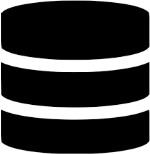
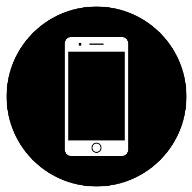
Project Design Phase-II Technology Stack (Architecture & Stack)

|  |  |
| --- | --- |
| Date | 03 October 2022 |
| Team ID | PNT2022TMID32046 |
| Project Name | SMARTFARMER – IoT ENABLED SMART FARMING APPLICATION |
| Maximum Marks | 4 Marks |

Technical Architecture:

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

USER DEVICE



**IBM CLOUD**

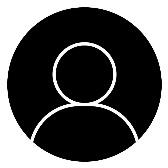
**IOT**

**APPLICATION**

**LOGIN**

**Humidity sensor**

**DATABASE**





Temperature sensor

Moisture sensor

Table-1 : Components & Technologies:

|  |  |  |  |
| --- | --- | --- | --- |
| **S.No** | **Component** | **Description** | **Technology** |
| 1. | User Interface | How user interacts with application e.g. Web UI, Mobile App, Chatbot etc. | HTML, CSS, JavaScript / Angular Js / React Js etc. |
| 2. | Application Logic-1 | Logic for a process in the application | Java / Python |
| 3. | Application Logic-2 | Logic for a process in the application | IBM Watson STT service |
| 4. | Application Logic-3 | Logic for a process in the application | IBM Watson Assistant |
| 5. | Database | Data Type, Configurations etc. | MySQL, NoSQL, etc. |
| 6. | Cloud Database | Database to store the data fetch by sensors connected with IoT device. | IBM DB2, IBM Cloudant etc. |
| 7. | File Storage | The user can store the data in any format | IBM Block Storage or Other Storage  Service or Local Filesystem |
| 8. | External API-1 | Because of farming land it will be need to monitoring weather, so the weather API are used. | IBM Weather API. |
| 9. | Machine Learning Model | It is necessary to monitor and identify the disease  infection. | Object Recognition Model. |
| 10. | Infrastructure (Server / Cloud) | Application Deployment on Local System / Cloud Local Server Configuration:  Cloud Server Configuration : | Local, Cloud Foundry, Kubernetes, etc. |

Table-2: Application Characteristics:

|  |  |  |  |
| --- | --- | --- | --- |
| **S.No** | **Characteristics** | **Description** | **Technology** |
| 1. | Open-Source Frameworks | Node red, MIT App | Connecting Web UI and Applications |
| 2. | Security Implementations | List all the security / access controls implemented, use of firewalls etc. | e.g. SHA-256, Encryptions, IAM Controls, OWASP etc. |
| 3. | Scalable Architecture | Justify the scalability of architecture (3 – tier, Micro-services) | Technology used |
| 4. | Availability | Justify the availability of application (e.g. use of load balancers, distributed servers etc.) | Technology used |
| 5. | Performance | Design consideration for the performance of the application (number of requests per sec, use of  Cache, use of CDN’s) etc. | Technology used |