

Personalized Learning with Generative AI and LMS Integration

EduTutor AI - Project Report

INTRODUCTION

1.1 Overview

EduTutor AI is a lightweight, AI-powered learning assistant developed to enhance student education through personalized content generation. Designed with simplicity and accessibility in mind, the system uses IBM's Watsonx Granite large language models (LLMs) to provide interactive learning experiences directly through a FastAPI backend.

The platform enables students to:

- *Understand academic topics through natural language explanations.*
- *Generate custom quizzes from text input or uploaded PDF materials.*
- *Practice English or Hindi grammar and vocabulary interactively.*

Unlike traditional tutoring systems or static e-learning platforms, EduTutor AI uses prompt-based interactions with a generative model to create on-demand educational content tailored to each user's input. The application does not depend on third-party libraries like Hugging Face; instead, it connects directly to IBM Watsonx APIs, offering a secure and scalable deployment route.

With modules for concept understanding, automated assessment, and bilingual language support, EduTutor AI brings together modern AI capabilities to deliver a more engaging and personalized student learning experience.

1.2 Purpose

The purpose of EduTutor AI is to bridge the gap between traditional teaching methods and modern AI-based educational tools. The key goals of the system are:

- *To assist students in learning complex topics by generating simple and accurate explanations.*
- *To enable automated quiz creation from both typed topics and uploaded study material in PDF form, saving time and ensuring consistent practice.*
- *To provide interactive English and Hindi language learning support, including grammar correction and vocabulary improvement.*

By integrating IBM's enterprise-grade LLMs into a focused educational assistant, EduTutor AI aims to make quality tutoring and self-paced assessment tools accessible to all learners—especially in resource-constrained or bilingual environments.

2. Ideation Phase

2.1 Define the Problem Statements

| | |
|---------------|---------------------------------------------------------------------------|
| Date | 26 June 2025 |
| Team ID | LTVIP2025TMID31818 |
| Project Name | EduTutor AI: Personalized Learning with Generative AI and LMS Integration |
| Maximum Marks | 2 Marks |

Customer Problem Statement EduTutor AI:

In today's fast-paced education system, students often face the challenge of understanding complex subjects without proper guidance. Many rely on online resources, but these are often generic, not personalized to the learner's pace or style. Meanwhile, educators are overwhelmed with the repetitive task of creating and evaluating quizzes manually for every student.

EduTutor AI aims to solve this by offering a personalized AI tutor that provides:

- Concept explanations simplified for different age levels
- Grammar and language learning support in English and Hindi
- Automated quiz and test generation from PDFs and topics

This solution empowers students to learn at their own pace with clarity, and enables educators to scale personalized learning experiences without increased workload.

| | |
|---------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| I am | A high school or college student preparing for competitive exams or internal assessments. I am tech-savvy, self-driven, but often lack access to quality personal tutors or learning support. |
| I'm trying to | Understand difficult subjects like Generative AI, Math, or English grammar on my own and prepare effectively for upcoming quizzes and exams. |
| But | I struggle to find clear explanations, quality practice material, or feedback that helps me know what I'm doing right or wrong. Generic online content doesn't adapt to my needs. |
| Because | Traditional learning systems are one-size-fits-all. Teachers have limited time and cannot offer one-on-one support. Also, many apps don't allow uploading my material to get personalized help. |
| Which makes me feel | Anxious and demotivated, especially when I get stuck or perform poorly in tests. It feels like I'm not learning the right way, and I fear being left behind. |

Example:



| Problem Statement (PS) | I am (Customer) | I'm trying to | But | Because | Which makes me feel |
|------------------------|-------------------------------------------|----------------------------------------------------|------------------------------------------------------------------|--------------------------------------------------------------------|-----------------------------------------------|
| PS-1 | a high school student preparing for exams | understand difficult topics and revise efficiently | I can't grasp the concepts fully and get no feedback | online resources are generic and teachers are not always available | frustrated, confused, and unsupported |
| PS-2 | a teacher managing multiple students | generate personalized | I don't have time to prepare and evaluate tests for each student | manual creation and review takes too much effort | overwhelmed and limited in my teaching impact |

2 Ideation Phase

2.2 Brainstorm & Idea Prioritization Template


| | |
|---------------|---------------------------------------------------------------------------|
| Date | 26 June 2025 |
| Team ID | LTVIP2025TMID31818 |
| Project Name | EduTutor AI: Personalized Learning with Generative AI and LMS Integration |
| Maximum Marks | 4 Marks |

Brainstorm & Idea Prioritization EduTutor AI:

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.

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

Template



EduTutor AI: Personalized Learning with Generative AI and LMS Integration

EduTutor AI is a generative AI-powered educational platform that offers personalized concept explanations, grammar assistance in Hindi and English, and automatic quiz/test generation from user-provided topics or PDFs. It features login-based classroom access and session tracking per user.

🕒 10 minutes to prepare

🕒 1 hour to collaborate

👤 2-8 people recommended

●

Before you collaborate

A little bit of preparation goes a long way with this session. Here's what you need to do to get going.

🕒 10 minutes

📌

Team gathering

Our team includes 4 members, each with a background in computer science and an interest in educational technology. We decided to explore ways to enhance the learning process using artificial intelligence. Each member came prepared with real-world issues students face during independent study.

🎯

Set the goal

The goal of this session is to identify a meaningful problem in the education sector and generate ideas on how AI can help. We wanted to focus on a solution that could assist students in understanding complex concepts, practicing language, and generating self-assessments automatically.

📖

Learn how to use the facilitation tools

We reviewed basic brainstorming techniques like sticky-note clustering and digital whiteboarding using tools like Miro and Google Jamboard.

Open article →

1

Define your problem statement

What problem are you trying to solve? Frame your problem as a How Might We statement. This will be the focus of your brainstorm.

