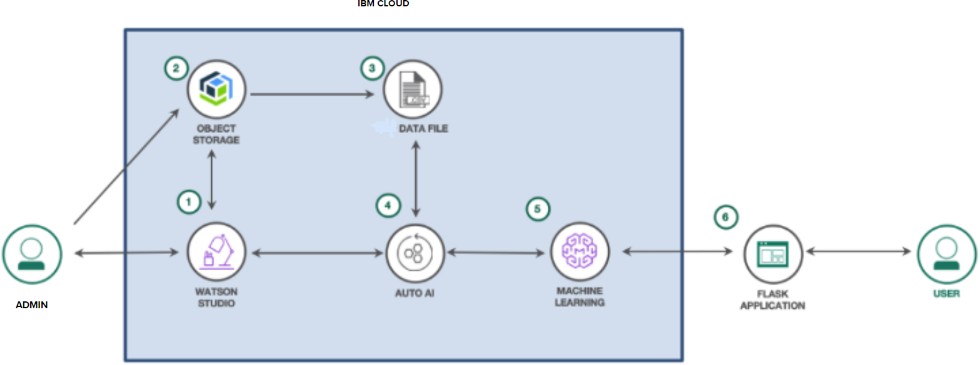
Project Design Phase-II Technology Stack (Architecture & Stack)

|  |  |
| --- | --- |
| Date | 18 October 2022 |
| Team ID | PNT2022TMID31228 |
| Project Name | Project - Detecting Parkinson’s Disease using Machine Learning. |
| Maximum Marks | 4 Marks |

# Technical Architecture:

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



# Table-1: Application Components:

|  |  |  |  |
| --- | --- | --- | --- |
| **S.No** | **Component** | **Description** | **Technology** |
| 1. | User Interface | How the user interacts with the application e.g. Web UI | HTML, CSS, Python flask. |
| 2. | Application Logic-1 | Register and Login page | HTML, CSS, Python flask. |
| 3. | Application Logic-2 | Home Page | HTML, CSS. |
| 4. | Application Logic-3 | Test vital page | HTML, CSS, Python flask. |
| 5. | Database | Data Type, Configurations, etc. | MySQL. |
| 6. | Cloud Database | Database Service on Cloud | IBM Database. |
| 7. | File Storage | File Storage requirements | IBM Cloud Object Storage |
| 8. | External API-1 | Purpose of External API used in the  application | IBM API Connect. |
| 9. | External API-2 | Purpose of External API used in the  application | NIL |
| 10. | Machine Learning Model | Train the classification model using the Random forest classification algorithm. | IBM Watson Studio. |
| 11. | Infrastructure (Server /  Cloud) | Application Deployment on Local  System / Cloud. | Local Server Configuration:  Local System.  Cloud Server Configuration: IBM Watson |

**Table-2: Application Characteristics:**

|  |  |  |  |
| --- | --- | --- | --- |
| **S.No** | **Characteristics** | **Description** | **Technology** |
| 1. | Open-Source Frameworks | List the open-source frameworks used | Python Flask, Jupyter Notebook, Tensorflow, and Python libraries. |
| 2. | Security Implementations | List all the security/access controls  implemented, use of firewalls, etc. | Through Password |
| 3. | Scalable Architecture | Justify the scalability of architecture (3  – tier, Micro-services) | Python Libraries. |
| 4. | Availability | Justify the availability of applications (e.g. use of load balancers, distributed  servers, etc.) | IBM Watson Machine Learning. |
| 5. | Performance | Design Considerations for the performance of the application (number of requests per sec, use of Cache, use of CDNs), etc. | Flask. |