AI - POWERED NUTRITION FOR FITNESS ENTHUSIASTS

TEAM ID: PNT2002MID34208

TEAM LEADER: SUSHMA R M

TEAM MEMBERS: SUSHMA R M (960219106139)

SHALINI S (960219106119)

SUNITHA K (960219106137)

SUVETHA M (960219106140)

DEPARTMENT: ELECTRONICS AND COMMUNICATION ENGINEERING

COLLEGE NAME:ARUNACHALA COLLEGE OF ENGINEERING FOR WOMEN

FACULTY MENTOR: STARLIN BEULA G

INDUSTRIAL MENTOR: SRI TULASI

TABLE OF CONTENTS

ABSTRACT:

AI Powered Nutrition Analyzer For Fitness Enthusiasts suggests using an intelligent agent to create diet regimens depending on the user's inputs. Using information about a person's lifestyle and health requirements, the system generates a food plan. A system with artificial intelligence that understands human nutrition is known as an online artificial dietician.

People are taking different approaches to their health and food because of the modern lifestyle, careless attitude, and materialism. People now frequently experience various health issues and fitness issues, which is mostly attributable to an unbalanced diet. Health, wealth, and time are traded off in the current situation. They frequently lack knowledge of the ideal nutrient balance for a healthy body.Therefore, a need for software that can offer diet consultations to the people at their preferred time and via mobile phones without having to visit a dietician arises in order to facilitate them with a proper diet chart along with light exercises according to their lifestyle and cope up with their busy schedule.

The main goal of the software is to provide the user with a list of all feasible diet plans, along with the nutritional value of the food items, based on his or her lifestyle, taking into account the user's height, weight, working hours, and eating habits as inputs.

1.INTRODUCTION :

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.

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:

EXISTING PROBLEM:

1.AI powered nutrition Analyzer For Fitness Enthusiasts – Jeukendrup A.E.,Killer S.C. The myths surrounding preexercise 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 crosssectional 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 .

REFERENCES:

i. Davenport.T and Kalakota R.2019.The potential for Artificial Intelligence in healthcare-Future healthcare journal,2019- ncbi.nlm.nih.gov

The complexity and rise of data in healthcare means that Artificial Intelligence(AI) will increasingly be applied within the field. Several types of AI are already being employed by payers and providers of care, and life sciences companies. The key categories of applications involve diagnosis and treatment recommendations, patient engagement and adherence, and administrative activities.

ii.Amann J., Blasimme A., Vayena E., Frey D., Madai V.I., and Precise Q.C.2020. Explainability for Artificial Intelligence in Fitness: A multidisciplinary perspective.

Explainability is one of the most heavily debated topics when it comes to the application of Artificial intelligence(AI) in Fitness. Even though AI- driven systems have been shown to outperform humans in certain analytical tasks, the lack of explainability continues to spark criticism. It is not a pure technological issue, instead it invokes a host of medical, legal, ethical, and societal questions that require thorough exploration.

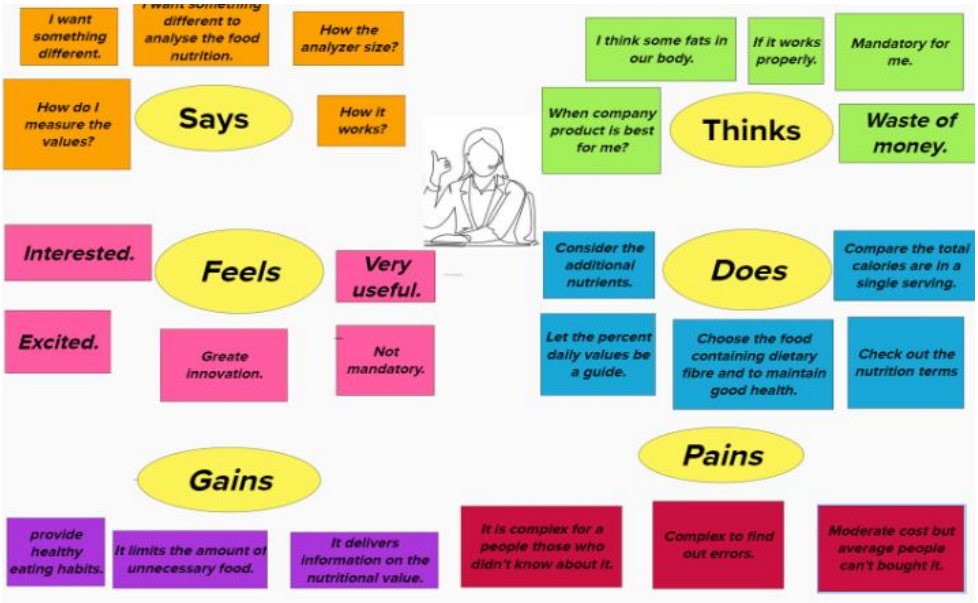
PROBLEM STATEMENT DEFINITION:

Due to change in food habits people do not get aware of food items. Our project is to get details about food nutritions, carbohydrate, protein and fat.Nutritional awareness is also related to knowledge of the interrelationships between nutritional matters and human life, which may have an effect on a person's life.

The World Health Organisation(WHO) data reveals that more than 60% of world's population is not physically active enough to induce health benefits.

3. IDEATION & PROPOSED SOLUTION:

EMPATHY MAP:

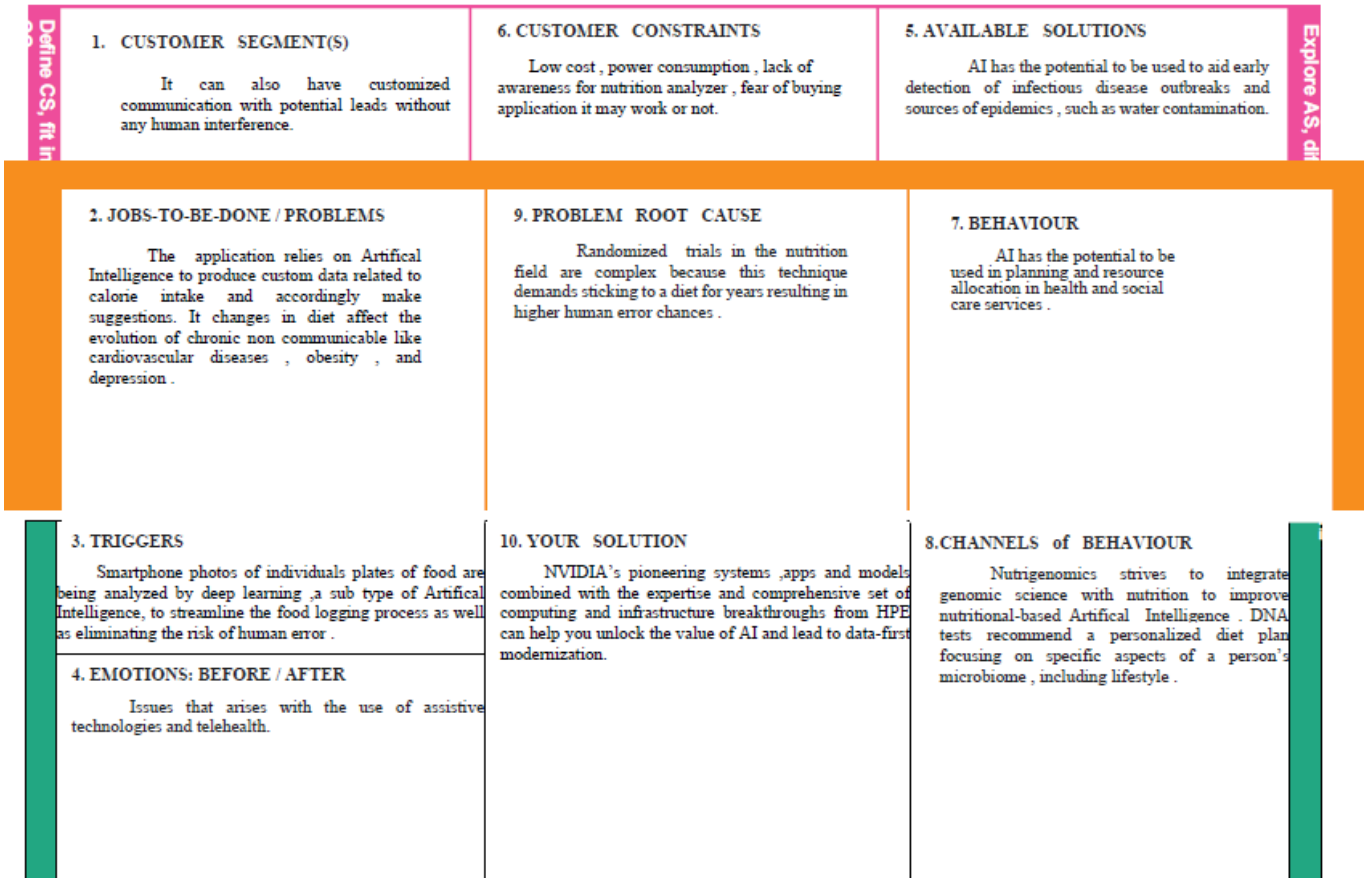


IDEATION AND BRAINSTROMING:

3 PROPOSED SOLUTION:

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, Fibres, Protein, Calories, etc.).

PROBLEM SOLUTION FIT :



4. REQUIREMENT ANALYSIS:

FUNCTIONAL REQUIREMENTS:

FR No.	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)

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	Certificate Requirements	Regulation Rules Profession wide.
FR-4	Authorization	Healthcare provider User group.
FR-5	Business Rules	Decision making Marketing.
FR-6	External Interfaces	Wide Area Network (WAN) Screen layouts.

