AI - POWERED NUTRITION ANALYZER FOR FITNESS ENTHUSIASTS

TEAM ID : PNT2022TMID49040

TEAM LEADER : SARAVANAKUMAR P (921619106053)

TEAM MEMBERS : TAMILARASAN S (921619106060)

SATHEES R (921619106054)

SANJAY S (921619106050)

DEPARTMENT : ELECTRONICS &COMMUNICATIONS ENGINEERING

COLLEGE NAME : SBM COLLEGE OF ENGINEERING &TECHNOLOGY

TABLE OF CONTENTS

CHAPTER TITLE PAGE.NO

1.INTRODUCTION

1.1 PROJECT OVERVIEW

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.

1.2 PURPOSE

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.).

2.LITERATURE SURVEY 2.1 EXISTING

PROBLEM

1.AI powered nutrition Analyzer For Fitness Enthusiasts –
Jeukendrup A.E.,Killer S.C. The myths surrounding pre-exercise carbohydrate feeding.

At present, the researchers are showing there great effort in the area of food nutrition. In this section is presenting some of the researchers work that must enlighten our study. A descriptive cross-sectional study has done among 144 children and they found that in rural areas, parameter Weight-for Height(WHZ) projected that 1.39% of children were severely malnourished, 1.39% moderately malnourished, 22.3% mildly malnourished and had no serious overweight, but in urban areas, 25%, 2.78% and 1.38% mild, moderate and severe over weight respectively.

For Weight-for-Age(WAZ), the results further stated that 38.8% children mildly underweight, 25% moderately underweight in rural areas and found the opposite result for urban area[1]. Another research has done with 80 street children with 90% boy and 10% gamines and the result shows that the underweight ratio 65%. About 77.5% of underweight children eat three meals a day and 22.5% of children eat only twice a day. Most of the children in the study, 85% developed the habit of washing their hands before eating. About 61.3% of them had been suffering from different diseases for the last 3 months before starting the study .

2.2 REFERENCES

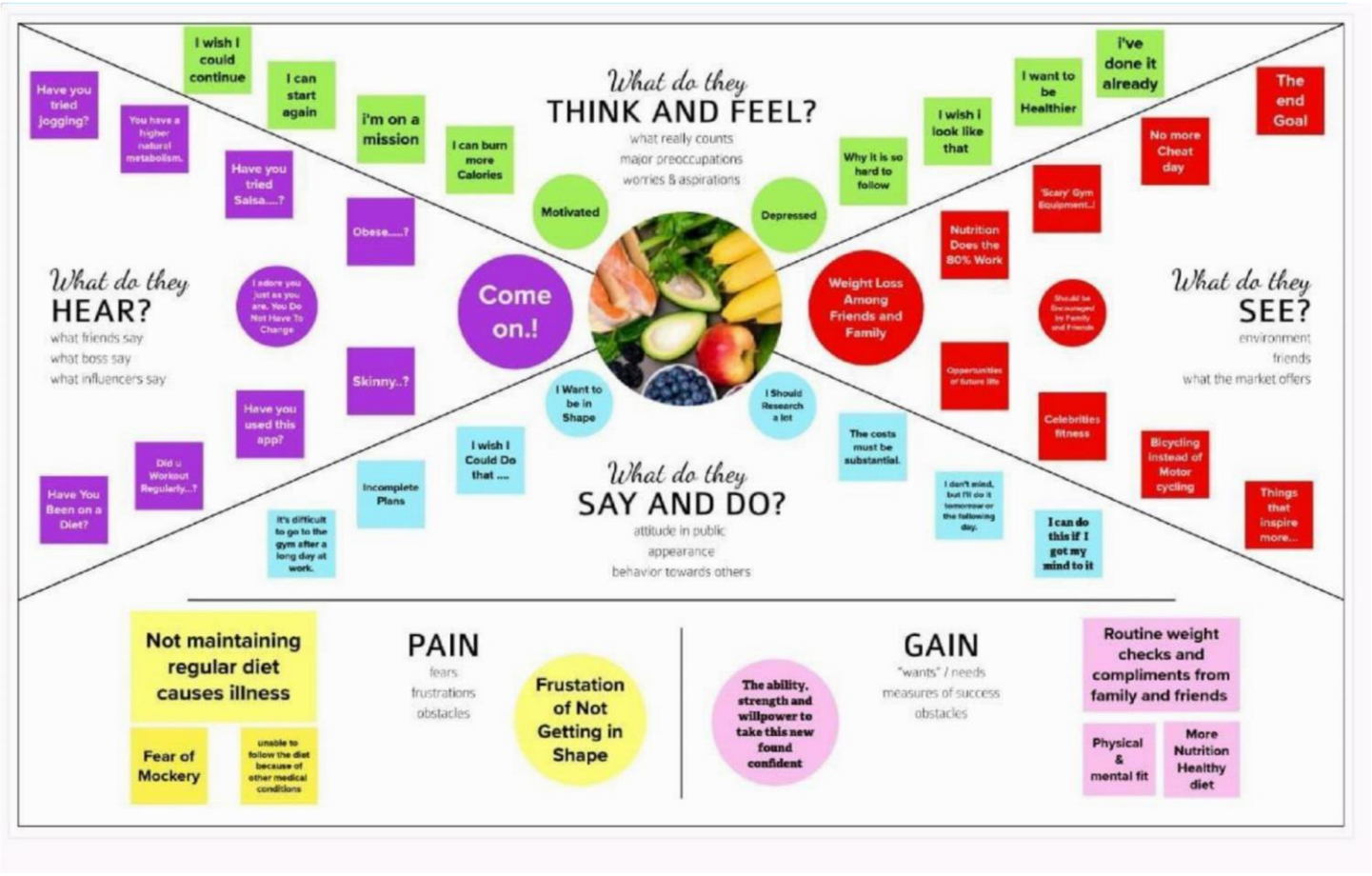
- Snap Meal App iPhone: Magical Meal Logging: https://apps.apple.com/us/app/mealsnap-photo-food-diary/id1431522193
- AI-Powered Nutrition Apps That Help Fitness Enthusiasts With Their Calorie Intake : https://analyticsindiamag.com/5-ai-powered-nutrition-apps-that- help-fitness-enthusiastswiththeir-calorie-intake/
- Watch what you eat, using your phone: https://www.deccanherald.com/content/384169/watch-you-eat-using- your.html

2.3 PROBLEM STATEMENT DEFINITION

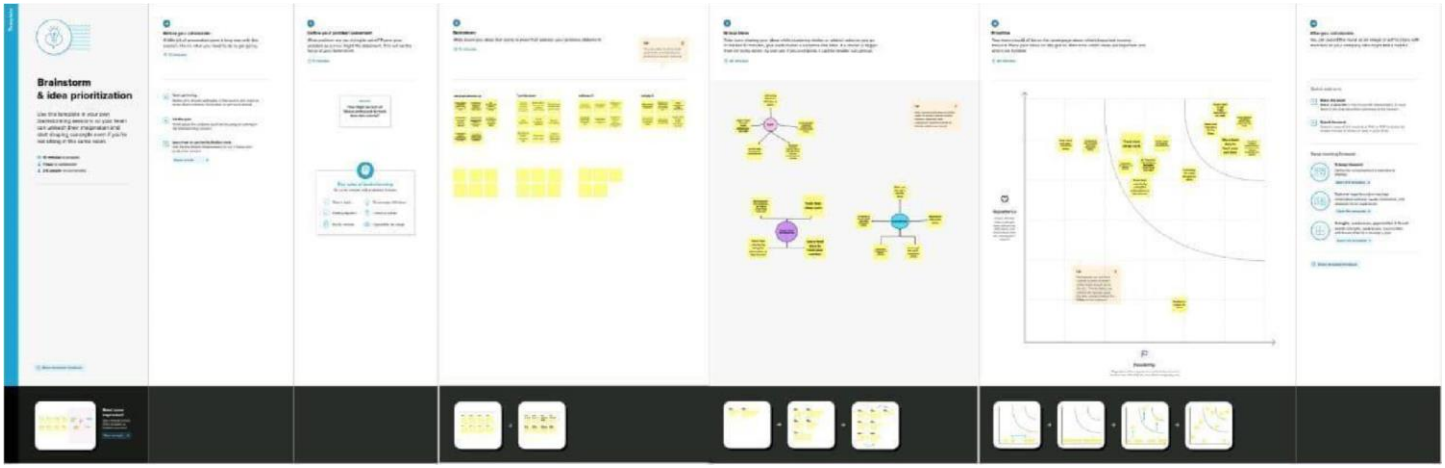
- Food is crucial for human life and has been the subject of numerous healthcare conventions. Nowadays, modern dietary assessment and nutrition analysis tools allow more options to Help people understand their daily eating habits, investigate nutrition trends and maintain a Healthy diet. Nutritional analysis is the method of determining the nutritional composition Of food. It is a critical aspect of analytical chemistry that offers information about the Chemical composition, processing, quality control and contamination of food. The major Purpose of the project would be to construct a model which is used for classifying the fruit Depending on the many features like color, shape, texture etc. Here the user can capture The photographs of different fruits and then the image will be provided to the trained Model. The model examines the image and identifies the nutrition depending on the fruit's As (Sugar, Fiber, Protein, Calories, etc).

3. IDEATION & PROPOSED SOLUTION

3.1 EMPATHY MAP CANVAS



3.2 IDEATION &BRAINSTORMING



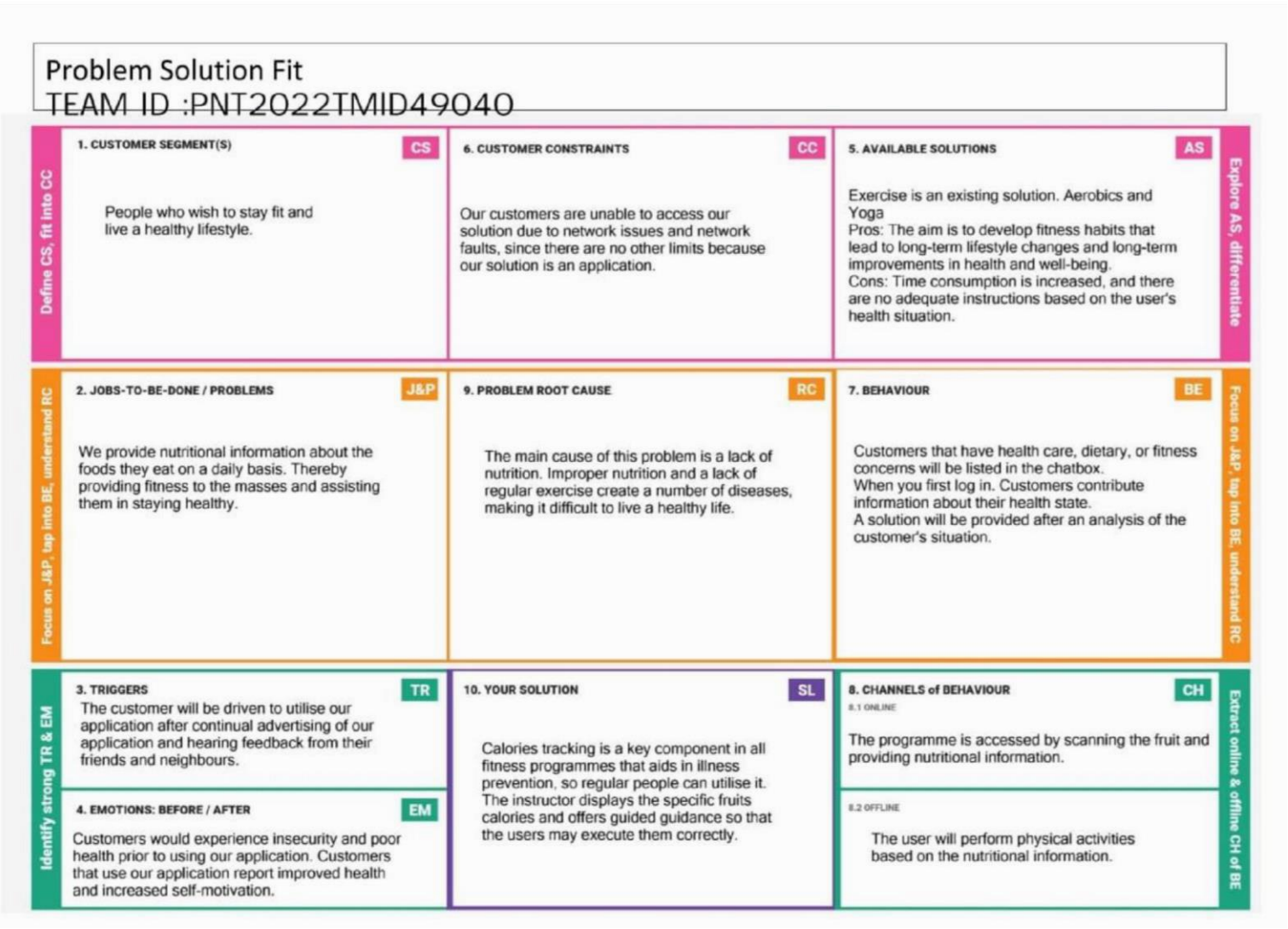
3.3 PROPOSED SOLUTION

S.NO	Parameter	Description
1.	Problem Statement (Problem to be solved)	A regular person must use cutting-edge AI- based analysing software to identify fruits and vegetables based on colour, texture, form, and other characteristics. At the time of identification, the user must also be aware of the nutritional content of that specific edible.
2.	Idea / Solution description	**Main solution :** ● Clear and proper identification of the given input data. ● Provide nutritional facts based on the obtained data. ● Fitness analysis and maintenance as per the user's body conditions **Additional Benefits :** ● Analysis of daily dietary requirements ● Daily tracking of dietary consumption thoroughly.

3.	Novelty / Uniqueness	The availability of fitness plans with addon bonuses
		● Suggestion of home remedies and simple solutions for basic problems. ● An individualized food plan based on health condition and deficiency. ● Allowing for diet flexibility helps promote a healthy and effective eating pattern
4.	Social Impact / Customer Satisfaction	Healthy lifestyle development ● Constant calorie management monitoring results in a fitness mindset.

5.	Business Model (Revenue Model)	Consultation with nearest trainers and nutritionist for personalized plans. ● Adopt a specialized diet plan under the direction of an expert. ● Advertise and offer nutritional supplements and fitness gear. ● Promotion for fitness centers and hospitals.
6.	Scalability of the Solution	Improving accuracy by expanding the data collection using user input data ● Storage requirements of a specific food. ● User friendly UI for everyone to use and get benefit from it.

3.4 PROBLEM SOLUTION FIT



4.REQUIREMENT ANALYSIS

4.1 FUNCTIONAL REQUIREMENTS

Following are the functional requirements of the proposed solution.