🕒 5 minutes

PROBLEM

How might we help students learn independently through personalized AI-powered concept explanations, grammar assistance, and automated assessments?

🧠

Key rules of brainstorming

To run a smooth and productive session

🕒 Stay in topic.

🕒 Encourage wild ideas.

🕒 Defer judgment.

🕒 Listen to others.

🕒 Go for volume.

🕒 If possible, be visual.

Step-2: Brainstorm, Idea Listing and Grouping

2 Brainstorm

Write down any ideas that come to mind that address your problem statement.

10 minutes

Person 1

MCQ Quiz App
 Make a basic quiz app that lets students answer multiple-choice questions by topic.

Flashcard Generator
 Build a tool where students can enter key terms and definitions to generate flashcards.

Person 2

Subject Suggestion Tool
 Create a form that asks about a student's interests and suggests subjects or careers they might enjoy.

Progress Tracker
 Design a chart or simple app that tracks completed lessons and quiz scores.

3 Group Ideas

Take turns sharing your ideas while clustering similar or related notes as you go. Once all sticky notes have been grouped, give each cluster a sentence-like label. If a cluster is bigger than six sticky notes, try and see if you can break it up into smaller sub-groups.

20 minutes

Personalised Learning

Custom sequences of learning experiences that suit each student's progress and interests.

Topic Explanation

- You get to learn at your own pace.
- You focus more on what you need help with.
- You build confidence because you're learning in a way that fits you.

Practice MCQs: Personalized Learning

MCQs are a quick and effective way to check if students understand key concepts.

Step-3: Idea Prioritization

4 Prioritize

Your team should all be on the same page about what's important moving forward. Place your ideas on this grid to determine which ideas are important and which are testable.

30 minutes

Importance
 If each of these ideas would get done without any difficulty or with which would make the most positive impact?

Feasibility
 Separate other importance, which have more realistic than others? (Cost, time, effort, technology, etc.)

Tip
 Participants can use their numbers to point at values they think should go on the grid. The facilitator can confirm this spot by calling out their number making sure they are the highest.

2. Ideation Phase

| | |
|---------------|---------------------------------------------------------------------------|
| Date | 26 June 2025 |
| Team ID | LTVIP2025TMID31818 |
| Project Name | EduTutor AI: Personalized Learning with Generative AI and LMS Integration |
| Maximum Marks | 4 Marks |

2.3 Empathize & Discover

Empathy Map Canvas EduTutor AI:

SAYS

What the student/user verbally expresses:

- “I don’t understand this topic, even after watching videos.”
- “I need a simple explanation like a teacher would give.”
- “I wish I could practice more questions like in school.”

THINKS

What the user is thinking but might not say out loud:

- “Am I studying the right way?”
- “What if I fail the test because I missed something important?”
- “Other students probably have better help than I do.”

DOES

What actions or behaviors the student takes:

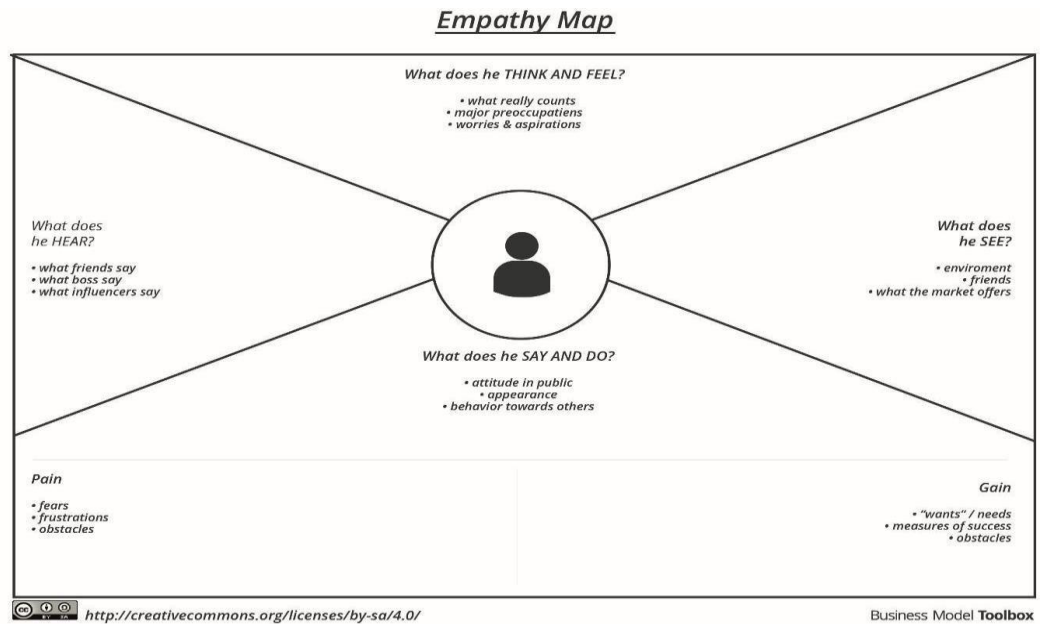
- Searches YouTube or websites for topics
- Uploads books or PDFs to extract questions
- Tries apps for grammar and test prep
- Attempts mock tests but gets no explanations

