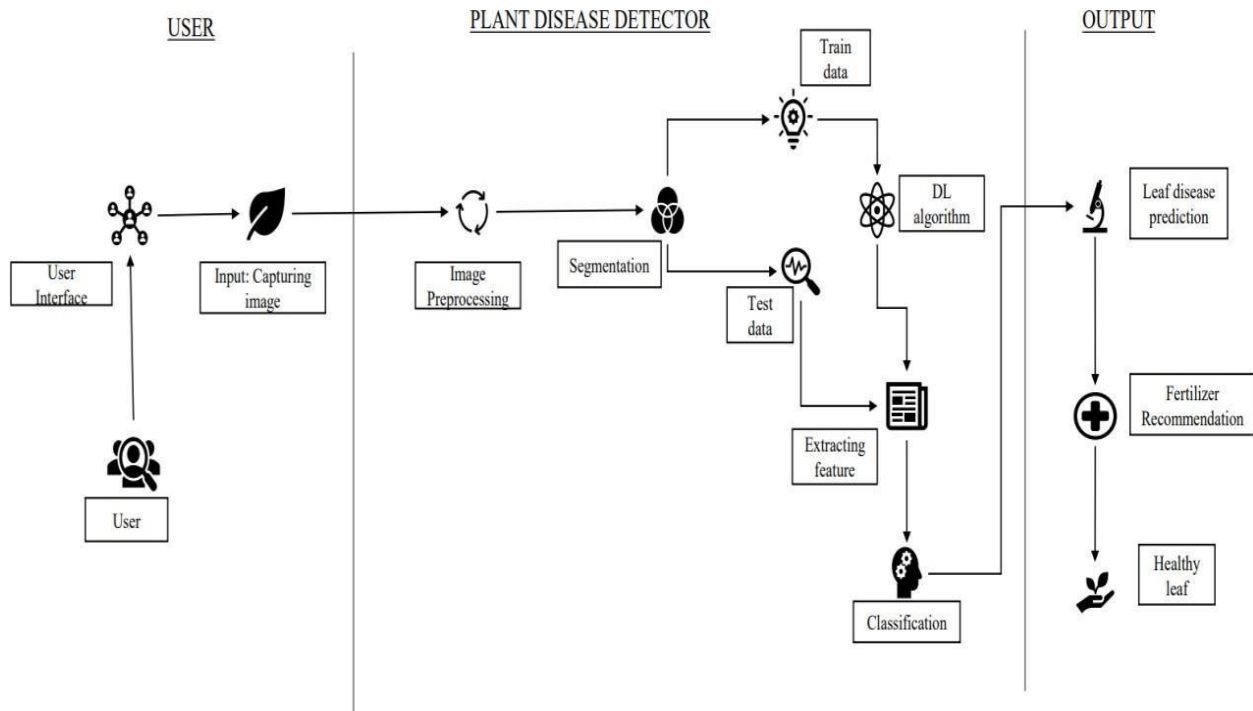


## PROJECT DESIGN PHASE-II

### TECHNOLOGY STACK (ARCHITECTURE & STACK)

|              |  |
|--------------|--|
| Date         | 16 October 2022  |
| Team ID      | PNT2022TMID00681   |
| Project Name | Fertilizers Recommendation System For Disease Prediction |

#### TECHNICAL ARCHITECTURE:



**TABLE -1: COMPONENTS & TECHNOLOGIES :**

| S.NO | Component                 | Description  | Technology   |
|------|---------------------------|--|--|
| 1,   | User Interface            | The user interact with the website by logging in and uploading the image (affected plant leaf).The data is sent through the Cloud by API and HTTP request ,response. | HTML, CSS, JavaScript, React, Angular, etc.,                               |
| 2,   | Disease Prediction        | Once the image has been uploaded by the user it is then processed by the model and the model gives the name of the disease affected by the plant.                    | Keras, CNN, Image processing, Machine Learning Algorithm,TensorFlow,etc, . |
| 3.   | Fertilizer Recommendation | After the disease has been predicted by the model, the fertilizer is recommended for the predicted disease   | User interface, HTML, CSS, JavaScript                                      |
| 4.   | Dataset                   | The training and testing dataare collectively stored in the IBM Watson Cloud.  | Kaggle.com, data.gov, UCI machine learning repository,etc.                 |
| 5.   | File Storage              | File storage requirements  | IBM, Local File system.  |
| 6,   | Modules                   | Purpose of deep learningmodules  | Image Recognition Modules, etc.  |
| 7.   | Infrastructure(Server)    | Application development on Local server<br>configuration:<br>4GB RAM<br>512GB SSD<br>64-bit Operating System<br>AMD PRO A4-3350BAPU                                  | Local File system.   |

**TABLE – 2: APPLICATION CHARACTERISTICS:**

| <b>S.NO</b> | <b>Characteristics</b> | <b>Description</b>  | <b>Technology</b>   |
|-------------|------------------------|---|---|
| 1.          | Opensource Framework   | List of the opensource framework used are TensorFlow, Keras, OpenCV   | Open source-PyCharm, anaconda navigator, flask ,OpenCV, CNN, Keras, TensorFlow. |
| 2.          | Secure Implementation  | Provides highly Secure Environment for the details provided.  | Security – OWASP,SHA-256,Encryptions,etc.,                                      |
| 3.          | Scalable Architecture  | As this architecture is Scalable in nature .Whenever a new feature is added to the model, it will not decrease the performance of the model.  | flask, OpenCV, CNN, TensorFlow, Keras.  |
| 4.          | Availability           | The website is completely available and provide 24/7 availability .The user can access the information irrespective of the location.  | Web application access to all the resources.                                    |
| 5.          | Performance            | Performance of the website is high when the image that is to be processed is available in the Cache. Performance is bit low when the image to be processed is not available in the cache. | Machine Learning Algorithm,Keras, Tensorflow,CNN, etc.,                         |