

# **AI-Powered Nutrition Analyzer For Fitness Enthusiasts**

## **IBM PROJECT REPORT**

**TEAM ID – PNT2022TMID46130**

**Submitted by**

**KRISHNASAMY R      (815119104023)**

**SAKTHIVEL S      (815119104039)**

**SATHISHKUMAR S      (815119104040)**

**SHAHEEN AHMAD S      (815119104042)**

**RANJITH R      (815119104034)**

in partial fulfillment for the award of the degree of,

**BACHELOR OF ENGINEERING IN**

**COMPUTER SCIENCE AND ENGINEERING**

**DHANALAKSHMI SRINIVASAN INSTITUTE OF TECHNOLOGY-TRICHY**

**ANNA UNIVERSITY: CHENNAI 600 025**

# **TABLE OF CONTENTS**

## **1. INTRODUCTION**

- 1.1 Project Overview
- 1.2 Purpose

## **2. LITERATURE SURVEY**

- 2.1 Existing problem
- 2.2 References
- 2.3 Problem Statement Definition

## **3. IDEATION & PROPOSED SOLUTION**

- 3.1 Empathy Map Canvas
- 3.2 Ideation & Brainstorming
- 3.3 Proposed Solution
- 3.4 Problem Solution fit

## **4. REQUIREMENT ANALYSIS**

- 4.1 Functional requirement
- 4.2 Non-Functional requirements

## **5. PROJECT DESIGN**

- 5.1 Data Flow Diagrams
- 5.2 Solution & Technical Architecture
- 5.3 User Stories

## **6. PROJECT PLANNING & SCHEDULING**

- 6.1 Sprint Planning & Estimation

## **7. CODING & SOLUTION**

- 7.1 Feature 1
- 7.2 Feature 2

## **8. TESTING**

8.1 Test Cases

8.2 User Acceptance Testing

## **9. RESULTS**

9.1 Performance Metrics

9.2 Software Requirements

9.3 Packages

## **10. ADVANTAGES & DISADVANTAGES**

## **11. CONCLUSION**

## **12. FUTURE SCOPE**

## **13. APPENDIX**

Source Code

GitHub & Project Demo Link

# **1.INTRODUCTION**

Food is essential for human life and has been the concern of many healthcare conventions. Nowadays new dietary assessment and nutrition analysis tools enable more opportunities to help people understand their daily eating habits, exploring nutrition patterns and maintain a healthy diet. Nutritional analysis is the process of determining the nutritional content of food. It is a vital part of analytical chemistry that provides information about the chemical composition, processing, quality control and contamination of food.

The main aim of the project is to building a model which is used for classifying the fruit depends on the different characteristics like colour, shape, texture etc. Here the user can capture the images of different fruits and then the image will be sent the trained model. The model analyses the image and detect the nutrition based on the fruits like (Sugar, Fibre, Protein, Calories,Etc..,).

## **1.1.PROJECT OVERVIEW**

1. Should have a thorough knowledge of picture data as well as the basic principles and methods of convolutional neural networks.
2. To use various data pre-processing techniques to preprocess/clean the data.
3. Utilize the Flask framework to create a web application. The user interacts with the user interface and inputs the image. The input image is then provided to our flask programme, where we categorise the result using the model we built and display it on the user interface.

## **1.2.PURPOSE**

The complicated and non-linear relationship between nutritionrelated data and health outcomes may be better understood and predicted with the use of AI algorithms. These nutrition analysis tools provide consumers additional chances to grasp their regular eating behaviours, explore nutrition trends, and keep up a healthy diet.

## 2.LITERATUTE SURVEY

### AI-POWERED NUTIRITION ANALYZER FOR FITNESS ENTHUSIASTS

#### LITERATURE SURVEY

SL.NO	TITLE	TECHNIQUES	MERITS	DEMERITS
1	HealthifyMe	HealthifyMe is a leading Indian health and fitness app whose artificial intelligence powered virtual nutritionist, Ria, helps its users regarding their queries around fitness and nutrition in both audio and text in more than 10 languages.	HealthifyMe's 250 million tracked foods, workouts and 10 million message exchanges between coaches and clients. HealthifyMe supposedly owns the largest data set in this regard and are compatible with popular fitness wearables devices currently available in India	Worst fitness app and services ever provided.  Don't value the customer time.  Don't even have knowledgeable dietician.
2	Neutrino	Neutrino delivers nutrition-based data services and analytics to its users and wants to turn into a leading source of the nutrition-related platform. The platform employs NLP and mathematical models from the optimization theory as well as predictive analysis to enable individualized data compilation.	It further enables its partners to purchase data regarding nutrition, using API and SDK integrations, to help improve their product offering and services.	They are only weakly interaction, so it takes about a light year of lead to stop a beam of moderately low energy (-MeV) neutrinos.
3	FitGenie	The app heavily relies on AI to produce customised data regarding calorie intake and make food suggestions accordingly.	Their advanced diet analysis and combines tools of calorie counter with to make dynamic and adaptive macronutrient adjustments thus providing high-quality nutrient plan each week for its users which is generated from its 1+ million foods.	Don't value the customer time.  Don't even have knowledgeable dietician.

REFERAL LINK: <https://analyticsindiamag.com/5-ai-powered-nutrition-apps-that-help-fitness-enthusiasts-with-their-calorie-intake/>

## 2.1.EXISTING PROBLEM

One of the most bothersome nutrition issues people face is a loss of appetite or decreased hunger caused by people who are unaware of the nutritional content of the foods they are consuming. This leads to nutritional disorders, diseases connected to nutrition, and illnesses that affect people. They consist of dietary excesses or deficiencies, obesity, eating disorders, and chronic illnesses like diabetes mellitus.

## 2.2.REFERENCES

[1] KB, Mirra, and R. Rajakumari. "Classification of Fruits Using Deep Learning Algorithms." Available at SSRN 4068366.

[2] Khatun, Mehenag, et al. "Fruits Classification using Convolutional Neural Network." GRD Journals-Global Research and Development Journal for Engineering 5.8 (2020).

[3] Kumar Saini, D.; rabbi, S.; Chhabra, D.; Shukla, P. Phycobiliproteins from *Anabaena variabilis* CCC421 and its production enhancement strategies using combinatorial evolutionary algorithm approach. *Bioresour. Technol.* 2020, 309, 123347.

[4] Rozga M, Latulippe ME, Steiber A. "Advancements in personalized nutrition technologies: guiding principles for registered dietitian nutritionists" *Jornal of the academy of nutrition and dietetics*.

[5] Oka, R.; Nomura, A.; Kometani, M.; Gondoh, Y.; Yoshimura, K.; Yoneda, T. Study protocol for the effects of artificial intelligence { AI } – supported automated nutritional intervention on glycemic control inpatients with 2 diabetes mellitus