FEELS

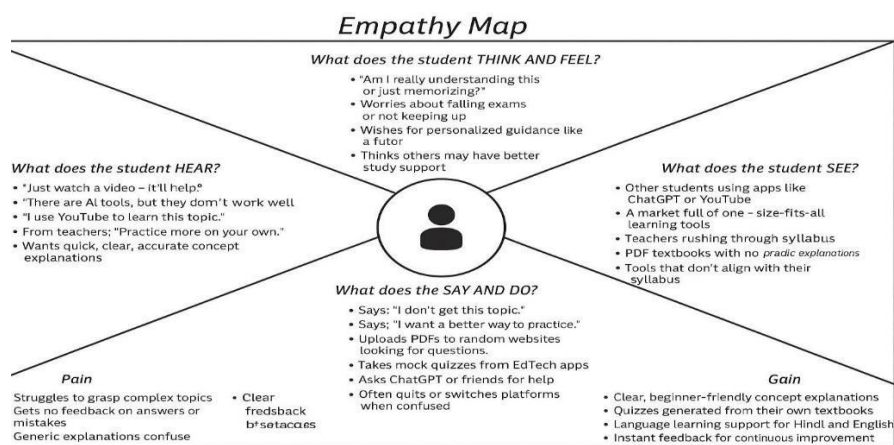
What emotions the user experiences:

- Anxious and overwhelmed before exams
- Relieved when they finally understand a topic
- Disappointed by one-size-fits-all apps
- Excited when learning becomes personalized

Example:



Example: EduTutor AI



3. Requirement Analysis

3.1 Solution Requirements (Functional & Non-functional)

| | |
|---------------|---------------------------------------------------------------------------|
| Date | 27 June 2025 |
| Team ID | LTVIP2025TMID31818 |
| Project Name | EduTutor AI: Personalized Learning with Generative AI and LMS Integration |
| Maximum Marks | 4 Marks |

Functional Requirements

The following are the core functional requirements for the EduTutor AI application, based strictly on the current feature set outlined in the README.

| FR No. | Functional Requirement (Epic) | Sub Requirement (Story / Sub-Task) |
|--------|-------------------------------|-----------------------------------------------------------------------------------------------------|
| FR-1 | User Input Handling | Accept topic input from user through a text field |
| FR-2 | Explanation Generation | Send topic to IBM Watsonx model Display explanation using Streamlit |
| FR-3 | MCQ Generation | Generate multiple-choice questions (MCQs) from the given topic Display MCQs with correct answers |
| FR-4 | Credential Configuration | Allow user to enter IBM Watsonx API key and project ID in app.py |

Non-Functional Requirements

The following are the non-functional requirements for the current version of EduTutor AI.

| NFR No. | Non-Functional Requirement | Description |
|---------|----------------------------|----------------------------------------------------------------------------------------|
| NFR-1 | Usability | The Streamlit interface should be simple and intuitive for quick interaction. |
| NFR-2 | Security | IBM Watsonx credentials should be kept secure and not hard-coded for public use. |
| NFR-3 | Reliability | System should consistently produce valid explanations and MCQs for well-formed inputs. |
| NFR-4 | Performance | Output should be generated within 2–4 seconds under normal load. |
| NFR-5 | Deployment Simplicity | App should run with minimal setup using 'streamlit run app.py'. |

3. Requirement Analysis

3.2 Data Flow Diagram & User Stories

| | |
|---------------|---------------------------------------------------------------------------|
| Date | 27 June 2025 |
| Team ID | LTVIP2025TMID31818 |
| Project Name | EduTutor AI: Personalized Learning with Generative AI and LMS Integration |
| Maximum Marks | 4 Marks |

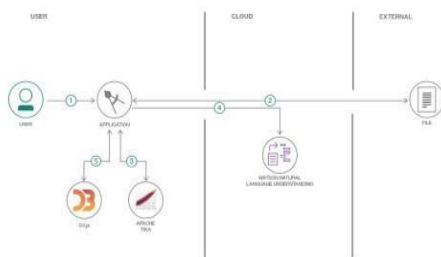
Data Flow Diagrams:

A Data Flow Diagram (DFD) is a traditional visual representation of the information flows within a system. A neat and clear DFD can depict the right amount of the system requirement graphically. It shows how data enters and leaves the system, what changes the information, and where data is stored.

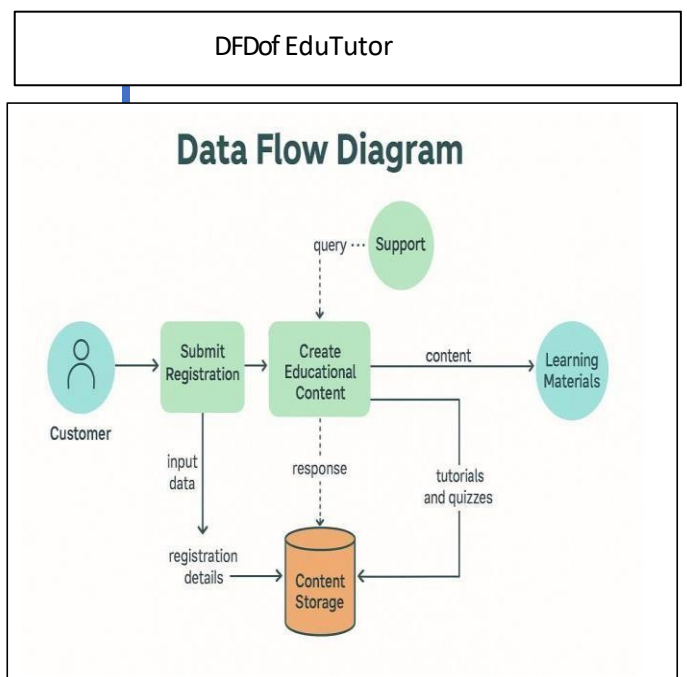
A **Data Flow Diagram (DFD)** is a graphical tool used to model the flow of data within a system. It shows how data is **input, processed, stored, and output** in a clear and systematic way. For EduTutor AI, the DFD outlines how users interact with the system through the interface and how their data is processed by the AI model and other components.

Example: [\(Simplified\)](#)

Flow



1. User configures credentials for the Watson Natural Language Understanding service and starts the app.
2. User selects data file to process and load.
3. Apache Tika extracts text from the data file.
4. Extracted text is passed to Watson NLU for enrichment.
5. Enriched data is visualized in the UI using the D3.js library.



