Project Design Phase-II Technology Stack (Architecture & Stack)

|  |  |
| --- | --- |
| Date | 03 October 2022 |
| Team ID | PNT2022TMIDxxxxxx |
| Project Name | Project - xxx |
| Maximum Marks | 4 Marks |

Technical Architecture:

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

A diagram of a diagram

Description automatically generated

Table-1 : Components & Technologies:

|  |  |  |  |
| --- | --- | --- | --- |
| **S.No** | **Component** | **Description** | **Technology** |
| 1. | User Interface | How user interacts with application: Web UI | HTML, CSS, JavaScript. |
| 2. | Web Application (Backend) | It connects the web interface to the ML model. | Flask. |
| 3. | Image preprocessing | It is necessary to ensure that uploaded images are in the required format for the machine learning model. | ImageDataGenerator Class in Keras. |
| 4. | Machine Learning Model | The core component responsible for identifying pest threats in images. | VGG16 (CNN) Model, TensorFlow |
| 5. | Application Logic | Logic for processes in application. | Python |
| 6. | Development Environments | Used for local development and data management. | Anaconda Navigator, Google Colab. |
| 7. | Dataset | Dangerous Farm Insects Threats Dataset | Kaggle |

Table-2: Application Characteristics:

|  |  |  |  |
| --- | --- | --- | --- |
| **S.No** | **Characteristics** | **Description** | **Technology** |
| 1. | Open-Source Frameworks | It is used to develop and train the machine learning model for insect classification. | TensorFlow |
| 2. | Security Implementations | Data encryption can be implemented using secure communication protocols like HTTPS (SSL/TLS). | Flask - SSLify |
| 3. | Scalable Architecture | The architecture may not involve traditional scalability components like databases. |  |

|  |  |  |  |
| --- | --- | --- | --- |
| **S.No** | **Characteristics** | **Description** | **Technology** |
| 4. | Availability | While Flask is not inherently a load balancing or distributed system, it still offers a degree of fault tolerance. | Flask |
| 5. | Performance | Design consideration for the performance of the application:   1. Performance can be optimized by ensuring that the model's inference process is efficient and responsive to user requests. 2. Implement basic in-memory caching in your Flask application to store and serve frequently accessed results. | Flask |

References:

<https://c4model.com/>

<https://developer.ibm.com/patterns/online-order-processing-system-during-pandemic/> <https://www.ibm.com/cloud/architecture>

<https://aws.amazon.com/architecture>

<https://medium.com/the-internal-startup/how-to-draw-useful-technical-architecture-diagrams-2d20c9fda90d>