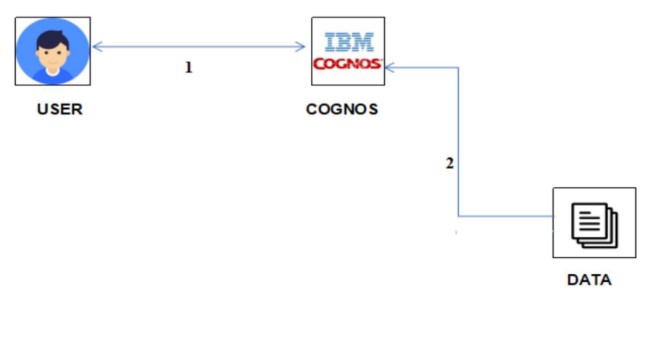
**Project Design Phase-II Technology Stack (Architecture & Stack)**

|  |  |
| --- | --- |
| Date | 12-11-2022 |
| Team ID | PNT2022TMID24269 |
| Project Name | Estimate The Crop Yield Using Data Analytics |
| Maximum Marks | 4 Marks |

**Technical Architecture:**

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



**Example: Order processing during pandemics for offline mode**

**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 | Logging in as a farmer (common user) in the application | Java / Python |
| 3. | Application Logic-2 | Logging in as an admin in the application | IBM Watson STT service |
| 4. | Application Logic-3 | Logging in as a merchant the application | IBM Watson Assistant |
| 5. | Database | Data about the crops is stored in database. | MySQL, NoSQL, etc. |
| 6. | Cloud Database | IBM Watson cloud is used for storage | IBM DB2, IBM Cloudant etc. |
| 7. | External API-1 | Purpose of External API used in the application | IBM Weather API, etc. |
| 8. | Machine Learning Model | Purpose of Machine Learning Model | Object Recognition Model, etc. |
| 9. | 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 | List the open-source frameworks used | Technology of Opensource framework |
| **S.No** | **Characteristics** | **Description** | **Technology** |
| 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 |

**Is the system robust?**

Yes, the software that is being developed for data analysis is robust

**Is it highly modifiable?**

Yes, the system is user friendly and ready for developers to make changes and enhance it

**Is it scalable?**

Yes, the system can be scaled up when there is a need and is flexible.

**Is it buildable?**

Yes, it is feasible to build at a low budget