User Stories

Use the below template to list all the user stories for the product.

| User Type | Functional Requirement (Epic) | User Story Number | User Story / Task | Acceptance Criteria | Priority | Release |
|------------------------|-------------------------------|-------------------|------------------------------------------------------------------------------------------|-------------------------------------------------|----------|----------|
| Customer (Mobile user) | Streamlit UI | USN-1 | As a user, I can register for the application by entering my name and password | I can access my account/dashboard | High | Sprint-1 |
| | | USN-2 | As a user, I can log into the application using name & password. | Login should redirect me to the dashboard | High | Sprint-1 |
| | Concept Explanation | USN-3 | As a user, I can enter a concept and get a simplified explanation. | The concept is clearly explained in an easy way | High | Sprint-1 |
| | MCQ Generation | USN-4 | As a user, I can enter a topic it generates questions for practice. | A quiz with relevant questions is generated | High | Sprint-2 |
| | | USN-5 | As a user, I can choose a language (English/Hindi) to learn grammar and parts of speech. | Language learning content is displayed | Medium | Sprint-2 |

3. Requirement Analysis

3.3 Technology Stack (Architecture & Stack)

| | |
|---------------|---------------------------------------------------------------------------|
| Date | 27 June 2025 |
| Team ID | LTVIP2025TMID31818 |
| Project Name | EduTutor AI: Personalized Learning with Generative AI and LMS Integration |
| Maximum Marks | 4 Marks |

Architecture Layers & Flow

| Layer | Technology | Description |
|---------------------------------|------------------------------------|------------------------------------------------------------------------------------|
| User Interface Layer (Local) | Streamlit | Collects user input such as concept/topic; displays explanations and MCQs |
| Application Logic Layer (Local) | Python | Handles prompt generation, session management, and quiz formatting logic |
| AI Model Layer (Cloud) | IBM Watsonx via ibm-watsonx-ai SDK | Uses Granite model (granite-3-8b-instruct) to generate topic explanations and MCQs |
| Data Storage Layer (Local) | In-memory Dictionary (Python) | Temporarily stores session data and results during runtime |
| External Interfaces | IBM Watsonx API Credentials | Secure communication with IBM foundation model APIs |

Table 1: Components & Technologies

| S.No | Component | Description | Technology |
|------|------------------------|------------------------------------------------------------|------------------------------------------------------|
| 1 | User Interface | Interactive UI for input and output display | Streamlit |
| 2 | Application Logic | Prompt construction, logic handling, and output formatting | Python |
| 3 | Session Handling | Tracks user inputs/output per session | Python in-memory structures |
| 4 | External API | Accesses IBM Watsonx foundation model | IBM Watsonx AI SDK (ibm-watsonx-ai) |
| 5 | Machine Learning Model | Performs natural language generation | granite-3-8b-instruct |
| 6 | Infrastructure | Execution environment for app | Local Runtime / Cloud-hosted via Hugging Face Spaces |

Table 2: Application Characteristics

| S.no | Characteristic | Description | Technology |
|------|--------------------------|------------------------------------------------------------------|------------------------------|
| 1 | Open-Source Frameworks | Built using open tools | Streamlit, Python, IBM SDK |
| 2 | Security Implementations | Credentials stored securely (locally or in Hugging Face secrets) | API Key + Project ID |
| 3 | Scalability | Modular layers (UI → Logic → Model); extendable | Streamlit + API-based logic |
| 4 | Availability | Run locally or deploy on Hugging Face Spaces or Streamlit Cloud | Python + Streamlit Cloud |
| 5 | Performance | Fast text generation via optimized cloud APIs | IBM Watsonx optimized models |

4. Project Design

4.1 Problem – Solution Fit Template

| | |
|---------------|---------------------------------------------------------------------------|
| Date | 27 June 2025 |
| Team ID | LTVIP2025TMID31818 |
| Project Name | EduTutor AI: Personalized Learning with Generative AI and LMS Integration |
| Maximum Marks | 2 Marks |

Problem – Solution Fit EduTutor AI:

Problem

Many students struggle to understand complex academic concepts on their own and lack the ability to assess their knowledge effectively. Despite the abundance of content online, these materials are often generic, not tailored to individual learning styles, languages, or needs. Simultaneously, educators lack tools to personalize instruction or quickly generate and evaluate assessments for each student. This gap leads to confusion, low confidence, poor academic performance, and overburdened educators.

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 automatically generate quizzes and tests from topics or uploaded PDFs. It includes:

- Concept explanations tuned for age and subject level
- MCQ'S generators

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

| | | | |
|-----------------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------|
| Define CS / IK into CC Focus on J in up into Be RC | 1. CUSTOMER SEGMENT(S) High school & college students, self-learners, teachers CS | 6. CUSTOMER CONSTRAINTS Limited time, no access to tutors, no feedback, poor app quality CC | 5. AVAILABLE SOLUTIONS YouTube videos, ChatGPT textbook quizzes—but not personalized or syllabus specific CC |
| | 2. JOBS-TO-BE-DONE / PROBLEMS Want a personalized learning tool to understand topics and get assessed easily JP | 9. PROBLEM ROOT CAUSE Jump between platforms, ask friends, use ChatGPT, watch BE | 7. BEHAVIOUR Jump between platforms, ask friends, use ChatGPT watch videos AS |
| | 3. TRIGGERS Struggle in test preparation lack of teacher feedback, poor results TR | 4. EMOTIONS: BEFORE / AFTER Generic tools don't adapt to individual learning styles or content needs EM | 8. CHANNELS OF BEHAVIOUR Online: Google, YouTube, EdTech forums Offline: Peer study groups, handwritten notes CH |
| | 4. EMOTIONS: BEFORE / AFTER Before: Confused, anxious about exams After: Confident, motivated to learn EM | 10. YOUR SOLUTION EduTutor AI; Personalized AI tutor for concept explanation, grammar, and quiz generation EM | 10. YOUR SOLUTION EduTutor AI: Personalized AI tutor for concept explanation, grammar, and quiz generation (PDF/topic-based) SL |