FR.NO	Functional Requirements (Epic)	Sub Requirement(Story/SubTask)
FR-1	User Registration	Registration through Form Registration through Gmail Registration through LinkedIn
FR-2	User Confirmation	Confirmation via Email Confirmation via OTP
FR-3	User Login	Login through Google Login through Email
FR-4	Choose Package	Selection of desired package
FR-5	Generate the Daily Plan	Daily plans will be generated by dietician
FR-6	Manage progress -report	Gathering information from database and generating
FR-7	Query	The user can ask for changes in plan

4.2.NONFUNCTIONAL REQUIREMENTS

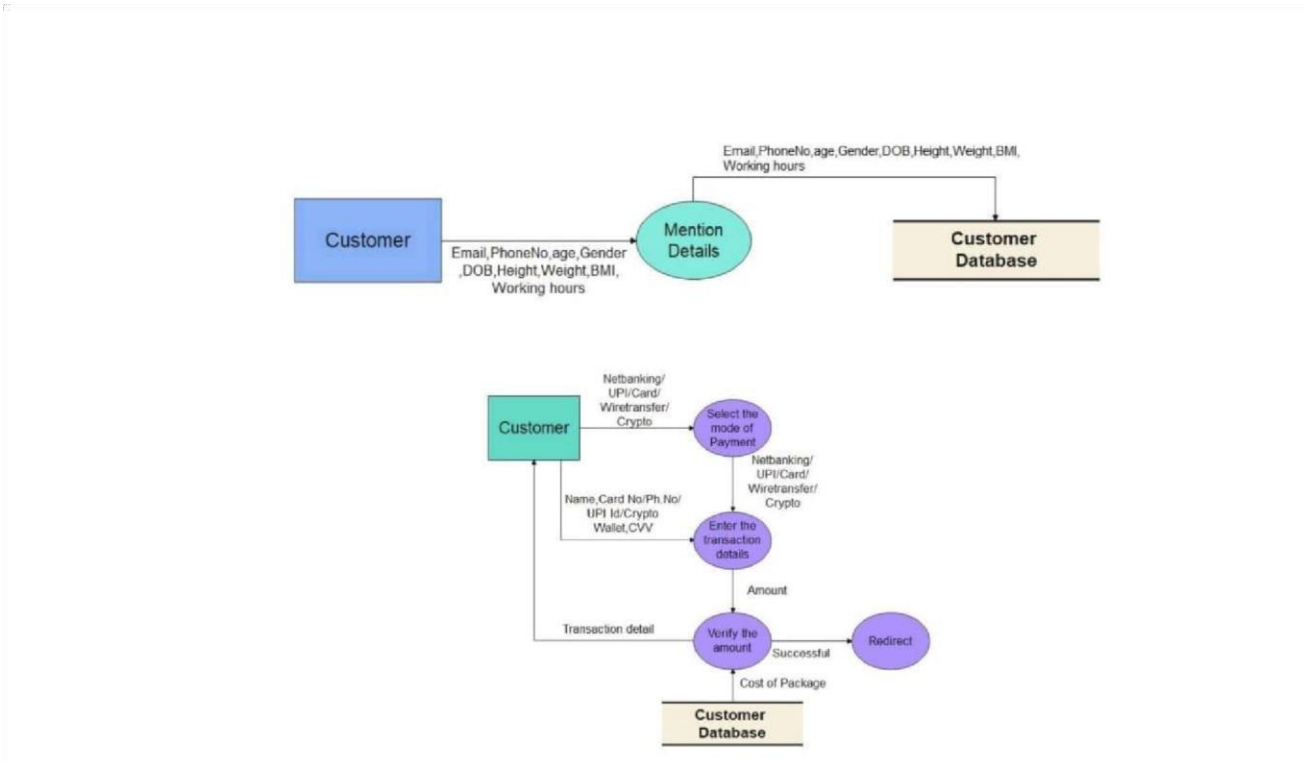
Following are the non-functional requirements of the proposed solution

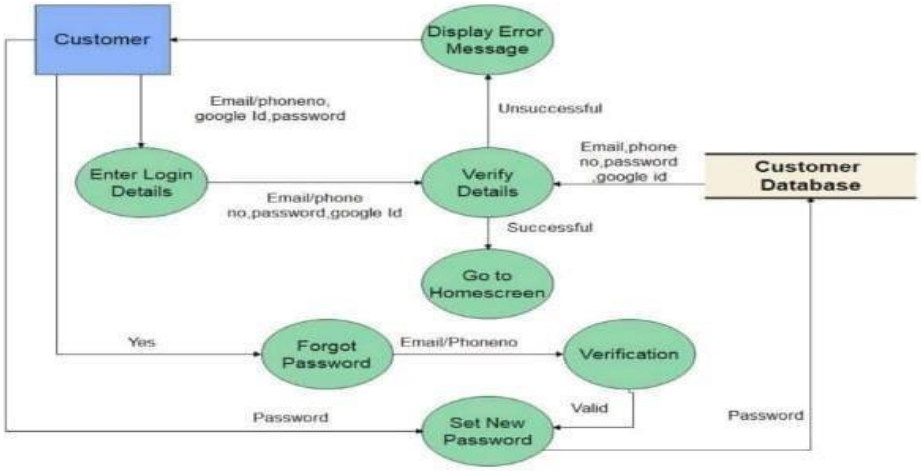
NFR.NO	Non Functional Requirements	Description
NFR-1	Usability	Easy to use with interactive User Interface.

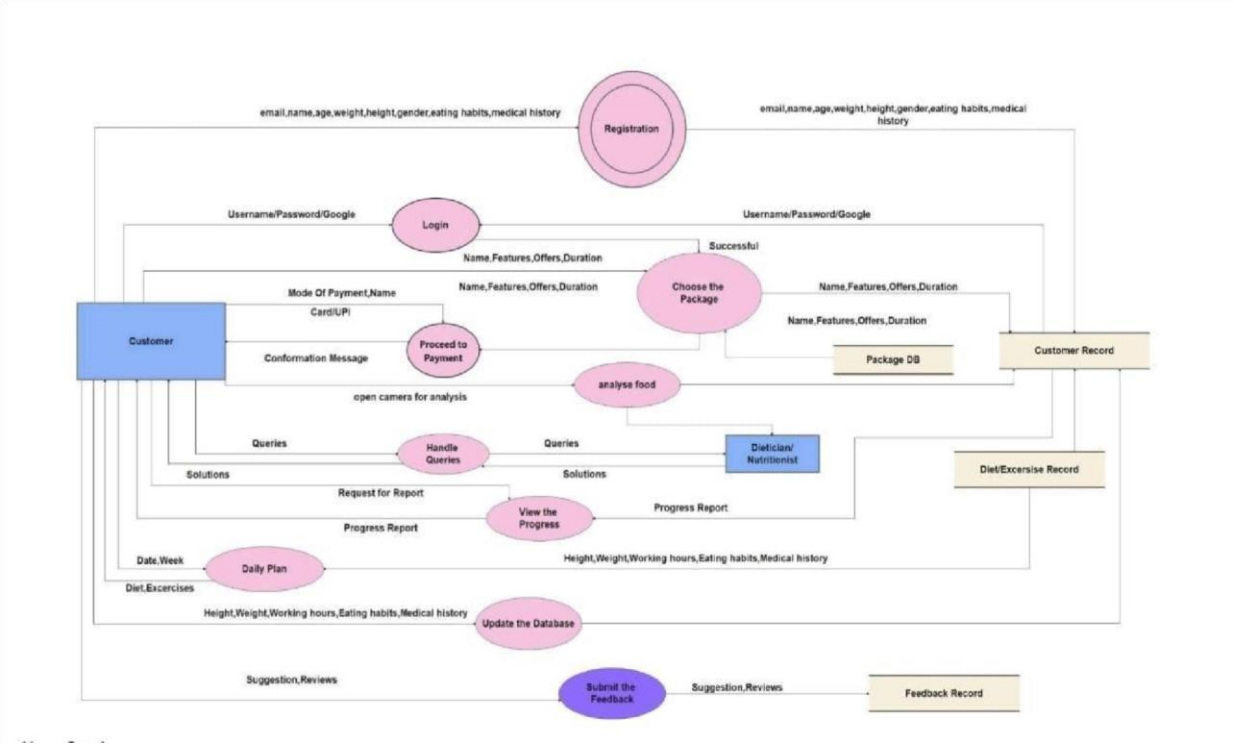
NFR-2	Security	User can access only their personal information and not that of other users.
NFR-3	Reliability	The average time of failure shall be 7 days.
NFR-4	Performance	The results has to be shown within 10 sec
NFR-5	Availability	The dietician shall be available to users 24 hours a day, 7 days a week.
NFR-6	Scalability	Supports various food items.

5.PROJECT DESIGN

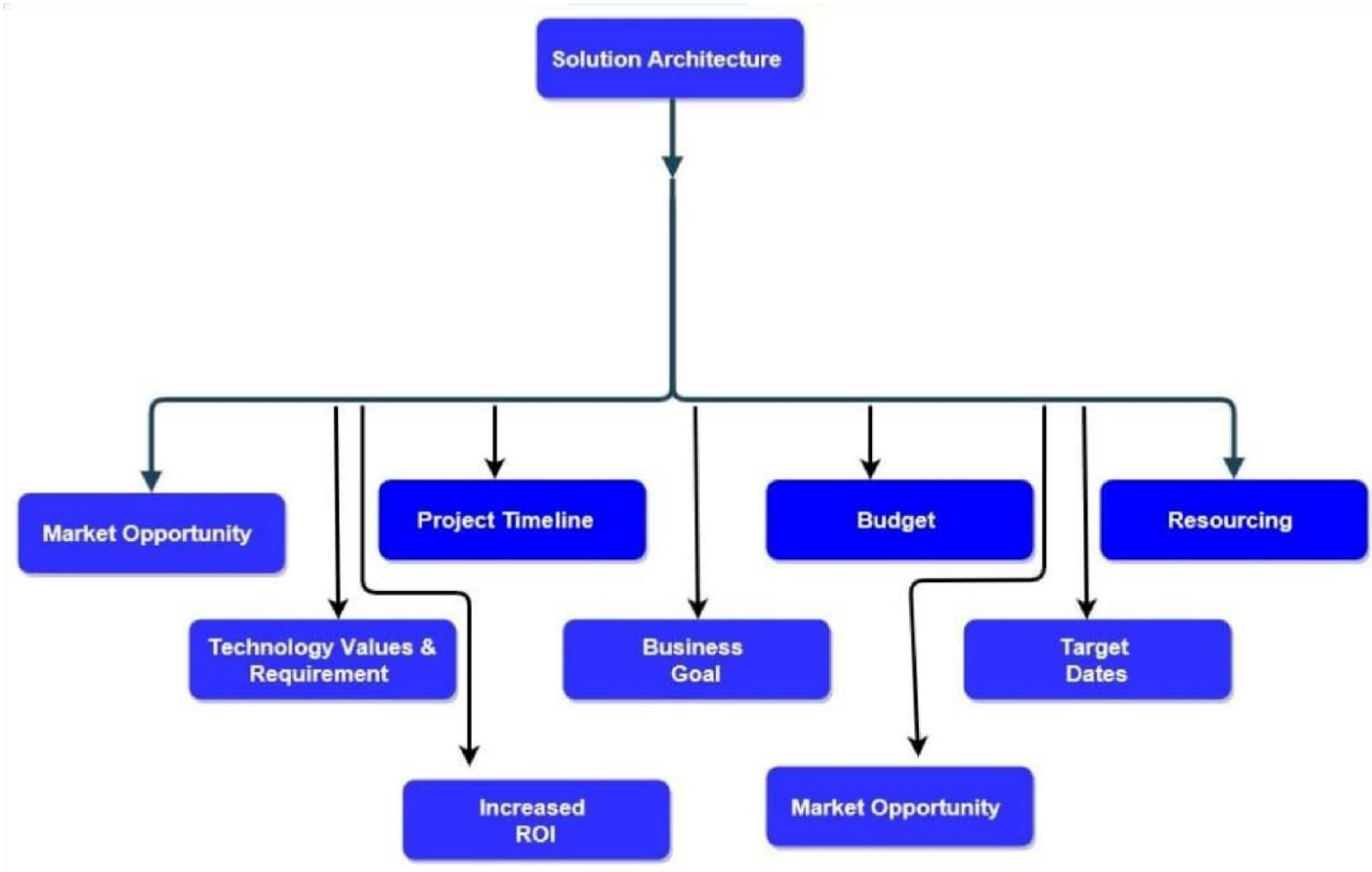
5.1 DATA FLOW DIAGRAMS



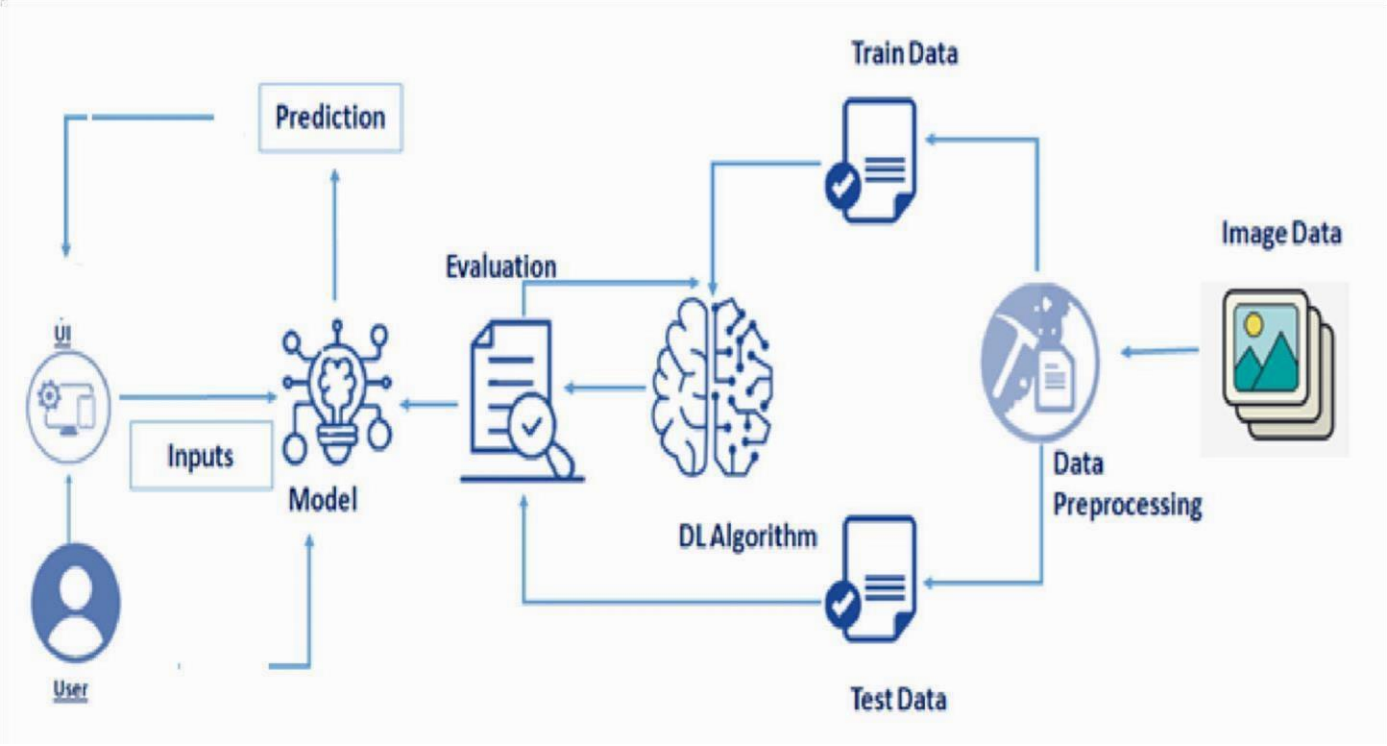




5.2 SOLUTION & TECHNICAL ARCHITECTURE


Solution Architecture
Market Opportunity
Project Timeline
Budget
Resourcing
Technology Values & Requirement
Business Goal
Target Dates
Increased ROI
Market Opportunity


