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

Date	20th October 2022
Team ID	PNT2022TMID51683
Project Name	Nutrition Assistant Application
Maximum Marks	4 Marks

Technical Architecture:

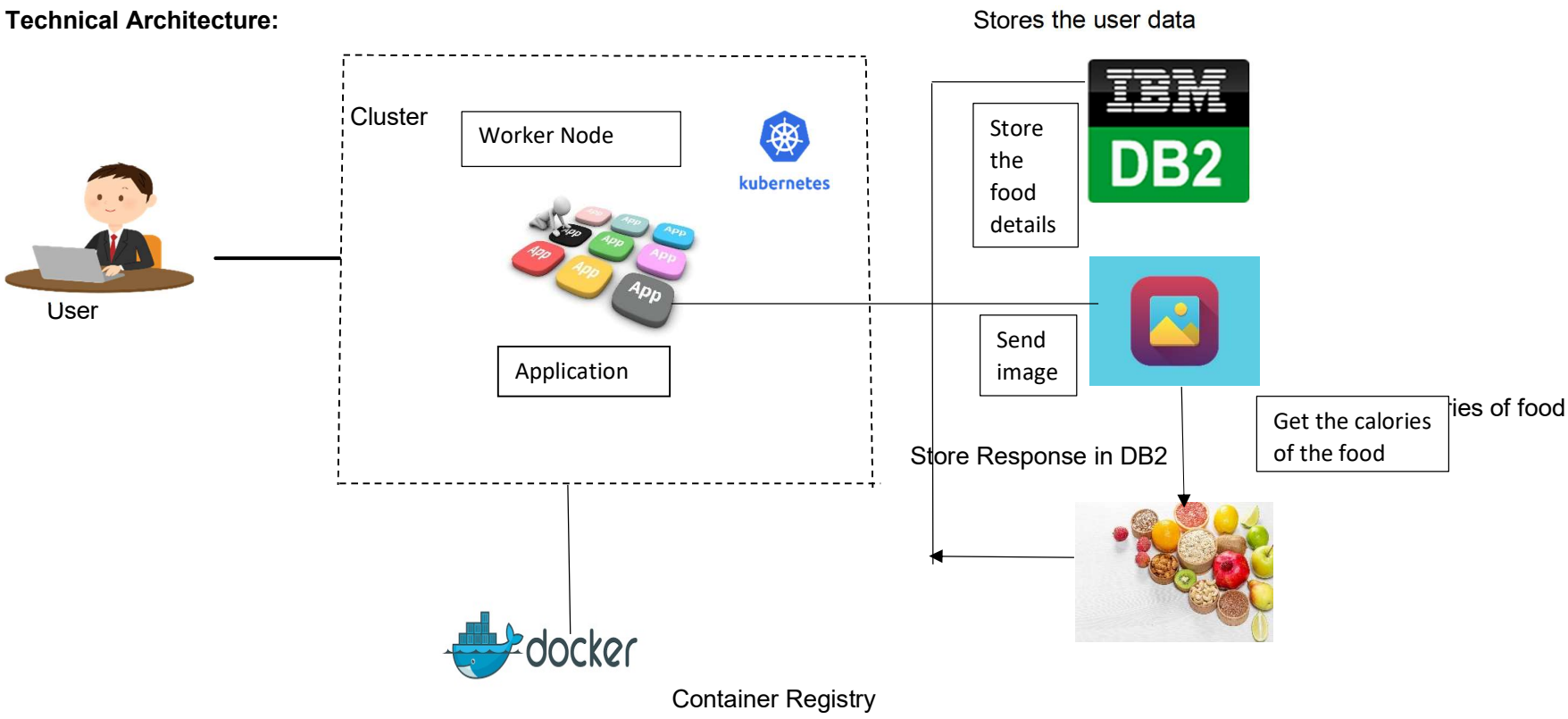

Table-1 : Components & Technologies:

S.No	Component	Description	Technology
1.	User Interface	How user interacts with application e.g. Web UI, Mobile App, Chatbot etc.	HTML, CSS, JavaScript, Flask, Python
2.	Application Logic-1: Creating an account	User registers their personal details such as name, age, current weight etc. These details are stored in the IBM cloud DB2.	Flask App running using Kubernetes Cluster, IBM DB2
3.	Application Logic-2: Logging in	Verify credentials and land the user to their home page. Perform OTP authentication if needed.	Flask App running using Kubernetes Cluster, IBM DB2
4.	Application Logic-3: Creating a custom meal	User enters details/ingredients and amount required to make a certain meal. The total calorie count is shown once the user clicks 'finish meal'.	Flask App running using Kubernetes Cluster, IBM DB2
5.	Application Logic-4: Purchasing Premium Subscription	User is redirected to payment portal to complete the purchase.	Flask App running using Kubernetes Cluster, IBM DB2
6.	Application Logic-5:Image Recognition	User can take or upload a picture to automatically detect a food item which is already available in the database.	Flask App running using Kubernetes Cluster, IBM DB2
7.	Application Logic-6: Viewing Dashboard	User can track their past records and visualise their calorie consumption and analyse their trends.	Flask App running using Kubernetes Cluster, IBM DB2
8.	Application Logic-7: Daily Reminders	Notification is sent to the User on a daily basis to remind them to add their daily consumption of food and track calories.	Flask App running using Kubernetes Cluster, IBM DB2
9.	Application Logic-8: In-App Social Network	Users can add friends and set goals together. User can also post their progress and view others progress.	Flask App running using Kubernetes Cluster, IBM DB2
10.	Application Logic-9: Blogs and Articles of Nutrition Experts	Articles and blogs by nutrition experts are added periodically to the app for users to view.	Flask App running using Kubernetes Cluster, IBM DB2
11.	Application Logic-10: Setting a daily calorie limit	User is alerted if they intake over calories.	Flask App running using Kubernetes Cluster, IBM DB2
12.	Database	Data Type, Configurations etc.	MySQL
13.	Cloud Database	Database Service on Cloud	IBM DB2,
14.	File Storage	File storage requirements	IBM Block Storage or Other Storage Service or Local Filesystem

15.	External API-1	The SendGrid service will be used to alert users of various notifications etc., as defined by the user.	SendGrid
16.	External API-2	The service will be used for image recognition.	Nutrition API
17.	Machine Learning Model	Pre trained model available through the API to recognise food items.	Object Recognition Model
18.	Infrastructure (Server / Cloud)	Application Deployment on Local System / Cloud Local Server Configuration: The application will run on the local server to allow user to interact with Web UI. Cloud Server Configuration: The application will be hosted on the cloud for the user to user. This is done through containerization of the application using Docker, stored in the container registry and will be run by Kubernetes.	IBM Cloud Registry, IBM Cloud Object Storage, IBM DB2, Docker, Kubernetes

Table-2: Application Characteristics:

S.No	Characteristics	Description	Technology
1.	Open-Source Frameworks	Flask	Backend Framework built on python.
2.	Security Implementations	TLS- Transport Layer Security	All external communications is encrypted and user data is protected.
3.	Scalable Architecture	The containerised app can be scaled easily to a large user base.	IBM DB2, IBM Cloud Object Storage, Kubernetes
4.	Availability	IBM Cloud guarantees availability with very minimal downtime. The app loses minimal functionality if the external APIs are not available.	IBM Cloud Object Storage, Kubernetes, Docker Images, IBM DB2, SendGrid
5.	Performance	Performance depends on the availability of compute power in the cloud.	IBM Cloud Object Storage, Kubernetes, Docker Images, IBM DB2, SendGrid