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

Date	03 October 2022
Team ID	PNT2022TMID44856
Project Name	Project - AI-Powered Nutrition Analyzer For Fitness Enthusiasts
Maximum Marks	4 Marks

Technical Architecture:

The Deliverable shall include the architectural diagram as below and the information as per the table1 & table 2

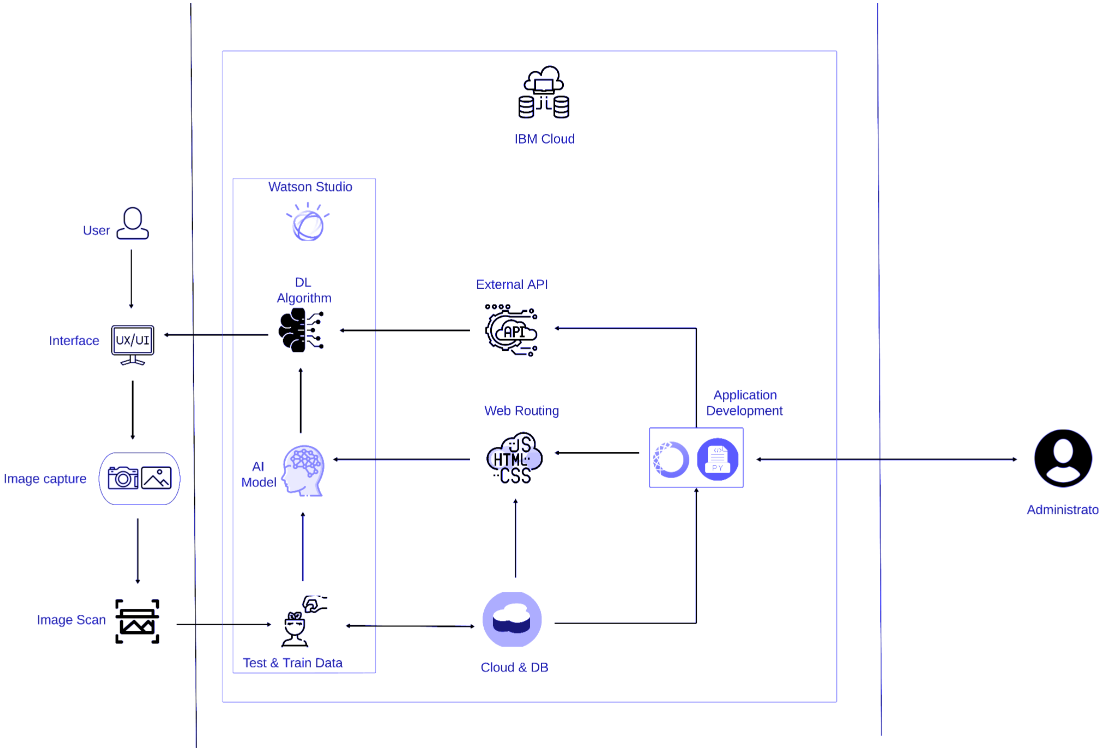

Table-1 : Components & Technologies:

S.No	Component	Description	Technology
1.	User Interface	How the user interacts with the application to enable identification of images given by the user. e.g. Web UI, Mobile App, Chatbot etc.	HTML, CSS, JavaScript / Angular Js / React Js ,Flask etc.
2.	Application Logic-1	Create HTML pages and build python code.	CSS, JS, Flask, JupyterLab, Jupyter Notebook, QtConsole,
3.	Application Logic-2	Creating a Flask application and loading the model by importing libraries.	IBM Watson Assistant, Cloudant DB, Flask
4.	Application Logic-3	Routing to HTML page for web application	Python Flask, RapidAPI, POST method, Conda
5.	Application Logic-4	Run the application.	Anaconda Prompt,IBM Cloudant
5.	Database	Image Data, Data Type, Configurations etc.	Cloudant, NoSQL, etc.
6.	Cloud Database	Database Service on Cloud	IBM DB2, IBM Cloudant, NoSQLetc.
7.	File Storage	File storage requirements	IBM Block Storage
8.	External API-1	Purpose of External API used in the application	IBM Weather API, etc.
9.	External API-2	Purpose of External API used in the application	Rapid API etc.
10.	Machine Learning Model	Purpose of Machine Learning Model	Object Recognition Model, etc.
11.	Infrastructure (Server / Cloud)	Application Deployment on Local System / Cloud Local Server Configuration: Cloud Server Configuration :	Local, Cloud Foundry, Kubernetes, etc.

Table-2: Application Characteristics:

S.No	Characteristics	Description	Technology
1.	Open-Source Frameworks	List the open-source frameworks used	Flask,
2.	Security Implementations	List all the security / access controls implemented, use of firewalls etc.	e.g. SHA-256, Encryptions, IAM Controls, OWASP etc.
3.	Scalable Architecture	Justify the scalability of architecture (3 – tier, Micro-services)	Technology used
4.	Availability	Justify the availability of applications (e.g. use of load balancers, distributed servers etc.)	Technology used
5.	Performance	Design consideration for the performance of the application (number of requests per sec, use of Cache, use of CDN’s) etc.	Technology used