Create Indian Food Dataset
Detect & Classify Meal
Map to Nutrition Database
Integrate into Mobile Application


Train Data
Prediction
Image Data
Evaluation
UI
Inputs
Model
DL Algorithm
Data
Preprocessing
User
Test Data

S.NO	COMPONENT	DESCRIPTION	TECHNOLOGY
1.	User Interface	Predicts the user interaction with Application	HTML, CSS, Javascript
2.	Application Logic-1	A fitness tool is used for analysing the nutrient	Python
3.	Application Logic-2	IBM Watson Health is a digital tool that helps the healthcare services through AI	IBM Watson STT service
4.	Database	Datatype, Configurations, Data, etc.,	MSSQL
5.	Cloud Database	Cloud Database Service	IBM DB2, IBM Cloudant
6.	Notification	Nutrition notification will be Sent from the server	Grid
7.	File Storage	File storage requirements	IBM Block Storage or Other Storage Services
8.	External API	External API is used in the Application	IBM Weather API, Aadhar API
9.	Machine Learning Model	Detect and identify the image and objects	Python Colab
10.	Infrastructure (Server / Cloud)	Application Deployment, Local Server Configuration, Cloud Server Configuration	Local, Cloud Foundry, Kubernetes, etc.,

APPLICATION CHARACTERISTICS

S.No	Characteristics	Description	Technology
1.	Open-Source Frameworks	Flask framework	Artificial Intelligence
2.	Security Implementations	Request Encryption, authentication, firewalls Security controls ,etc.,	
3.	Scalable	Supports high workloads	Artificial Intelligence
	Architecture	Use of load, distributed Servers	Artificial Intelligence
5.	Performance	The application predicts the image up to 6000 per second	Artificial Intelligence

5.3 USER STORIES

User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority	Release
Customer (Mobile user)	Registration	USN-1	I may sign up for the programme as a user by providing my email address, a password and a password confirmation	I can login my dashboard of account	High	Sprint-1
	Login	USN-2	When I register for the application as a user, I will get a confirmation email	When I register for the application as a user, I will get a confirmation email	High	Sprint-1
	Registration	USN-3	I may sign up for the application as a user through Facebook	I may use Facebook to sign up and view the dashboard	Low	Sprint-2
	Registration	USN-4	I may sign up for the application as a user using Gmail	I can sign up via mail	Medium	Sprint-1
	Login	USN-5	I may access the application as a user by providing my email address and password	I have continuous access to the website as a user	High	Sprint-1
	Access	USN-6	As a user, I can give access to camera	I can give access	Medium	Sprint-1
	Webpage	USN-7	As a user, I can upload the input fruit image to the website	I can upload the images	High	Sprint-2
	Calorie Tracker	USN-8	As a user, I have the option of manually entering my food consumption or fvie daily camera picture captures	Everyday my food consumption is calculated and analysed	Medium	Sprint-2
	Diet Plan	USN-9	As a user, I am able to create my own diet plan using the vital components provided	The AI model determines if my food has the necessary amounts of nutrients	Low	Sprint-3
Customer (Web user)	Registration	USN-10	I may sign up for the programme as a user by providing my email address, a password and a password confirmation	I can login my account or dashboard	High	Sprint-3
Customer Care Executive	Solving customer queries	USN-11	In the event that the application was unsuccessful, I should be able to contact customer service for assistance	I can get suggestions & replies from it	Medium	Sprint-2

Administrator	Database maintenance	US-12	I can manage all the user data & picture datasets collected by the AI model in my capacity as an administrator.	I can give numerous assurances on user security and data safety	High	Sprint-4

6. PROJECT PLANNING & SCHEDULING

6.1 SPRINT PLANNING & ESTIMATION

Sprint	Functional Requirement(Epic)	User Story Number	User Story/ Task	Story Points	Priority	Team Member
Sprint-1		USN-0	As a developer I have to collect different type of data supporting the model	5	High	Saravanakumar P
Sprint-1		USN-1	As a user, I can register for the application by entering my email, password, and confirming my password	5	High	Tamilarasan S

Sprint-1		USN-2	As a user, I will receive confirmation email once I	5	High	Saravanakumar P
			have registered for the application			
Sprint-2		USN-3	As a user, I will receive confirmation email once I have registered for the application	3	Low	Sathees R
Sprint-1		USN-4	As a user, I can register for the application through Gmail	3	Medium	Sanjay S
Sprint-1	Login	USN-5	As a user, I can log into the application by entering email & password	5	High	Tamilarasan S

Sprint-2	Model Building	USN-6	As a user, I can log into the application by entering	5	High	Saravanakumar P
			email & password			
Sprint-2	Main Interface	USN-7	As a user I can view my calorie intake by clicking photo of the food I eat	5	High	Sathees R
Sprint-2	Package, Dashboard	USN-8	As a user I can choose variety of packages based on my requirement	4	Medium	Sanjay S
Sprint-3	Diet plan for free users	USN-9	As a dietitian I provide daily plans for the betterment of the user	5	High	Tamilarasan S

Sprint-3	Personalized food habitbased diet plan for premium users	USN-10	As a Premium User, I can choose to follow diet plan based on my food habits or the generalized one	3	Medium	Saravanakumar P
Sprint-2	User image analysis	USN-11	As a user I can track my calorie intake, and know about my food in detail	5	High	Sathees R
Sprint-3	Improve efficienc y of AI model	-	As a developer I have to give a better model that will analyse food precisely and provide accurate results	3	Medium	Sanjay S

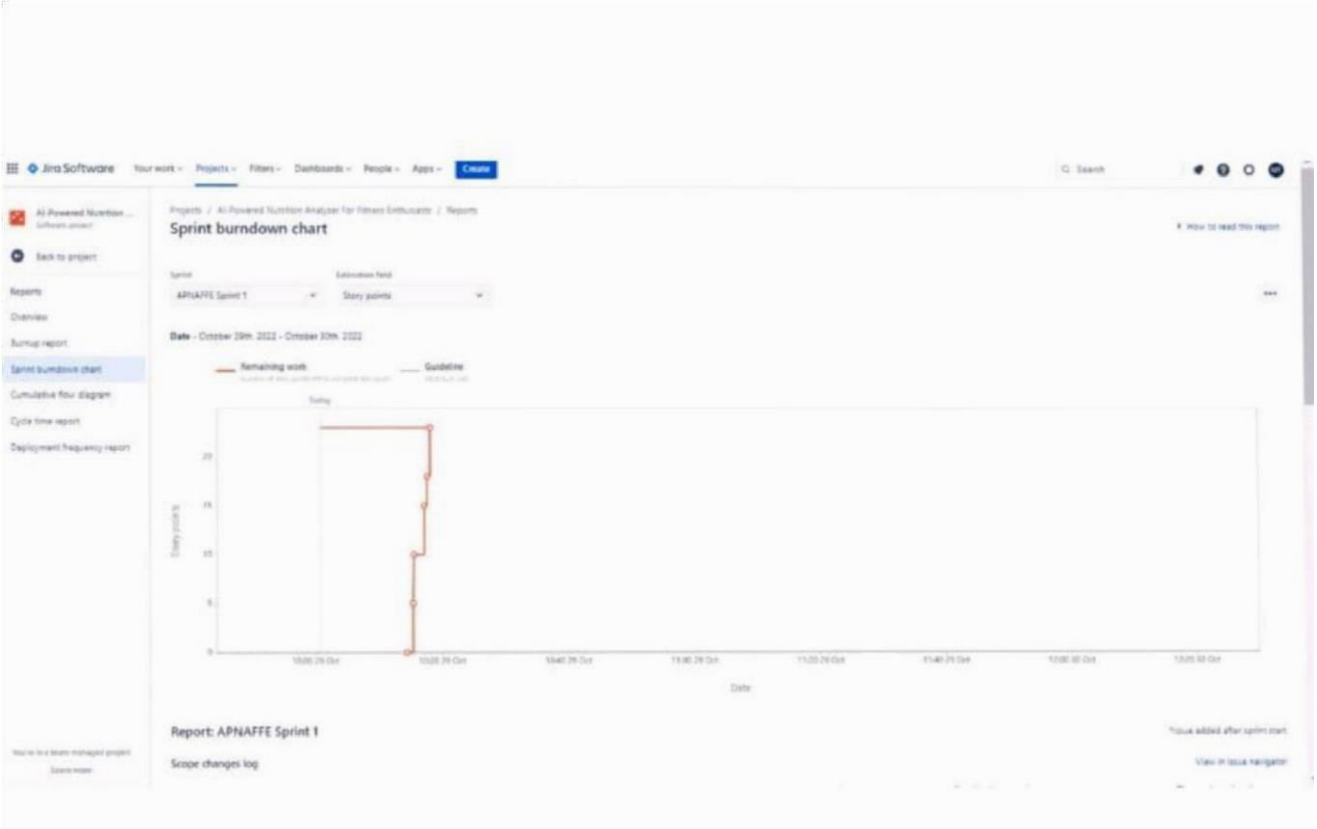
Sprint-2	User Analysis record	USN-12	As a user, I can check the previous records and I can analyse my food habits	4	Medium	Tamilarasan S
Sprint-4	Fitness tips and basic exercises	USN-13	As a user I can follow some fitness tips and I can maintain	5	Medium	Saravanakumar P
			weight as required			
Sprint-4	Home remedies	USN-14	As a user I can follow some natural home remedies for common diseases like (cold, cough, fever) and treat myself	5	High	Sathees R

Sprint-4	Optimize the user experience with the app		As a developer I have to provide clean and smooth interface to my user	5	High	Sanjay S
Sprint-4	Payment Gateway for purchasing package		As a developer I have to create a environment which makes user feel ease to complete his/her Payments with various	3	Medium	Tamilarasan S
			Payment options			

6.2 SPRINT DELIVERY SCHEDULE

Sprint	Total Story Points	Duration	Sprint Start Date	Sprint End Date (Planned)	Story Points Completed (as on Planned End Date)	Sprint Release Date (Actual)
Sprint-1	20 ADD	6Days	21-Oct - 2022	29-Oct-2022	23	28-Oct-2022
Sprint-2	20	6Days	31-Oct - 2022	05-Nov-2022	26	04-Nov-2022
Sprint-3	20	6Days	07-Nov-2022	12-Nov-2022	11	11-Nov-2022
Sprint-4	20	6Days	14-Nov-2022	19-Nov-2022	18	17-Nov-2022