NON FUNCTIONAL REQUIREMENTS:

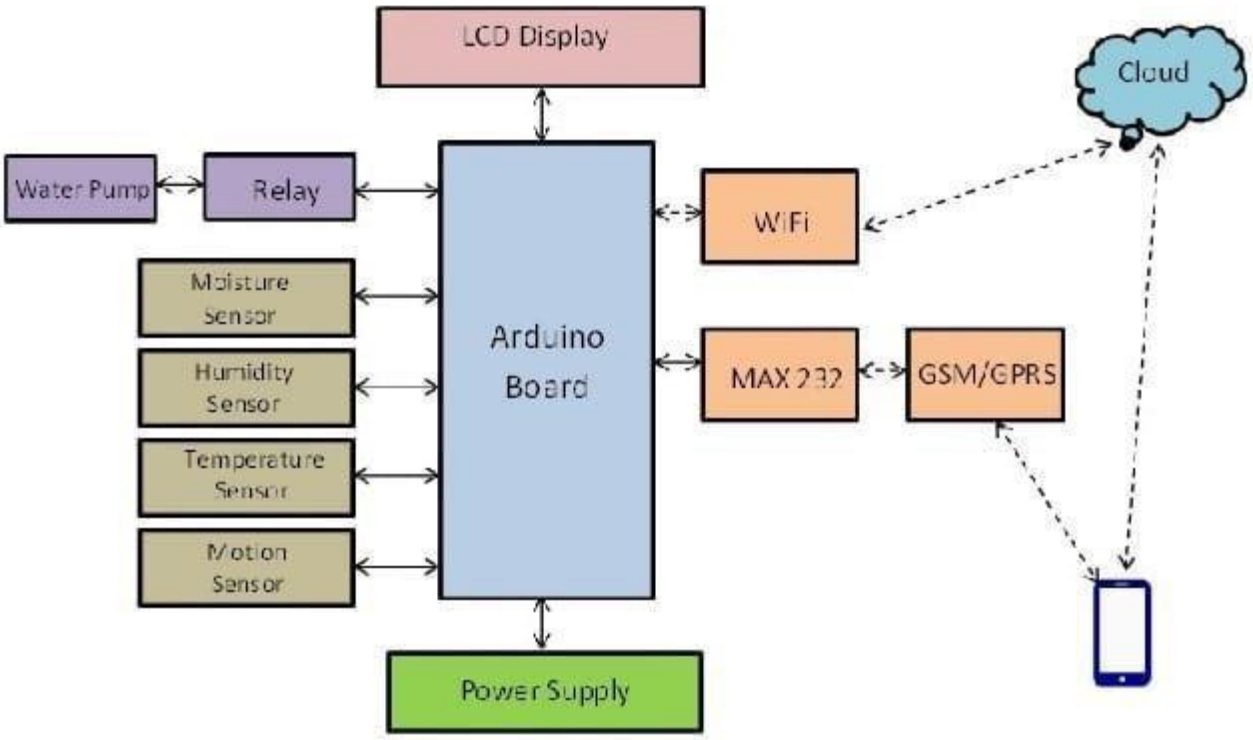
FR No.	**Non-Functional Requirement**	**Description**
NFR-1	**Usability**	Used to determine the nutritional content of food.

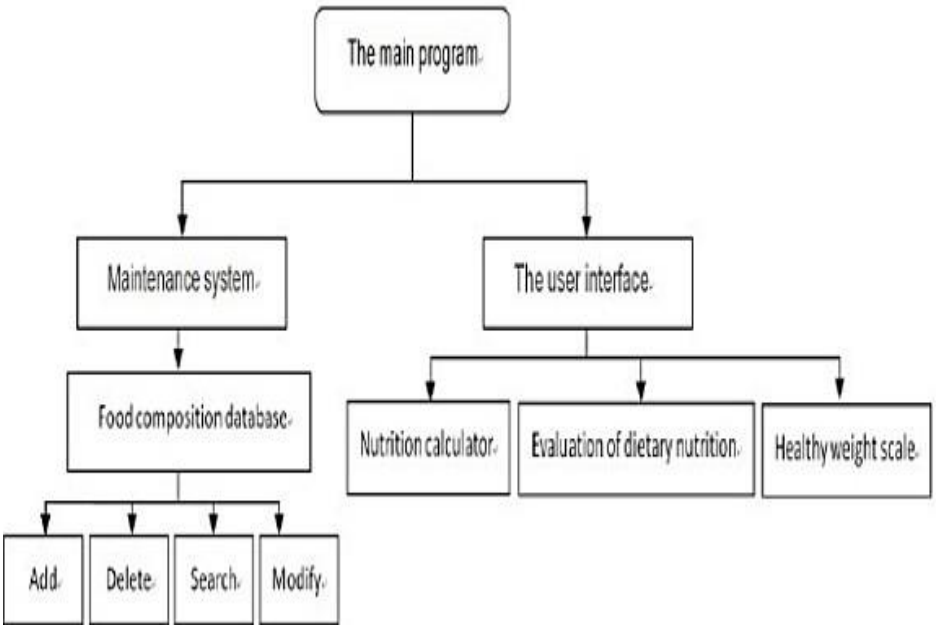
NF R-2	**Security**	Analysis of quality control andcontaminati on of food.
NF R-3	**Reliability**	Change eating habits to prevent non communicable diseases.
NF R-4	**Performan ce**	Prevention from severe health related problems.
NF R-5	**Availability**	Streamline the food logging process as well as eliminating the risk of human error.
NF R-6	**Scalability**	Support and monitor the personalized supply of nutrients.

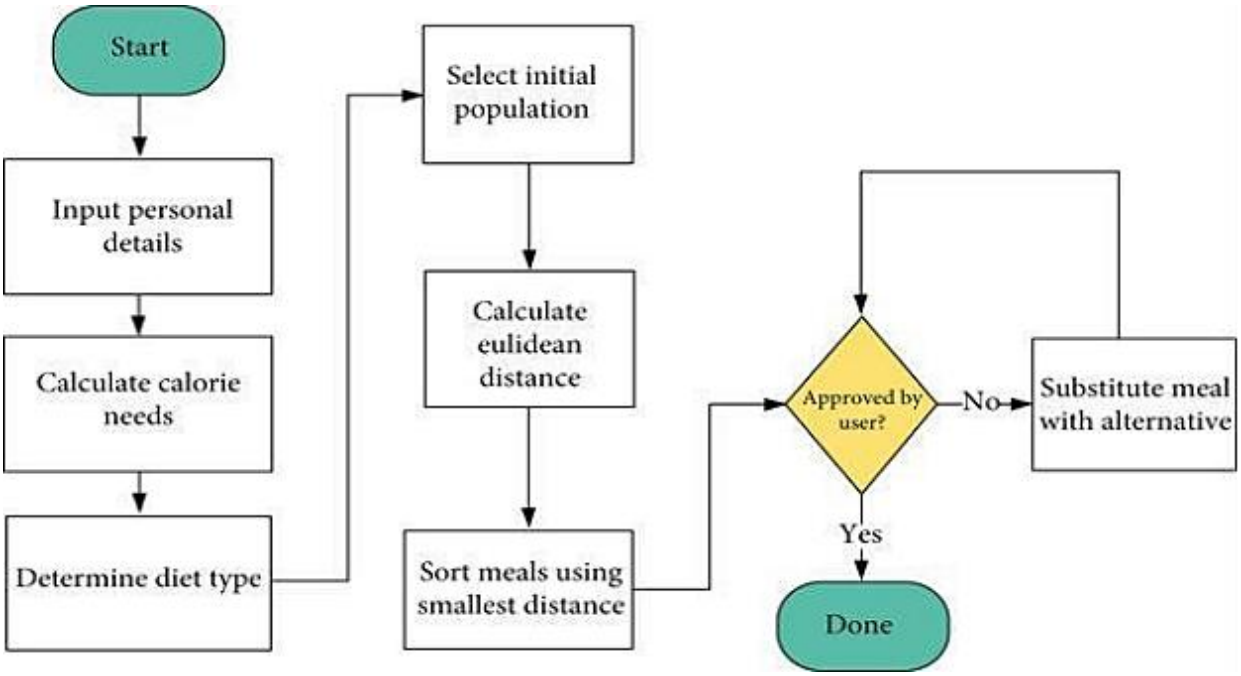
5.PROJECT DESIGN:

