**Project Design Phase-II**

**Technology Stack (Architecture & Stack)**

|  |  |
| --- | --- |
| Date | 31 January 3035 |
| Team ID | LTVIP2025TMID34781 |
| Project Name | Smart Sorting |
| Maximum Marks | 4 Marks |

|  |  |  |  |
| --- | --- | --- | --- |
| S.No | Characteristics | Description | Technology |
| 1 | Application Logic-1 | Core logic for image preprocessing and classification | Python (TensorFlow, Keras) |
| 2 | Application Logic-2 | Integration of speech feedback for classification (optional) | IBM Watson Speech-to-Text |
| 3 | Application Logic-3 | User interaction via chatbot (optional use case) | IBM Watson Assistant |
| 4 | Database | Store labels, image metadata, and logs | MySQL / SQLite |
| 5 | Cloud Database | Cloud-based data storage and retrieval | IBM Cloudant / IBM DB2 |
| 6 | File Storage | Image files and results storage | Local filesystem / IBM Block Storage |
| 7 | External API-1 | Get environmental data like temperature/humidity (optional) | IBM Weather API |
| 8 | External API-2 | Identity verification for secure access (optional) | Aadhar API (via UIDAI) |
| 9 | Machine Learning Model | Transfer Learning Model for Fresh vs Rotten Classification | MobileNetV2 / ResNet50 (TensorFlow) |
| 10 | Infrastructure (Server / Cloud) | Deployment of application on a local machine or cloud platform | Localhost / IBM Cloud Foundry / AWS |

**Table-2: Application Characteristics:**

|  |  |  |  |
| --- | --- | --- | --- |
| S.No | Characteristics | Description | Technology |
| 1 | Open-Source Frameworks | Frameworks and libraries used | TensorFlow, Keras, OpenCV, Flask |
| 2 | Security Implementations | Authentication, encryption, and secure data handling | SHA-256, HTTPS, API Keys, IAM Controls |
| 3 | Scalable Architecture | Supports growth in data and users via microservices or containerization | Flask (API), Docker, Kubernetes |
| 4 | Availability | Ensures continuous access and fault tolerance | Load Balancer, Redundancy, Cloud Hosting (IBM/AWS) |

**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)

|  |  |  |  |
| --- | --- | --- | --- |
| **S.No** | **Characteristics** | **Description** | **Technology** |
| 1. | Open-Source Frameworks | List the open-source frameworks used | Technology of Opensource framework |
| 2. | Security Implementations | List all the security / access controls implemented, use of firewalls etc. | e.g. SHA-256, Encryptions, IAM Controls, OWASP etc. |
| 3. | Scalable Architecture | Justify the scalability of architecture (3 – tier, Micro-services) | Technology used |
| 4. | Availability | Justify the availability of application (e.g. use of load balancers, distributed servers etc.) | Technology used |