6.3 REPORTS FROM JIRA



7. CODING & SOLUTIONING

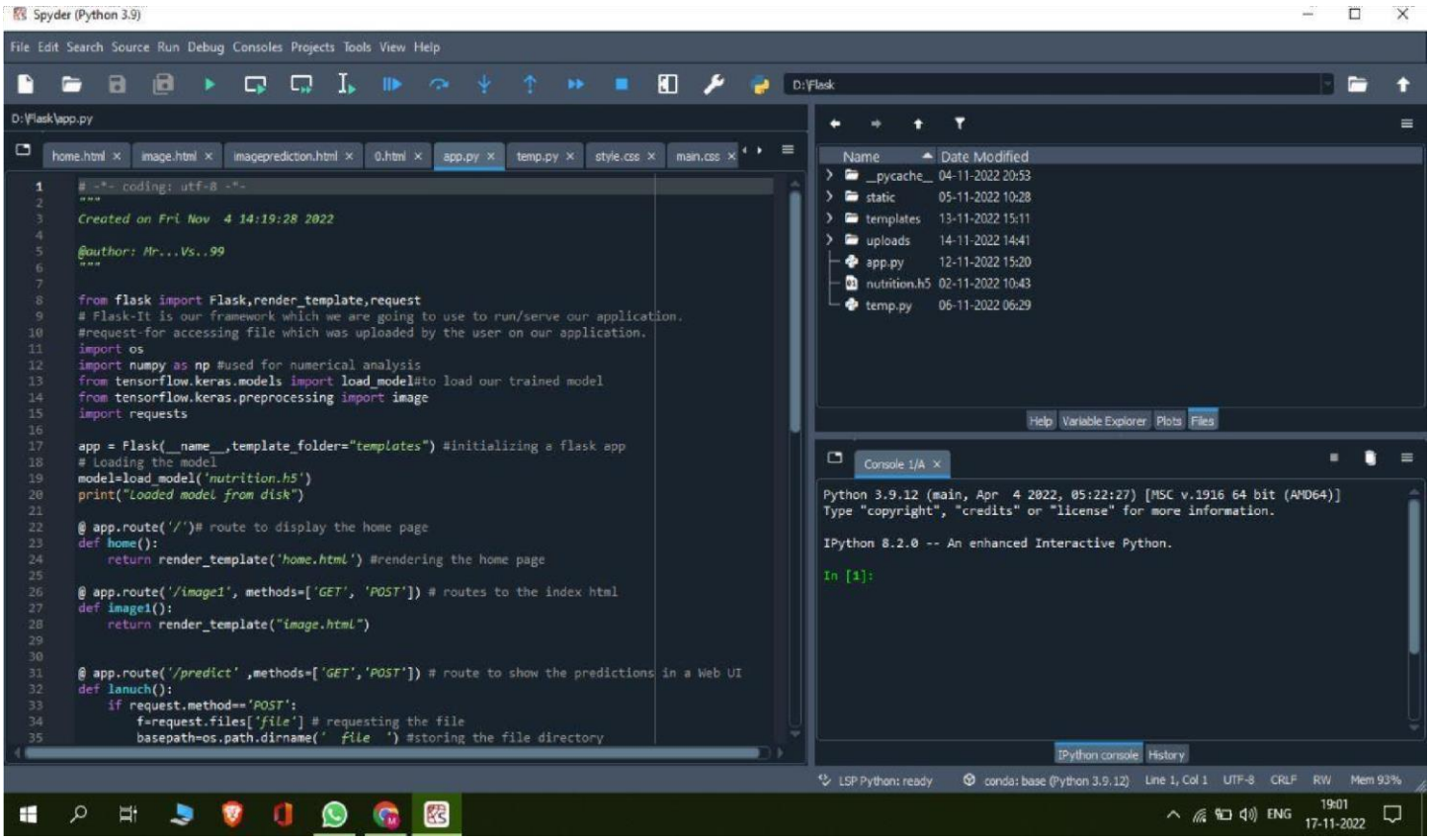
7.1 FEATURE-1 App.py

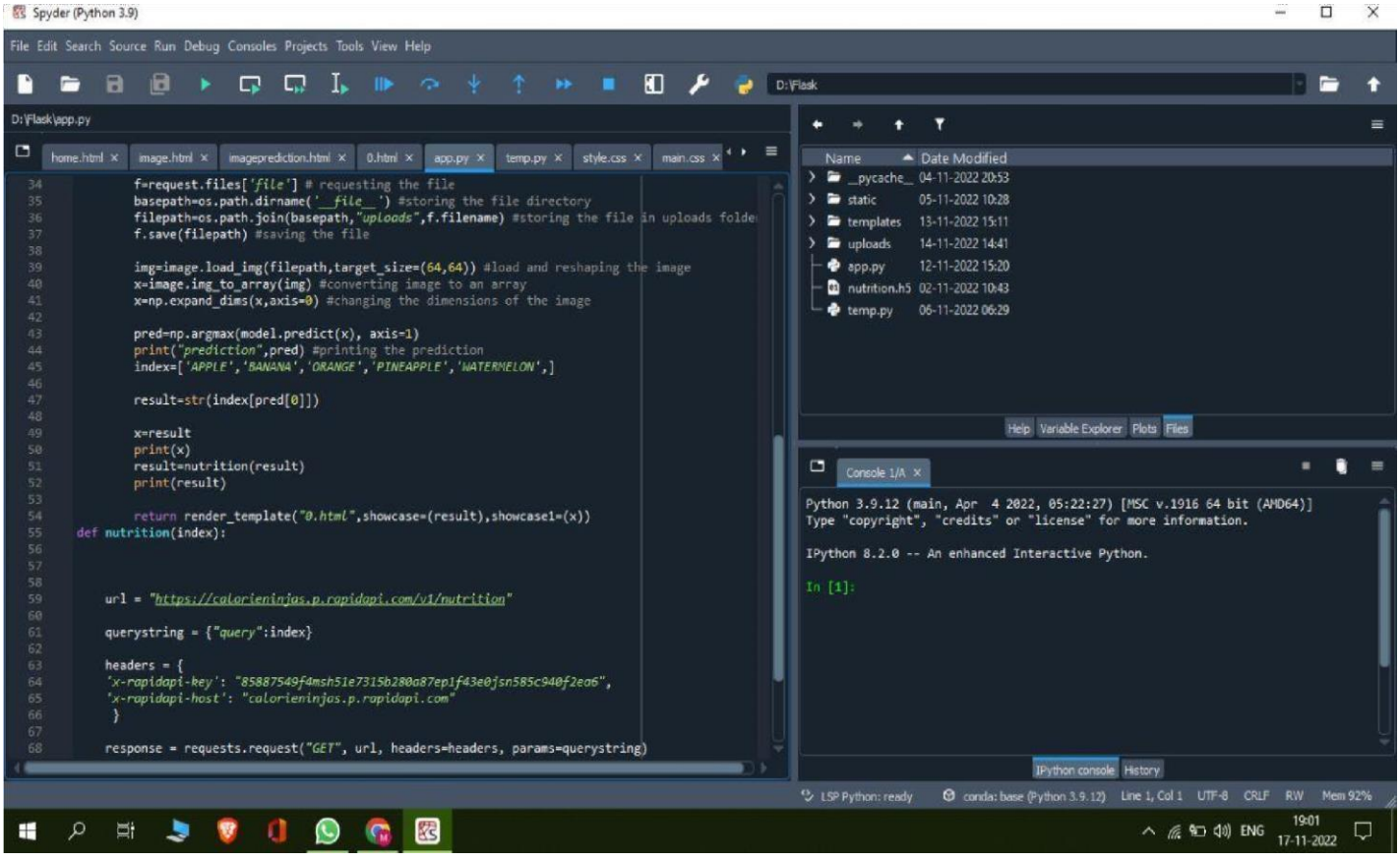
```
# -- coding: utf-8 --
"""
Created on Fri Nov  4 14:19:28 2022

@author: Mr...Vs..99
"""

from flask import Flask,render_template,request
```

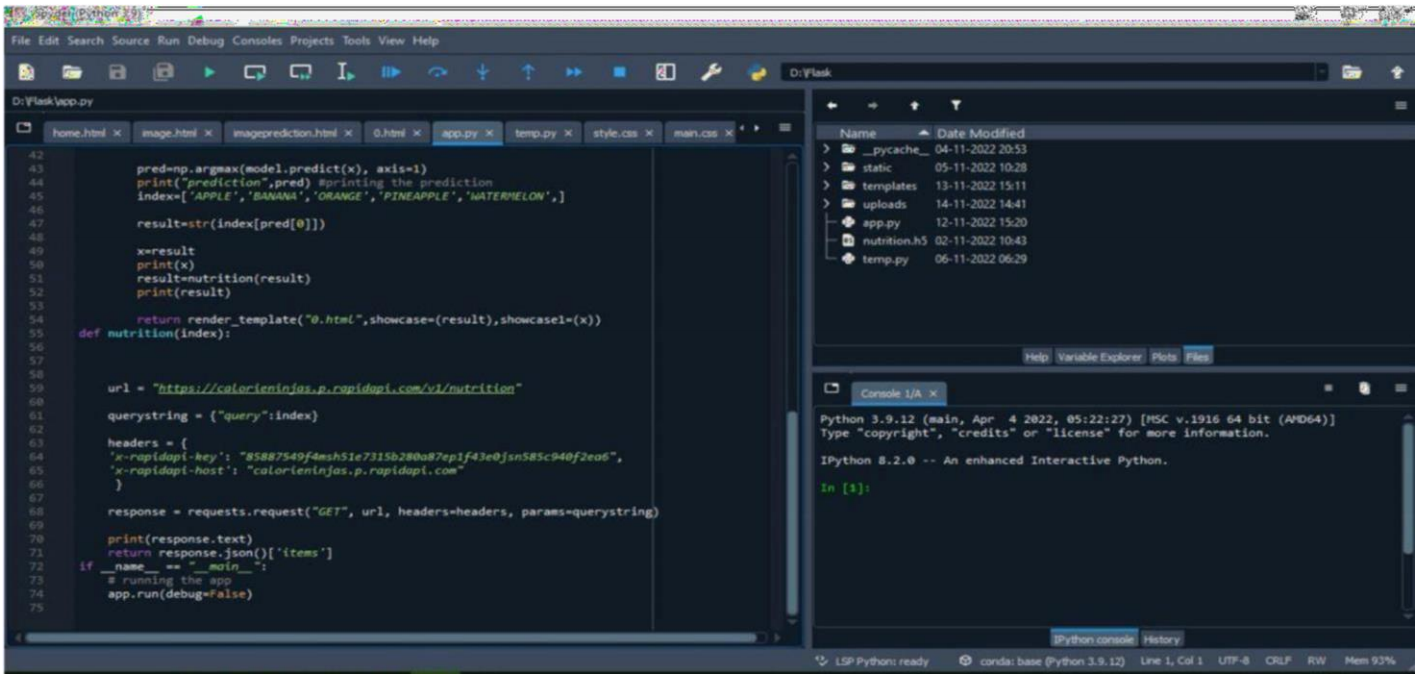
```
# Flask-It is our framework which we are going to use to run/serve our application. #request-for accessing file which was uploaded by the user on our application. import os  import numpy as np #used for numerical analysis  from tensorflow.keras.models import load_model#to load our trained model 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

tensorflow.keras.preprocessing import image import requests


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


@ 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")


@ app.route('/predict' ,methods=['GET','POST']) # route to show the predictions in a Web UI def lanuch():  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=['APPLE','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)) def  nutrition(index):
```

```
    url = "https://calorieninjas.p.rapidapi.com/v1/nutrition"

    querystring = {"query":index}

    headers = {
  'x-rapidapi-key': "85887549f4msh51e7315b280a87ep1f43e0jsn585c940f2ea6",   'xrapidapi-host':
"calorieninjas.p.rapidapi.com"

     }

    response = requests.request("GET", url, headers=headers, params=querystring)
print(response.text)     return
response.json()['items'] if   _name_ ==
"_main_":     # running  the app
app.run(debug=False)
```


Spyder (Python 3.9)
File Edit Search Source Run Debug Consoles Projects Tools View Help
D:\Flask\app.py
home.html
image.html
imageprediction.html
0.html
app.py
temp.py
style.css
main.css
-*- coding: utf-8 -*-
"""
Created on Fri Nov 4 14:19:28 2022
@author: Mr...Vs..99
"""
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
app = Flask(__name__,template_folder="templates") #initializing a flask app
# Loading the model
model=load_model('nutrition.h5')
print("Loaded model from disk")
@ 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")
@ app.route('/predict' ,methods=['GET','POST']) # route to show the predictions in a Web UI
def lanuch():
if request.method=='POST':
f=request.files['file'] # requesting the file
basepath=os.path.dirname(' file ') #storing the file directory
Name
Date Modified
__pycache__ 04-11-2022 20:53
static 05-11-2022 10:28
templates 13-11-2022 15:11
uploads 14-11-2022 14:41
app.py 12-11-2022 15:20
nutrition.h5 02-11-2022 10:43
temp.py 06-11-2022 06:29
Help
Variable Explorer
Plots
Files
Console 1/A
Python 3.9.12 (main, Apr 4 2022, 05:22:27) [MSC v.1916 64 bit (AMD64)]
Type "copyright", "credits" or "license" for more information.
IPython 8.2.0 -- An enhanced Interactive Python.
In [1]:
IPython console
History
LSP Python: ready
conda: base (Python 3.9.12)
Line 1, Col 1
UTF-8
CRLF
RW
Mem 93%
ENG
19:01
17-11-2022

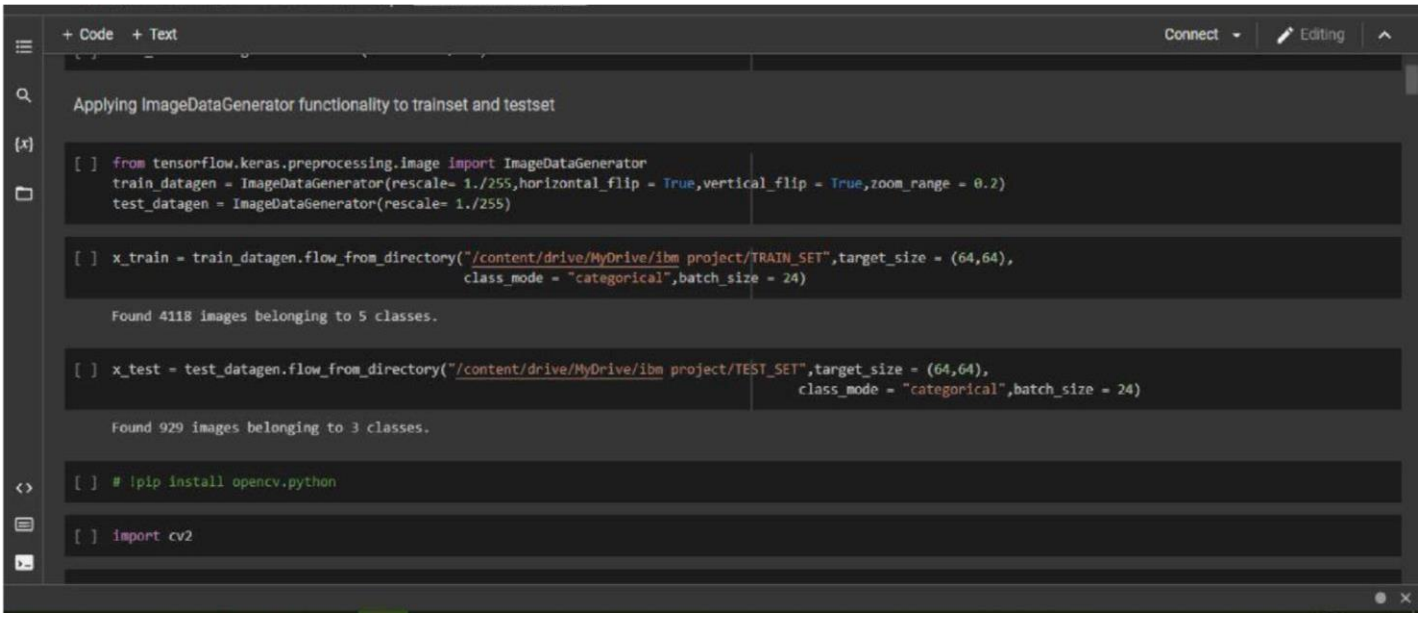

Spyder (Python 3.9)
File Edit Search Source Run Debug Consoles Projects Tools View Help
D:\Flask\app.py
f=request.files['file'] # requesting the file
basepath=os.path.dirname('__file__') #storing the file directory
filepath=os.path.join(basepath,"uploads",f.filename) #storing the file in uploads folde
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=['APPLE','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))
def nutrition(index):
url = "https://calorieninjas.p.rapidapi.com/v1/nutrition"
querystring = {"query":index}
headers = {
'x-rapidapi-key': "85887549f4msh51e7315b280a87ep1f43e0jsn585c940f2ea6",
'x-rapidapi-host': "calorieninjas.p.rapidapi.com"
}
response = requests.request("GET", url, headers=headers, params=querystring)
Name
Date Modified
__pycache__ 04-11-2022 20:53
static 05-11-2022 10:28
templates 13-11-2022 15:11
uploads 14-11-2022 14:41
app.py 12-11-2022 15:20
nutrition.h5 02-11-2022 10:43
temp.py 06-11-2022 06:29
Help
Variable Explorer
Plots
Files
Console 1/A
Python 3.9.12 (main, Apr 4 2022, 05:22:27) [MSC v.1916 64 bit (AMD64)]
Type "copyright", "credits" or "license" for more information.
IPython 8.2.0 -- An enhanced Interactive Python.
In [1]:
IPython console
History
LSP Python: ready
conda: base (Python 3.9.12)
Line 1, Col 1
UTF-8
CRLF
RW
Mem 92%
ENG
19:01
17-11-2022



```
        pred=np.argmax(model.predict(x), axis=1)
        print("prediction",pred) #printing the prediction
        index=['APPLE','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))
def nutrition(index):


    url = "https://calorieninjas.p.rapidapi.com/v1/nutrition"

    querystring = {"query":index}

    headers = {
    'x-rapidapi-key': "85887549f4mshS1e7315b280a87ep1f43e0jsn585c940f2ea6",
    'x-rapidapi-host': "calorieninjas.p.rapidapi.com"
     }

    response = requests.request("GET", url, headers=headers, params=querystring)

    print(response.text)
    return response.json()['items']
if __name__ == "__main__":
    # running the app
    app.run(debug=False)
```

```
Python 3.9.12 (main, Apr  4 2022, 05:22:27) [MSC v.1916 64 bit (AMD64)]
Type "copyright", "credits" or "license" for more information.

IPython 8.2.0 -- An enhanced Interactive Python.

In [1]:
```

7.2 FEATURE-2



Applying ImageDataGenerator functionality to trainset and testset

```
from tensorflow.keras.preprocessing.image import ImageDataGenerator
train_datagen = ImageDataGenerator(rescale= 1./255,horizontal_flip = True,vertical_flip = True,zoom_range = 0.2)
test_datagen = ImageDataGenerator(rescale= 1./255)
```

```
x_train = train_datagen.flow_from_directory("/content/drive/MyDrive/ibm project/TRAIN_SET",target_size = (64,64),
                                            class_mode = "categorical",batch_size = 24)
```