## 2.3.PROBLEM STATEMENT

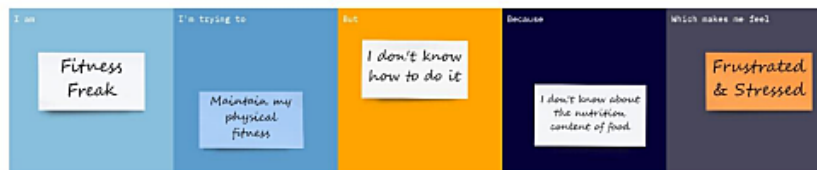
### AI-POWERED NUTRITION ANALYZER FOR FITNESS ENTHUSIASTS

#### PROBLEM STATEMENTS

##### PROBLEM STATEMENT 1



##### PROBLEM STATEMENT 2

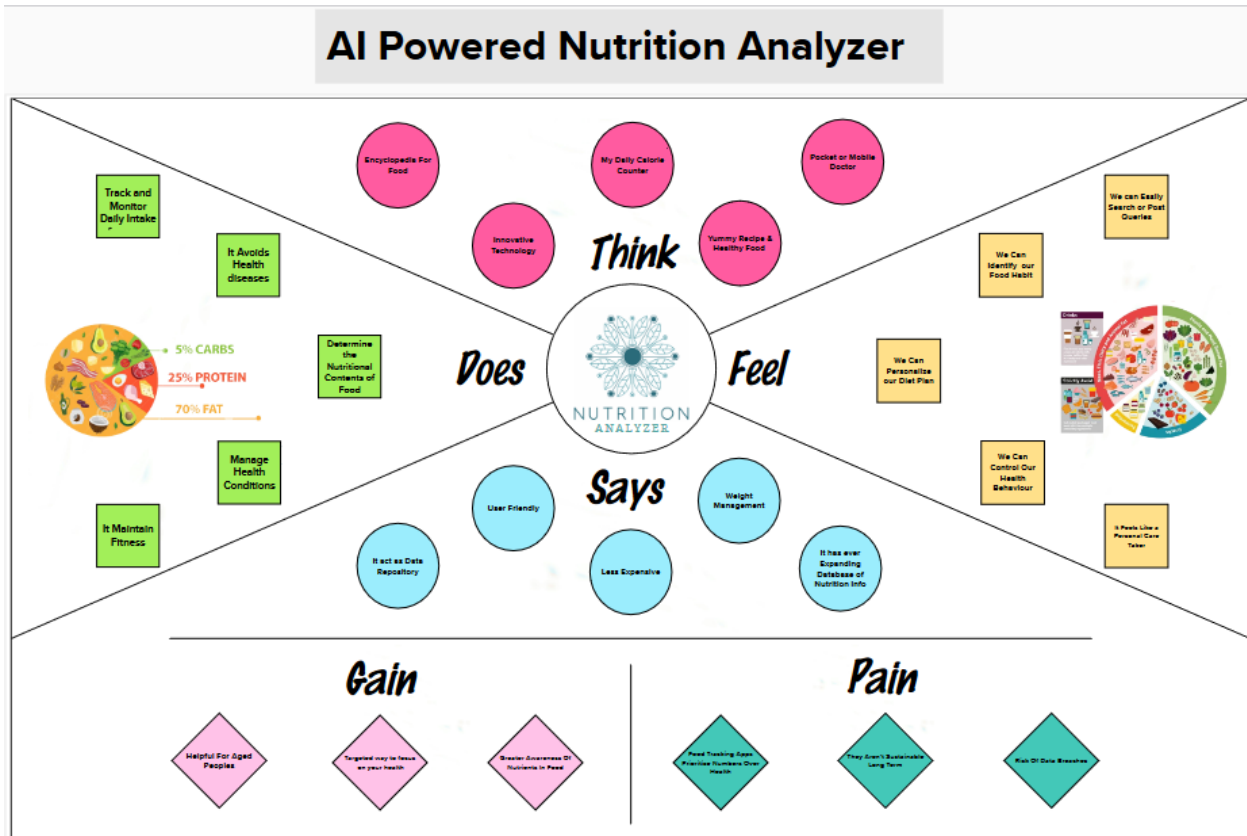


Problem Statement (PS)	I am (Customer)	I'm trying to	But	Because	Which makes me feel
PS – 1	Sportsmen	Maintain my physique	I'm unable to concentrate on the nutrition diet	I have more commitments and complexity	Tired & Exhausted
PS – 2	Fitness Freak	Maintain my physical fitness	I don't know how to do it	I don't know about the nutrition content of food	Frustrated & Stressed

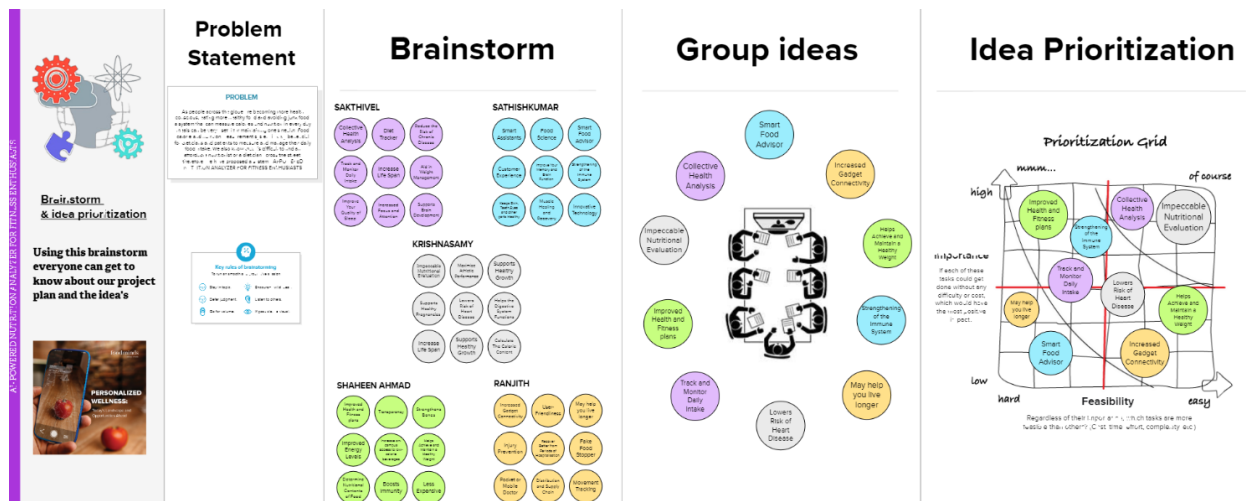
---

### 3.IDEATION & PROPOSED SOLUTION

#### 3.1.EMPATHY MAP CANVAS



#### 3.2.IDEATION & BRAINSTORMING





### 3.3.PROPOSED SOLUTION

#### AI-POWERED NUTRITION FOR FITNESS AND ENTHUSIAST

##### PROPOSED SOLUTION

S.No.	Parameter	Description
1.	Problem Statement (Problem to be solved)	<p>Poor eating habits include under- or over-eating, not having enough of the healthy foods we need each day, or consuming too many types of food and drink, which are low in fiber or high in fat, salt and/or sugar.</p> <p>These unhealthy eating habits can affect our nutrient intake, including energy (or kilojoules) protein, carbohydrates, essential fatty acids, vitamins and minerals as well as fiber and fluid.</p>
2.	Idea / Solution description	<p><b>Use AI-Powered Nutrition For Fitness &amp; Enthusiast Application</b></p> <p>Have a good variety of healthy foods from the five food groups each day. For more information see the Healthy eating for different ages and stages and Healthy Eating tips sections.</p> <p>Aim for two serves of fruit and five serves of vegetables each day. Only occasionally eat sugary, fatty or salty food, and then only in small amounts. Drink fresh, clean tap water instead of sugary drinks.</p> <p>Switch over to healthy recipes that look and taste good. Plan your meals ahead and shop for healthy ingredients. Enjoy cooking and eating healthy food with family or friends and without distractions such as the television.</p>
3.	Novelty / Uniqueness	<p>Nutrition can help enhance athletic performance. An active lifestyle and exercise routine, along with eating well, is the best way to stay healthy. Eating a good diet can help provide the energy you need to finish a race, or just enjoy a casual sport or activity.</p> <p>You are more likely to be tired and perform poorly during sports when you do not get</p>

		<p>enough:</p> <ul style="list-style-type: none"> <li>• Calories</li> <li>• Carbohydrates</li> <li>• Fluids</li> <li>• Iron, vitamins, and other minerals</li> <li>• Protein</li> </ul>
4.	Social Impact / Customer Satisfaction	<p>Largest database available in a diet tracker includes many restaurant foods.</p> <p>Users can download recipes online and calculate the calorie content of each serving.</p> <p>Ability to quick add calories when you don't have time to log the whole meal.</p>
5.	Business Model (Revenue Model)	<p>First, it is a lifestyle tracker calorie counter.</p> <p>Second, is their social feed, which allows the user to find their kind of people with similar goals and problems.</p> <p>The third, and most innovative aspect is technology augmented coaching, which uses a combination of artificial intelligence and human trainers to help users stay healthy and engage with the app.</p>
6.	Scalability of the Solution	<p>The modern AI-based apps are your fitness assistants. They instantly connect you to a nutritionist, provide customized workouts based on your health data, and offer solutions for all your fitness needs.</p> <ol style="list-style-type: none"> <li>1. Collective health analysis</li> <li>2. Improved health and fitness plans</li> <li>3. Increased gadget connectivity</li> <li>4. Smart Assistants</li> <li>5. Movement tracking</li> <li>6. Impeccable nutritional evaluation</li> </ol>

### 3.4.PROBLEM SOLUTION FIT



## 4.REQUIREMENT ANALYSIS

### 4.1 FUNCTIONAL REQUIREMENTS

#### Functional Requirement :

Following are the functional requirement for the proposed solution

Fr.no	Functional Requirement	Sub Requirement(Story/subtask)
Fr-1	User Registration	Registration through Form Registration through Gmail
Fr-2	User Confirmation	Confirmation via OTP Confirmation via Gmail
Fr-3	Capturing Image	Capture the image of the food And check the parameter captured image
Fr-4	Image Processing	Upload the image for the prediction of the nutrition in the food
Fr-5	Food Identification	Identify the food and predict the nutrition in the food
Fr-6	Image Description	Suggesting the best food with enriched nutrition to Eat

## 4.2.NON-FUNCTIONAL REQUIREMENTS

### Non-Functional Requirement :

Following are the Non-Functional requirement for the proposed solution

Nfr.no	Non-Functional Requirement	Description
Nfr-1	Usability	Datasets of all the food items is used to detecting the nutrition that present in the food
Nfr-2	Security	The Information belongs to the user and Nutrition are secured highly
Nfr-3	Reliability	The Image Quality of food is important for predicting the Nutrition in the food
Nfr-4	Performance	The performance is based on the Image Quality of food used for nutrition prediction
Nfr-5	Availability	It is available for all user to predict the nutrition in the food
Nfr-6	Scalability	Increasing the prediction of the Nutrition in the food

## 5.PROJECT DESIGN

### 5.1.DATA FLOW DIAGRAM

The classic visual depiction of how information moves through a system is a data flow diagram (DFD). A tidy and understandable DFD may visually represent the appropriate quantity of the system demand. It demonstrates how information enters and exits the system, what modifies the data, and where information is kept.