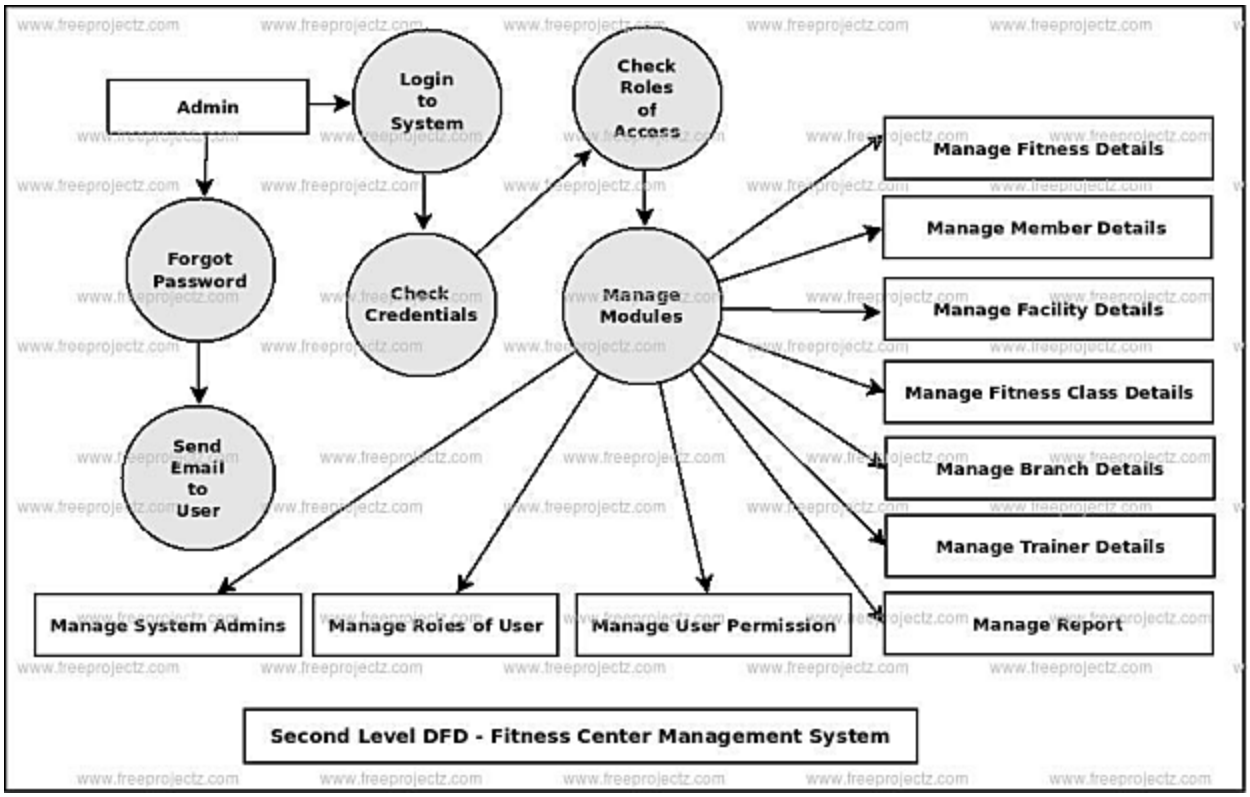
DATA FLOW DIAGRAMS:

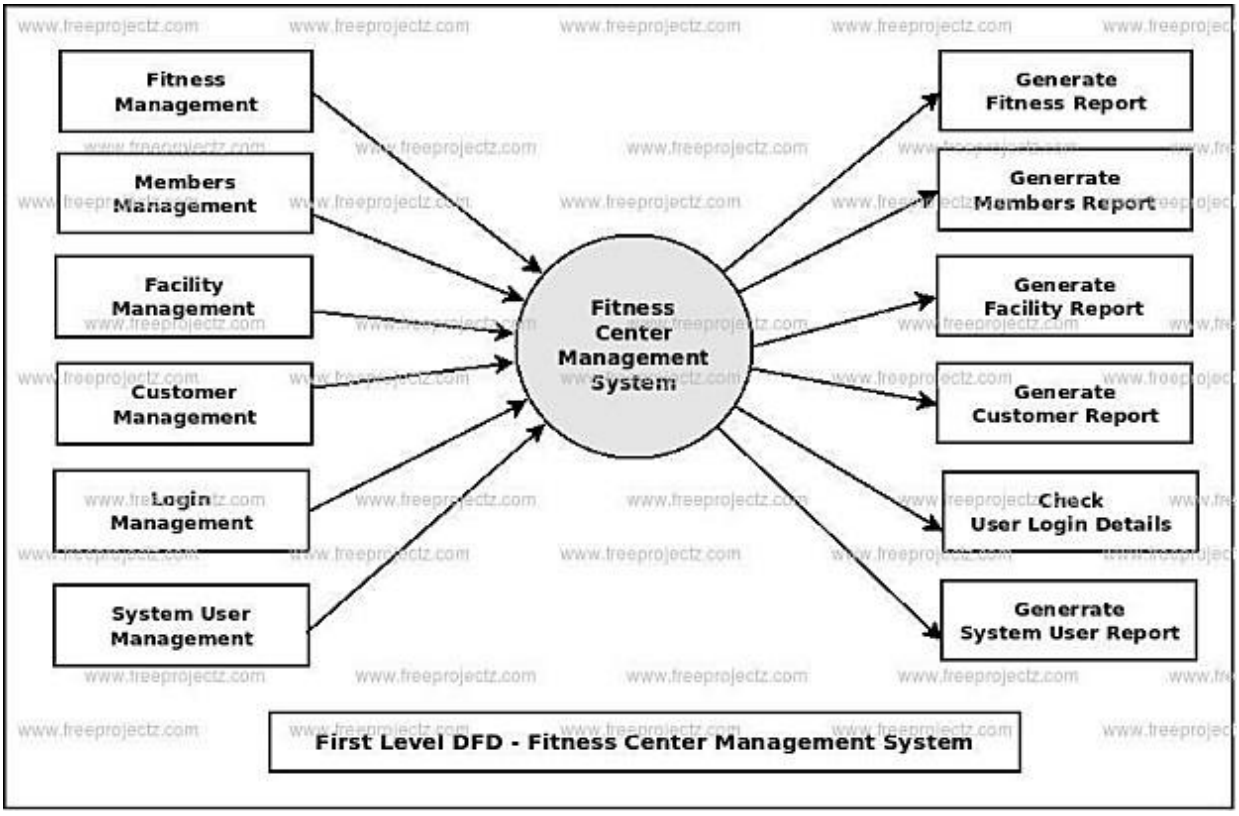
A Data Flow Diagram (DFD) is a traditional visual representation of the information flows within a system. A neat and clear DFD can depict the right amount of the system requirement graphically. It shows how data enters and leaves the system, what changes the information, and where data is stored.


LCD Display
Cloud
Water Pump
Relay
Arduino Board
WiFi
Moisture Sensor
Humidity Sensor
Temperature Sensor
Motion Sensor
MAX 232
GSM/GPRS
Power Supply

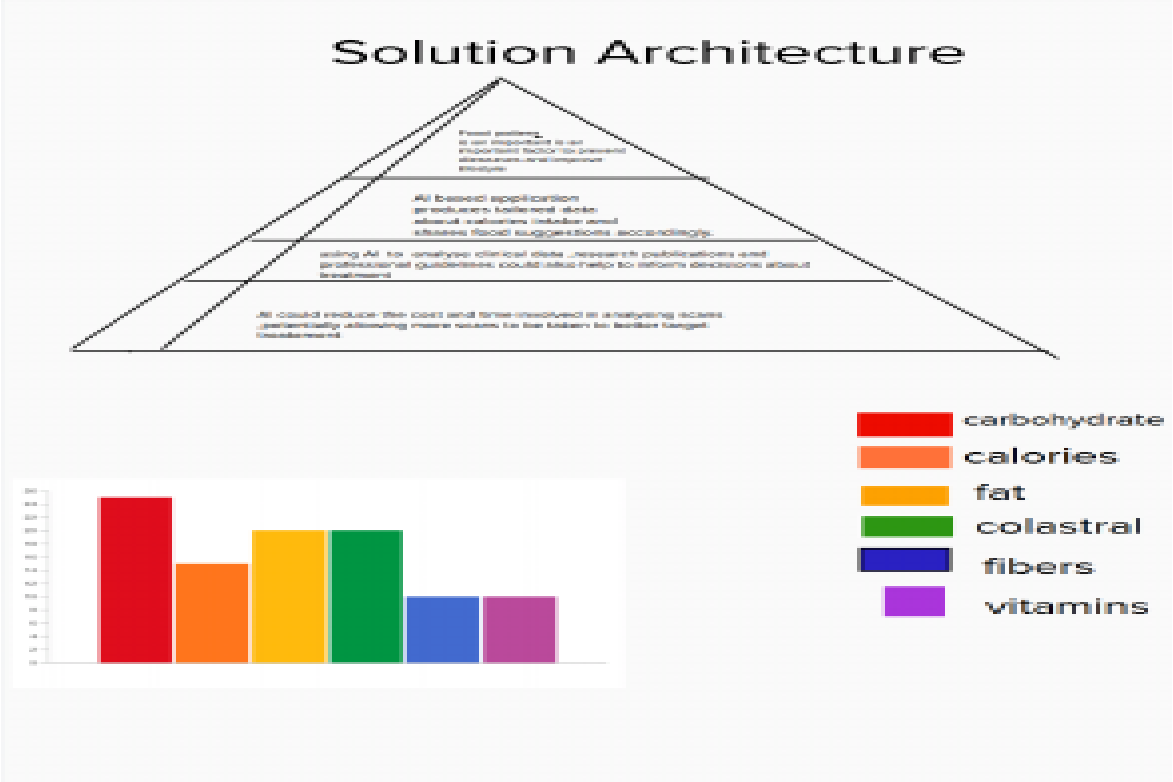

The main program
Maintenance system
The user interface
Food composition database
Nutrition calculator
Evaluation of dietary nutrition
Healthy weight scale
Add
Delete
Search
Modify

