Project Design Phase-II

Technology Stack(Architecture & Stack)

Date	18 NOV 2022
Team ID	PNT2022TMID39006
Project Name	Nutrition Assistant Application
Maximum Marks	4 Marks

Technical Architecture:

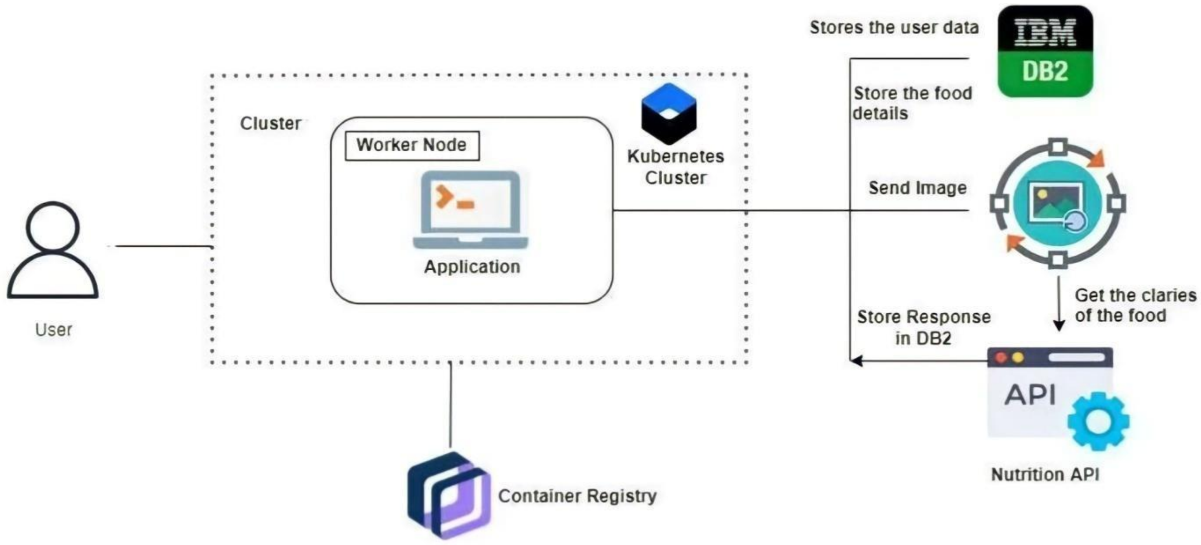

Table-1 : Components & Technologies:

S.No.	Component	Description	Technologies
1.	User Interface	How user interacts with application e.g. Web UI, Mobile App, Chatbot etc.	HTML, CSS, Javascript, Flask, Python
2.	Application Logic	Logic for a process in the application.	Python

3.	Database	Data Type, Configurations etc.	MySQL
4.	Cloud Database	Database Service on Cloud.	IBM DB2, IBM Cloudant
5.	File Storage	File storage requirements.	IBM Block Storage or Other Storage Service or Local File system
6.	External API-1-SendGrid	The SendGrid service will be used to alert users of various notifications etc as defined by the user.	SendGrid
7.	External API-2-NutritionAPI	The service will be used for image recognition.	NutritionAPI
8.	Machine Learning Model	Pre trained model available through the API to recognize food items.	Object Recognition Model
9.	Deployment	Application Deployment on Local System / Cloud Local Server Configuration: It will run on the local server/client side to allow user to interact with Web UI. Cloud Server Configuration: It 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	Technologies
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	SHA-256, Encryptions, IAM Controls
3.	Scalable Architecture	Justify the scalability of architecture (3 – tier, Micro-services)	IBM DB2, IBM Cloud Object Storage, Kubernetes
4.	Availability	Justify the availability of application (e.g. use of load balancers, distributed servers etc.)	Justify the availability of application (e.g. use of load balancers, distributed servers etc.)
5.	Performance	Performance depends on the availability of compute power in the cloud.	IBM Cloud Object Storage, Kubernetes, Docker Images, IBM DB2, SendGrid