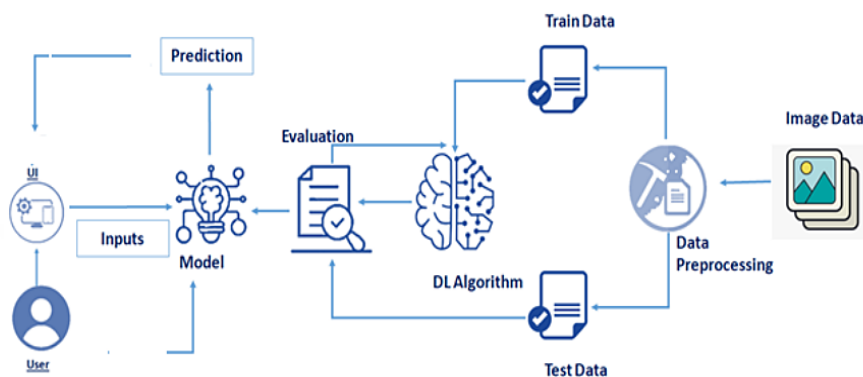
The Data Flow Diagram is presented below:

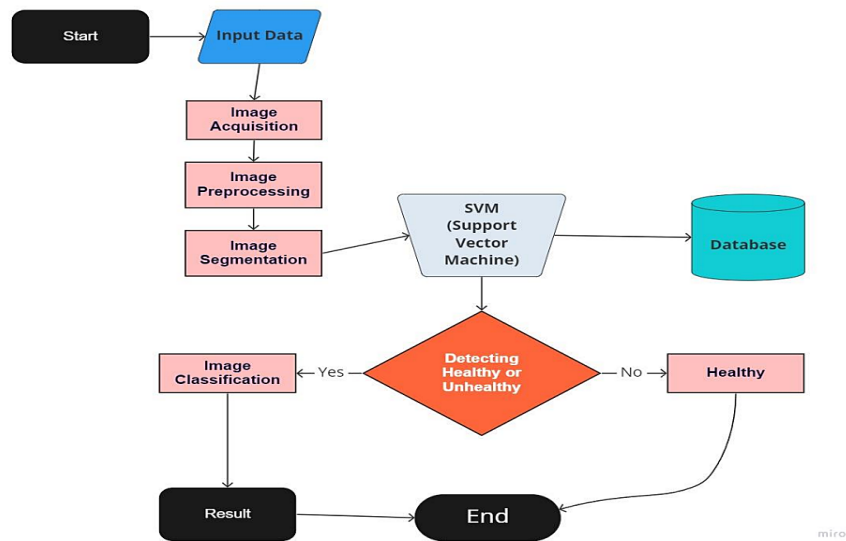
**AI-powered Nutrition Analyzer for Fitness Enthusiasts**  
**Data Flow Diagram & User Stories**

Date	28 October 2022
Team ID	PNT2022TMID46130
Project Name	AI-powered Nutrition Analyzer for Fitness Enthusiasts
Maximum Marks	4 Marks

#### Data Flow Diagrams:

A Data Flow Diagram (DFD) is a traditional visual representation of the information flows within a system. A neat and clear DFD can depict the right amount of the system requirement graphically. It shows how data enters and leaves the system, what changes the information, and where data is stored.





#### User Stories:

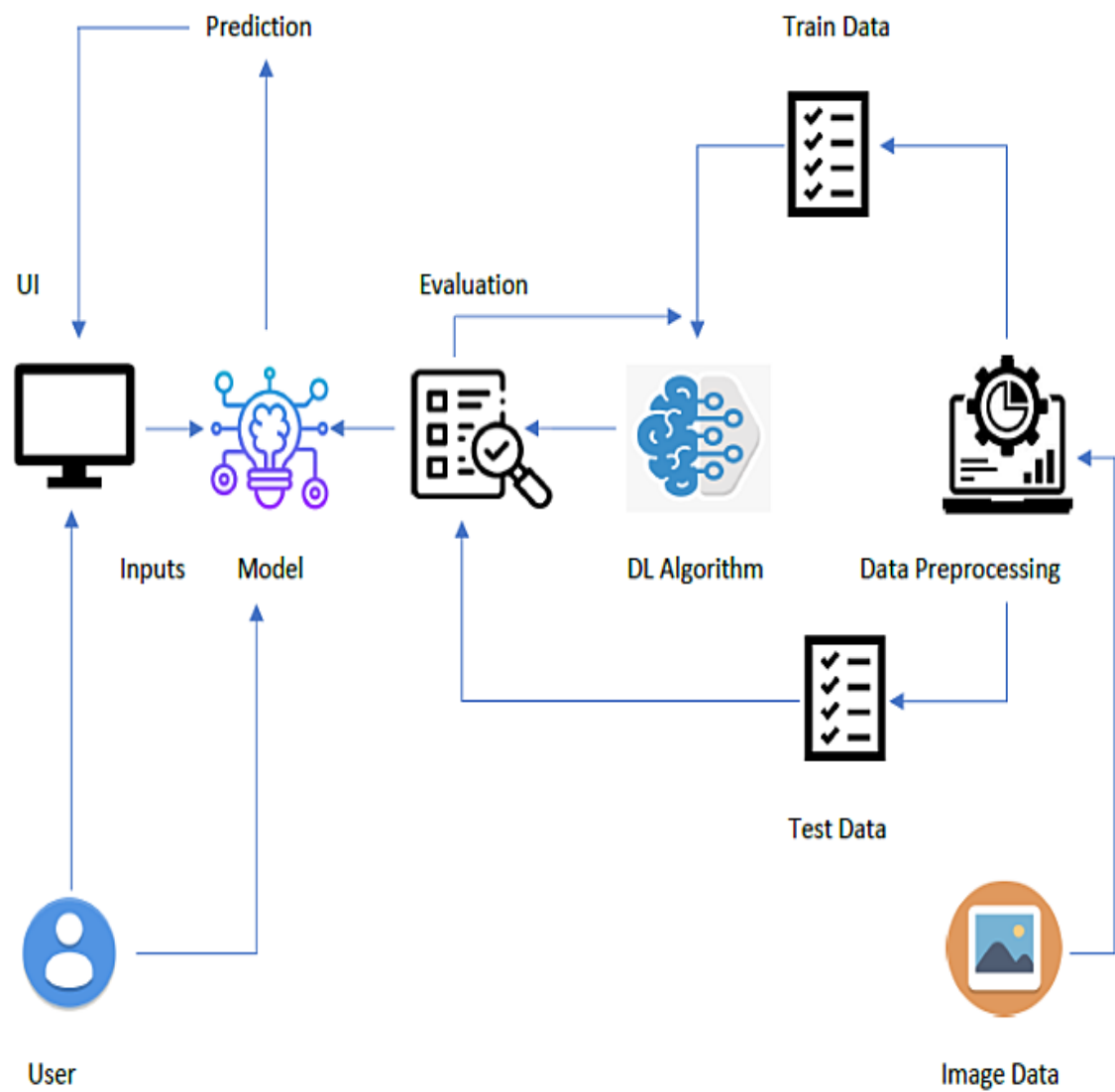
User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority	Release
Customer (Web user)	Registration	USN-10	As a user, I can register for the application by entering my email, password, and confirming my password.	I can access my account / dashboard	High	Sprint-1
		USN-11	As a user, I will receive confirmation email once I have registered for the application	I can receive confirmation email & click confirm	High	Sprint-1
		USN-12	As a user, I can register for the application through Face book	I can register & access the dashboard with Face book Login	Low	Sprint-3
		USN-13	As a user, I can register for the application through Gmail	I can register and access the dashboard with Gmail login	Medium	Sprint-1
	Login	USN-11	As a user, I can log into the application by entering email & password	I can register and access the dashboard with email and password	High	Sprint-1
	Nutrition Analyzer	USN-15	As a user, I can view the nutrition level of the fruit by scanning the fruit	I can view the nutrition level of the fruit by scanning the fruit	High	Sprint-2
		USN-16	As a user, I can able to calculate the nutritional value of the different fruits at the same time	I can able to calculate the nutritional value of the different fruits at the same time	High	Sprint-2
		USN-17	As a user, I can able to view fruits (input data) details by related web site	I can able to view fruits (input data) details by related web site	Medium	Sprint-3
Customer Care Executive	Customer Support	USN-18	As a customer care executive, I can view the complaints on chat box	I can view the complaints on chat box	Medium	Sprint-4
		USN-19	As a customer, I should be able solve and reply for the customers queries	I can reply to customer queries in the chat thread	Low	Sprint-4