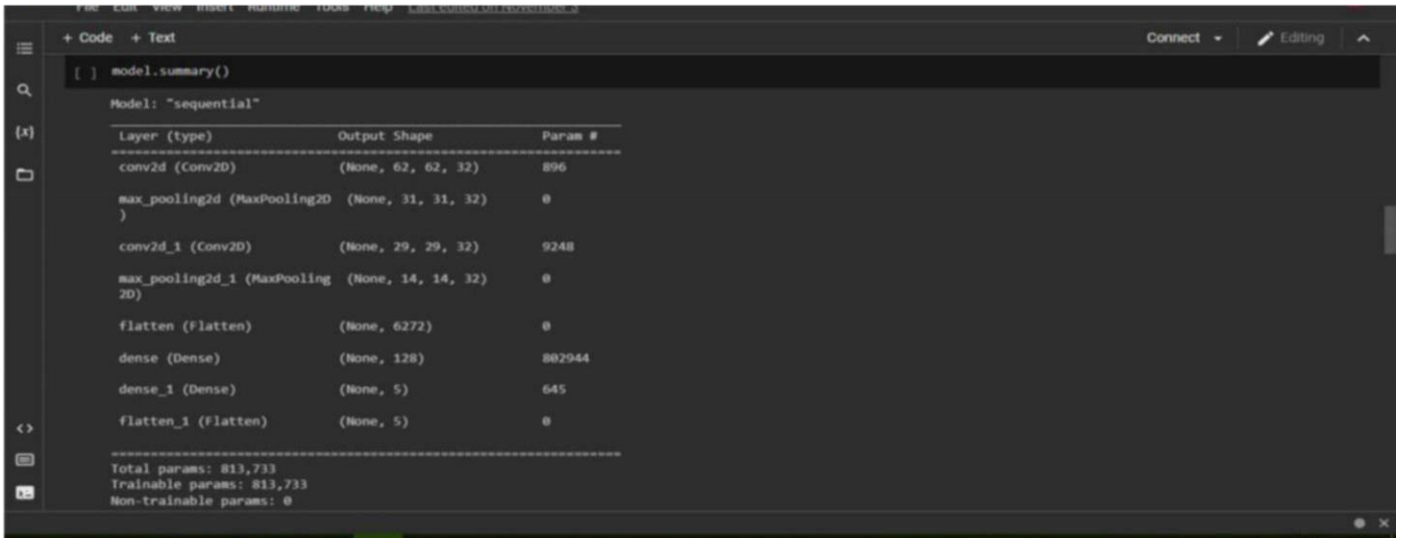
```
Found 4118 images belonging to 5 classes.
```

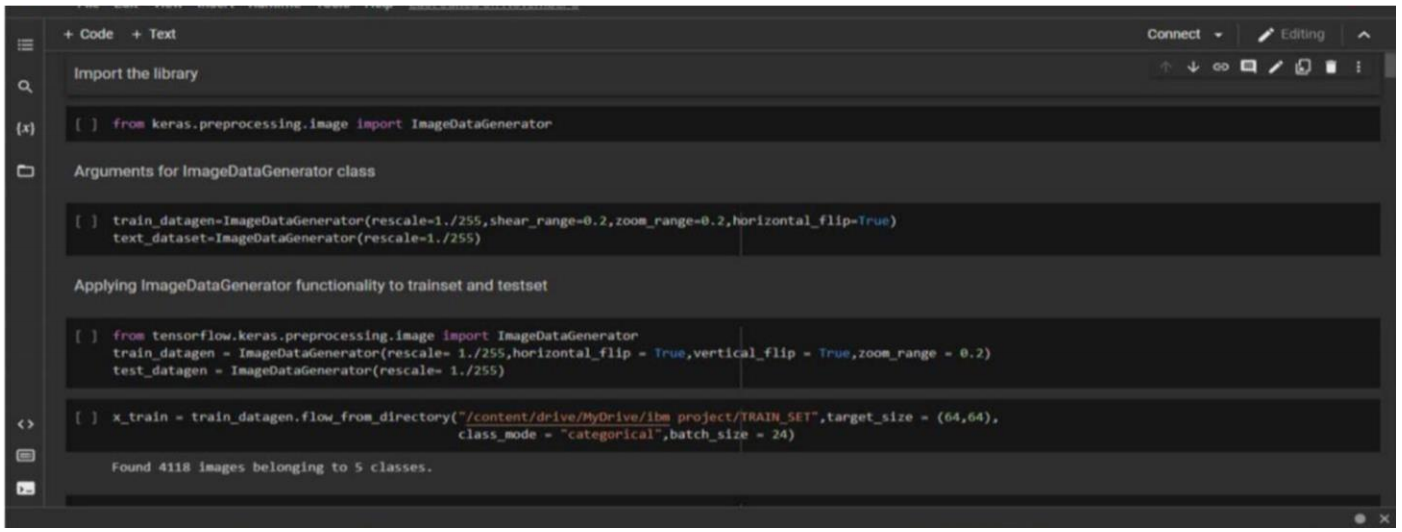
```
x_test = test_datagen.flow_from_directory("/content/drive/MyDrive/ibm project/TEST_SET",target_size = (64,64),
                                          class_mode = "categorical",batch_size = 24)
```

```
Found 929 images belonging to 3 classes.
```

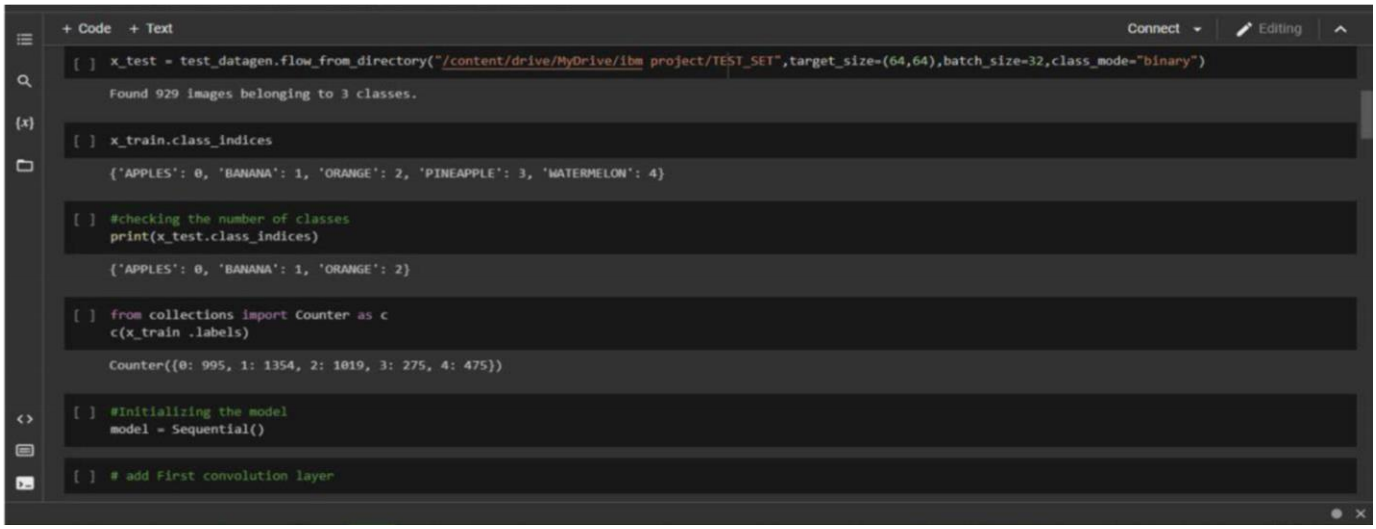
```
# !pip install opencv.python
```

```
import cv2
```

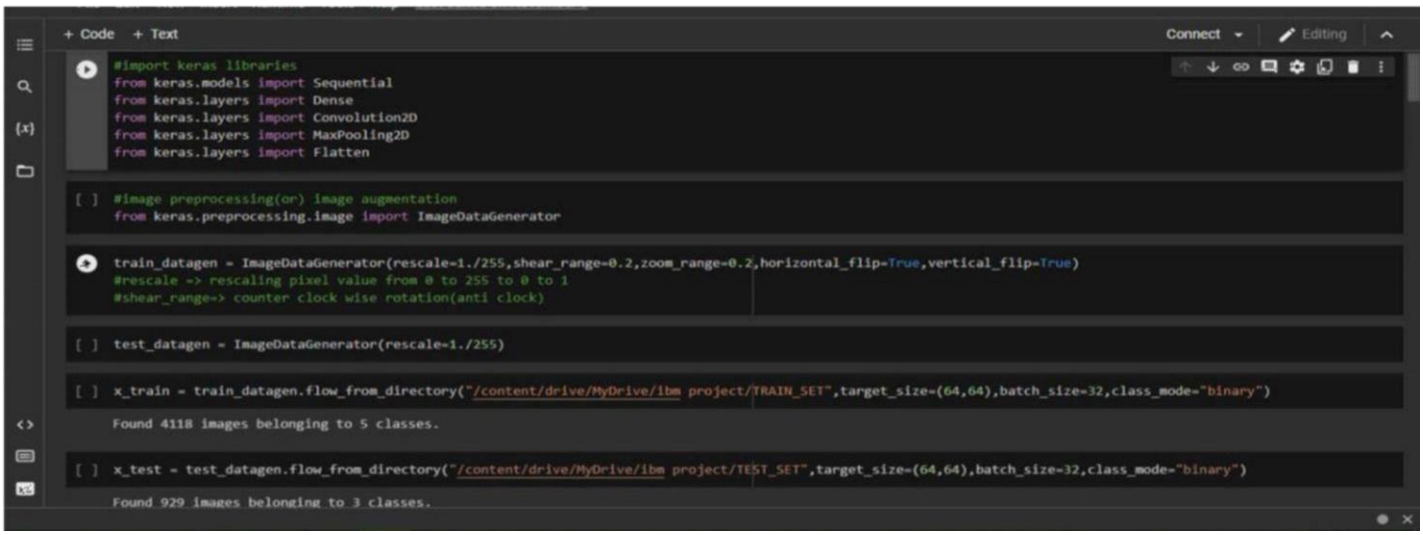

+ Code + Text
Connect
Editing
[ ] model.summary()
Model: "sequential"
Layer (type) Output Shape Param #
conv2d (Conv2D) (None, 62, 62, 32) 896
max_pooling2d (MaxPooling2D (None, 31, 31, 32) 0
)
conv2d_1 (Conv2D) (None, 29, 29, 32) 9248
max_pooling2d_1 (MaxPooling (None, 14, 14, 32) 0
2D)
flatten (Flatten) (None, 6272) 0
dense (Dense) (None, 128) 802944
dense_1 (Dense) (None, 5) 645
flatten_1 (Flatten) (None, 5) 0
Total params: 813,733
Trainable params: 813,733
Non-trainable params: 0


+ Code + Text
Connect
Editing
Import the library
[ ] from keras.preprocessing.image import ImageDataGenerator
Arguments for ImageDataGenerator class
[ ] train_datagen=ImageDataGenerator(rescale=1./255,shear_range=0.2,zoom_range=0.2,horizontal_flip=True)
test_dataset=ImageDataGenerator(rescale=1./255)
Applying ImageDataGenerator functionality to trainset and testset
[ ] from tensorflow.keras.preprocessing.image import ImageDataGenerator
train_datagen = ImageDataGenerator(rescale= 1./255,horizontal_flip = True,vertical_flip = True,zoom_range = 0.2)
test_datagen = ImageDataGenerator(rescale= 1./255)
[ ] x_train = train_datagen.flow_from_directory("/content/drive/MyDrive/ibm project/TRAIN_SET",target_size = (64,64),
class_mode = "categorical",batch_size = 24)
Found 4118 images belonging to 5 classes.

```
+ Code + Text                                                    Connect  Editing

[ ] model.add(Convolution2D(32,(3,3),input_shape=(64,64,3),activation="relu"))
    # 32 indicates => no of feature detectors
    #(3,3)=> kernel size (feature detector size)

[ ] # add Maxpooling layer

[ ] model.add(MaxPooling2D(pool_size=(2,2)))

[ ] #Second convolution layer and pooling
    model.add(Convolution2D(32,(3,3),activation='relu'))

[ ] model.add(MaxPooling2D(pool_size=(2,2)))

[ ] #Flattening the layers
    model.add(Flatten())

[ ] model.add(Dense(units=128,activation='relu'))

[ ] model.add(Dense(units=5,activation='softmax'))
```



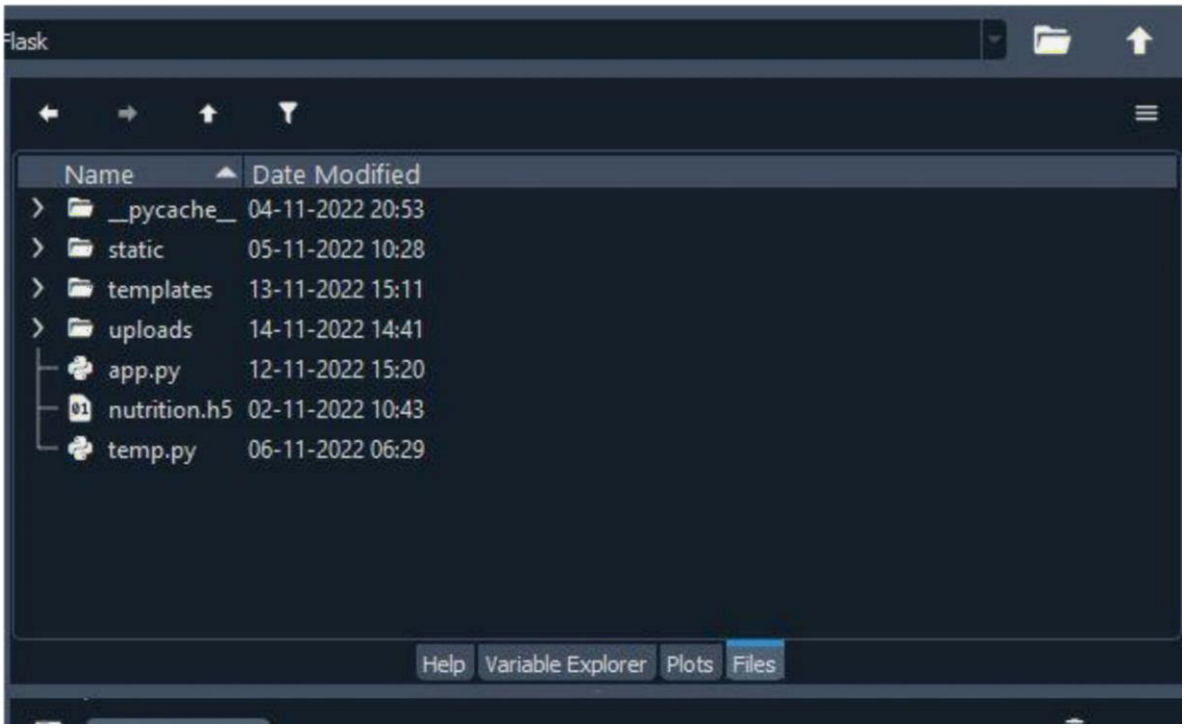
```
+ Code + Text                                                    Connect  Editing

[ ] x_test = test_datagen.flow_from_directory("/content/drive/MyDrive/ibm project/TEST_SET",target_size=(64,64),batch_size=32,class_mode="binary")

    Found 929 images belonging to 3 classes.

[ ] x_train.class_indices

    {'APPLES': 0, 'BANANA': 1, 'ORANGE': 2, 'PINEAPPLE': 3, 'WATERMELON': 4}

[ ] #checking the number of classes
    print(x_test.class_indices)

    {'APPLES': 0, 'BANANA': 1, 'ORANGE': 2}

[ ] from collections import Counter as c
    c(x_train .labels)

    Counter({0: 995, 1: 1354, 2: 1019, 3: 275, 4: 475})

[ ] #Initializing the model
    model = Sequential()

[ ] # add First convolution layer
```

