**Technology Stack**

|  |  |
| --- | --- |
| **Date** | **24 June 2025** |
| **Team ID** |  |
| **Project Name** |  |
| **Maximum Marks** | **4 Marks** |

**Table-1: Components & Technologies**

|  |  |  |  |
| --- | --- | --- | --- |
| **S.no** | **Component** | **Description** | **Technology** |
| 1. | User Interface | Web interface for image upload and displaying prediction result | HTML, CSS, JavaScript, Bootstrap |
| 2. | Application Logic-1 | Backend server managing request routing, file upload, and serving prediction results | Python, Flask |
| 3. | Application Logic-2 | AI model integration and disease prediction logic | TensorFlow, Keras (Transfer Learning) |
| 4. | File Storage | Temporarily holds uploaded images during processing | Local filesystem (Flask static/uploads folder) |
| 5. | Machine Learning Model | Performs disease classification on poultry images | TensorFlow / Keras (MobileNet / ResNet / EfficientNet) |
| 6. | Infrastructure (Server/Cloud) | Application deployment environment | Flask Local Server / Render Cloud Hosting |
| 7. | Development Tools | Tools for coding, testing, and model development | Jupyter Notebook, Google Colab, VS Code |

**Table-2: Application Characteristics**

|  |  |  |  |
| --- | --- | --- | --- |
| S.no | Characteristics | Description | Technology / Approach |
| 1. | Open-Source Frameworks | Frameworks, libraries, and platforms used | Flask, TensorFlow, Keras, Bootstrap |
| 2. | Security Implementations | Basic app security — input validation, file type checks, safe upload handling | Flask Security, Secure File Handling (werkzeug secure\_filename) |
| 3. | Scalable Architecture | Simple modular 2-tier architecture (Presentation + Logic layer) | Flask MVC structure, AI model service integration |
| 4. | Availability | Deployable on both local and cloud servers | Render Cloud Hosting, Flask Local Server |
| 5. | Performance | Optimized AI inference using pretrained models for minimal prediction delay | TensorFlow / Keras Transfer Learning |