		USN-20		I can close the complaint after assisting	Low	Sprint-4
Administrator		USN-21	As a Administrator, I would take care of registrations and maintenance of accounts	I can take care of registrations and maintenance of accounts	High	Sprint-3
		USN-22	As a Administrator, I Would resolve issues on Access	I can resolve issues in Access	High	Sprint-2

## 5.2.SOLUTION AND TECHNOLOGY ARCHITECTURE

### SOLUTION ARCHITECTURE

Solution Architecture – AI Powered Nutrition Analyzer For Fitness Enthusiasts





# TECHNICAL ARCHITECTURE

**Table 1: Components and Technologies**

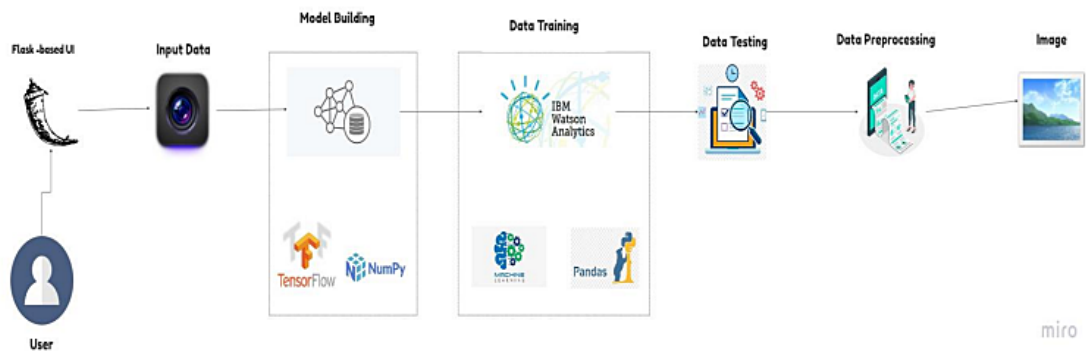
S. No	Component	Description	Technology
1.	User Interface	How the user interacts with application. e.g. Web UI	HTML and CSS
2.	Application Logic-1	Handle all the user requests done through the Web UI / Display the results after process	Python Flask Server
3.	Application Logic-2	Process the image provided by the user via Web UI	Python
4.	Application Logic-3	Train the model and provide the classification result for the image given as input	IBM Watson Studio
5.	Cloud Database	Database Service on Cloud	IBM DB2, IBM Cloudant etc.
6.	File Storage	File storage requirements	Local Filesystem
7.	Machine Learning Model	Purpose of Machine Learning Model	VGG16 Pre-Trained Model
8.	Infrastructure (Server / Cloud)	Application Deployment on Local System	Local

**Table-2: Application Characteristics:**

S. No	Characteristics	Description	Technology
1.	Open-Source Frameworks	List the open-source frameworks used	Flask, TensorFlow, Keras , NumPy, OpenCV
2.	Security Implementations	List all the security / access controls implemented, use of firewalls etc.	IAM Controls

3.	Scalable Architecture	Justify the scalability of architecture (3 – tier, Micro-services)	3-tier type (Web server, App server and DB server).
4.	Availability	Justify the availability of application (e.g. use of load balancers, distributed servers etc.)	Local: Available based on computer's specs. Cloud: Web server, DB server available when requested. App server requires high requirements compared with other 2 servers, thereby availability is bit less but can be compensated by cloud
5.	Performance	Design consideration for the performance of the application (number of requests per sec, use of Cache, use of CDN's) etc.	Accuracy of model: >85% (Expected) Number of requests per second: 250 – 1000 (based on network traffic, 250 is default as targeted user group is moderate)

### Technical Architecture:



## 5.3.USER STORIES

### User Stories:

User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority	Release
Customer (Web user)	Registration	USN-10	As a user, I can register for the application by entering my email, password, and confirming my password.	I can access my account / dashboard	High	Sprint-1
		USN-11	As a user, I will receive confirmation email once I have registered for the application	I can receive confirmation email & click confirm	High	Sprint-1
		USN-12	As a user, I can register for the application through Face book	I can register & access the dashboard with Face book Login	Low	Sprint-3
		USN-13	As a user, I can register for the application through Gmail	I can register and access the dashboard with Gmail login	Medium	Sprint-1
	Login	USN-11	As a user, I can log into the application by entering email & password	I can register and access the dashboard with email and password	High	Sprint-1
	Nutrition Analyzer	USN-15	As a user, I can view the nutrition level of the fruit by scanning the fruit	I can view the nutrition level of the fruit by scanning the fruit	High	Sprint-2
		USN-16	As a user, I can able to calculate the nutritional value of the different fruits at the same time	I can able to calculate the nutritional value of the different fruits at the same time	High	Sprint-2
		USN-17	As a user, I can able to view fruits (input data) details by related web site	I can able to view fruits (input data) details by related web site	Medium	Sprint-3
Customer Care Executive	Customer Support	USN-18	As a customer care executive, I can view the complaints on chat box	I can view the complaints on chat box	Medium	Sprint-4
		USN-19	As a customer, I should be able solve and reply for the customers queries	I can reply to customer queries in the chat thread	Low	Sprint-4

		USN-20		I can close the complaint after assisting	Low	Sprint-4
Administrator		USN-21	As a Administrator, I would take care of registrations and maintenance of accounts	I can take care of registrations and maintenance of accounts	High	Sprint-3
		USN-22	As a Administrator, I Would resolve issues on Access	I can resolve issues in Access	High	Sprint-2

## **6.PROJECT PLANNING & SCHEDULING**

### **6.1.SPRINT PLANNING & ESTIMATION**

#### **SPRINT PLANNING**

Food is a necessity for human life and has been addressed in several medical conventions. Modern dietary evaluation and nutrition analysis technologies provide consumers additional possibilities to explore nutrition patterns, comprehend their daily eating habits, and keep up a balanced diet. Finding out a food's nutritional value is done through nutritional analysis. Information on the chemical make-up, processing, quality assurance, and contamination of food is a crucial component of analytical chemistry.

The primary goal of the project is to develop a model that will be used to categorise fruits according to their many attributes, such as colour, shape, and texture. Here, users may take pictures of various fruits, which are subsequently uploaded to a trained algorithm for analysis. The model examines the picture to determine the nutrient content of fruits

#### **ESTIMATION**

1. Planning is a crucial role in project management because it allows team members to schedule their time on the project.