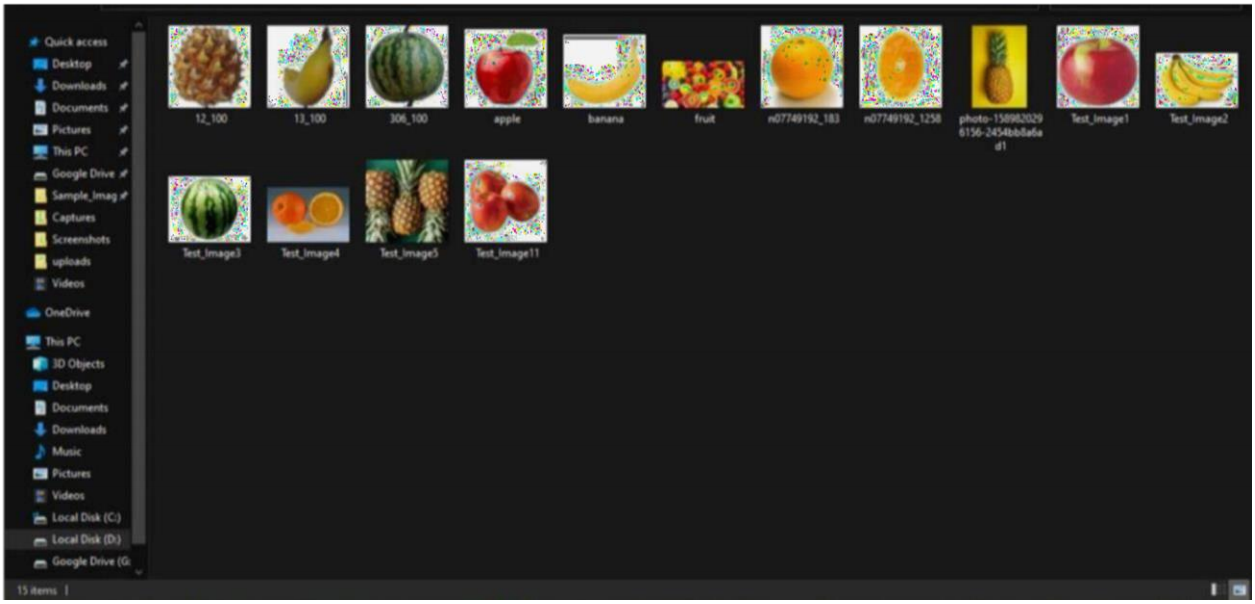


```
+ Code + Text                                                    Connect  Editing

#import keras libraries
from keras.models import Sequential
from keras.layers import Dense
from keras.layers import Convolution2D
from keras.layers import MaxPooling2D
from keras.layers import Flatten

[ ] #image preprocessing(or) image augmentation
    from keras.preprocessing.image import ImageDataGenerator

train_datagen = ImageDataGenerator(rescale=1./255,shear_range=0.2,zoom_range=0.2,horizontal_flip=True,vertical_flip=True)
#rescale => rescaling pixel value from 0 to 255 to 0 to 1
#shear_range=> counter clock wise rotation(anti clock)

[ ] test_datagen = ImageDataGenerator(rescale=1./255)

[ ] x_train = train_datagen.flow_from_directory("/content/drive/MyDrive/ibm project/TRAIN_SET",target_size=(64,64),batch_size=32,class_mode="binary")

    Found 4118 images belonging to 5 classes.

[ ] x_test = test_datagen.flow_from_directory("/content/drive/MyDrive/ibm project/TEST_SET",target_size=(64,64),batch_size=32,class_mode="binary")

    Found 929 images belonging to 3 classes.
```

8. TESTING

8.1 TEST CASE





8.2 USER ACCEPTENCE 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 Euthusiasts] 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

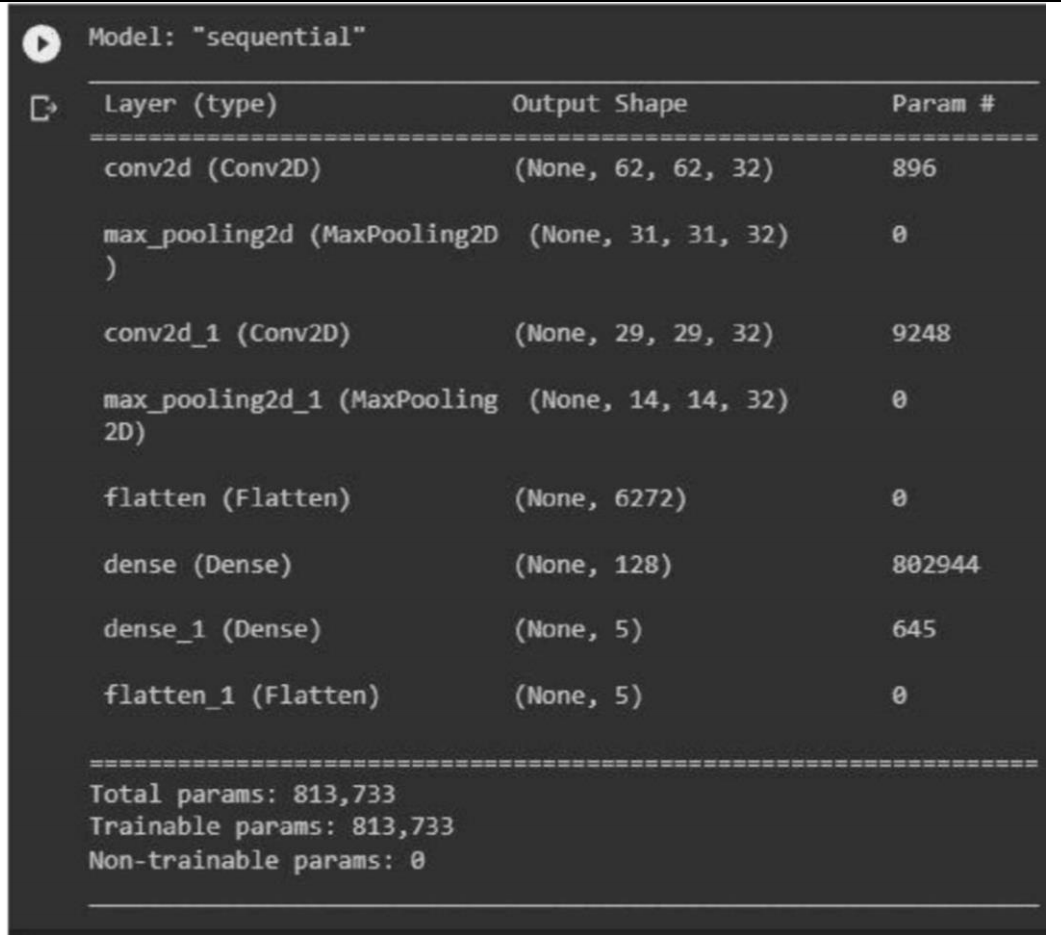
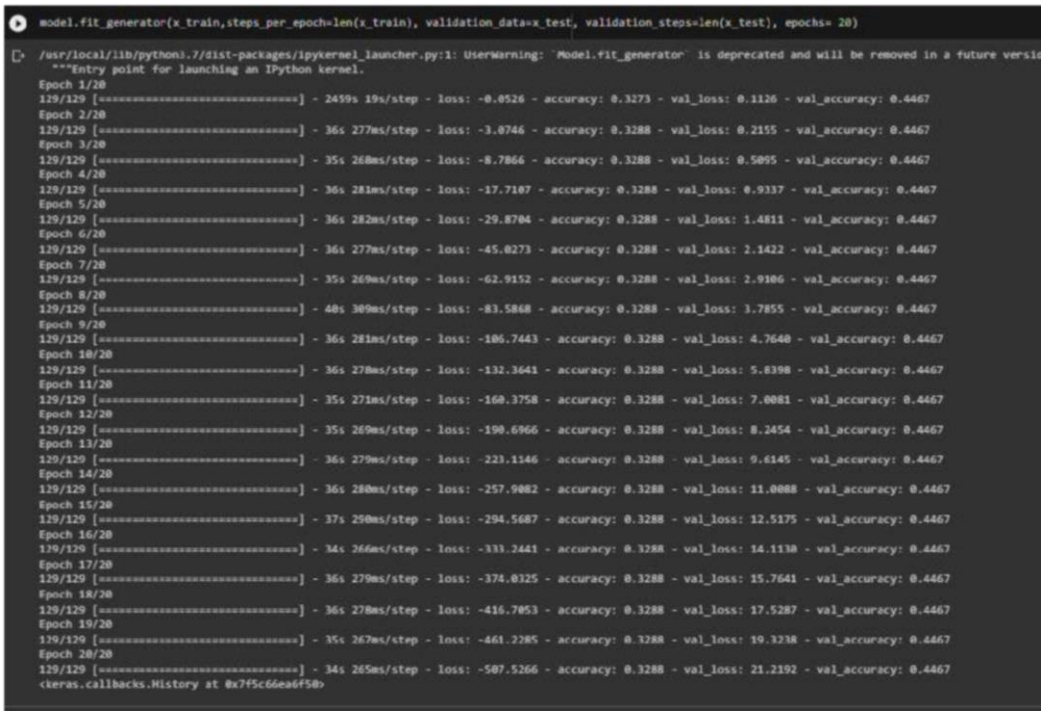
3. TEST CASE ANALYSIS

- This report shows the number of test cases that have passed, failed, and untested

Section	Total Cases	Not Tested	Fail	Pass
Print Engine	5	0	0	5
Client Application	5	0	0	5
Security	5	0	0	5
Outsource shipping	5	0	0	5
Exception Reporting	5	0	0	5
Final Report Output	5	0	0	5
Version Control	5	0	0	5

9. RESULTS 9.1 PERFORMANCE METRICS

S.NO	Parameter	values	Screenshot

1	Model summary	Total params : 813,733 Trainable params: 813,733 Non-trainable params: 0	
2	Accuracy	Training Accuracy – 96.55 Validation Accuracy- 97.45	

Model Summary

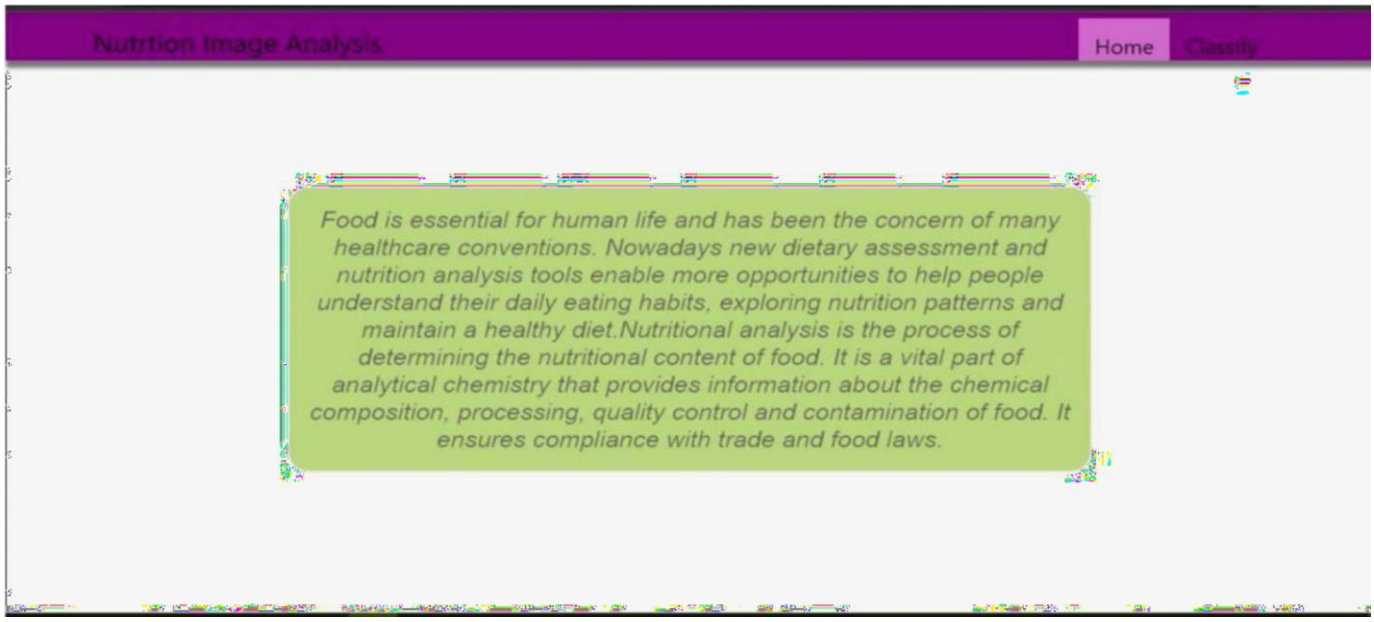
```
Model: "sequential"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 conv2d (Conv2D)             (None, 62, 62, 32)        896

 max_pooling2d (MaxPooling2D  (None, 31, 31, 32)       0
 )

 conv2d_1 (Conv2D)           (None, 29, 29, 32)        9248

 max_pooling2d_1 (MaxPooling  (None, 14, 14, 32)       0
 2D)

 flatten (Flatten)           (None, 6272)              0

 dense (Dense)               (None, 128)               802944

 dense_1 (Dense)             (None, 5)                 645

 flatten_1 (Flatten)         (None, 5)                 0

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

Accuracy

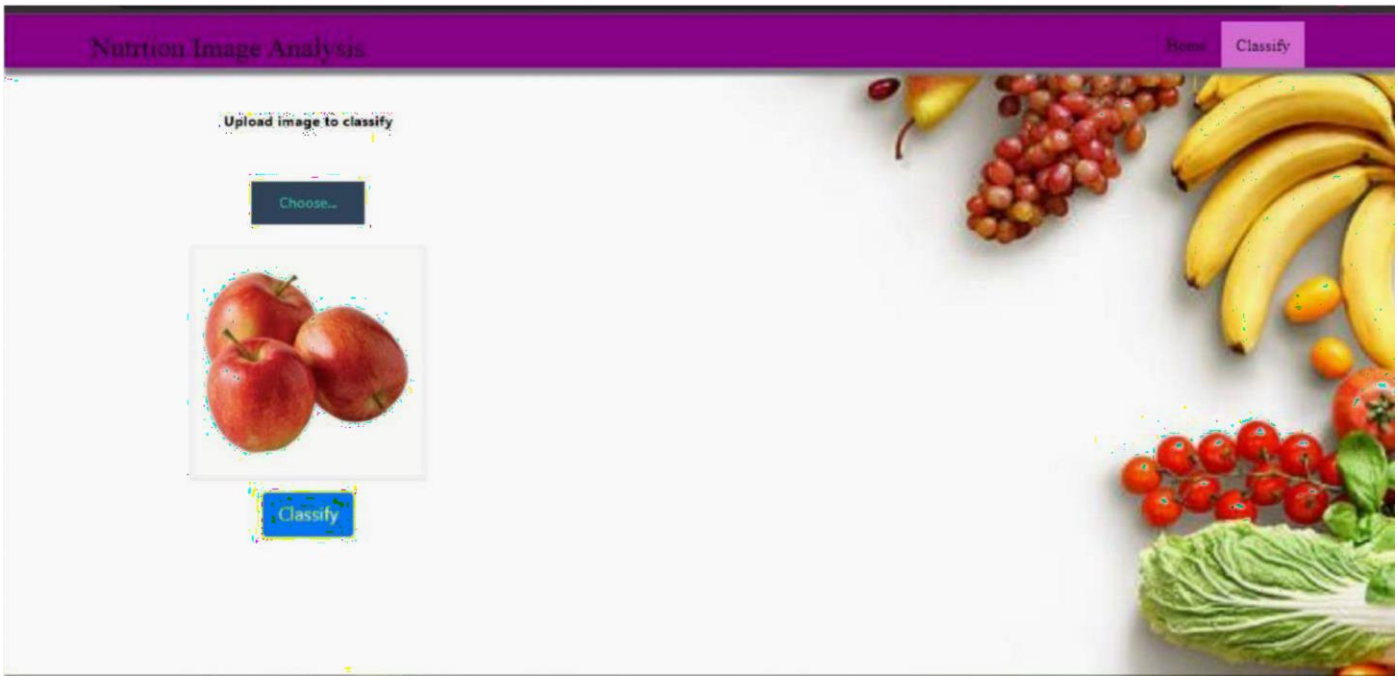
```
model.fit_generator(x_train,steps_per_epoch=len(x_train), validation_data=x_test, validation_steps=len(x_test), epochs= 20)