4. Project Design Phase

4.2 Proposed Solution Template

| | |
|---------------|---------------------------------------------------------------------------|
| Date | 27 June 2025 |
| Team ID | LTVIP2025TMID31818 |
| Project Name | EduTutor AI: Personalized Learning with Generative AI and LMS Integration |
| Maximum Marks | 2 Marks |

Proposed Solution EduTutor AI:

Project team shall fill the following information in the proposed solution template.

| S.No. | Parameter | Description |
|-------|------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 1. | Problem Statement (Problem to be solved) | Students face difficulties in understanding complex topics independently and lack access to personalized learning . Teachers also struggle with time-consuming quiz generation and individual evaluation. |
| 2. | Idea / Solution description | EduTutor AI is a generative AI-powered educational platform that offers personalized concept explanations, grammar assistance in Hindi and English, and MCQ'S generation from user-provided topics . It features login-based classroom access and session tracking per user. |
| 3. | Novelty / Uniqueness | Unlike generic platforms, EduTutor AI can offer desirable explanation and generate personalized quizzes. It combines LMS features with generative AI for both concept delivery and adaptive assessment in multiple languages. |
| 4. | Social Impact / Customer Satisfaction | It enhances accessibility to quality education for students without tutors, supports regional languages, and reduces the workload on teachers. Students gain confidence and clarity, while teachers benefit from automation. |
| 5. | Business Model (Revenue Model) | Freemium model: free access to basic features (concepts, quizzes), with premium plans for advanced features like PDF uploads, progress analytics, custom LMS integration for schools/colleges. Additional revenue through institutional licensing. |
| 6. | Scalability of the Solution | The solution is cloud-based and can support millions of users with the same backend. It can scale across languages, educational levels, and institutions globally, and easily integrate with other EdTech platforms or LMS providers. |

4. Project Design Phase

4.3 Solution Architecture

| | |
|---------------|---------------------------------------------------------------------------|
| Date | 27 June 2025 |
| Team ID | LTVIP2025TMID31818 |
| Project Name | EduTutor AI: Personalized Learning with Generative AI and LMS Integration |
| Maximum Marks | 4 Marks |

Solution Architecture

Solution architecture outlines how the system components are organized to achieve the project's functional goals. It bridges the gap between business requirements and technical implementation in a structured and scalable way.

Purpose

EduTutor AI is designed to offer a personalized learning experience using generative AI capabilities provided by IBM Watsonx. The platform provides users with the ability to input a topic and receive either a clear explanation or a set of quiz questions. The entire application is built using Python and Streamlit, without any complex backend frameworks or third-party language model integrations such as Hugging Face.

Architecture Overview

The system architecture is composed of three primary layers:

1. User Interface Layer (Streamlit)

This layer is responsible for interacting with the user through a web-based UI built entirely in Streamlit. It provides:

- A text input field for entering concepts or topics
- A mode selector to switch between "Explain" and "Quiz"
- A display area to show either the explanation or the generated quiz
- A simple, responsive interface that runs locally or in the cloud

2. Logic & Processing Layer (Python)

This layer contains the core application logic, which includes:

- Handling user input and selection of modes
- Formatting prompts for explanation or quiz generation
- Managing user session data using Python dictionaries (if applicable)
- Communicating with the IBM Watsonx API
- Optionally handling file input and parsing via PyPDF2 (for PDF-to-quiz functionality)

3. AI Model Layer (IBM Watsonx Granite API)

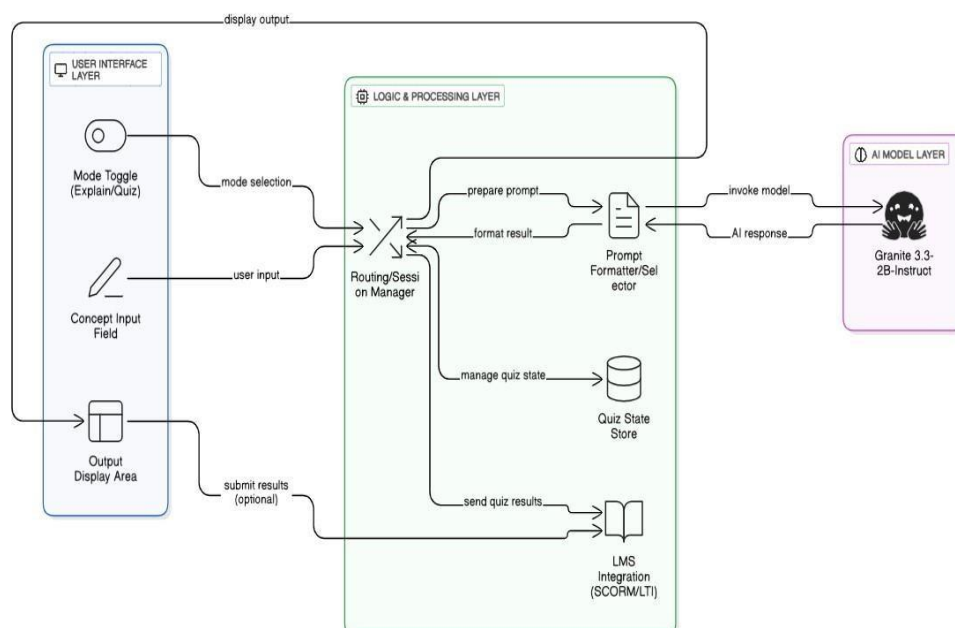
This layer integrates directly with IBM Watsonx, utilizing the Granite 3.3-2B-Instruct model via secure API calls. It performs:

- Concept explanation generation based on natural language input
- Topic-based quiz generation (5 MCQs with options and correct answers)
- Language-specific responses when needed (e.g., English/Hindi grammar lessons)

Data Flow Summary

1. The user interacts with the Streamlit interface to enter a concept or topic.
2. The Python backend detects the selected mode (Explain or Quiz) and prepares the appropriate prompt.
3. The prompt is sent to the IBM Watsonx API using the provided credentials.
4. The AI model processes the input and returns a response.
5. The response is formatted and displayed to the user in the Streamlit interface.