2. This activity demonstrates how the team members assigned and completed various tasks.

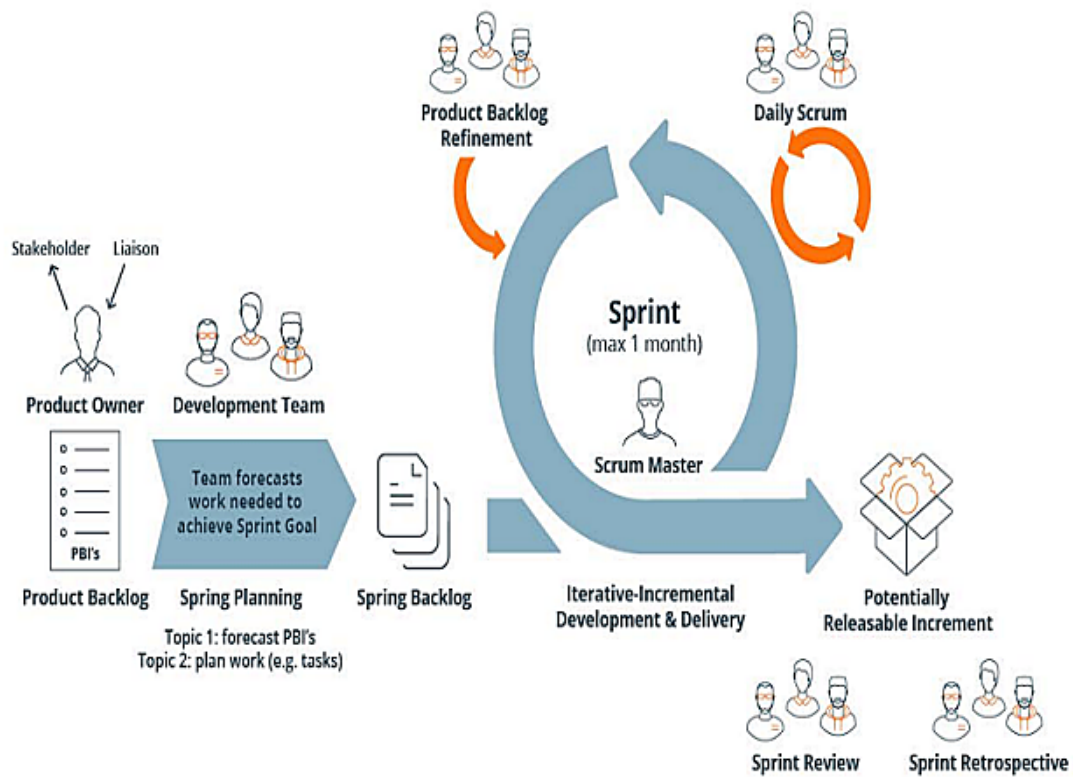
3. In Project we can Split into the Four Step of Phases are

Phase 1: Data Collection & retrieval and Requirement Analysis

Phase 2: Project Planning and Developing Modules

Phase 3: Implementing the High Accuracy Deep Learning Algorithm to  
Perform the give task smoothly.

Phase 4: Deploying the Model on Cloud and Testing the Mode



## 7.CODING & SOLUTION

### 7.1.FEATURE 1[MODEL BUILDING]

#### 1. Importing The Model Building Libraries

##### ### Importing Neccessary Libraries

```
import numpy as np#used for numerical analysis
import tensorflow #open source used for both ML and DL for computation
from tensorflow.keras.models import Sequential #it is a plain stack of layers
from tensorflow.keras import layers #A layer consists of a tensor-in tensor-out computation function
#Dense Layer is the regular deeply connected neural network layer
from tensorflow.keras.layers import Dense,Flatten
#Flatten-used for flattening the input or change the dimension
from tensorflow.keras.layers import Conv2D,MaxPooling2D,Dropout #Convolutional layer
#MaxPooling2D-for downsampling the image
from keras.preprocessing.image import ImageDataGenerator
```

#### 2. Initializing The Model

Keras has 2 ways to define a neural network:

- Sequential
- Function API

The Sequential class is used to define linear initializations of network layers which then, collectively, constitute a model. In our example below, we will use the Sequential constructor to create a model, which will then have layers added to it using the add() method.

```
model=Sequential()
```

### 3. Adding CNN Layers

- As the input image contains three channels, we are specifying the input shape as (64,64,3).
- We are adding a two convolution layer with activation function as "relu" and with a small filter size (3,3) and the number of filters (32) followed by a max-pooling layer.
- Max pool layer is used to down sample the input.(Max pooling is a pooling operation that selects the maximum element from the region of the feature map covered by the filter)
- Flatten layer flattens the input. Does not affect the batch size.

#### ### Creating the model

```
# Initializing the CNN
classifier = Sequential()

# First convolution layer and pooling
classifier.add(Conv2D(32, (3, 3), input_shape=(64, 64, 3), activation='relu'))
classifier.add(MaxPooling2D(pool_size=(2, 2)))

# Second convolution layer and pooling
classifier.add(Conv2D(32, (3, 3), activation='relu'))

# input_shape is going to be the pooled feature maps from the previous convolution layer
classifier.add(MaxPooling2D(pool_size=(2, 2)))

# Flattening the layers
classifier.add(Flatten())
```

### 4. Adding Dense Layers

A dense layer is a deeply connected neural network layer. It is the most common and frequently used layer.

```
# Adding a fully connected layer
classifier.add(Dense(units=128, activation='relu'))
classifier.add(Dense(units=5, activation='softmax')) # softmax for more than 2
```

The number of neurons in the Dense layer is the same as the number of classes in the training set. The neurons in the last Dense layer, use softmax activation to convert their outputs into respective

probabilities.

Understanding the model is a very important phase to properly using it for training and prediction purposes. Keras provides a simple method, a summary to get the full information about the model and its layers.

```
classifier.summary()#summary of our model
```

```
Model: "sequential"
```

Layer (type)	Output Shape	Param #
conv2d (Conv2D)	(None, 62, 62, 32)	896
max_pooling2d (MaxPooling2D)	(None, 31, 31, 32)	0
conv2d_1 (Conv2D)	(None, 29, 29, 32)	9248
max_pooling2d_1 (MaxPooling2D)	(None, 14, 14, 32)	0
flatten (Flatten)	(None, 6272)	0
dense (Dense)	(None, 128)	802944
dense_1 (Dense)	(None, 5)	645

```
Total params: 813,733  
Trainable params: 813,733  
Non-trainable params: 0
```

## 5. Configure The Learning Process

- The compilation is the final step in creating a model. Once the compilation is done, we can move on to the training phase. The loss function is used to find errors or deviations in the learning process. Keras requires loss function during the model compilation process.
- Optimization is an important process that optimizes the input weights by comparing the prediction and the loss function. Here we are using adam optimizer
- Metrics are used to evaluate the performance of your model. It is similar to the loss function, but not used in the training process



### ### Compiling the model

```
# Compiling the CNN
# categorical_crossentropy for more than 2
classifier.compile(optimizer='adam', loss='sparse_categorical_crossentropy', metrics=['accuracy'])
```

## 6. Train The Model

Now, let us train our model with our image dataset. The model is trained for 20 epochs and after every epoch, the current model state is saved if the model has the least loss encountered till that time. We can see that the training loss decreases in almost every epoch till 20 epochs and probably there is further scope to improve the model.

fit\_generator functions used to train a deep learning neural network

Arguments:

- steps\_per\_epoch: it specifies the total number of steps taken from the generator as soon as one epoch is finished and the next epoch has started. We can calculate the value of steps\_per\_epoch as the total number of samples in your dataset divided by the batch size.
- Epochs: an integer and number of epochs we want to train our model for.
- validation\_data can be either:
  - an inputs and targets list
  - a generator
  - inputs, targets, and sample\_weights list which can be used to evaluate the loss and metrics for any model after any epoch has ended.
- validation\_steps: only if the validation\_data is a generator then only this argument

can be used. It specifies the total number of steps taken from the generator before it is stopped at every epoch and its value is calculated as the total number of validation data points in your dataset divided by the validation batch size.

### ## Fitting the model

```
classifier.fit_generator(  
    generator=x_train, steps_per_epoch = len(x_train),  
    epochs=20, validation_data=x_test, validation_steps = len(x_test)) # No of images in test set
```

## 7. Save The Model

The model is saved with .h5 extension as follows

An H5 file is a data file saved in the Hierarchical Data Format (HDF). It contains multidimensional arrays of scientific data.

```
### Saving our model

# Save the model
classifier.save('nutrition.h5')
```

## 8. Test The Model

Evaluation is a process during the development of the model to check whether the model is the best fit for the given problem and corresponding data.

Load the saved model using load\_model

```
### Predicting our results

from tensorflow.keras.models import load_model
from keras.preprocessing import image
model = load_model("nutrition.h5") #Loading the model for testing
```

Taking an image as input and checking the results

```
img = image.load_img(r"C:\Users\DELL\Desktop\Desk Files\Nutrition Analysis Using Image Classification\
Sample_Images\Test_Image5.jpg",
                    grayscale=False, target_size= (64,64)) #Loading of the image
x = image.img_to_array(img) #Image to array
x = np.expand_dims(x, axis = 0) #changing the shape
pred = model.predict_classes(x) #predicting the classes
pred
```

By using the model we are predicting the output for the given input image

```
index=['APPLES', 'BANANA', 'ORANGE', 'PINEAPPLE', 'WATERMELON']
result=str(index[pred[0]])
result

'PINEAPPLE'
```

The predicted class index name will be printed here.

