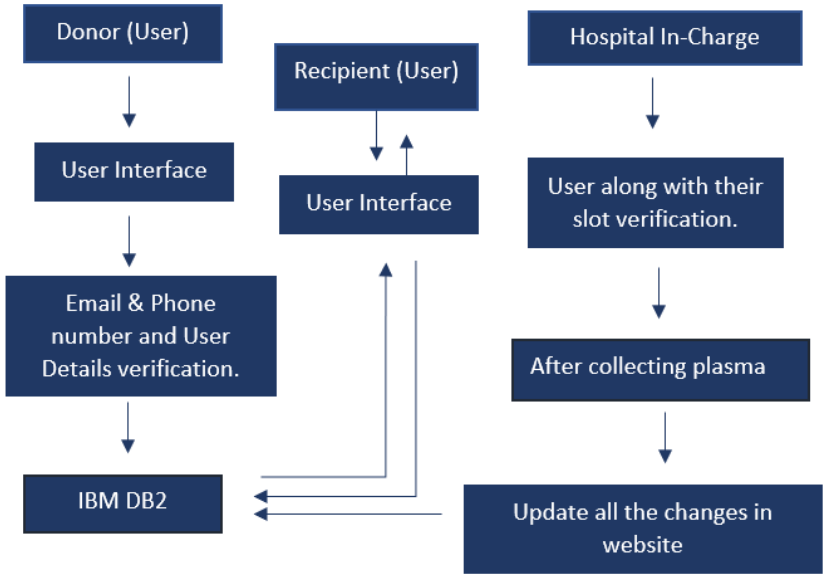
Project Design Phase-II
Technology Stack (Architecture & Stack)

Date	15 October 2022
Team ID	PNT2022TMID15338
Project Name	Plasma Donor Application
Maximum Marks	4 Marks

Technical Architecture:

The Architecture consists of storing the database on db2 and files on object storage.



Steps:

1. Donor/Recipient/Hospital In-Charge can enter their details and check their eligibility. (HTML, CSS, Bootstrap, JavaScript and Python Flask)
2. Data to be stored in Cloud database storage (IBM DB2)
3. Chat Bot for user's convenience (Watson Assistant/Chat Bot)
4. Documents and Pictures are stored in Object Storage in IBM CLOUD.

Table-1: Components & Technologies:

S.No	Component	Description	Technology
1.	User Interface	User interacts with application with three kinds of logins.	HTML, CSS, JavaScript, Bootstrap.
2.	Donor Registration and Login	Using Email/SMSs verification	Python and Python Flask, SendGrid
3.	Recipient Registration and Login	Using Email/SMSs verification	Python and Python Flask, SendGrid
4.	Hospital In-Charge Registration and Login	Using Email/SMSs verification	Python and Python Flask, SendGrid
5.	Database	Storing all the data in the database.	MySQL.
6.	Cloud Database	Database Service on Cloud	IBM DB2.
7.	File Storage	File storage requirements	IBM Cloud Object Storage
8.	Chat Bot / Assistant	Helps the users to give certain instructions based on the experience.	IBM Watson Assistant
9.	Hospital API-1	To used for provide nearby hospitals and details.	Hospitals API, etc.
10.	Aadhar API-1	To check the entered Aadhar number is valid one or not.	Aadhar API, etc.
11.	Infrastructure (Server / Cloud)	Application Deployment on System. Cloud Server Configuration: Docker / Kubernetes.	Docker, Kubernetes, etc.

Table-2: Application Characteristics:

S. No	Characteristics	Description	Technology
1.	Open-Source Frameworks	Used Web technologies for Front end and Back end.	HTML, CSS, JS, Python, Flask
2.	Security Implementations	User verification through Email Service	SendGrid
3.	Scalable Architecture	Run the app in Local and Cloud System	Docker and Kubernetes
4.	Availability	Justify the availability of application (e.g., use of load balancers, distributed servers etc.)	Docker, IBM Cloud
5.	Performance	Design consideration for the performance of the application (number of requests per sec, use of Cache, use of CDN's) etc.	IBM Cloud, Kubernetes Cluster, Container Registry.

References:

https://www.ibm.com/cloud/architecture

https://flask.palletsprojects.com/en/2.2.x/

https://www.ibm.com/products/db2

https://cloud.ibm.com/objectstorage/create

https://cloud.ibm.com/catalog/services/watson-assistant

https://docs.docker.com/

https://kubernetes.io/docs/home/

https://docs.sendgrid.com/