

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

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

Technical Architecture:

The deliverable shall include the architectural diagram as below.

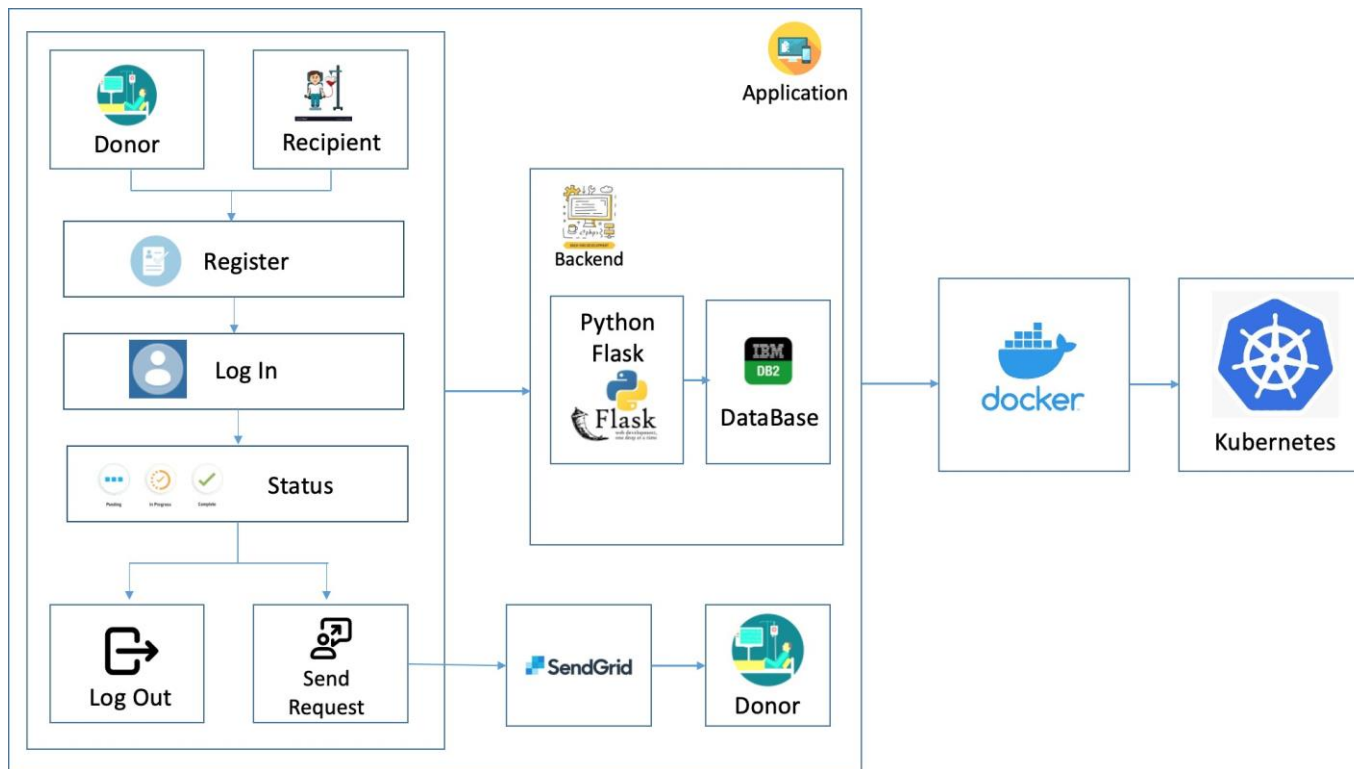


Table-1 : Components & Technologies:

| S.No | Component | Description | Technology |
|------|---------------------------------|---|---|
| 1. | User Interface | How user interacts with application e.g. Web UI, Mobile App, Chatbot etc. | HTML, CSS, JavaScript / AngularJs |
| 2. | Application Logic-1 | New User registers in the application by giving the genuine contact details which will be stored in the database. | Java,Flask,HTML,CSS |
| 3. | Application Logic-2 | Users login into the application by providing the username and password. | Flask,IBM DB2 |
| 4. | Application Logic-3 | Stats page displays the blood unit count available and the number of donors available for each blood group | IBM Watson Assistant |
| 5. | Application Logic-4 | A request page that collects the name,contact number,gender and the blood group needed.Finally the request is sent to a donor whose blood group matches with the request. | Sendgrid |
| 6. | Database | Characters,Integers,String,Long, Configurations | IBM DB2, MySQL |
| 7. | Cloud Storage | Database service on cloud | IBM DB2, IBM Block Storage or Other Storage Service or Local Filesystem |
| 8. | External API-1 | Authentication, used to store, manage and deploy container images. | Flask, Container registry |
| 9. | External API-2 | Sending request to donors | Sendgrid |
| 10. | Infrastructure (Server / Cloud) | Application Deployment | Kubernetes, cloud foundry |

Table-2: Application Characteristics:

| S.No | Characteristics | Description | Technology |
|-------------|--------------------------|--|---|
| 1. | Open-Source Frameworks | List the open-source frameworks used | Python Flask |
| 2. | Security Implementations | List all the security / access controls implemented, use of firewalls etc. | Doctor content Trust (DCT), Transport Layer Security(TLS), Container registry |
| 3. | Scalable Architecture | Justifying the scalability of architecture (3 – tier, Micro-services) Kubernetes prevents hardware problems like downtime error. | Docker, Kubernetes cluster |
| 4. | Availability | Use of load balancers, distributed servers. Kubernetes provide all time availability. | Kubernetes |
| 5. | Performance | Application performance is improved by Docker | Docker |