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

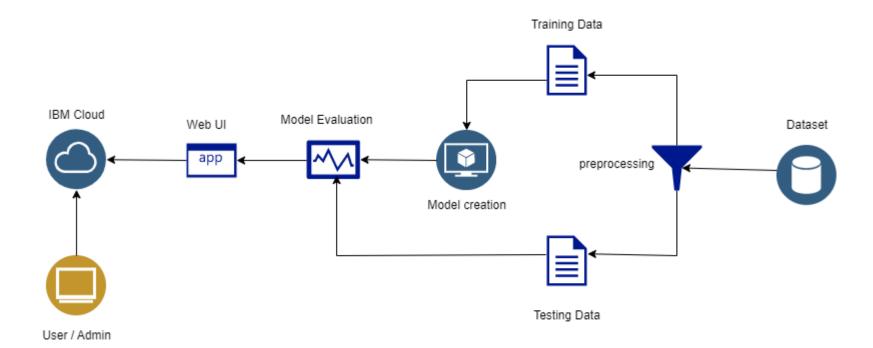| Date	15 October 2022
Team ID	PNT2022TMID21262
Project Name	Project - Early Detection of Chronic Kidney Disease using Machine Learning
Maximum Marks	4 Marks

Technical Architecture:

Table-1 : Components & Technologies:

S.No	Component	Description	Technology
1.	User Interface	User Interact with our application through web user interface	HTML, CSS , Flask , *React (Subsidiary)*
2.	Application Logic-1 (Registration)	User is redirected to register page for registering themselves by providing valid details	HTML, CSS , Flask , *React (Subsidiary)*
3.	Application Logic-2 (Login)	Once the user is registered he is now able to login to access the web service . There is an external login button to redirect to login page	HTML, CSS , Flask , *React (Subsidiary)*
4.	Application Logic-3 (Test / Prediction)	The test or prediction page is present for the user who has logged in and can predict the disease by providing input in the form.	HTML, CSS , Flask , *React (Subsidiary)*
5.	Database	Data Type - String , Numbers	MySQL
6.	Cloud Database	Database Service on Cloud	IBM DB2, IBM Cloudant etc.
7.	File Storage	File storage requirements	IBM Block Storage or Other Storage Service or Local Filesystem
8.	External API-1	Role Based Access is provided for using the API	Backend API
9.	External API-2	Purpose of External API used in the application	NIL
10.	Machine Learning Model	To predict the output based on the training and testing of the data from dataset	Data Recognition Model, etc.
11.	Infrastructure (Server / Cloud)	Application Deployment on Local System / Cloud Local Server Configuration: Cloud Server Configuration :	NIL

Table-2: Application Characteristics:

S.No	Characteristics	Description	Technology
1.	Open-Source Frameworks	Frameworks are used in both in making web app and for model creation	Flask (micro web framework) , python (web framework) , Scikit-learn (ML framework)
2.	Security Implementations	passwords are hashed for the user , as well roles are provided for access based control system	SHA 256
3.	Scalable Architecture	The Scalability can be improvised by using three-tier architecture	Three tier architecture
4.	Availability	Scalability includes availability,the service must be available even if there are more user request ,load balancer is needed to do the above task	Load Balancer
5.	Performance	Performance is key for increased revenue , handling multiple requests and expanding it can be done using Load Balancer.	Load Balancer