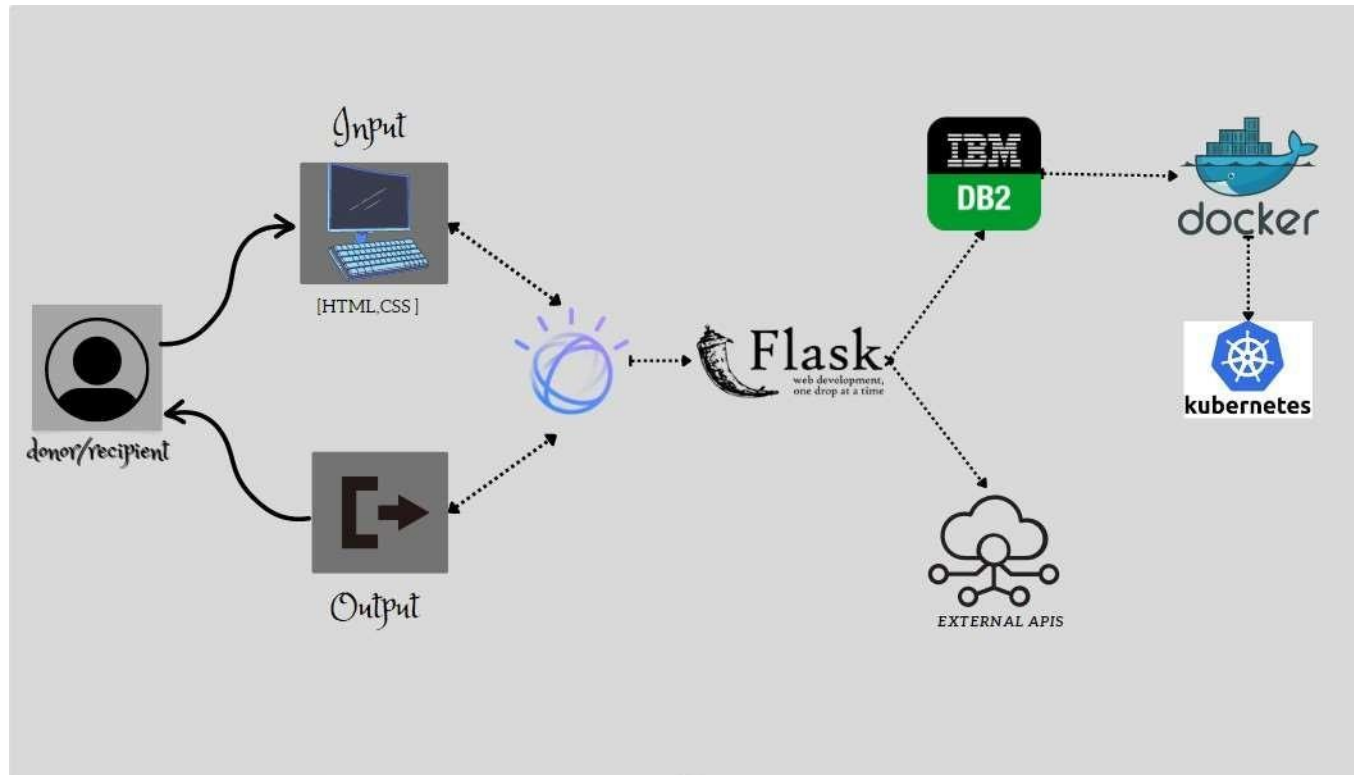


## Project Design Phase-II Technology Stack (Architecture & Stack)

|               |                                    |
|---------------|------------------------------------|
| Date          | 12 October 2022                    |
| Team ID       | PNT2022TMID14320                   |
| Project Name  | Project – Plasma Donor Application |
| Maximum Marks | 4 Marks                            |

### Technical Architecture:



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

| S. No | Component                       | Description   | Technology                       |
|-------|---------------------------------|---|----------------------------------|
| 1.    | User Interface                  | How user interacts with application   | HTML, CSS, JavaScript / React Js |
| 2.    | Application Logic-1             | Registration with verification and Login to the app.  | Python                           |
| 3.    | Application Logic-2             | Dashboard with donors and plasma availability details for recipient and requests for donors | Python-Flask                     |
| 4.    | Application Logic-3             | Chatbot for FAQs, raising requests and other services                                       | IBM Watson Assistant             |
| 5.    | Database                        | String, integer, long, allowed values   | MySQL or PostgreSQL              |
| 6.    | Cloud Database                  | Database Service on Cloud   | IBM DB2, IBM Cloud               |
| 7.    | External API                    | Containerize the application  | Docker, Container Registry.      |
| 8.    | Infrastructure (Server / Cloud) | Application Deployment on Local System / Cloud  | Kubernetes, Cloud Foundry        |

**Table-2: Application Characteristics:**

| <b>S. No</b> | <b>Characteristics</b>   | <b>Description</b>   | <b>Technology</b>   |
|--------------|--------------------------|--|---|
| 1.           | Open-Source Frameworks   | Open Source Backend Framework to create API Endpoints  | Python-Flask  |
| 2.           | Security Implementations | Prevents data leakage and secures medical records of the users.  | Docker content Trust (DCT), Transport Layer Security(TLS), Container registry |
| 3.           | Scalable Architecture    | Kubernetes Cluster allow containers to run across multiple machines and environments   | Kubernetes Cluster, Docker  |
| 4.           | Availability             | Kubernetes and IBM Cloud being run by multinational organizations have a very less chance of going down, hence always available.           | Kubernetes Cluster, IBM Cloud   |
| 5.           | Performance              | Kubernetes and Docker are known and used widely, even by fortune 500 companies, for their exceptional performance, all factors considered. | Kubernetes Cluster, IBM Cloud, Docker   |