## 7.2.FEATURE 2[PYTHON CODE]

### 7.2 Feature 2[Python Code]:

#### Build Python Code

##### 1. Importing Libraries

The first step is usually importing the libraries that will be needed in the program.

```
from flask import Flask,render_template,request
# Flask-It is our framework which we are going to use to run/serve our application.
#request-for accessing file which was uploaded by the user on our application.
import os
import numpy as np #used for numerical analysis
from tensorflow.keras.models import load_model#to load our trained model
from tensorflow.keras.preprocessing import image
import requests
```

Importing the flask module into the project is mandatory. An object of the Flask class is our WSGI application. Flask constructor takes the name of the current module (\_\_name\_\_) as an argument Pickle library to load the model file.

##### 2. Creating Our Flask Application And Loading Our Model By Using Load\_model Method

```
app = Flask(__name__,template_folder="templates") # initializing a flask app
# Loading the model
model=load_model('nutrition.h5')
print("Loaded model from disk")
```

##### 3. Routing To The Html Page

Here, the declared constructor is used to route to the HTML page created earlier.

In the above example, the '/' URL is bound with the home.html function. Hence, when the home page of the webserver is opened in the browser, the HTML page is rendered. Whenever you enter the values from the HTML page the values can be retrieved using the POST Method.

Here, "home.html" is rendered when the home button is clicked on the UI

```
@app.route('/')# route to display the home page
def home():
    return render_template('home.html')#rendering the home page

@app.route('/image1',methods=['GET','POST'])# routes to the index html
def image1():
    return render_template("image.html")
```

When "image is uploaded "on the UI, the launch function is executed

```
@app.route('/predict',methods=['GET', 'POST'])# route to show the predictions in a web UI
def launch():
```

It will take the image request and we will be storing that image in our local system then we will convert the image into our required size and finally, we will be predicting the results with the help of our model which we trained and depending upon the class identified we will showcase the class name and its properties by rendering the respective html pages.

```
@app.route('/predict',methods=['GET', 'POST'])# route to show the predictions in a web UI
def launch():
    if request.method=='POST':
        f=request.files['file'] #requesting the file
        basepath=os.path.dirname('__file__')#storing the file directory
        filepath=os.path.join(basepath,"uploads",f.filename)#storing the file in uploads folder
        f.save(filepath)#saving the file

        img=image.load_img(filepath,target_size=(64,64)) #load and reshaping the image
        x=image.img_to_array(img)#converting image to an array
        x=np.expand_dims(x,axis=0)#changing the dimensions of the image

        pred=np.argmax(model.predict(x), axis=1)
        print("prediction",pred)#printing the prediction
        index=['APPLES','BANANA','ORANGE','PINEAPPLE','WATERMELON']

        result=str(index[pred[0]])

        x=result
        print(x)
        result=nutrition(result)
        print(result)

        return render_template("0.html",showcase=(result),showcase1=(x))
```

API Integration:

Here we will be using Rapid API

Using RapidAPI, developers can search and test the APIs, subscribe, and connect to the APIs — all with a single account, single API key and single SDK. Engineering teams also use RapidAPI to share internal APIs and microservice documentation.

[Reference link](#)

API used: [Link](#)

```
def nutrition(index):  
  
    url = "https://calorieninjas.p.rapidapi.com/v1/nutrition"  
  
    querystring = {"query":index}  
  
    headers = {  
        'x-rapidapi-key': "5d797ab107mshe668f26bd044e64p1ffd34jsnf47bfa9a8ee4",  
        'x-rapidapi-host': "calorieninjas.p.rapidapi.com"  
    }  
  
    response = requests.request("GET", url, headers=headers, params=querystring)  
  
    print(response.text)  
    return response.json()['items']
```

Finally, Run the application

This is used to run the application in a localhost. The local host runs on port number 5000.(We can give different port numbers)

```
if __name__ == "__main__":  
    # running the app  
    app.run(debug=False)
```

## 8.TESTING

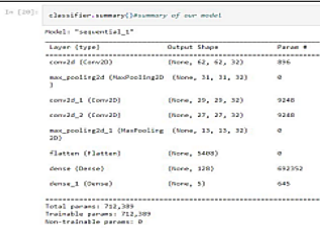
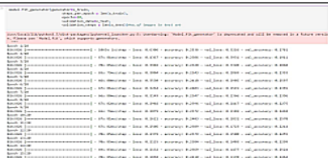
### 8.1.TEST CASES

Test case ID	Feature Type	Component	Test Scenario	Pre-Requisite
HomePage_TC_OO1	Functional	Home Page	Verify user is able to see the home page or not.	
HomePage_TC_OO2	UI	Home Page	Verify the UI elements in Home Page	
PredictPage_TC_OO3	Functional	Predict page	Verify user is able to redirect to predict page or not.	
PredictPage_TC_OO4	UI	Predict page	Verify the UI elements in Predict Page	
PredictPage_TC_OO5	Functional	Predict page	Verify user is able to select the dropdown value or not.	
PredictPage_TC_OO6	Functional	Predict page	Verify user is able to upload the image or not.	
PredictPage_TC_OO7	Functional	Predict page	Verify whether the image is predicted correctly or not	

Steps To Execute	Test Data	Expected Result	Actual Result	Status
1.Enter URL and click go 2.verify whether the user is able to see the home page.	Enter URL and click go	User able to see the home page	Working as expected	Pass
1.Enter URL and click go 2.Verify the UI elements in Home Page.	Enter URL and click go	Application should show below UI elements: Home Tab & Classify Tab	Working as expected	pass
1.Enter URL and click go 2.Click on Predict button 3.Verify whether the user to redirect to predict page or not.	Click the predict button in home page	User should navigate to Predict page	Working as expected	pass
1.Enter URL and click go 2.Verify the UI elements in Predict Page.	Click the predict button and redirect to predict page	Application should show below UI elements: Dropdown List , Upload file Button, Predict button.	Working as expected	pass
1.Enter URL and click go 2.Click on Predict button 3.Verify whether the user to redirect to predict page or not. 4.Verify user is able to select the dropdown value or not.	Fruit or Vegetable	Application should show the user, the uploaded image for verification of given specified fruit or vegetable.	Working as expected	pass
1.Enter URL and click go 2.Click on Predict button 3.Verify whether the user to redirect to predict page or not. 4.Verify user is able to select the dropdown value or not. 5.Verify user is able to upload the images or not	Images to be Uploaded	Application should shows the uploaded image.	Working as expected	pass
1.Enter URL and click go 2.Click on Predict button 3.Verify whether the user to redirect to predict page or not. 4.Verify user is able to select the dropdown value or not. 5.Verify user is able to upload the images or not 6. Verify whether the Image is predicted correctly or not	Click the Predict Button	Application shows the nutrition details classified output	Working as expected	pass

## Model Performance Testing

Project team shall fill the following information in model performance testing template.

S.No.	Parameter	Values	Screenshot
1.	Model Summary	Total params: 712,389 Trainable params:712,389 Non-trainable params: 0	 <pre> In [30]: classifier.summary(summary of our model)  Model: "sequential_1" Layer (type)                Output Shape              Param # ----- conv2d (Conv2D)             (None, 62, 62, 32)       896 max_pooling2d (MaxPooling2D) (None, 31, 31, 32)       0 conv2d_1 (Conv2D)           (None, 27, 27, 32)       9248 conv2d_2 (Conv2D)           (None, 27, 27, 32)       9248 max_pooling2d_1 (MaxPooling2D) (None, 13, 13, 32)       0 flatten (Flatten)           (None, 5408)              0 dense (Dense)               (None, 100)               532752 dense_1 (Dense)             (None, 1)                 645 Total params: 712,389 Trainable params: 712,389 Non-trainable params: 0 </pre>
2.	Accuracy	Training Accuracy – 96.55  Validation Accuracy – 97.45	 <pre> Epoch 1/10: 100% 10/10 [&gt;] 0.9655 loss: 0.0000 Epoch 2/10: 100% 10/10 [&gt;] 0.9655 loss: 0.0000 Epoch 3/10: 100% 10/10 [&gt;] 0.9655 loss: 0.0000 Epoch 4/10: 100% 10/10 [&gt;] 0.9655 loss: 0.0000 Epoch 5/10: 100% 10/10 [&gt;] 0.9655 loss: 0.0000 Epoch 6/10: 100% 10/10 [&gt;] 0.9655 loss: 0.0000 Epoch 7/10: 100% 10/10 [&gt;] 0.9655 loss: 0.0000 Epoch 8/10: 100% 10/10 [&gt;] 0.9655 loss: 0.0000 Epoch 9/10: 100% 10/10 [&gt;] 0.9655 loss: 0.0000 Epoch 10/10: 100% 10/10 [&gt;] 0.9655 loss: 0.0000 </pre>

## 8.2.USER ACCEPTANCE TESTING

**1. Purpose of Document:** The purpose of this document is to briefly explain the test coverage and open issues of the AI-powered Nutrition Analyzer for Fitness Enthusiasts project at the time of the release to User Acceptance Testing (UAT).