Example - Solution Architecture Diagram:



5. Project Planning Phase

5.1 Project Planning Template (ProdBacklog, Sprint Planning, Stories, Story points)

| | |
|---------------|---------------------------------------------------------------------------|
| Date | 27 June 2025 |
| Team ID | LTVIP2025TMID31818 |
| Project Name | EduTutor AI: Personalized Learning with Generative AI and LMS Integration |
| Maximum Marks | 5 Marks |

Product Backlog, Sprint Schedule, and Estimation (4 Marks)

Use the below template to create product backlog and sprint schedule

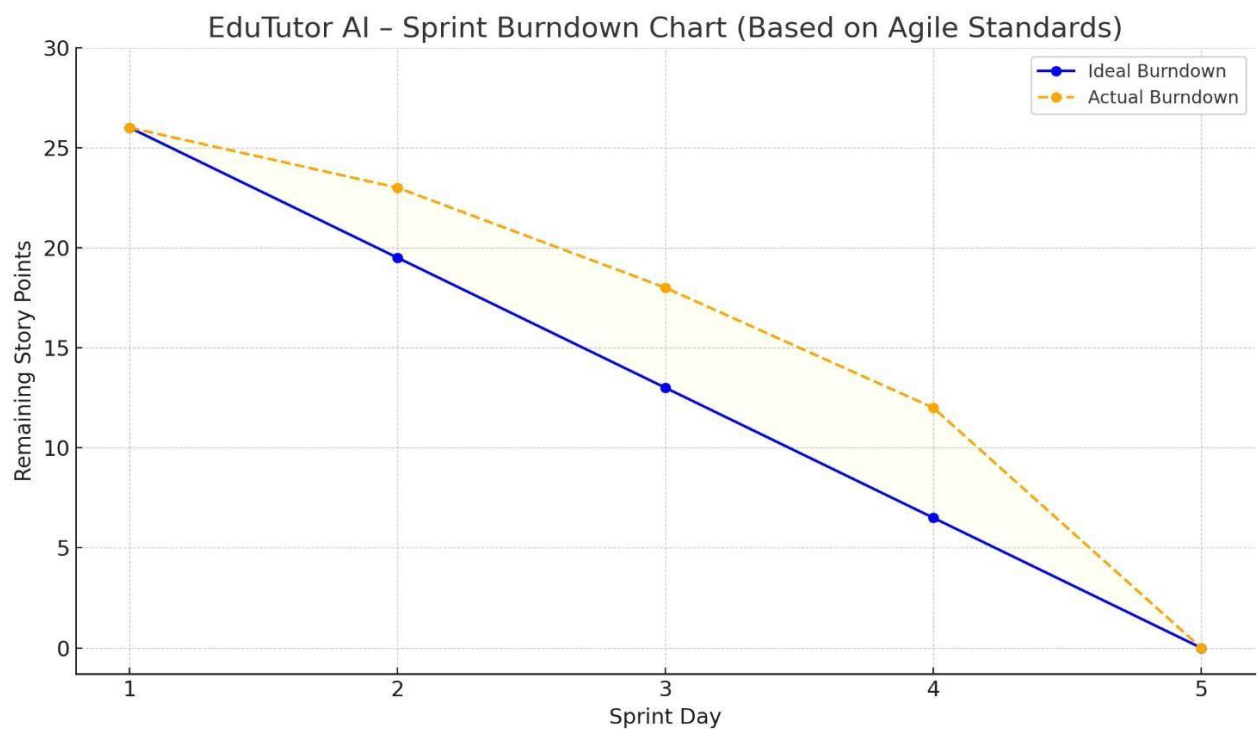
| Sprint | Functional Requirement (Epic) | User Story Number | User Story / Task | Story Points | Priority | Team Members |
|----------|-------------------------------|-------------------|------------------------------------------------------------------------|--------------|----------|----------------|
| Sprint-1 | Streamlit UI | USN-1 | As a user, I can register using a username and password. | 2 | High | varsha ,nazima |
| Sprint-1 | Output Formatting | USN-2 | As a user, I can log in to my account with name and password. | 2 | High | varsha |
| Sprint-1 | Mode Switching Logic | USN-3 | As a user, I can track my sessions after login. | 2 | Medium | nazima |
| Sprint-2 | MCQ Generator Prompting | USN-4 | As a user, I can enter a concept and get AI generated explanation. | 3 | High | varsha |
| Sprint-2 | Answer Formatting | USN-5 | As a user, I can choose Hindi or English and learn grammar basics. | 3 | High | nazima |
| Sprint-3 | Watsonx Integration | USN-6 | As a user, I can upload a PDF and receive a quiz based on the content. | 5 | High | varsha |
| Sprint-3 | Streamlit Deployment | USN-7 | As a user, I can enter a topic and receive a custom quiz. | 3 | Medium | nazima |
| Sprint-3 | Error Handling | USN-8 | As a developer, I can create a multi-tab UI using Gradio. | 3 | High | varsha |

Imagine we have a 5-day sprint duration, and the velocity of the team is 13 (points per sprint). Let's calculate the team's average velocity (AV) per iteration unit (story points per day)

$$AV = \text{sprint duration} / \text{velocity} = 13 / 5 = 2.6$$

Burndown Chart:

A burn down chart is a graphical representation of work left to do versus time. It is often used in agile software development methodologies such as Scrum. However, burn down charts can be applied to any project containing measurable progress over time.



