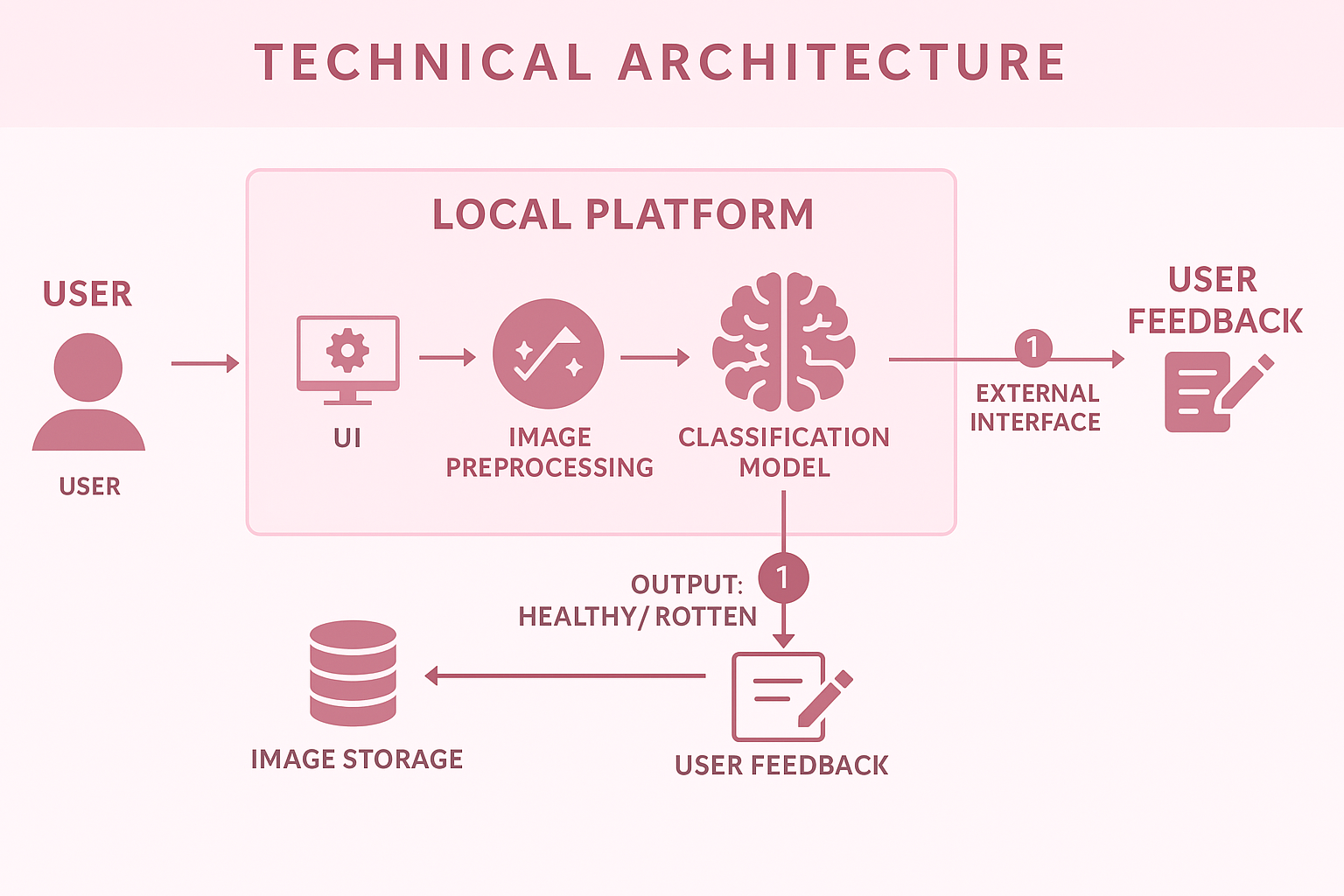
**Project Design Phase-II**

**Technology Stack (Architecture & Stack)**

|  |  |
| --- | --- |
| Date | 28 January 2025 |
| Team ID | LTVIP2025TMID41140 |
| 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

****

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 Used** |
| --- | --- | --- | --- |
| **1.** | **User Interface** | **Web UI for uploading images & displaying results** | **HTML, CSS, Jinja2 Templates (Flask)** |
| **2.** | **Application Logic-1** | **Handles routing, image upload, prediction, and feedback** | **Python (Flask)** |
| **3.** | **Application Logic-2** | **Image preprocessing before prediction** | **Keras utilities (load\_img, img\_to\_array)** |
| **4.** | **Application Logic-3** | **AI model prediction logic** | **TensorFlow / Keras with VGG16** |
| **5.** | **Local File Storage** | **Saves uploaded images and feedback locally** | **Local filesystem (static/, .json files)** |
| **6.** | **Machine Learning Model** | **VGG16 model for fruit/vegetable classification** | **.keras or .h5 model (TensorFlow/Keras)** |
| **7.** | **Infrastructure** | **Application hosted locally for development/testing** | **Flask via Anaconda, running on Localhost** |

**Table-2: Application Characteristics:**

| **S.No** | **Characteristic** | **Description** | **Technology** |
| --- | --- | --- | --- |
| **1.** | **Open-Source Frameworks** | **Uses only free and open-source libraries** | **Python, Flask, Keras, TensorFlow** |
| **2.** | **Scalable Architecture** | **Can be scaled later; currently a simple local Flask app** | **3-tier: UI → Logic → Model** |
| **3.** | **Availability** | **Runs on a local system only** | **Flask server on localhost** |
| **4.** | **Performance** | **Uses pre-trained VGG16, good prediction speed** | **TensorFlow/Keras** |