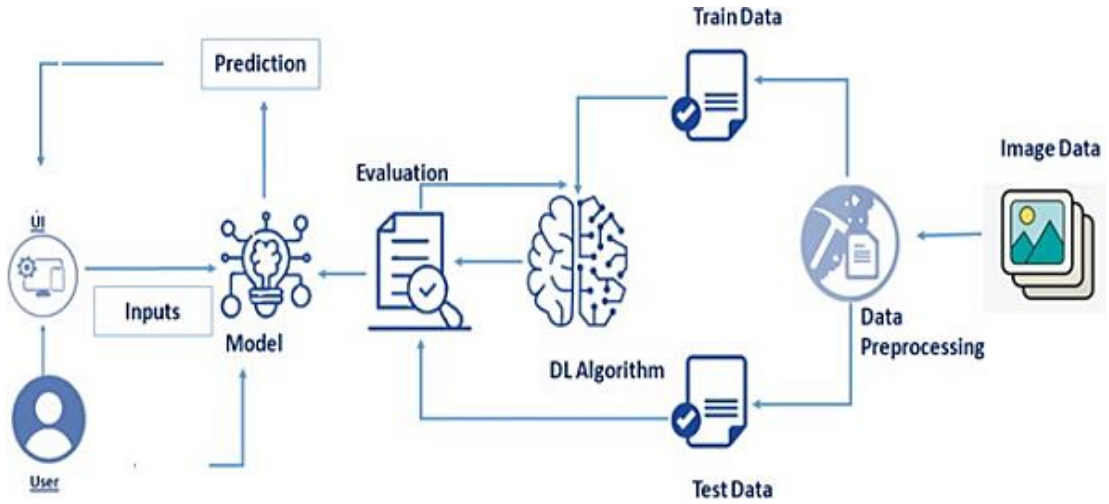

Start
Input personal details
Calculate calorie needs
Determine diet type
Select initial population
Calculate eulidean distance
Sort meals using smallest distance
Approved by user?
No
Substitute meal with alternative
Yes
Done





SOLUTION AND TECHNICAL ARCHITECTURE:





TECHNICAL ARCHITECTURE:

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

Technology Stack (Architecture & Stack)

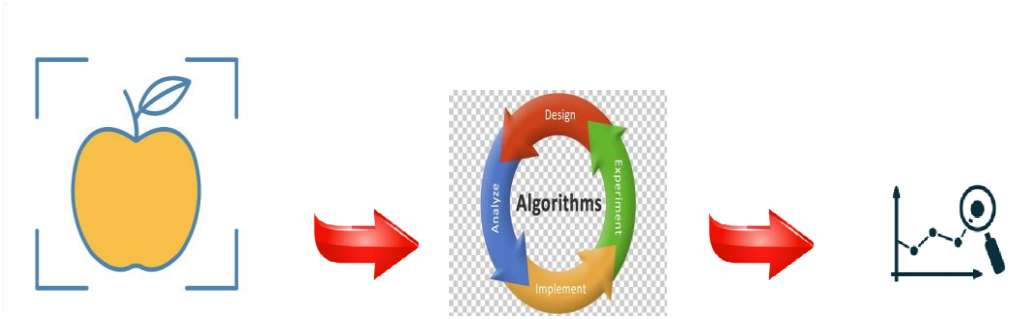
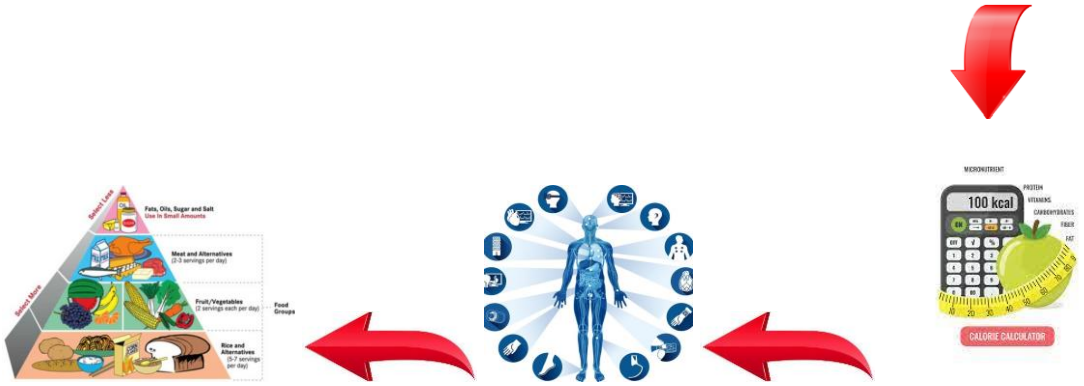


IMAGE RECOGNITION **DL ALGORITHMS**
PREDICTION ANALYSIS



DIETARY PLAN **CALCULATE USER** **CALCULATE**
ACTIVITY STATUSCALORIES AND NUTRITION

FINAL ANALYSIS

Components & Technologies:

S.N o	Component	Description	Technology
1.	User Interface	How user interacts with application e.g. Web UI, Mobile App, Chatbot etc.	HTML, CSS,

2.	Application Logic-1	Logic for a process in the application	Python
3.	Application Logic-2	Logic for a process in the application	IBM Watson STT service
4.	Application Logic-3	Logic for a process in the application	IBM Watson Assistant
5.	Database	Data Type, Configurations etc.	MySQL
	Cloud Database	Database Service on Cloud	IBM Cloudant
6.	File Storage	File storage requirements	IBM Block Storag or Local Filesystem
7.	External API-1	Purpose of External API used in the application	IBM Weather API, etc.
8.	External API-2	Purpose of External API used in the application	Aadhar API, etc.
9.	Machine Learning Model	Purpose of Machine Learning Model	Object Recognition Model, etc.
10	Infrastructure (Server / Cloud)	Application Deployment on Local System / Cloud	Local, Cloud Foundry, Kubernetes, etc.

		Local Server Configuration: Cloud Server Configuration :	

Application Characteristics:

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

		use of Cache, use of CDN's) etc.	

USER STORIES:

User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority	Release
Customer (Mobile user)	Registration	USN-1	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-2	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-3	As a user, I can register for the application through Facebook	I can register & access the dashboard with Facebook Login	Low	Sprint-2
		USN-4	As a user, I can register for the application through Gmail		Medium	Sprint-1
	Login	USN-5	As a user, I can log into the application by entering email & password		High	Sprint-1
	Dashboard					
Customer (Web user)						
Customer Care						
Executive Administrator						

6.PROJECT PLANNING AND SCHEDULING:

SPRINT PLANNING AND ESTIMATION:

Use the below template to create product backlog and sprint schedule

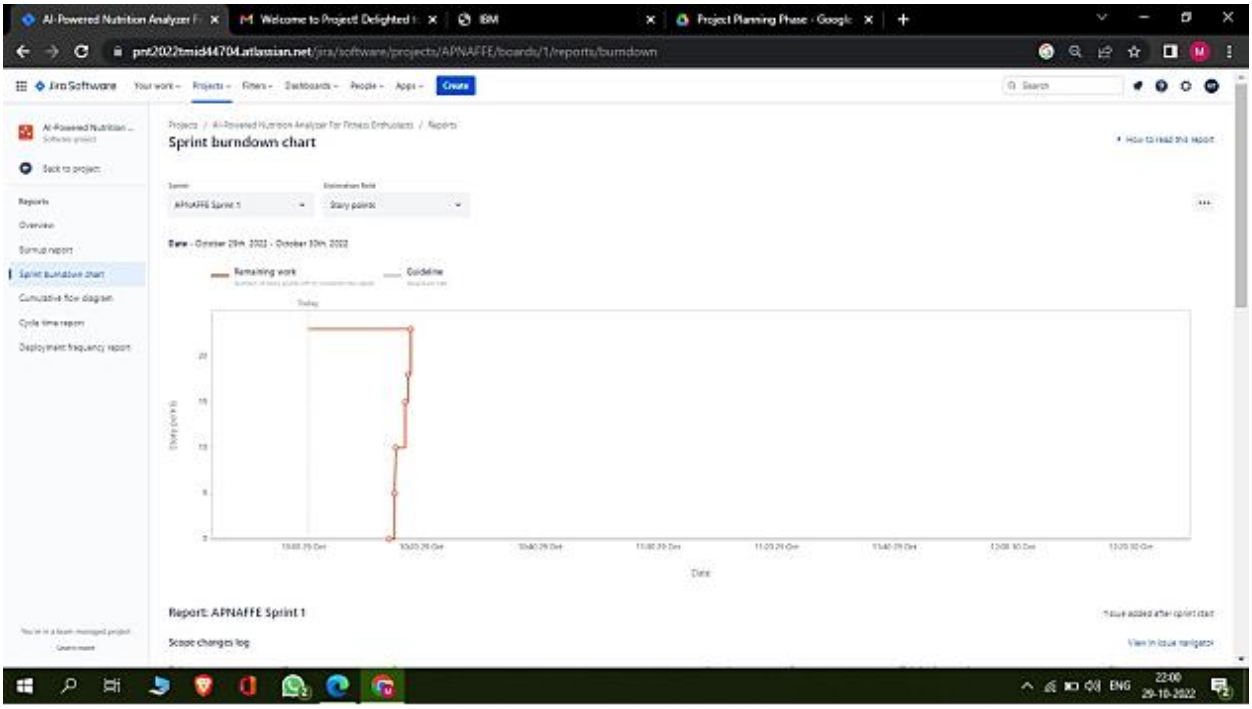
Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-1	Registration	USN-1	As a user, I can register for the application by entering my email, password, and confirming my password.	2	High	SHALINI.S, SUNITHA.K, SUSHMA.R.M, SUVETHA.M
Sprint-1		USN-2	As a user, I will receive confirmation email once I have registered for the application	1	High	SHALINI.S, SUNITHA.K, SUSHMA.R.M, SUVETHA.M
Sprint-2		USN-3	As a user, I can register for the application through Facebook	2	Low	SHALINI.S, SUNITHA.K, SUSHMA.R.M, SUVETHA.M
Sprint-1		USN-4	As a user, I can register for the application through Gmail	2	Medium	SHALINI.S, SUNITHA.K, SUSHMA.R.M, SUVETHA.M

