**Project Design Phase-II**

**Technology Stack (Architecture & Stack)**

|  |  |
| --- | --- |
| Date | 28 June 2025 |
| Team ID | LTVIP2025TMID46247 |
| Project Name | Classifying Fabric Patterns Using Deep Learning |
| Maximum Marks | 4 Marks |

**Technical Architecture:**

The Deliverable shall include the architectural diagram as below and the information as per the table1 & table 2

**Example: Order processing during pandemics for offline mode**

**Reference:** [**https://developer.ibm.com/patterns/ai-powered-backend-system-for-order-processing-during-pandemics/**](https://developer.ibm.com/patterns/ai-powered-backend-system-for-order-processing-during-pandemics/)

Guidelines:

Include all the processes (As an application logic / Technology Block)

Provide infrastructural demarcation (Local / Cloud)

Indicate external interfaces (third party API’s etc.)

Indicate Data Storage components / services

Indicate interface to machine learning models (if applicable)



**Table-1 : Components & Technologies:**

|  |  |  |  |
| --- | --- | --- | --- |
| **S.No** | **Component** | **Description** | **Technology** |
| 1 | User Interface | Web interface for uploading fabric images and displaying results | Streamlit |
| 2 | Application Logic-1 | Image preprocessing and prediction pipeline | Python (TensorFlow, NumPy) |
| 3 | Application Logic-2 | Model training with data augmentation | TensorFlow / Keras |
| 4 | Application Logic-3 | Data augmentation logic | Keras Preprocessing |
| 5 | Database | No structured DB; uses directory-based organization | N/A |
| 6 | Cloud Database | Not applicable | N/A |
| 7 | File Storage | Stores uploaded images and trained model files | Local File System |
| 8 | External API-1 | Not used | N/A |
| 9 | External API-2 | Not used | N/A |
| 10 | Machine Learning Model | CNN for fabric pattern classification | TensorFlow / Keras |
| 11 | Infrastructure (Server / Cloud) | Local deployment of Streamlit app | Local Machine |

**Table-2: Application Characteristics:**

| **S.No** | **Characteristics** | **Description** | **Technology** |
| --- | --- | --- | --- |
| 1 | Open-Source Frameworks | Uses open-source frameworks for AI and UI | TensorFlow, Streamlit, NumPy |
| 2 | Security Implementations | Local-only usage; no web exposure | N/A |
| 3 | Scalable Architecture | Model can be retrained with larger datasets | TensorFlow |
| 4 | Availability | App is available on local system at all times | Localhost |
| 5 | Performance | Lightweight model and fast classification on local machine | TensorFlow, Streamlit |

**References:**

[**https://c4model.com/**](https://c4model.com/)

[**https://developer.ibm.com/patterns/online-order-processing-system-during-pandemic/**](https://developer.ibm.com/patterns/online-order-processing-system-during-pandemic/)

[**https://www.ibm.com/cloud/architecture**](https://www.ibm.com/cloud/architecture)

[**https://aws.amazon.com/architecture**](https://aws.amazon.com/architecture)

[**https://medium.com/the-internal-startup/how-to-draw-useful-technical-architecture-diagrams-2d20c9fda90d**](https://medium.com/the-internal-startup/how-to-draw-useful-technical-architecture-diagrams-2d20c9fda90d)