**Project Design Phase-II**

Technology Stack (Architecture & Stack)

Date: 27 JUNE 2025

Team ID: LTVIP2025TMID31213

Project Name:

EDUTUTOR-AI-PERSONALIZED-LEARNING-WITH-GENERATIVE-LMS-INTEGRATION

Maximum Marks: 4 Marks

**Technical Architecture**

The deliverable includes the architectural diagram, customer journey map, and detailed information as per the following structure.

**Table-1: Technology Stack**

|  |  |  |  |
| --- | --- | --- | --- |
| **S.No** | **Component** | **Description** | **Technology Used** |
| 1 | Application Logic - 1 | Quiz Generation and Personalized Learning | LogicPython (Flask / FastAPI) |
| 2 | Application Logic - 2 | Speech-to-Text for student input | IBM Watson STT |
| 3 | Application Logic - 3 | AI Assistant for Query Handling | IBM Watson Assistant |
| 4 | Database | Stores User Data, Quiz Data, Results | MySQL + MongoDB |
| 5 | Cloud Database | Cloud-based secure storage of data | IBM DB2, IBM Cloudant |
| 6 | File Storage | File storage for reports, resources | IBM Cloud Object Storage |
| 7 | External API - 1 | Weather-based insights | IBM Weather API |
| 8 | External API - 2 | Student Identity Verification | Aadhar API |
| 9 | Machine Learning Model | Personalized Recommendation Engine | OpenAI GPT-4 |
| 10 | Infrastructure (Server/Cloud) | Application Deployment and Hosting | IBM Cloud Foundry / Kubernetes |

**Table-2: Application Characteristics**

|  |  |  |  |
| --- | --- | --- | --- |
| **S.No** | **Characteristics** | **Description** | **Technology Used** |
| 1 | Open-Source Frameworks | Backend and Frontend frameworks | Flask / FastAPI, ReactJS |
| 2 | Security Implementations | Access Controls, Encryption | OAuth2, SHA-256, IAM, OWASP |
| 3 | Scalable Architecture | Microservices-based modular and scalable | systemDocker, Kubernetes |
| 4 | Availability | Load Balancing and Fault Tolerance | IBM Cloud Load Balancer, Distributed Serve |
| 5 | Performance | Caching and CDN for optimized performanc | eRedis, IBM CDN |

**Customer Journey Map**

\*\*Stage 1: Awareness\*\*

* Student/Teacher discovers EduTutor AI via web or recommendation.

\*\*Stage 2: Onboarding\*\*

* User logs in using Google authentication.
* Syncs Google Classroom or uploads course details.

\*\*Stage 3: Engagement\*\*

* Students access personalized quizzes and study resources.
* Teachers track student progress via dashboard.

\*\*Stage 4: Interaction\*\*

* Students interact with AI Assistant for doubts.
* Real-time feedback provided using AI models.

\*\*Stage 5: Evaluation\*\*

* Student's performance evaluated automatically.
* Reports and analytics generated.

\*\*Stage 6: Continuous Learning\*\*

* Personalized recommendations offered for improvement.
* Teachers update content or quizzes based on analytics.

**References**

* https://c4model.com/

-

https://developer.ibm.com/patterns/ai-powered-backend-system-for-order-processing-during-pande mics/

* https://www.ibm.com/cloud/architecture
* https://aws.amazon.com/architecture

-

https://medium.com/the-internal-startup/how-to-draw-useful-technical-architecture-diagrams-2d20c9f da90d