SPRINT DELIVERY SCHEDULE:

S.No	ACTIVITY TITLE	ACTIVITY DESCRIPTION	DURATION
		Assign the team members and create repository in the Github	

1.	UNDERSTANDING THE PROJECT REQUIREMENTS	assign the task to each member and teach how to use and open and access the Github and IBM career education.	1 week
2.	STARTING OF PROJECT	Advice students to attend classes of IBM portal create and develop an rough diagram based on project description and gather of information on AI IBM project and team leader assign task to each member of the project.	1 Week
3.	ATTEND CLASSES	Team members and team lead must watch and learn from classes provided by IBM and NALAYA THIRAN and must gain access of MIT licence for their project.	4 Week
4.	BUDGET AND SCOPE	Budget and analyze the use of AI in the project and discuss with team for budget prediction to predict the favourability for the customer to buy.	1 Week

REPORT FROM JIRA:



7.CODING AND SOLUTIONING:
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 modelfrom 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")

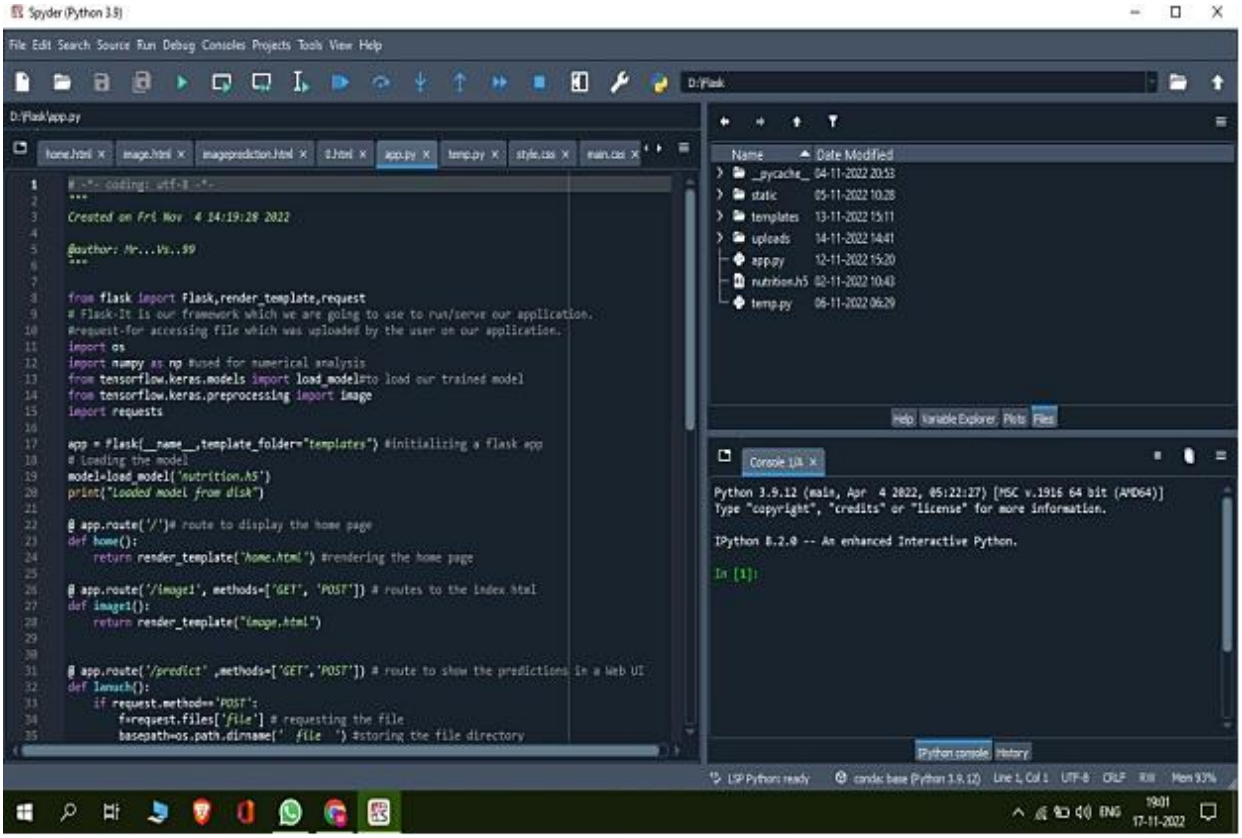
@ app.route('/')# route to display the home pagedef home():

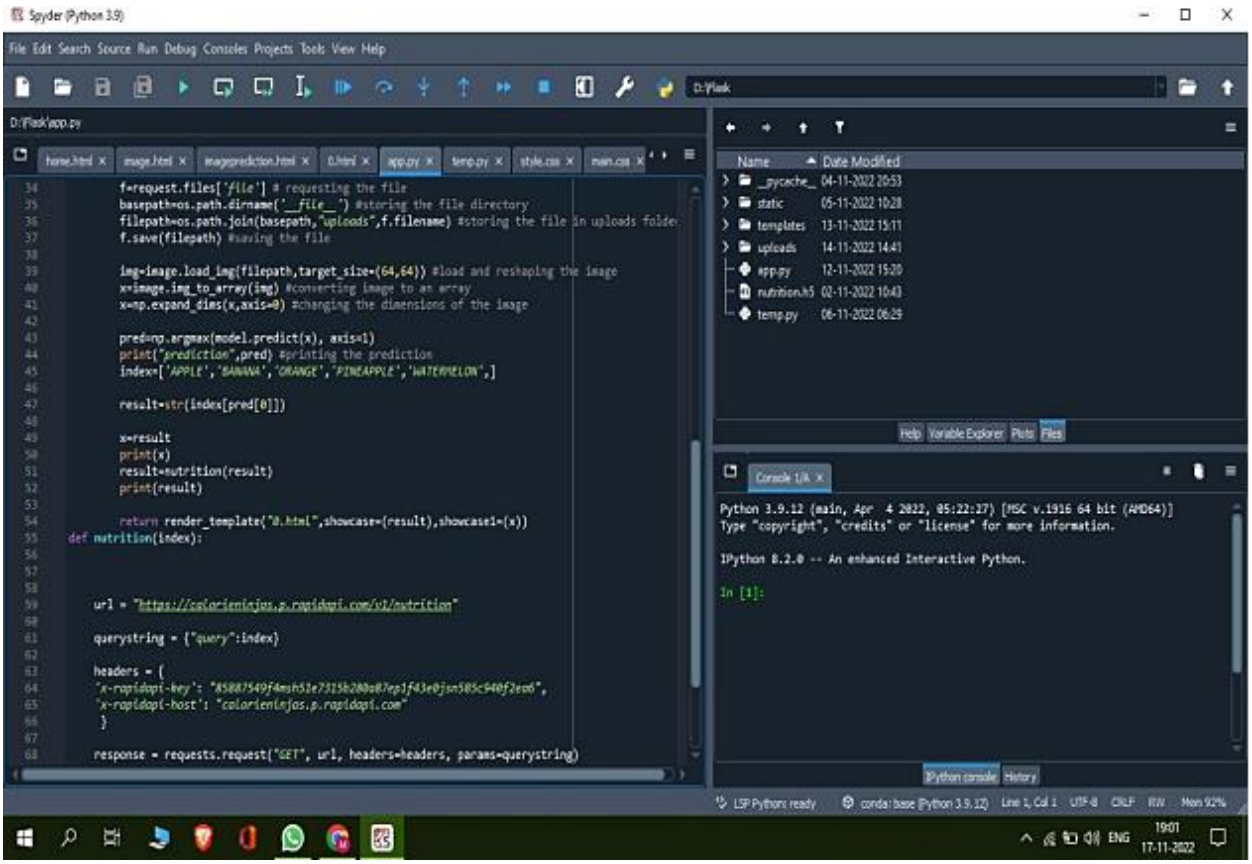
    return render_template('home.html') #rendering the home page

