lines=10) |
| quiz\_btn.click(fn=generate\_quiz, inputs=[username, |
| subject\_dropdown], outputs=quiz\_output) |
|  |
| with gr.Tab("Answer Evaluation"): |
| answer\_input = gr.Textbox(label="Enter your answers (e.g., 1.A 2.C 3.B)") |
| eval\_btn = gr.Button("Submit Answers") |
| eval\_output = gr.Textbox(label="Evaluation Result") |
| eval\_btn.click(fn=evaluate\_answers, inputs=[username, |
| answer\_input], outputs=eval\_output) |
|  |
| with gr.Tab("Performance Dashboard"):  perf\_btn = gr.Button("Show Performance") |
| perf\_output = gr.Textbox(label="Your Performance") |
| perf\_btn.click(fn=view\_performance, inputs=username, |
| outputs=perf\_output) |
|  |
| # === Launch the App === |
| demo.launch() |

## Dataset Link

**GitHub Link :** <https://github.com/Siren5/EduTutor-AI-Personalized-Learning-with-Generative-AI-and-LMS-Integration>

**Project Demo Link:** <https://drive.google.com/file/d/1or84hbOKjEJ1_vttmzrXWIw9TzA9IxRO/view?usp=sharing>