**2. Defect Analysis:** This report shows the number of resolved or closed bugs at each severity level, and how they were resolved.

Resolution	Severity 1	Severity 2	Severity 3	Severity 4	Subtotal
By Design	0	0	1	0	1
Duplicate	1	3	2	2	8
External	2	3	0	0	5
Fixed	4	4	4	4	16
Not Reproduced	0	0	0	1	1
Skipped	0	0	0	0	0
Won't Fix	0	0	0	0	0
Totals	7	10	7	7	31

**3. Test Case Analysis:** This report shows the number of test cases that have passed, failed, and untested.



## **9. RESULTS**

### **9.1.PERFORMANCE METRICS**

#### **Software Requirements**

To complete this project, you should have the following software and packages.

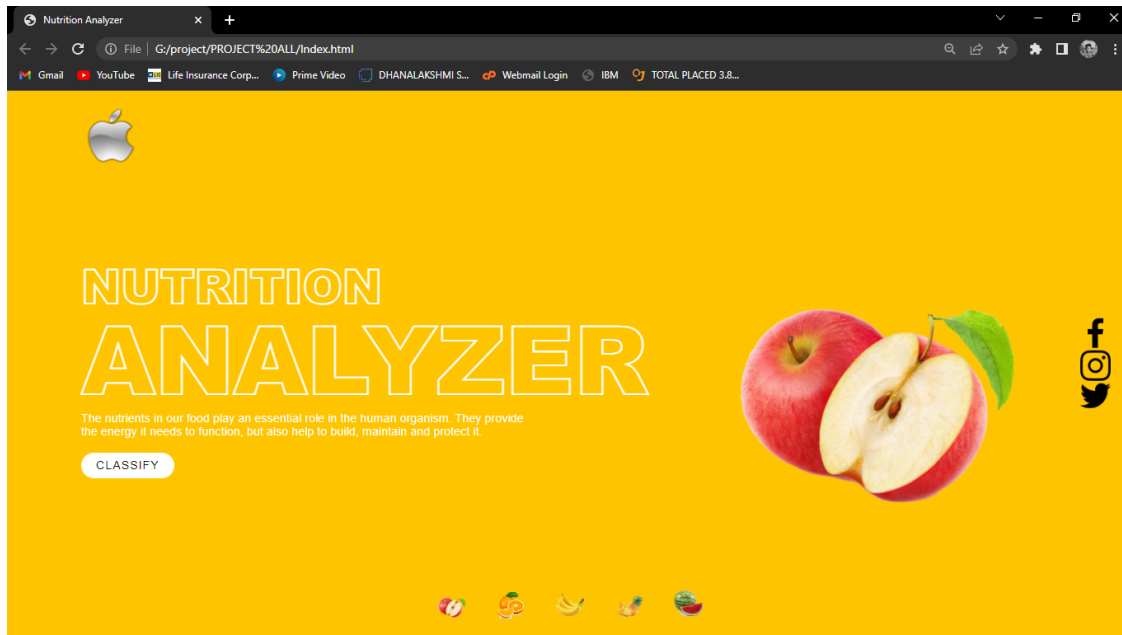
##### **Softwares**

- Anaconda Navigator
- py charm
- Visual studio code
- Jupiter notebook

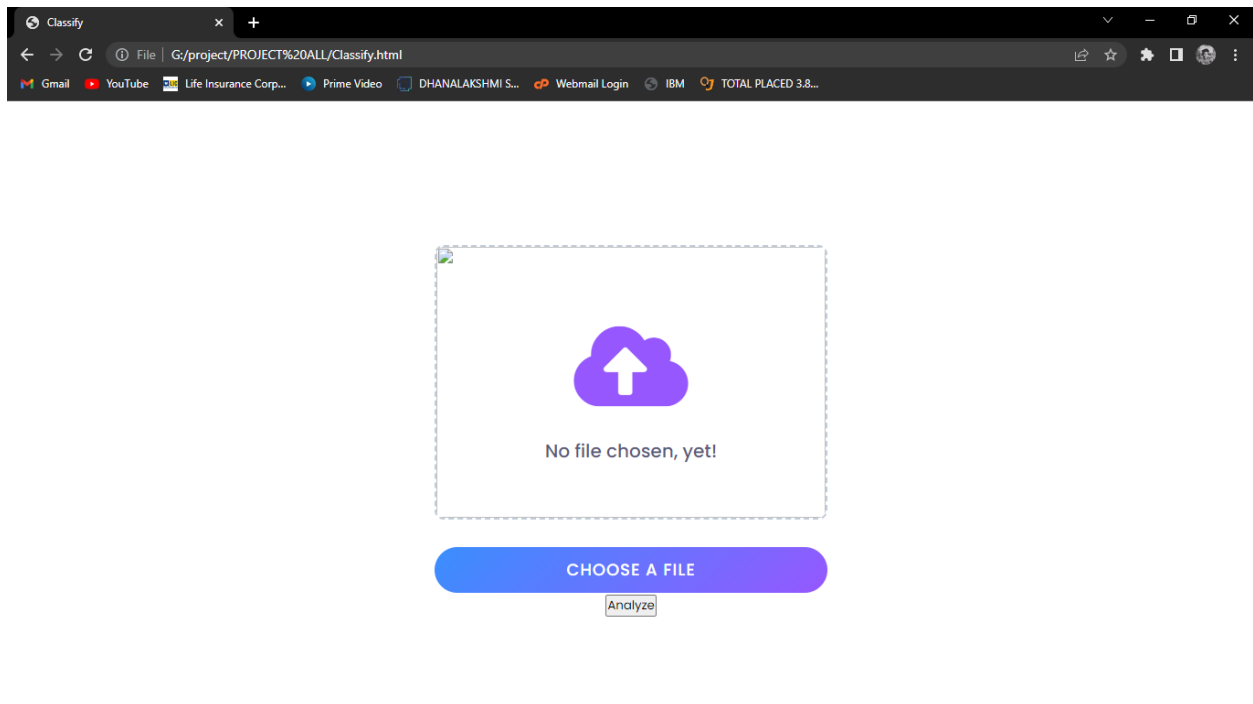
##### **IBM Watson studio Packages**

- Tensor flow
- Keras
- Flask
- numpy
- Pandas

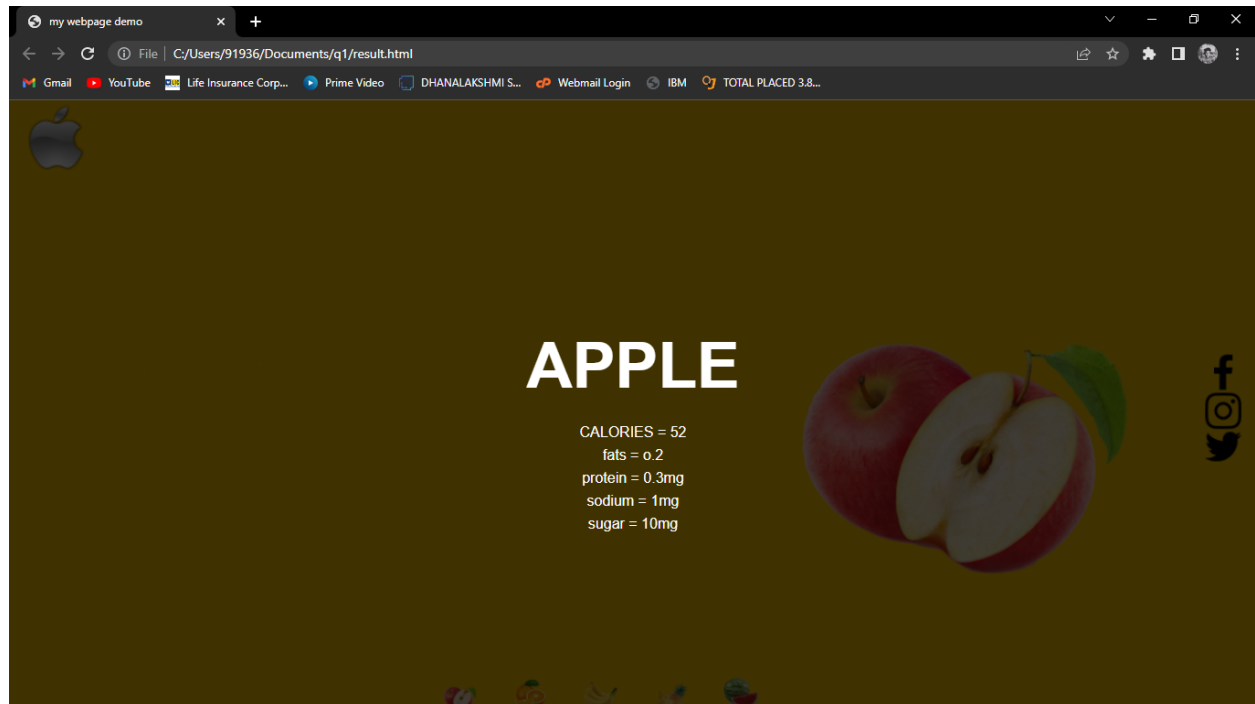
## OUTPUT



## UPLOAD PAGE



## ANALYZER PAGE



## **10.ADVANTAGES & DISADVANTAGES**

### **ADVANTAGES**

1.The AI Nutrition Analyzer was designed to direct users toward a balanced diet and help them reach their health objectives.