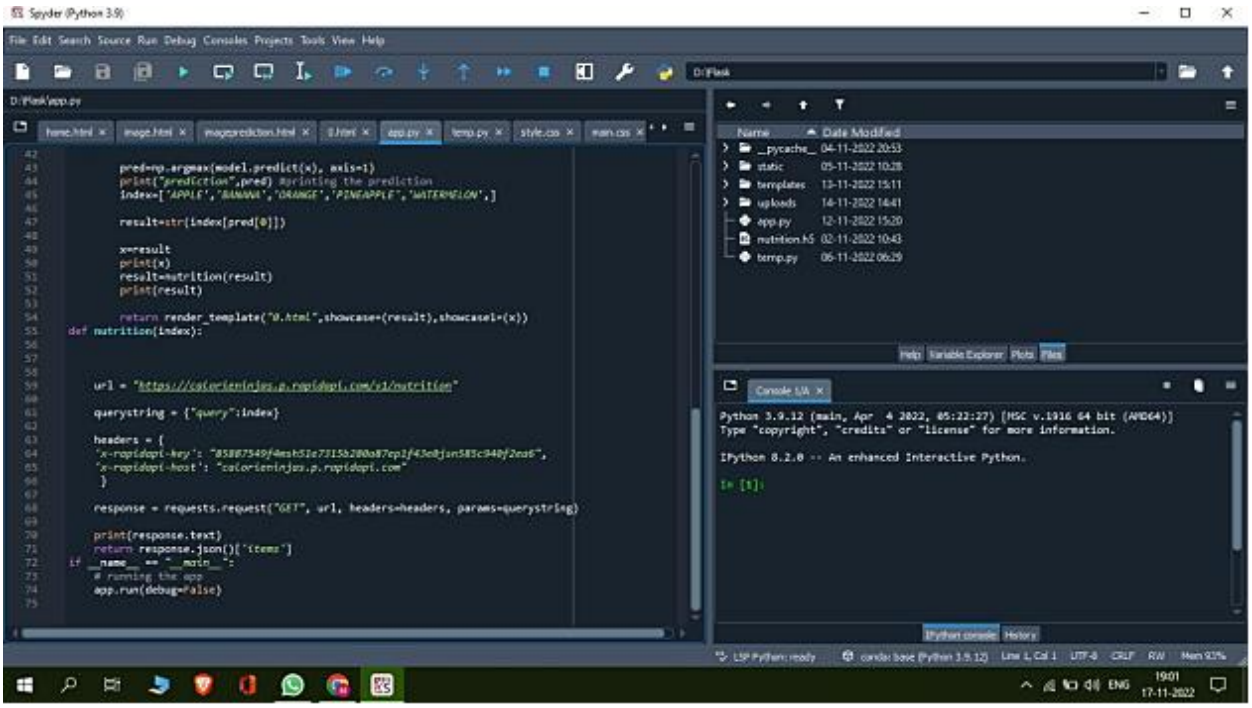
```
@ app.route('/image1', methods=['GET', 'POST']) # routes to the

index htmldef image1():

    return render_template("image.html")


@ app.route('/predict' ,methods=['GET','POST']) # route to show the
predictions ina 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 inuploads folder

        f.save(filepath) #saving the file



        img=image.load_img(filepath,target_size=(64,64)) #load and
reshaping theimage

        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
        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"
```

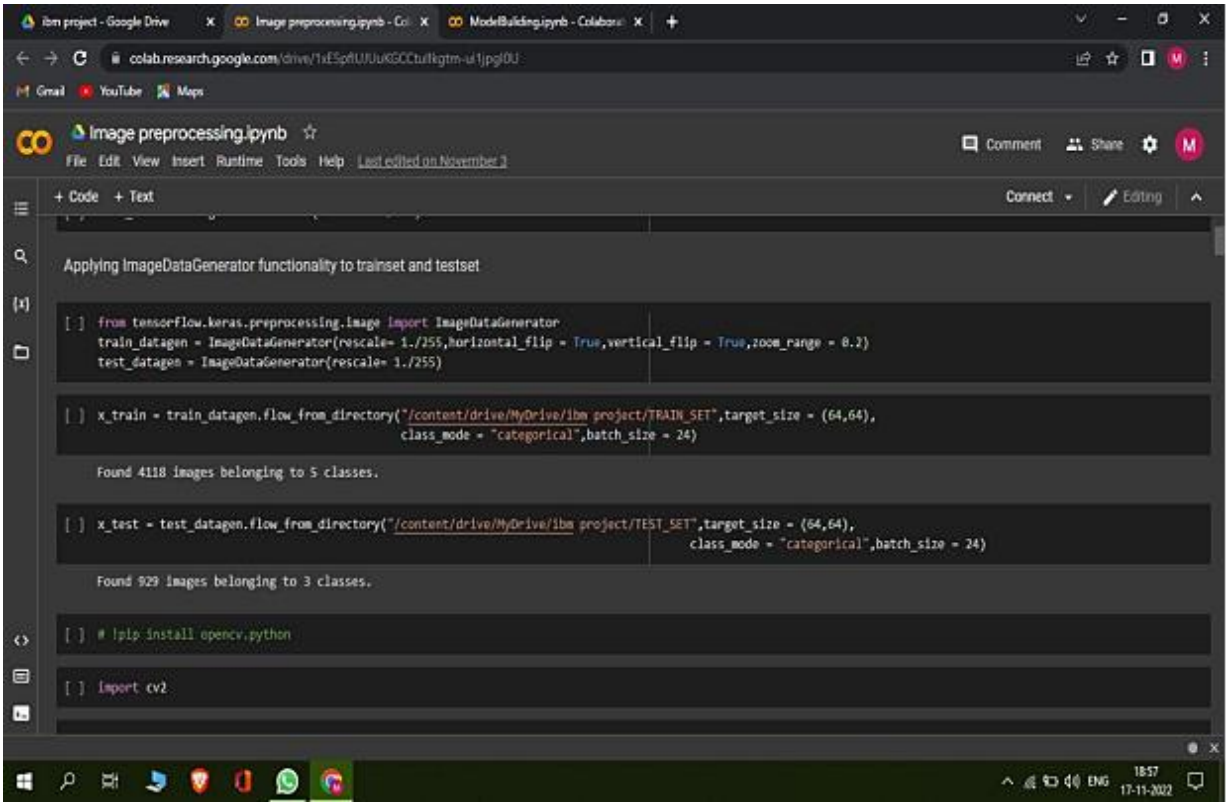
```
    }
    response = requests.request("GET", url, headers=headers,
    params=querystring)



    print(response.tex

    t)return
response.json()['items']
if _name_
==
"_main_":
    # running
    theapp
    app.run(deb
    ug=False)
```

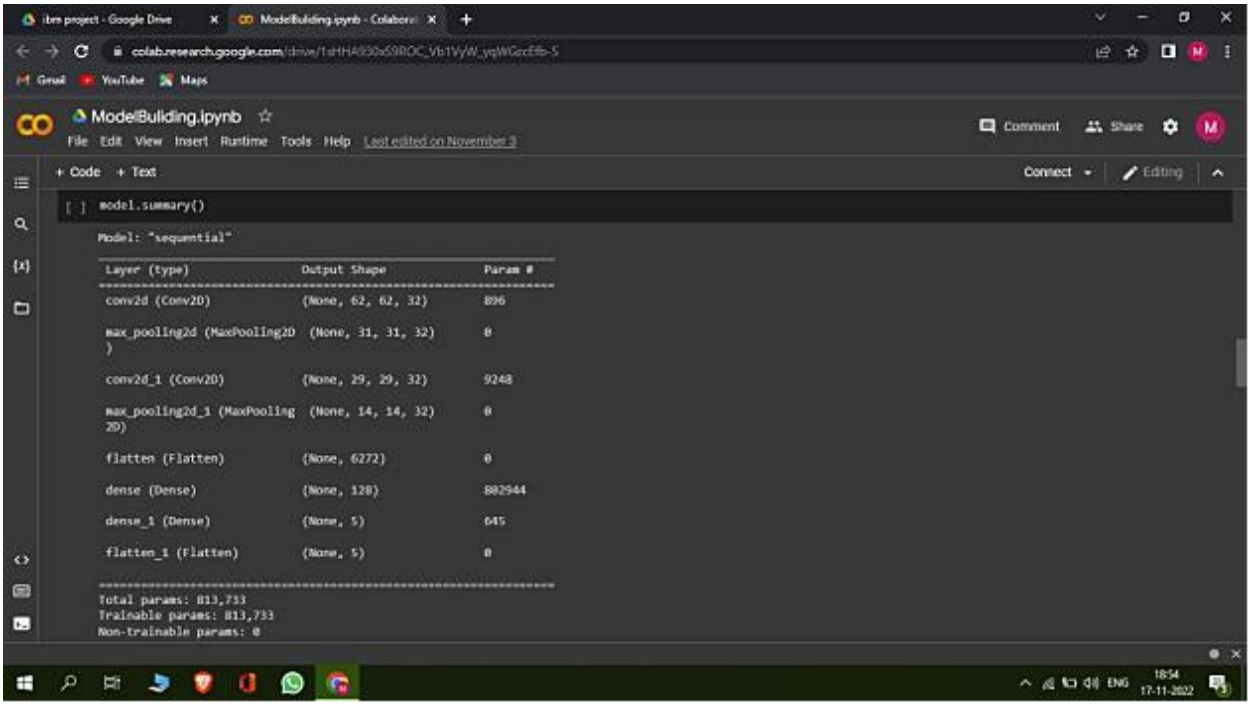

Spyder (Python 3.9)
File Edit Search Source Run Debug Consoles Projects Tools View Help
D:\Flask\app.py
home.html
image.html
imageprediction.html
0.html
app.py
temp.py
style.css
main.css
-*- coding: utf-8 -*-
"""
Created on Fri Nov 4 14:19:28 2022
@author: Mr...Vs..99
"""
from flask import Flask,render_template,request
# 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 launch():
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 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': "45887549f4msh51e7315b280a87ep1f43e0jsn585c940f2ea6",
'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


Spyder (Python 3.9)
File Edit Search Source Run Debug Consoles Projects Tools View Help
D:\Flask\app.py
home.html
image.html
imagepreddiction.html
0.html
app.py
temp.py
style.css
main.css
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': "85887549f4eskSIe7315b280a87ep1f45e8jsn585c940f2ea6",
'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)
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 97%
ENG
19:01
17-11-2022

