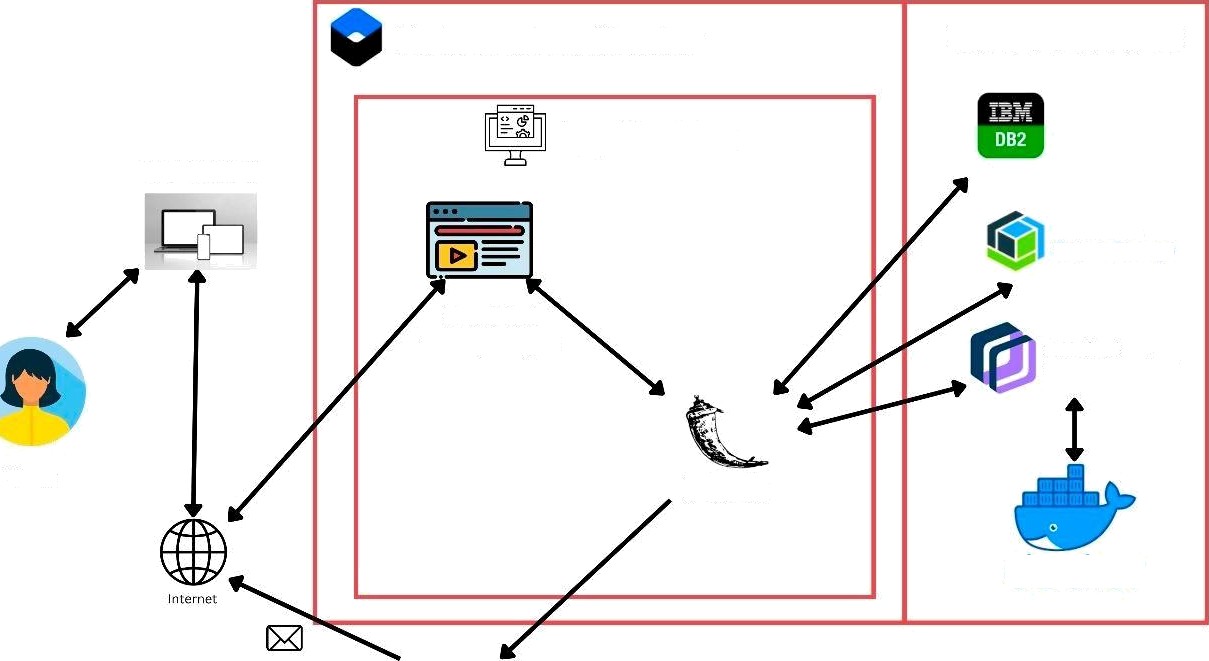
**Project Design Phase-II Technology Stack (Architecture & Stack)**

|  |  |
| --- | --- |
| Date | 18 October 2022 |
| Team ID | PNT2022TMID24120 |
| Project Name | Project – Plasma Donor Application |
| Maximum Marks | 4 Marks |



lasma Donor Ap fication Techn Architecture



Kubernetes Cluster

Cloud Services

Application

User System

IBM Cloud

Ob ect Storage

Front End

HTM L, CSS, JS

IBN Cloud Conlainer Registry

User

Flask

SendGrid

Table-1 : Components & Technologies:

|  |  |  |  |
| --- | --- | --- | --- |
| **S.No** | **Component** | **Description** | **Technology** |
| 1. | User Interface | For user onboarding such as Login and Dashboard functions | HTML, CSS, JavaScript |
| 2. | Verifying and filtering matching donors | Database operations to get data and perform operations and give user the appropriate details | Python |
| 3. | IBM Watson | Chatbot to enable instant help for user | Watson Assistant by IBM |
| 4. | Database | Stores all data including donor and user  information | MySQL, NoSQL. |
| 5. | Cloud Database | Cloud is used to store all the data in the database  for elasticity and security | IBM DB2, IBM Cloudant. |
| 6. | File Storage | File storage requirements have to be met here | IBM Cloud object storage |
| 7. | External API: To send email SendGrid | Notifying users through e-mail when required to pass critical information | SendGrid |
| 8. | Infrastructure (Server / Cloud) | For Application Deployment in Cloud | IBM - Docker – container, Cloud  Foundry, Kubernetes container |

Table-2: Application Characteristics:

|  |  |  |  |
| --- | --- | --- | --- |
| **S.No** | **Characteristics** | **Description** | **Technology** |
| 1. | Open-Source Frameworks | Using flask as open source framework to deploy the backend features and basic app navigation. | Flask |
| 2. | Security Implementations | Stable architecture with secure application  services and its functionalities. | IAM Controls, SHA-256, Encryptions |
| 3. | Scalable Architecture | Extensible architecture with 3-tier, micro -services. | IBM cloud and Flask with front end |
| 4. | Availability | Availability maintained by use of Kubernetes and  load balancers, and also with distributed servers. | IBM DB2, Docker, Kubernetes, Cloud  Object storage. |
| 5. | Performance | Efficiency of the application in use | IBM Container registry |