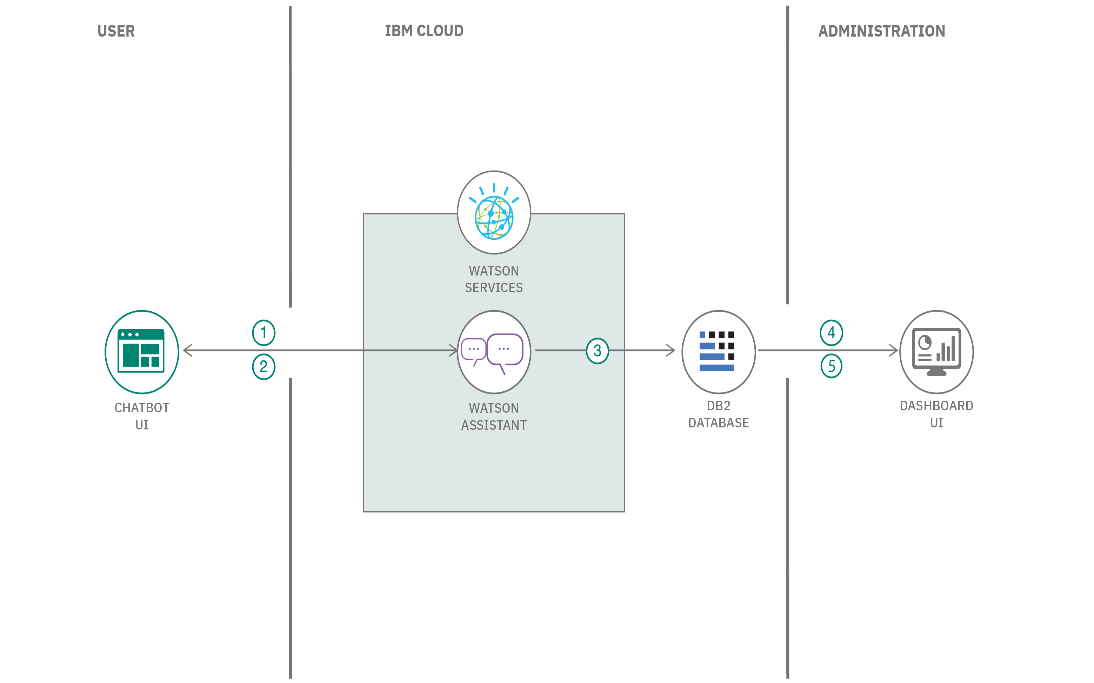
**Project Design Phase-II**

**Technology Stack (Architecture & Stack)**

|  |  |
| --- | --- |
| Date | 03 October 2022 |
| Team ID | PNT2022TMID23604 |
| Project Name | Project – Skill and Job Recommender |
| Maximum Marks | 4 Marks |

**Technical Architecture:**



1. Feed the data to the Watson Assistant service.
2. Convert the text into intents and entities, and enable the conversation.
3. Extract the order and customer details from the text.
4. The extracted attributes get stored on Db2 Database on IBM Cloud.
5. Visualize the order and customer details from the input text on a dashboard.

**Table-1 : Components & Technologies:**

|  |  |  |  |
| --- | --- | --- | --- |
| **S.No** | **Component** | **Description** | **Technology** |
|  | User Interface | Web UI, Chatbot etc. | HTML, CSS, JavaScript / Angular Js / React Js etc. |
|  | Application Logic-1 | Kubernetes Algorithm | Python |
|  | Application Logic-2 | Get insights through accelerated data optimization capabilities | IBM Watson STT service |
|  | Application Logic-3 | Deploy Webchat in minutes, or use our fully extensible architecture | IBM Watson Assistant |
|  | Database | Stores data efficiently | MySQL |
|  | Cloud Database | Database Service on Cloud | IBM DB2, IBM Cloudant etc. |
|  | File Storage | IBM Cloud object storage file access | IBM Block Storage or Other Storage Service or Local Filesystem |
|  | External API-1 | Using IBM Cloud monitoring | IBM Weather API, etc. |
|  | External API-2 | External API used in the application | Aadhar API, etc. |
|  | Machine Learning Model | Best practices for running containers and Kubernetes in prediction | Object Recognition Model, etc. |
|  | Infrastructure (Server / Cloud) | Flask,IBM DB2,IBM Object storage,IBM Watson assistant,IBM Cloud container register,IBM Cloud,Kubernetes | Local, Cloud Foundry, Kubernetes, etc. |

**Table-2: Application Characteristics:**

| **S.No** | **Characteristics** | **Description** | **Technology** |
| --- | --- | --- | --- |
|  | Open-Source Frameworks | Flask | python |
|  | Security Implementations | IBM Object storage and integration | SHA-256, Encryptions, IAM Controls, OWASP |
|  | Scalable Architecture | 3 – tier, Micro-services | IBM DB2, IBM Cloud storage |
|  | Availability | Use of load balancers, distributed servers | IBM Cloud |
|  | Performance | Design consideration for the performance of the application (number of requests per sec, use of Cache, use of CDN’s) etc. | IBM Kubernetes |

**References:**

[**https://c4model.com/**](https://c4model.com/)

[**https://developer.ibm.com/patterns/online-order-processing-system-during-pandemic/**](https://developer.ibm.com/patterns/online-order-processing-system-during-pandemic/)

[**https://www.ibm.com/cloud/architecture**](https://www.ibm.com/cloud/architecture)

[**https://aws.amazon.com/architecture**](https://aws.amazon.com/architecture)

[**https://medium.com/the-internal-startup/how-to-draw-useful-technical-architecture-diagrams-2d20c9fda90d**](https://medium.com/the-internal-startup/how-to-draw-useful-technical-architecture-diagrams-2d20c9fda90d)