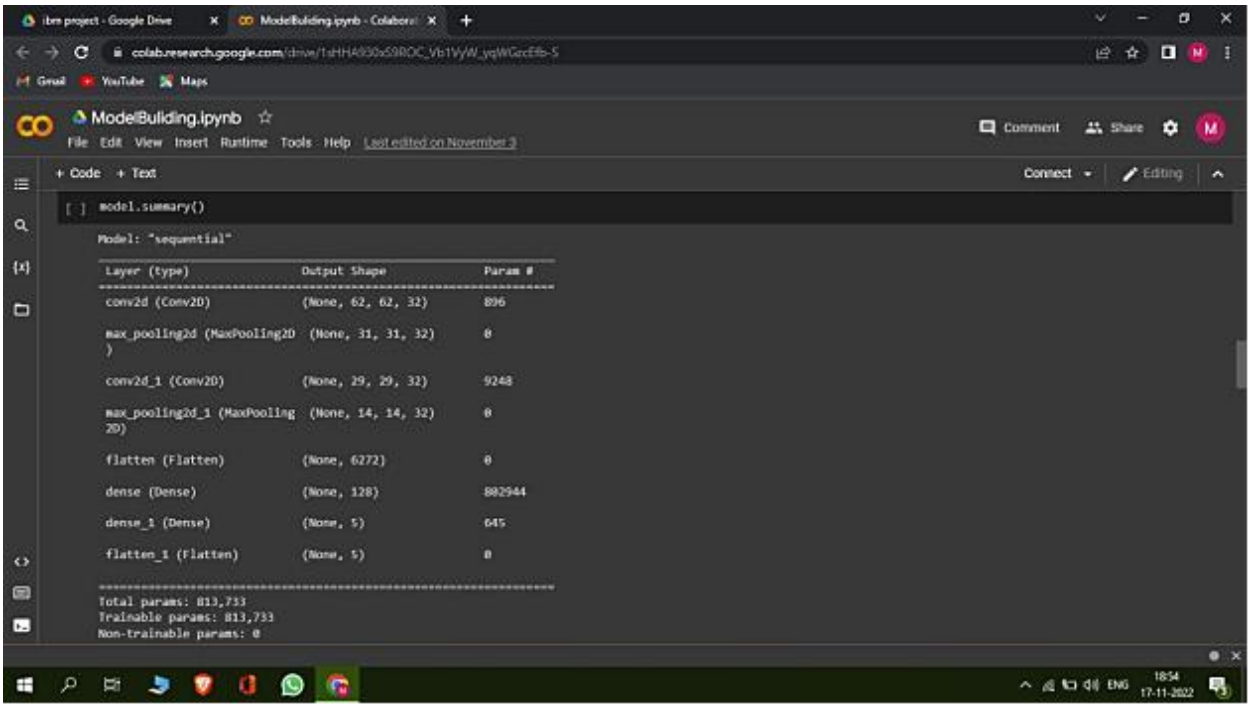
FEATURE 2 :


ibm project - Google Drive
Image preprocessing.ipynb - Co
ModelBuilding.ipynb - Colabora
colab.research.google.com/drive/1xESpfUUuKGCCtutkgtm-ui1jpgl0U
Gmail
YouTube
Maps
Image preprocessing.ipynb
File Edit View Insert Runtime Tools Help Last edited on November 3
Comment
Share
+ Code
+ Text
Connect
Editing
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
ENG
18:57
17-11-2022



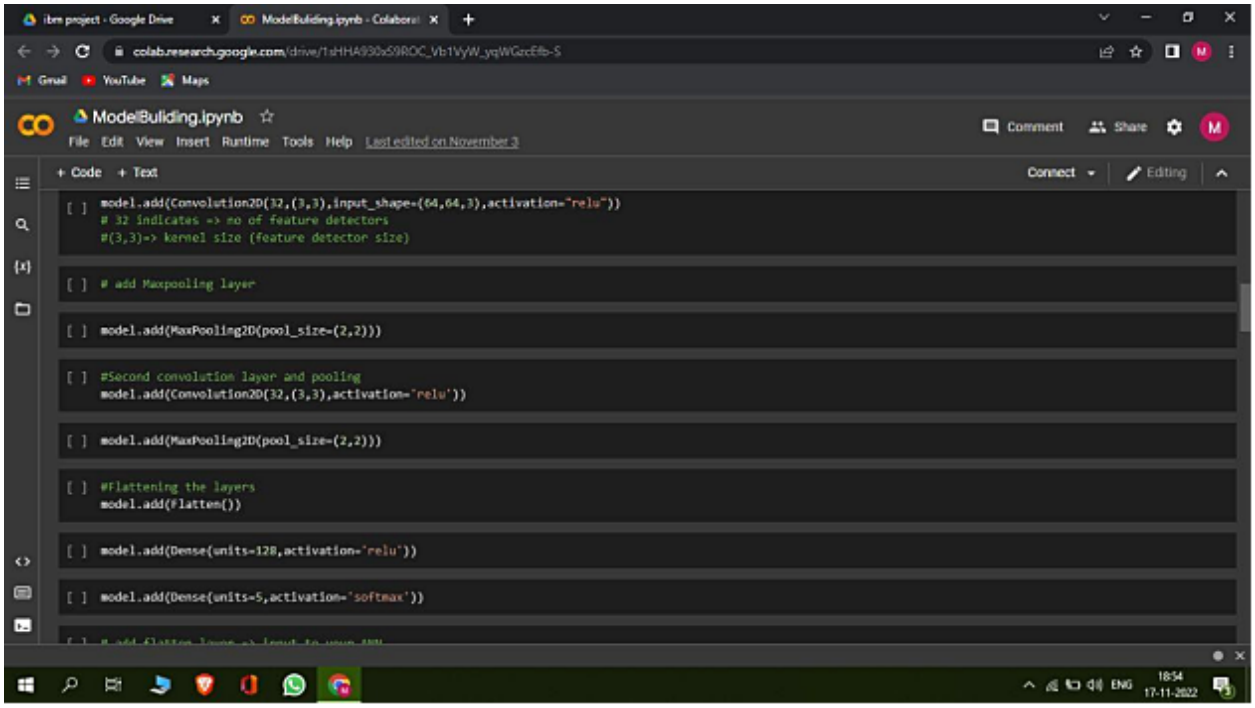
```
model.summary()
```

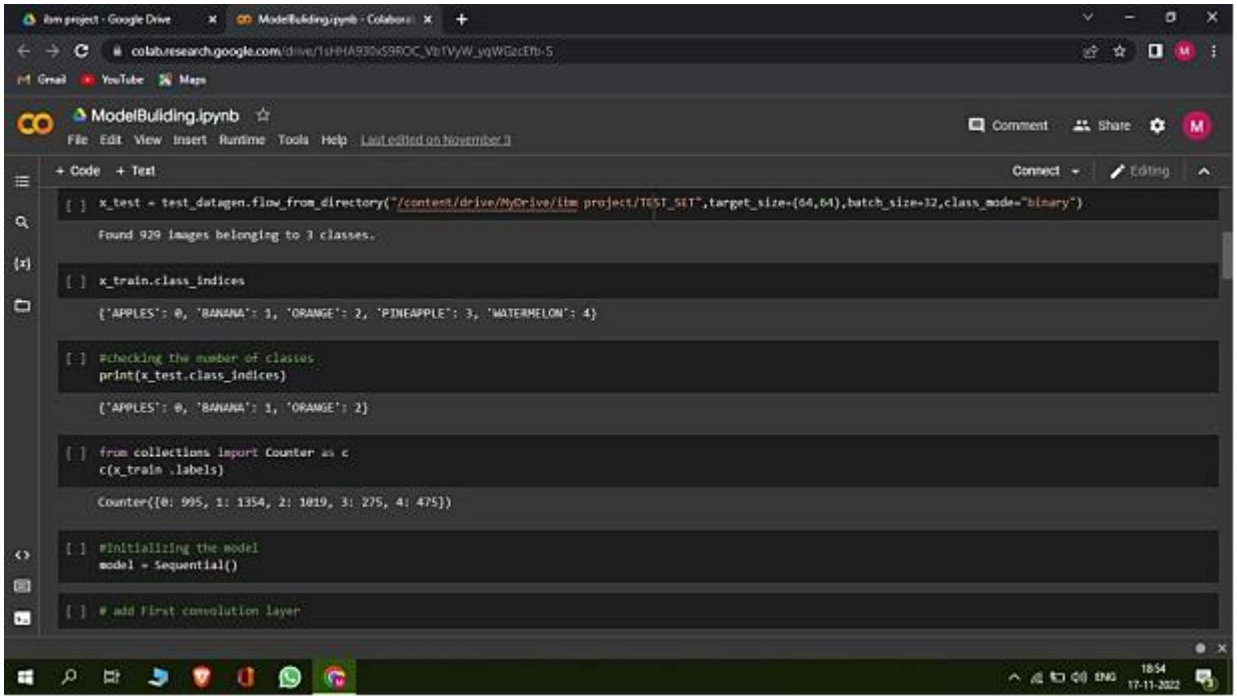
```
Model: "sequential"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
conv2d (Conv2D)              (None, 62, 62, 32)        896

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
```

