**Project Design Phase-II**

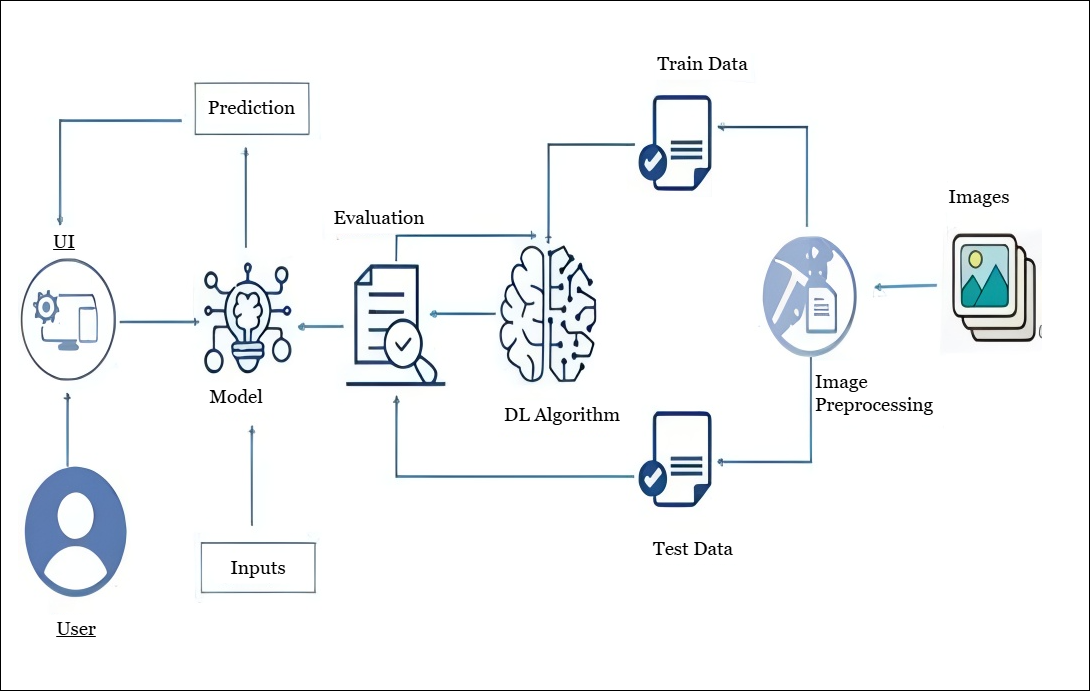
**Technology Stack (Architecture & Stack)**

|  |  |
| --- | --- |
| Date | 31 January 3035 |
| Team ID | LTVIP2025TMID59800 |
| Project Name | Smart Sorting: Transfer Learning for Identifying Rotten Fruits and Vegetables |
| Maximum Marks | 4 Marks |

**Technical Architecture:**

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

**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/)

****

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

|  |  |  |  |
| --- | --- | --- | --- |
| **S.No** | **Component** | **Description** | **Technology** |
|  | User Interface | How user interacts with application (Web Page) | HTML, CSS, Bootstrap, Flask (Python) |
|  | Application Logic | Logic for a process in the application | Python |
|  | File Storage | File storage requirements | Stores predicted images in Local Filesystem |
|  | Machine Learning Model | Purpose of Machine Learning Model | VGG16 |
|  | Data | Data used to train the model | Dataset from Kaggle |

**Table-2: Application Characteristics:**

| **S.No** | **Characteristics** | **Description** | **Technology** |
| --- | --- | --- | --- |
|  | User-Friendly Interface | Simple, intuitive web interface for image upload and result visualization. | HTML, CSS, Bootstrap, Flask (Python) |
|  | Real-Time Prediction | Immediate classification of produce as healthy or rotten. | Flask backend, TensorFlow model |
|  | |  | | --- | | Extendable Dataset Support |  |  | | --- | |  | | |  | | --- | |  |   New produce types can be added by updating the dataset and retraining. | ImageDataGenerator, Keras, TensorFlow |
|  | Efficient Processing | Optimized VGG16 model ensures fast and reliable predictions. | Pre-trained VGG16, Numpy |

**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)