# PROJECT DESIGN PHASE-II
# TECHNOLOGY STACK (ARCHITECTURE & STACK)

| | |
|---|---|
| **TEAM ID** | PNT2022TMID33788 |
| **PROJECT NAME** | PLASMA DONOR APPLICATION |
| **MAXIMUM MARKS** | 4 MARKS |

**TECHNICAL ARCHITECTURE:**

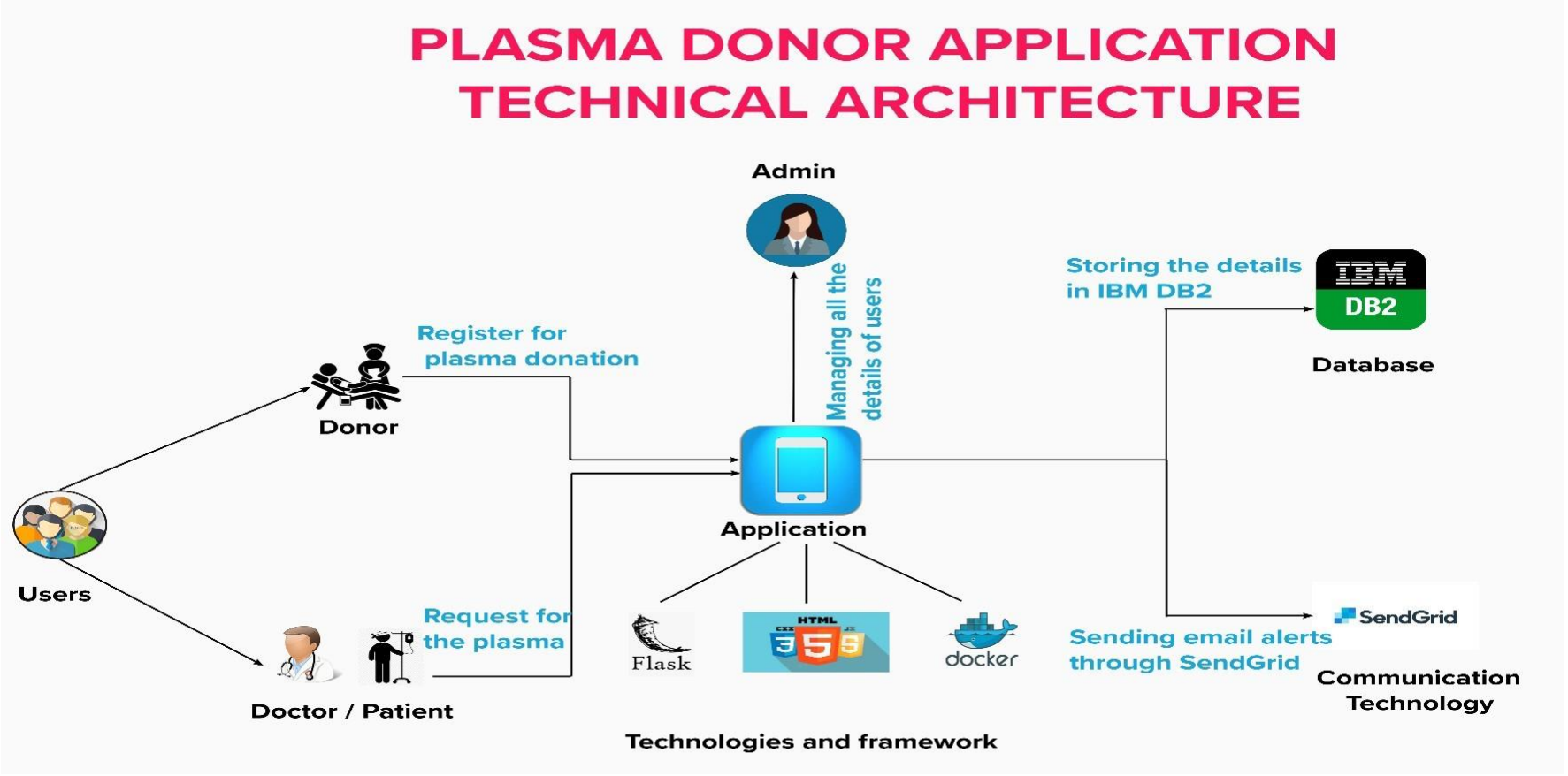

**TABLE-1: COMPONENTS & TECHNOLOGIES:**

| S.No | Component | Description | Technology |
|---|---|---|---|
| 1. | User Interface | The application interacts with the user using Captcha verification. | HTML, CSS, JavaScript. |
| 2. | Application Logic-1 | Framework used for design the application. | Python – flask |
| 3. | Application Logic-2 | Communication between users and the application via mails. | Send Grid |
| 4. | Application Logic-3 | Storing the details of the users both donors and patients. | IBM DB2 |
| 5. | Application Logic-4 | Docker is an open source platform for building, deploying, and managing containerized applications. | Docker |
| 6. | Database | Data Type, Configurations etc. | MySQL |
| 7. | Cloud Database | Database Service on Cloud | IBM DB2 |
| 8. | File Storage | File storage requirements | IBM Block Storage |
| 9. | External API-1 | They make it easier for your developers to store, manage and deploy container images. | Container Registry |
| 10. | Infrastructure (Server / Cloud) | Application Deployment on Cloud | Kubernetes |

**TABLE-2: APPLICATION CHARACTERISTICS:**

| S.No | Characteristics | Description | Technology |
|---|---|---|---|
| 1. | Open-Source Frameworks | Python – flask is an open-source framework used to develop the application. | Python -flask |
| 2. | Security Implementations | Container registry and Kubernetes Cluster are used for encryption of data. | Container registry and Kubernetes Cluster |
| 3. | Scalable Architecture | Kubernetes Cluster allow containers to run across multiple machines and environments. | Kubernetes Cluster |
| 4. | Availability | Kubernetes Cluster provides all time availability. | Kubernetes Cluster |
| 5. | Performance | Docker improves the application performance. | Docker |