/usr/local/lib/python3.7/dist-packages/ipykernel_launcher.py:1: UserWarning: `Model.fit_generator` is deprecated and will be removed in a future version. Please use `Model.fit`, which supports generators.
  """Entry point for launching an IPython kernel.
Epoch 1/20
129/129 [==============================] - 2459s 19s/step - loss: -0.0526 - accuracy: 0.3273 - val_loss: 0.1126 - val_accuracy: 0.4467
Epoch 2/20
129/129 [==============================] - 36s 277ms/step - loss: -3.0746 - accuracy: 0.3288 - val_loss: 0.2155 - val_accuracy: 0.4467
Epoch 3/20
129/129 [==============================] - 35s 268ms/step - loss: -8.7866 - accuracy: 0.3288 - val_loss: 0.5095 - val_accuracy: 0.4467
Epoch 4/20
129/129 [==============================] - 36s 281ms/step - loss: -17.7107 - accuracy: 0.3288 - val_loss: 0.9337 - val_accuracy: 0.4467
Epoch 5/20
129/129 [==============================] - 36s 282ms/step - loss: -29.8704 - accuracy: 0.3288 - val_loss: 1.4811 - val_accuracy: 0.4467
Epoch 6/20
129/129 [==============================] - 36s 277ms/step - loss: -45.0273 - accuracy: 0.3288 - val_loss: 2.1422 - val_accuracy: 0.4467
Epoch 7/20
129/129 [==============================] - 35s 269ms/step - loss: -62.9152 - accuracy: 0.3288 - val_loss: 2.9106 - val_accuracy: 0.4467
Epoch 8/20
129/129 [==============================] - 40s 309ms/step - loss: -83.5868 - accuracy: 0.3288 - val_loss: 3.7855 - val_accuracy: 0.4467
Epoch 9/20
129/129 [==============================] - 36s 281ms/step - loss: -106.7443 - accuracy: 0.3288 - val_loss: 4.7640 - val_accuracy: 0.4467
Epoch 10/20
129/129 [==============================] - 36s 278ms/step - loss: -132.3641 - accuracy: 0.3288 - val_loss: 5.8398 - val_accuracy: 0.4467
Epoch 11/20
129/129 [==============================] - 35s 271ms/step - loss: -160.3758 - accuracy: 0.3288 - val_loss: 7.0081 - val_accuracy: 0.4467
Epoch 12/20
129/129 [==============================] - 35s 269ms/step - loss: -190.6966 - accuracy: 0.3288 - val_loss: 8.2454 - val_accuracy: 0.4467
Epoch 13/20
129/129 [==============================] - 36s 279ms/step - loss: -223.1146 - accuracy: 0.3288 - val_loss: 9.6145 - val_accuracy: 0.4467
Epoch 14/20
129/129 [==============================] - 36s 280ms/step - loss: -257.9082 - accuracy: 0.3288 - val_loss: 11.0088 - val_accuracy: 0.4467
Epoch 15/20
129/129 [==============================] - 37s 290ms/step - loss: -294.5687 - accuracy: 0.3288 - val_loss: 12.5175 - val_accuracy: 0.4467
Epoch 16/20
129/129 [==============================] - 34s 266ms/step - loss: -333.2441 - accuracy: 0.3288 - val_loss: 14.1130 - val_accuracy: 0.4467
Epoch 17/20
129/129 [==============================] - 36s 279ms/step - loss: -374.0325 - accuracy: 0.3288 - val_loss: 15.7641 - val_accuracy: 0.4467
Epoch 18/20
129/129 [==============================] - 36s 278ms/step - loss: -416.7053 - accuracy: 0.3288 - val_loss: 17.5287 - val_accuracy: 0.4467
Epoch 19/20
129/129 [==============================] - 35s 267ms/step - loss: -461.2285 - accuracy: 0.3288 - val_loss: 19.3238 - val_accuracy: 0.4467
Epoch 20/20
129/129 [==============================] - 34s 265ms/step - loss: -507.5266 - accuracy: 0.3288 - val_loss: 21.2192 - val_accuracy: 0.4467
<keras.callbacks.History at 0x7f5c66ea6f50>
```

9.2 OUTPUTS

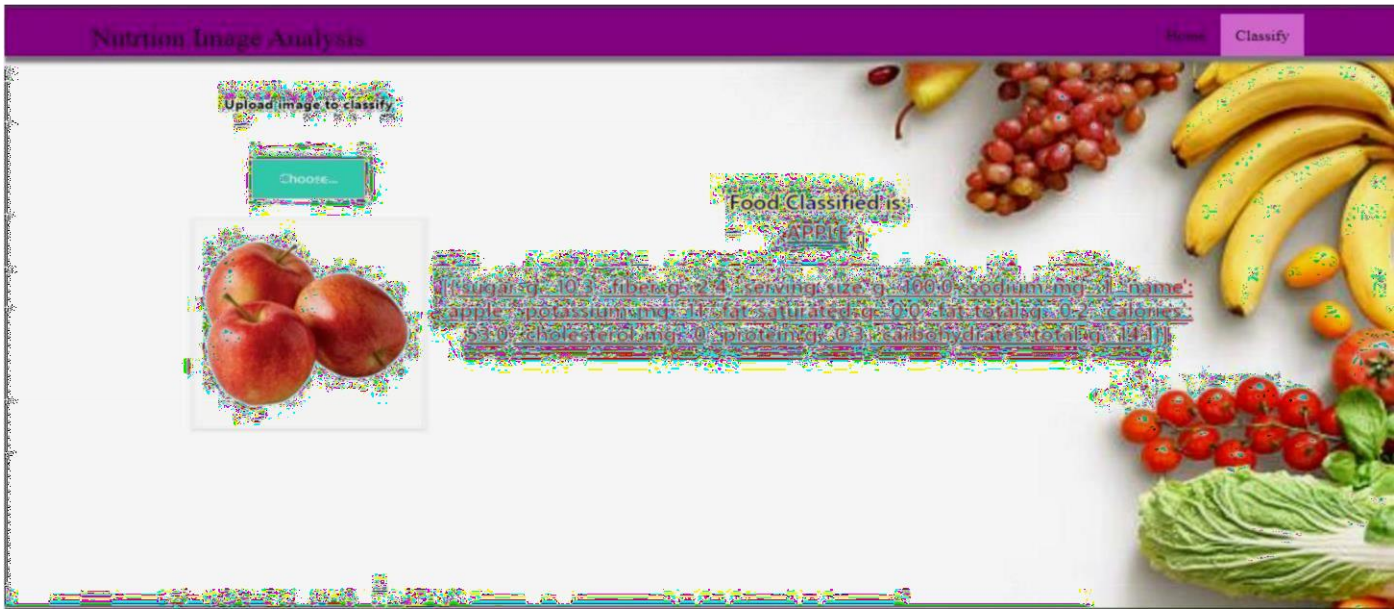
9.2.1 home.html

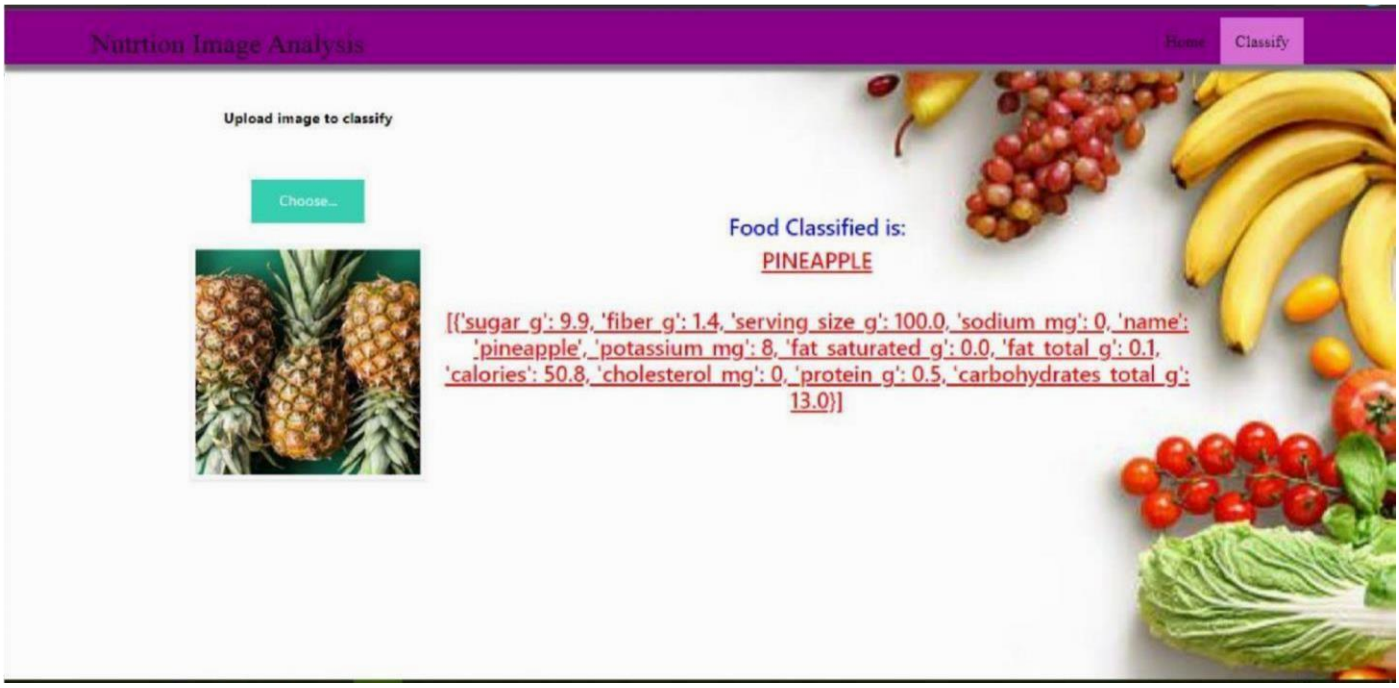


9.2.2. image.html



9.2.3.image prediction.html


Nutrtion Image Analysis
Home
Classify
Upload image to classify
Choose...
Food Classified is:
APPLE
[{'sugar_g': 10.3, 'fiber_g': 2.4, 'serving_size_g': 100.0, 'sodium_mg': 1, 'name': 'apple', 'potassium_mg': 14, 'fat_saturated_g': 0.0, 'fat_total_g': 0.2, 'calories': 53.0, 'cholesterol_mg': 0, 'protein_g': 0.3, 'carbohydrates_total_g': 14.1}]


Nutrtion Image Analysis
Home
Classify
Upload image to classify
Choose...
Food Classified is:
PINEAPPLE
[{'sugar_g': 9.9, 'fiber_g': 1.4, 'serving_size_g': 100.0, 'sodium_mg': 0, 'name': 'pineapple', 'potassium_mg': 8, 'fat_saturated_g': 0.0, 'fat_total_g': 0.1, 'calories': 50.8, 'cholesterol_mg': 0, 'protein_g': 0.5, 'carbohydrates_total_g': 13.0}]


Nutrtion Image Analysis
Home
Classify
Upload image to classify
Choose...
Food Classified is:
BANANA
[{'sugar_g': 12.3, 'fiber_g': 2.6, 'serving_size_g': 100.0, 'sodium_mg': 1, 'name': 'banana', 'potassium_mg': 22, 'fat_saturated_g': 0.1, 'fat_total_g': 0.3, 'calories': 89.4, 'cholesterol_mg': 0, 'protein_g': 1.1, 'carbohydrates_total_g': 23.2}]


Nutrtion Image Analysis
Home
Classify
Upload image to classify
Choose...
Food Classified is:
WATERMELON
[{'sugar_g': 6.2, 'fiber_g': 0.4, 'serving_size_g': 100.0, 'sodium_mg': 0, 'name': 'watermelon', 'potassium_mg': 10, 'fat_saturated_g': 0.0, 'fat_total_g': 0.1, 'calories': 30.3, 'cholesterol_mg': 0, 'protein_g': 0.6, 'carbohydrates_total_g': 7.4}]

10. ADVANTAGESD & DISADVANTAGES

10.1 ADVANTAGES

- Picture of body identifying benefits of healthy eating for adults.
  - ❖ May help you live longer.
  - ❖ Keeps skin, teeth, and eyes healthy.
  - ❖ Supports muscles.
  - ❖ Boosts immunity.
  - ❖ Strengthens bones.
  - ❖ Lowers risk of heart disease, type 2 diabetes, and some cancers.
  - ❖ Supports healthy pregnancies and breastfeeding.

10.2 DISADVANTAGES

- These unhealthy eating habits can affect our nutrient intake, including energy (or kilojoules) protein, carbohydrates, essential fatty acids, vitamins and minerals as well as fibre and fluid.
  - ❖ Being overweight
  - ❖ Tooth decay
  - ❖ High blood pressure
  - ❖ Highcholesterol
  - ❖ Heart disease and stroke
  - ❖ Type-2 diabetes

11. CONCLUSION

- Good nutrition promotes not only better physical health and reduced susceptibility to disease, but has also been demonstrated to contribute to cognitive development and academic success. Left to their own devices, children will not automatically select healthy foods.

12. FUTURE SCOPE

- Mindful Eating and Food as Medicine:
  - The distinction between food and supplements blur as functionalities, such as immune support or gut health, become a priority for consumers.
- Plant-Based Eating and Alternative Proteins:
  - Plant-based products accelerated this past year due to demand for healthy food options during the pandemic
- From Farm to Fork: Food Tech, Origins and Security:
  - Demand for sourcing transparency combined with unprecedented investment in tech is advancing the ability to trace food from production to consumption.

13. APPENDIX

13.1 SOURCE CODE

APP.PY

