Project Design Phase
Solution Architecture

Date	27 June 2025
Team ID	LTVIP2025TMID32976
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:

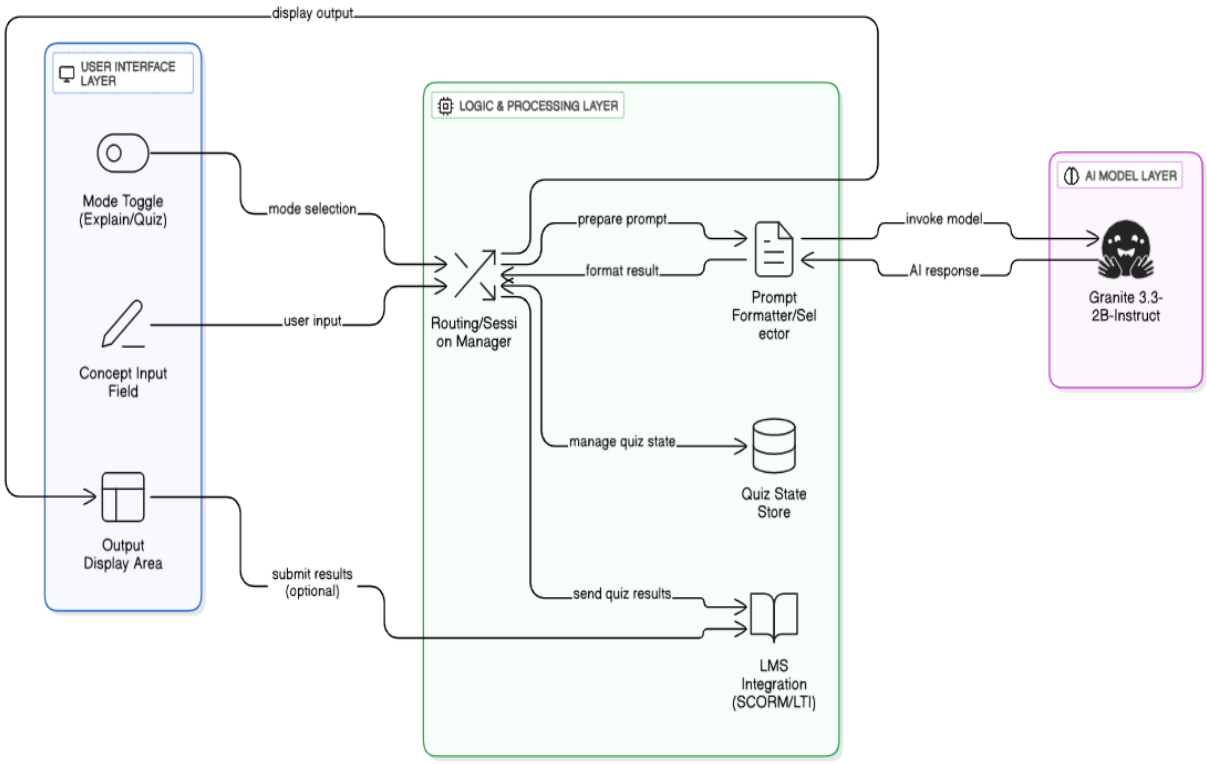

Figure 1: Architecture and data flow of the EduTutor AI: Personalized Learning with Generative AI and LMS Integration.