<https://www.visual-paradigm.com/scrum/scrum-burndown-chart/>

<https://www.atlassian.com/agile/tutorials/burndown-charts>

6. RESULTS

The screenshot displays the 'EduTutor Student Panel' interface. On the left, a sidebar contains links for 'app', 'google login', and 'students dashboard'. Below these is a 'Login as:' dropdown menu set to 'Student'. The main content area features a header with 'EduTutor Student Panel' and navigation links for 'Login', 'Register' (highlighted in red), 'Dashboard', and 'Quiz History'. The 'Student Registration' form includes input fields for 'Username', 'Email', and 'Password' (with a toggle for visibility), and a 'Register' button.

app
google login
students dashboard

Login as:
Student

EduTutor Student Panel

Login Register Dashboard Quiz History

Student Registration

Username

Email

Password

Register

The screenshot displays the 'EduTutor Student Panel' interface. On the left, a sidebar contains links for 'app', 'google login', and 'students dashboard'. Below these is a 'Login as:' dropdown menu set to 'Student'. The main content area features a header with 'EduTutor Student Panel' and navigation links for 'Login' (highlighted in red), 'Register', 'Dashboard', and 'Quiz History'. The 'Student Login' form includes input fields for 'Email' and 'Password' (with a toggle for visibility), and a 'Login' button.

app
google login
students dashboard

Login as:
Student

EduTutor Student Panel

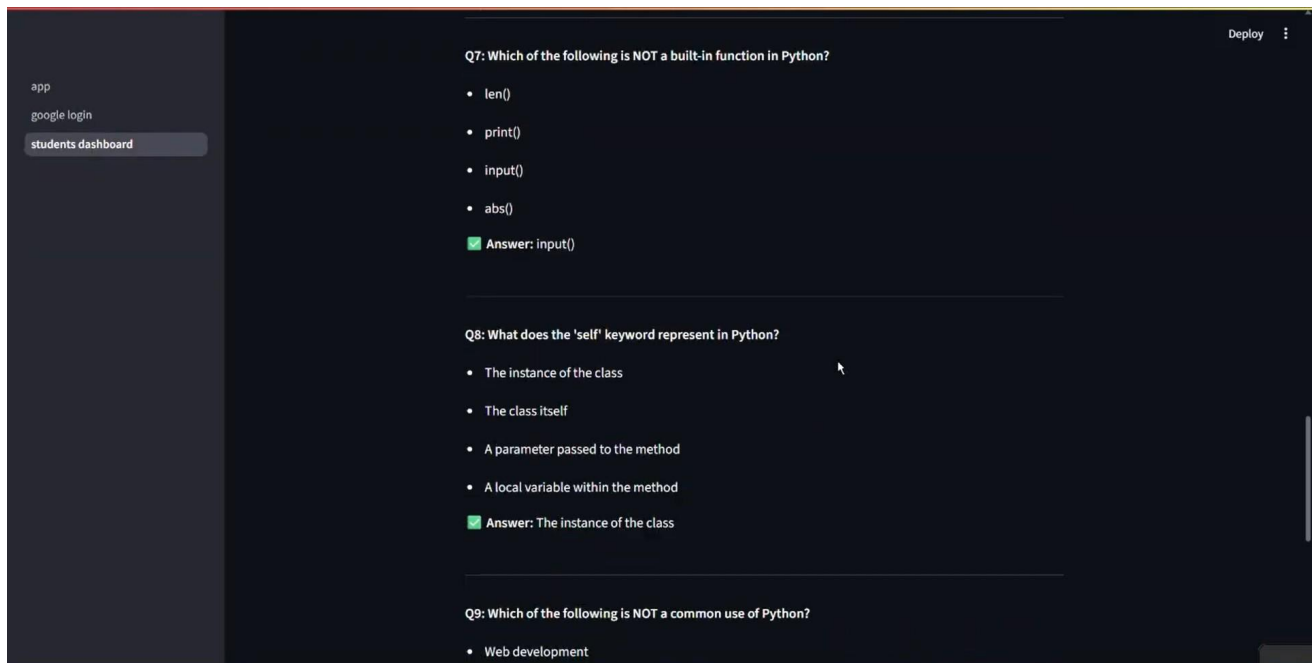
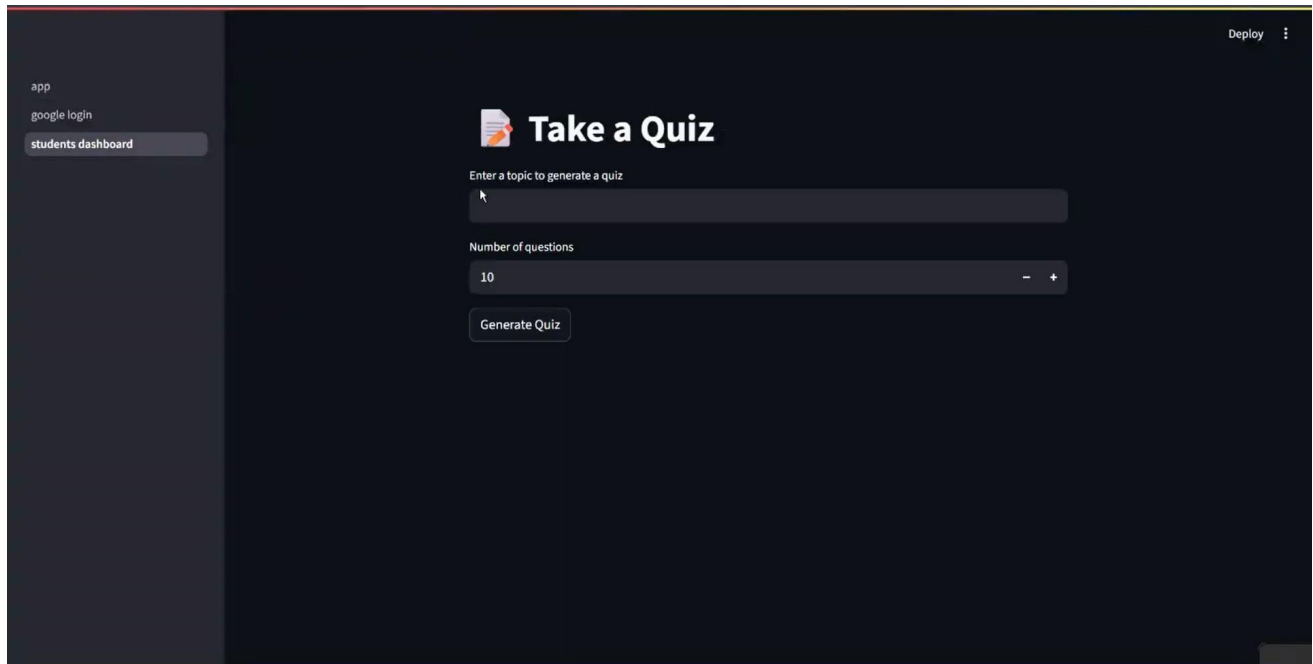
Login Register Dashboard Quiz History