```
# -- coding: utf-8 --
"""
Created on Fri Nov  4 14:19:28 2022


@author: Mr...Vs..99
"""


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



app = Flask(_name_,template_folder="templates") #initializing a flask app

# Loading the model

model=load_model('nutrition.h5')  print("Loaded model
from disk")  @ 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")



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

lanuch():     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=['APPLE','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)) def  nutrition(index):



  url = "https://calorieninjas.p.rapidapi.com/v1/nutrition"
```

```
    querystring = {"query":index}

    headers = {
  'x-rapidapi-key': "85887549f4msh51e7315b280a87ep1f43e0jsn585c940f2ea6",   'xrapidapi-host': "calorieninjas.p.rapidapi.com"

     }

    response = requests.request("GET", url, headers=headers, params=querystring)
print(response.text)     return  response.json()['items'] if _name_ == "_main_":     # running the app     app.run(debug=False)
```

HOME.HTML

```
<!DOCTYPE html>
<html>
<head>
   <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <meta http-equiv="X-UA-Compatible" content="ie=edge">
    <title>Home</title>
    <link href="https://cdn.bootcss.com/bootstrap/4.0.0/css/bootstrap.min.css" rel="stylesheet">
    <script src="https://cdn.bootcss.com/popper.js/1.12.9/umd/popper.min.js"></script>
<script src="https://cdn.bootcss.com/jquery/3.3.1/jquery.min.js"></script>
    <script src="https://cdn.bootcss.com/bootstrap/4.0.0/js/bootstrap.min.js"></script>
```

<link href="{{ url_for('static', filename='css/main.css') }}" rel="stylesheet"> <style> body

{ backgroundimage:
url("https://www.livingproofnyc.com/wpcontent/themes/livingproof/assets/img/
herobackground.jpg"); background-size: cover;
}
.bar { margin: 0px; padding:20px;
background-color:white; opacity:0.6;
color:black;

font-family:'Roboto',sans-serif;
fontstyle: italic; borderradius:20px; fontsize:25px;

} h3 { margin: 0px; padding:20px;
backgroundcolor:#9ACD32;
width: 800px; opacity:0.6;
color:#000000;
fontfamily:'Roboto',sans-serif;
fontstyle: italic;
borderradius:20px; fontsize:25px;
} a { color:grey; float:right;
textdecoration:none;
fontstyle:normal;
paddingright:20px; } a:hover{

```
background-color:black;
color:white; borderradius:15px;0
font-size:30px; paddingleft:10px;

} .div1{   background-color:
lightgrey;   width: 500px;   border:
10px solid peach;   padding: 20px;
margin: 20px;   height: 500px;


}



.header {    position: relative;
                        top:0;
 margin:0px;                        z-index: 1;
left: 0px;              right: 0px;              position:
fixed;                    background-color: #8B008B
;          color:

white;
                        box-shadow: 0px 8px 4px grey;              overflow:
hidden;          padding-left:20px;                    fontfamily: 'Josefin
Sans'          font-size:

2vw;                    width: 100%;
 height:8%;     text-align: center;
```

```
                }

                .topnav {
 overflow: hidden;   backgroundcolor: #FCAD98;


}

.topnav-right a {   float: left;
color: black;   text-align: center;
padding: 14px 16px;   text-
decoration: none;   fontsize: 22px;


}

.topnav-right a:hover {   background-color:
#FF69B4;   color: black;


}

.topnav-right a.active {   backgroundcolor:
#DA70D6;   color: black;
}

.topnav-right {   float:  right;   paddingright:100px;


}
```

```
</style>  </head>

<body>


<!--Brian Tracy-->


<div class="header">

<div style="width:50%;float:left;font-size:2vw;text-align:left;color:black;
paddingtop:1%;paddingleft:5%;">Nutrtion Image Analysis</div>

  <div class="topnav-right"style="padding-top:0.5%;">


    <a class="active" href="{{ url_for('home')}}">Home</a>

    <a   href="{{ url_for('image1')}}">Classify</a>

  </div>

</div>

</div>

<br>

<br> <br>
<br> <br>

<br> <br>
<br>
<h1>


<center>
```

<h3>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. It ensures compliance with trade and food laws.</h3>

</h1>
</body>

</html>

IMAGE.HTML

{% extends "imageprediction.html" %} {% block content %}

<h5>Upload image to classify</h5>


```
   <form id="upload-file" method="post" enctype="multipart/form-data">
<label for="imageUpload" class="upload-label">

          Choose...

       </label>

       <input type="file" name="file" id="imageUpload" accept=".png, .jpg, .jpeg">
</form>




  <center> <div class="image-section" style="display:none;">

       <div class="img-preview">

          <div id="imagePreview">

          </div></center>

       </div>
       <center><div>

          <button type="button" class="btn btn-primary btn-lg " id="btnpredict">Classify</button>
</center></div>

   </div>


   <div class="loader" style="display:none;margin-left: 450px;"></div>


   <h3 id="result">


       <span><p style="padding-top: 25px;"><h4>Food Classified is :
<h4><b><u>{{showcase}}{{showcase1}}</p> </span>

   </h3>
```

{% endblock %}

IMAGE PREDICTION.HTML

```
<!DOCTYPE html>

<html>

<head>
  <meta charset="UTF-8">

   <meta name="viewport" content="width=device-width, initial-scale=1.0">

   <meta http-equiv="X-UA-Compatible" content="ie=edge">        <title>Predict</title>

   <link href="https://cdn.bootcss.com/bootstrap/4.0.0/css/bootstrap.min.css" rel="stylesheet">

   <script src="https://cdn.bootcss.com/popper.js/1.12.9/umd/popper.min.js"></script>
<script src="https://cdn.bootcss.com/jquery/3.3.1/jquery.min.js"></script>

   <script src="https://cdn.bootcss.com/bootstrap/4.0.0/js/bootstrap.min.js"></script>
   <link href="{{ url_for('static', filename='css/main.css') }}" rel="stylesheet">     <style> body


{     background-image:
url("https://i.pinimg.com/originals/be/21/1a/be211ad5043a8d05757a3538bdd8f450.jpg");
background-size: cover;
```

```
}
.bar { margin: 0px; padding:20px;
background-color:white; opacity:0.6;
color:black;


font-family:'Roboto',sans-serif; font-style: italic; border-radius:20px;
font-size:25px;

} a { color:grey; float:right; textdecoration:none; font-
style:normal; padding-right:20px; } a:hover{ background-
color:black; color:white; borderradius:15px;0 font-
size:30px; paddingleft:10px;    }         .div1{
        background-color: lightgrey;   width: 500px;
border: 10px solid peach;   padding: 20px;   margin:

20px;    height:
500px;
}


.header {    position: relative;
                        top:0;
 margin:0px;                          z-index: 1;
left: 0px;              right: 0px;             position:
```

```
fixed;                  background-color: #8B008B

;       color:

white;



                box-shadow: 0px 8px 4px grey;
overflow: hidden;       padding-left:20px;          font-
family: 'Josefin Sans';             font-size:

2vw;                    width: 100%;

 height:8%;
                        text-align: center;
                }
                .topnav {
  overflow: hidden;   backgroundcolor: #FCAD98;


}

.topnav-right a {   float: left;

color: black;   text-align: center;

padding: 14px 16px;   text-

decoration: none;   fontsize: 18px;



}


.topnav-right a:hover {   background-color:
```

```
#FF69B4;   color: black;

}

.topnav-right a.active {   backgroundcolor:
#DA70D6;   color: black;

}

.topnav-right {   float: right;   paddingright:100px;
```

```
</style>
</head>

<body>

<div class="header">
<div style="width:50%;float:left;font-size:2vw;text-align:left;color:black;
paddingtop:1%;padding-left:5%;">Nutrtion Image Analysis</div>    <div class="topnav-
right"style="padding-top:0.5%;">



   <a href="{{ url_for('home')}}">Home</a>

   <a  class="active" href="{{ url_for('image1')}}">Classify</a>

  </div>

</div>

<br>



</div>

<div class="container">

      <center>

<div id="content" style="margin-top:2em">{% block content %}{% endblock %}</div></center>
   </div>

</body>



<footer>

   <script src="{{ url_for('static', filename='js/main.js') }}" type="text/javascript"></script>

</footer>
```

```
 }
</html>
```

MAIN.CSS

```
img-preview {     width: 256px;
height: 256px;     position: relative;     border: 5px solid

#F8F8F8;

   box-shadow: 0px 2px 4px 0px rgba(0, 0, 0, 0.1);     margin-top: 1em;
margin-bottom: 1em;


}

.img-preview>div {     width:
 100%;      height: 100%;


   background-size: 256px 256px;  background-repeat: no-repeat;        background-position:
center;

}

input[type="file"] {     display: none;

.upload-label{     display: inline-block;     padding: 12px 30px;     background:
```

```
#39D2B4;

   color: #fff;     fontsize: 1em;     transition: all

.4s;     cursor: pointer;

}

.upload-label:hover{     background:
#34495E;     color:

#39D2B4;

}

.loader {  border: 8px solid #f3f3f3; /* Light grey */       border-top: 8px solid #3498db; /* Blue */
borderradius: 50%;     width: 50px;     height: 50px;     animation: spin 1s linear infinite;



}
@keyframes spin {
   0% { transform: rotate(0deg); }
   100% { transform: rotate(360deg); }
}
```

}

STYLE.CSS

```
body{
        background-image:url(bg.jpg);     background-size: 400% auto;   backgroundrepeat: no-repeat;   backgroundposition:center;     color:#555;

        font-family:Arial, Helvetica, sans-serif;   font-size:16px;    line-height:1.6em;   margin:0;

}

.container{             width:80%;    margin:auto;   overflow:hidden;

}

.justify{     textalign:justify;     textjustify: auto;

.parallax {   /* The image used */      background-image: url("doc.jpg");

  /* Set a specific height */   minheight:
750px;
```

```
  /* Create the parallax scrolling effect */   backgroundattachment: fixed;
backgroundposition: center;   background-repeat: no-repeat;   backgroundsize: cover;



}

html {   scroll-behavior: smooth;

} #section2 {   height: 500px;   background: ; } div.background {
background: url("static/bgg2.jpg");

min-height:  5px;  backgroundattachment:  fixed;   backgroundposition:        center;
backgroundrepeat:

        no-repeat;   background-size: cover;
}

#navbar{   backgroundcolor:#fff;   color:#333;


}

#navbar ul{
        padding:0;

 list-style: none;

}
```

```
 }

#navbar li{     display:inline;


}

#navbar a{   color:#fff;   textdecoration:
none;   font-size:18px;    paddingright:15px;

}

#showcase{   minheight:300px;   marginbottom:30px;
```

```
}

#showcase h1{      width: 100%;    color:#333;   font- size:40px;
text-align: center;   line-height:
        1em;
paddingtop:10px;

}
#showcase h2{     width: 100%;

        color:#333;   font-  size:30px;   text-align: center;   lineheight:
1.6em;   padding-top:10px;

}

#main{        float:left;   color:#fff;   width:65%;   padding:0 30px;
box-sizing: border-box;

}

#sidebar{     float:right;   width:35%;
        background-color: #ffcccc;   color:#000;
padding-left:10px;   padding-right:10px;
paddingtop:1px;    box-sizing: borderbox;
```

```
}

.img-preview {     width: 10px;     height: 10px;     position: relative;
border: 5px solid #F8F8F8;     box-shadow: 0px 2px 4px 0px rgba(0,
0, 0, 0.1);     margin-top: 1em;     marginbottom: 1em;

}

.img-preview>div  {     width: 10%;     height: 10%;
background-size:

100px 10px;      background-
repeat:  norepeat;
background- position:
center;   input[type="file"] {     display: none;
}

.upload-label{     display: inline-block;     padding:

12px 30px;     background:  #39D2B4;

   color: #fff;     fontsize: 1em;     transition: all
```

```
}

.4s;     cursor: pointer;

}

.upload-label:hover{     background:
#34495E;     color:

#39D2B4;

}

.myButton {   border: none;   text-align: center;   cursor: pointer;
text-transform: uppercase;   outline: none;   overflow: hidden;
position: relative;   color: #fff;   font-weight:

700;   font-size: 12px;   background-color: #ff0000;   padding:
10px 15px;   margin: 0 auto;   box-shadow: 0 5px


15px rgba(0,0,0,0.20);

}

.myButton span {     position: relative;
zindex:

1;
```

```
}

.myButton:after {    content: "";
position: absolute;   left: 0;   top: 0;
height: 310%;   width: 150%;
background: #f2f2f2;

  -webkit-transition: all .5s ease-in-out;   transition: all
.5s ease-in-out;
  -webkit-transform: translateX(-98%) translateY(-25%)
rotate(45deg);   transform: translateX(-98%) translateY(-25%)
rotate(45deg);

.myButton:hover:after {

  -webkit-transform: translateX(-9%) translateY(-25%) rotate(45deg);
transform: translateX(-9%) translateY(-25%) rotate(45deg);

}

.loader {        border: 8px solid #f3f3f3; /* Light
grey */      border-top: 8px solid #ff0000; /* Red
*/     border-radius: 50%;     width: 50px;
height: 50px;       animation: spin 1s linear
infinite;
```

```
}

}

@keyframes spin {
   0% { transform: rotate(0deg); }
   100% { transform: rotate(360deg); }
}

#main-footer{
background: #333;
color:#fff;   text-align:
center;        padding:1px;
margintop:0px;

}

@media(max-width:600px){
        #main{          width:100%;

 float:none;
        }

        #sidebar{     width:100%;
```

```
 float:none;
        }
}
```

MAIN.JS

```
$(document).ready(function () {
    // Init
    $('.image-section').hide();         $('.loader').hide();
    $('#result').hide();
```

```
   // Upload Preview     function readURL(input)
{         if (input.files

&& input.files[0]) {              var reader
= new FileReader();               reader.onload
= function (e) {

                $('#imagePreview').css('background-image', 'url(' + e.target.result + ')');
                $('#imagePreview').hide();
                $('#imagePreview').fadeIn(650);
            }
            reader.readAsDataURL(input.files[0]);
        }
}
    $("#imageUpload").change(function () {
        $('.image-section').show();
        $('#btn-predict').show();
        $('#result').text('');
$('#result').hide();         readURL(this);

    });
```

```
    // Predict

    $('#btn-predict').click(function () {           var form_data = new
FormData($('#upload-file')[0]);


        // Show loading animation

        $(this).hide();

        $('.loader').show();


        // Make prediction by calling api /predict          $.ajax({            type:

'POST',           url: '/predict',           data:
form_data,           contentType: false,
cache: false,            processData: false,
async: true,           success: function (data)


{

                // Get and display the result

                $('.loader').hide();
                $('#result').fadeIn(600);
$('#result').html(data);               console.log('Success!');

            },          });
```

```
    });

});
```

GITHUB LINK

IBM-EPBL/IBM-Project-36345-1660294343 **PROJECT DEMO LINK**

https://drive.google.com/file/d/1cFBZ_kBqct0ZvKahGA1l9NBsv3K4Qt6/view?usp=drivesdk