2.A good way to gauge caloric intake

3.It guarantees that the meal has the right amount of vitamins and minerals.

4.Offer more precise values.

5.Cost effectiveness.

### **DISADVANTAGES**

1.It has limited applicability to other nutrients like proteins and vitamins.

2.Erroneous information

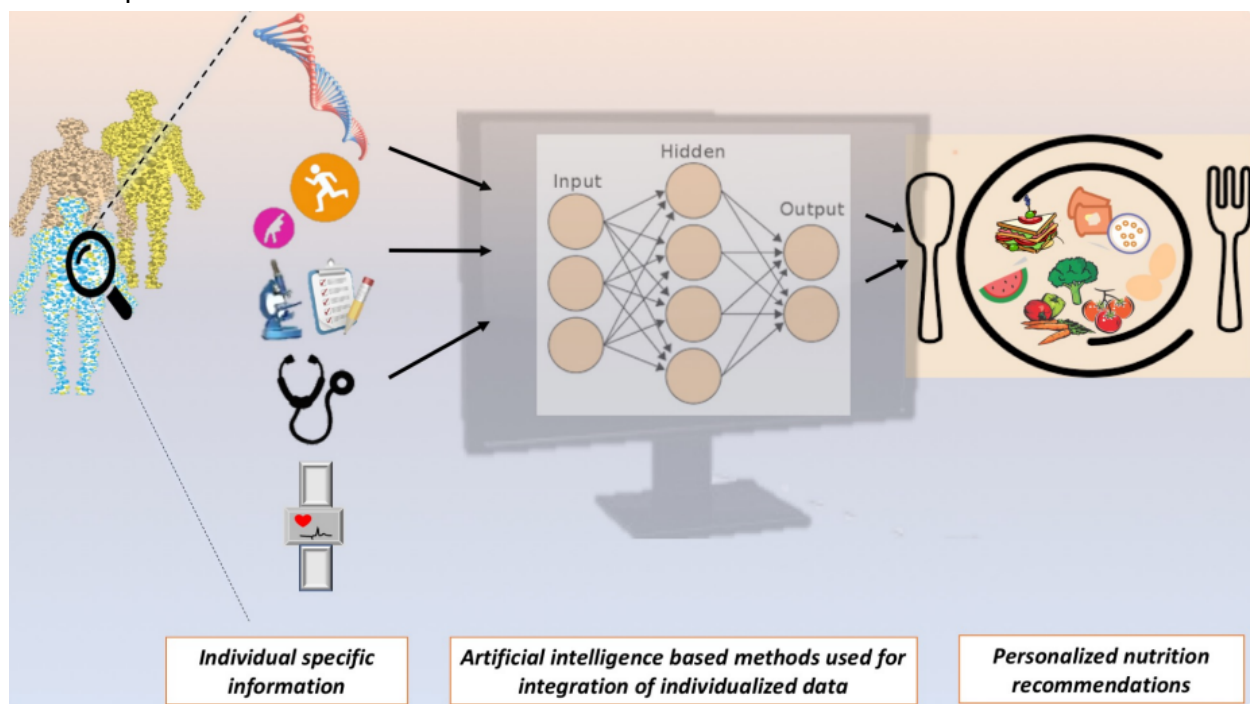
3.Takes a lot of time

## 11.CONCLUSION

We develop a useful deep learning algorithm for fitness enthusiasts based on the AI-driven Nutrition Analyser. In this method, the services are used by a deep learning algorithm-based food picture identification system. We are enhancing the algorithm's performance (in terms of detection accuracy). We will eventually include our technique into a genuine mobile device to improve cloud computing. This method uses the user's submitted image of the meal to classify it and then uses the estimated model to measure the item's properties. The outcomes are improved using flask application, model construction, and picture preprocessing in the IBM cloud. This method of categorization yields precise values. The system's accuracy and usefulness will be enhanced in the future

## 12.FUTURE SCOPE

Future plans for an AI-powered nutrition analyzer for exercise enthusiasts include handling health issues and difficulties as well as making it simple for regular individuals to accomplish their objectives. Increase the epochs and layer count to gather additional data. We can get more sophisticated features from it, like bespoke searches and better visualisation, for example



## 13.APPENDIX

### Source Code

#### [Python Code]:

```
#request-for accessing file which was uploaded by the user on our
application.
import os
import tensorflow as tf
global graph
graph = tf.compat.v1.get_default_graph()
import numpy as np # used for numerical analysis
from werkzeug.utils import secure_filename
from gevent.pywsgi import WSGIServer

from flask import Flask, render_template, request, url_for
from keras.preprocessing import image
from keras.models import load_model # to load our trained model

app = Flask(__name__, template_folder="templates") # initializing a flask
app
# Loading the model
model=load_model('Fruites.h5')
print("Loaded model from disk")

@app.route('/')# route to display the home page
def home():
    return render_template('Index.html')#rendering the home page

@app.route('/analyser', methods=['GET', 'POST'])# routes to the index html
def image1():
    return render_template("Analyzer.html")
```

```

@app.route('/ai',methods=['GET', 'POST'])# route to show the predictions
in a web UI
def launch():
    if request.method=='POST':
        f=request.files['image']
        print('current path')#requesting the file
        basepath=os.path.dirname('__file__')
        print('current path',basepath)#storing the file directory
        filepath=os.path.join(basepath,"uploads",f.filename)
        print('upload folder is',filepath)#storing the file in uploads
        folder
        f.save(filepath)#saving the file

        img=image.load_img(filepath,target_size=(64,64)) #load and
        reshaping the image
        x=image.img_to_array(img)#converting image to an array
        x=np.expand_dims(x,axis=0)#changing the dimensions of the image

        with graph.as_default():
            pred=model.predict_classes(x)
            print("prediction",pred)#printing the prediction
            index=['APPLES','BANANA','ORANGE','PINEAPPLE','WATERMELON']
            text = 'the predicted animal is: ' + str(index[pred[0]])

        return text
    else:
        return render_template('Result.html')

if __name__ == "__main__":
    # running the app
    app.run(debug=False)

```

[HTML Code]:

## Index Page:

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8">
    <title>Nutrition Analyzer</title>
    <link rel="stylesheet" href="/static/css/style.css">
  </head>
  <body>
    <section class="sec">
      <header>
        <a href="#"></a>
      </header>
      <div class="content">
        <div class="textBox">
          <h2>Nutrition<br><span>Analyzer</span></h2>
          <p>The nutrients in our food play an essential role in
the human organism. They provide the energy it needs to function, but also
help to build, maintain and protect it.</p>
          <a href="http://localhost:5000/analyser"
class="link">Classify</a>
        </div>
        <div class="imgBox">
          
        </div>
      </div>
      <ul class="thumb">
        <li></li>
        <li></li>
        <li></li>
```



```

        <li></li>
        <li></li>
    </ul>
    <ul class="sci">
        <li><a href="#"></a></li>
        <li><a href="#"></a></li>
        <li><a href="#"></a></li>
    </ul>
</section>
<script type="text/javascript">
    function imgSlider(anything){
        document.querySelector('.apple').src = anything;
    }
    function changeBgcolor(color){
        const sec = document.querySelector('.sec');
        sec.style.background = color;
    }
</script>
</body>
</html>

```

**Analyzing Page:**

**Git-Hub Link**

<https://github.com/IBM-EPBL/IBM-Project-29505-1660126397>

**Project Demonstration Link**

<https://drive.google.com/file/d/1IKQWPNx0elZ2RhrN2Wtx2D8NJQNxr0GT/view?usp=drivesdk>