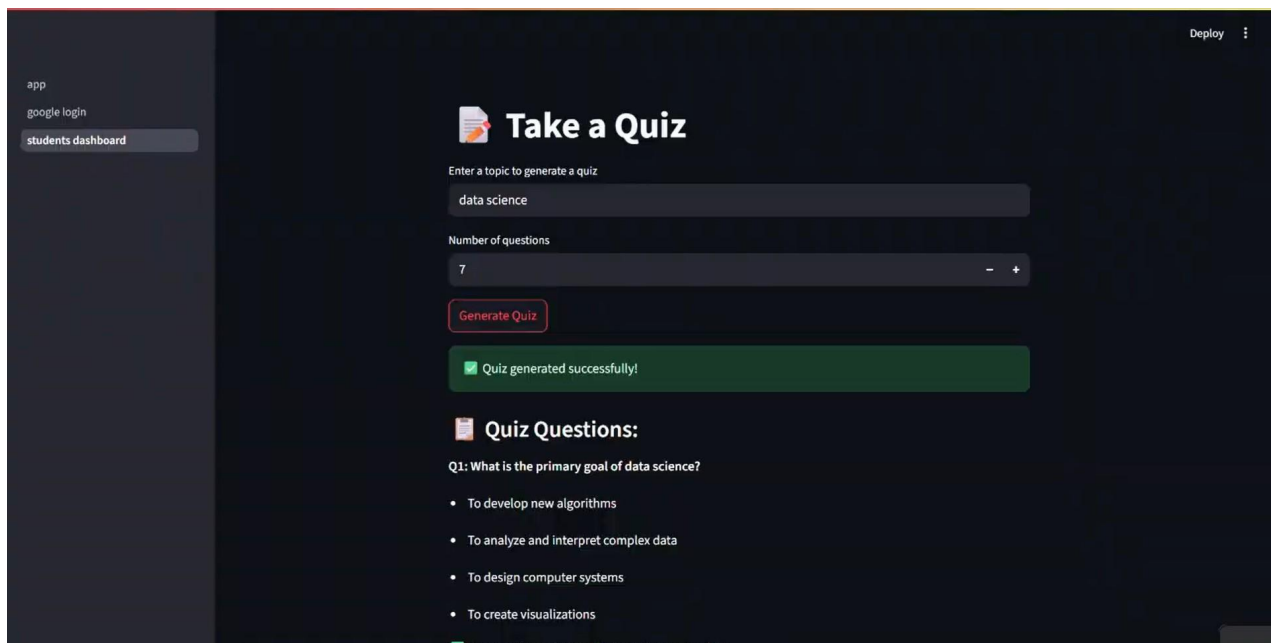
Student Login

Email

Password

Login





| EduTutor AI 0.1.0 OAS 3.1 | | |
|-----------------------------------------------------|----------------|-----------------|
| openapi.json | | |
| default ^ | | |
| GET | / | Root v |
| Quiz ^ | | |
| POST | /quiz/generate | Generate Quiz v |
| Google Auth ^ | | |
| GET | /auth/login | Login v |
| GET | /auth/callback | Auth Callback v |
| User ^ | | |
| POST | /user/register | Register v |
| POST | /user/login | Login v |
| Quiz Submission ^ | | |

8.ADVANTAGES & DISADVANTAGES

Advantages:

- Quick and personalized learning

- Minimal setup required
- AI-generated quizzes improve retention

Disadvantages:

- Dependency on internet and IBM API
- May require fine-tuning for specific subjects

9. CONCLUSION

EduTutor AI offers an innovative approach to personalized learning. It bridges the gap between traditional study methods and AI-powered solutions by providing instant explanations and quizzes. The platform enhances the learning process and makes education more interactive and effective.

10. FUTURE SCOPE

- Add support for diagram-based explanations
- Include multilingual quiz support
- Integrate with platforms like Google Classroom
- Track student progress and analytics

11. APPENDIX

GitHub & Project Demo Link: <https://github.com/riktha-13/EduTutor-AI-Personalized-Learning-with-Generative-AI-and-LMS-Integration/upload/main>