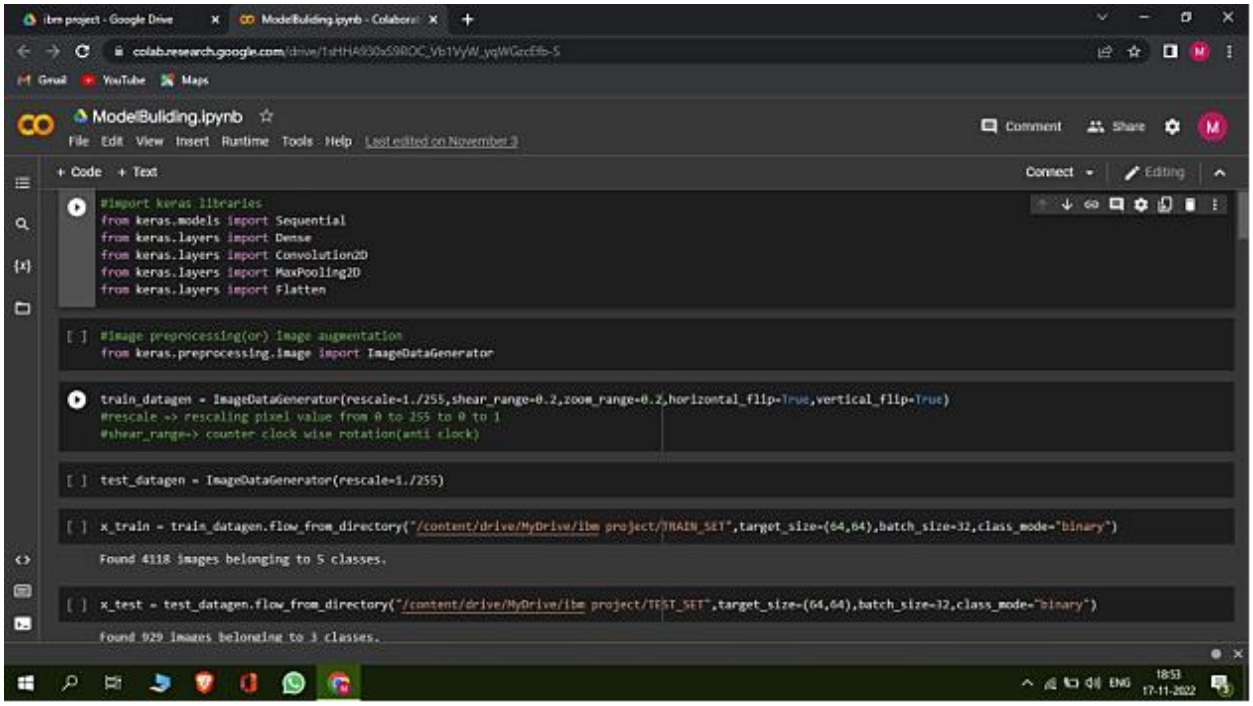



```
model.summary()
```

```
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
```

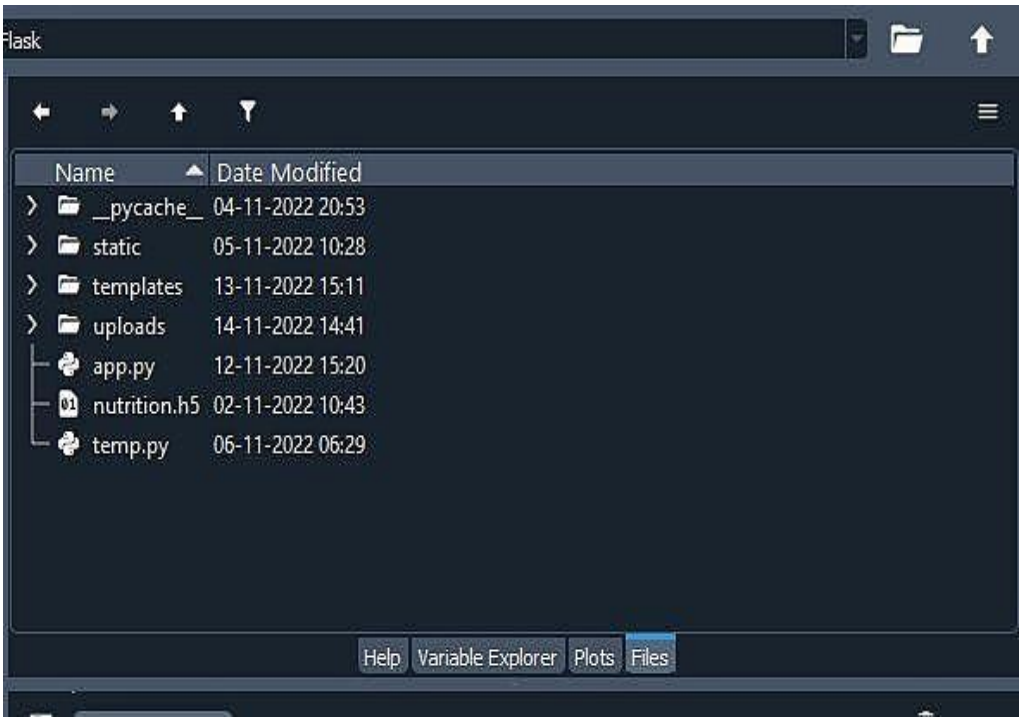

ibm project - Google Drive
ModelBuliding.ipynb - Colabora
colab.research.google.com/drive/1sHHA930xS9ROC_Vb1VyW_yqWGzcEfb-S
Gmail
YouTube
Maps
ModelBuliding.ipynb
File Edit View Insert Runtime Tools Help Last edited on November 3
Comment
Share
+ Code + Text
Connect
Editing
model.add(Convolution2D(32,(3,3),input_shape=(64,64,3),activation="relu"))
# 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'))
ENG
18:54
17-11-2022


ibm project - Google Drive
ModelBuliding.ipynb - Colabora
colab.research.google.com/drive/1sHHA930xS9ROC_Vb1VyW_yqWGzcEfb-S
Gmail
YouTube
Maps
ModelBuliding.ipynb
File Edit View Insert Runtime Tools Help Last edited on November 3
Comment
Share
+ 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
ENG
18:54
17-11-2022



TESTING:

TEST CASE:



USER ACCEPTANCE TESTING:

1.PURPOSE OF DOCUMENT:

The purpose of this document is to briefly explain the test coverage and openissues of the [AI-Powered Nutrition Analyzer For Fitness Euthusiasts] project at the time of the release to UserAcceptance Testing (UAT).

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	15	4	2	3	25
Duplicate	1	0	3	0	4
External	2	3	0	1	6
Fixed	11	2	4	20	37
Not Reproduced	0	0	1	0	1
Skipped	0	0	0	0	0
Won't Fix	0	0	0	0	0
Totals	24	14	13	26	77

TEST CASE ANALYSIS

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

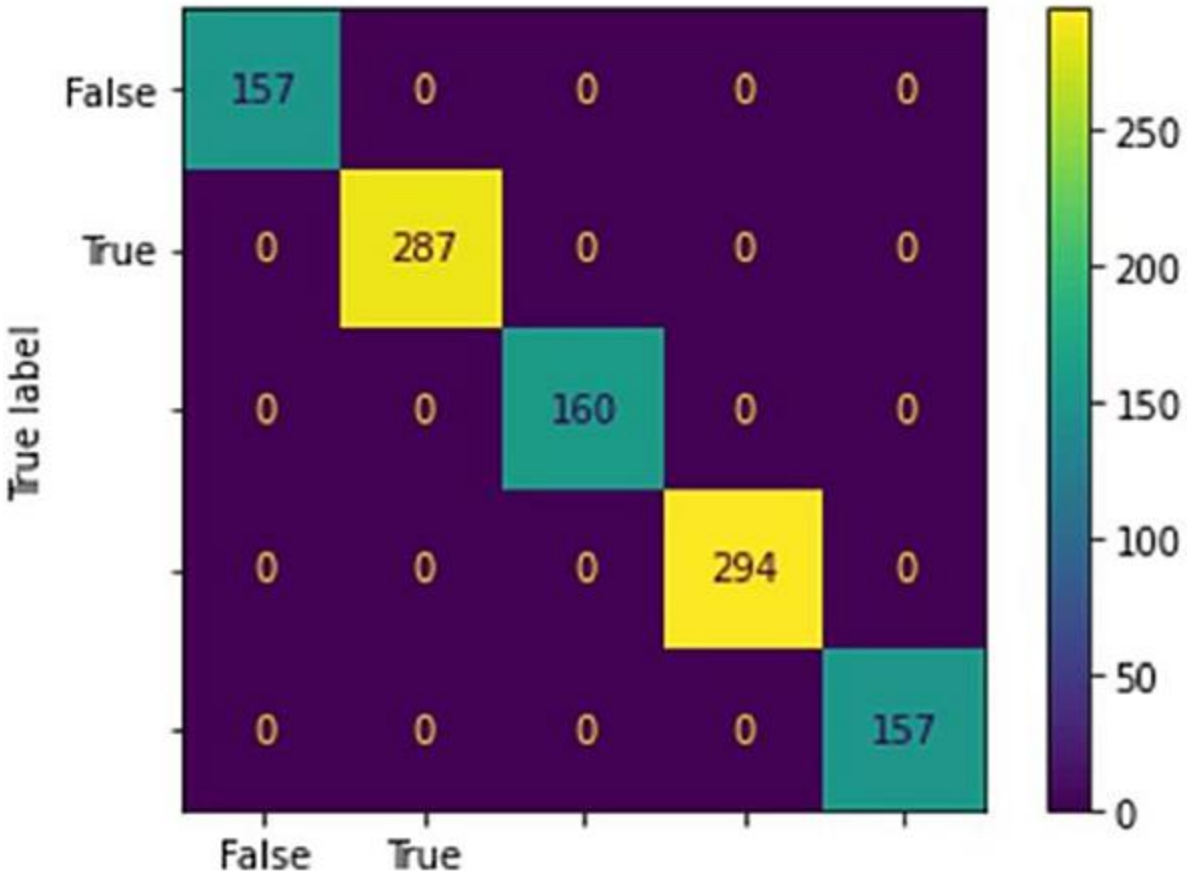
Section	Total Cases	Not Tested	Fail	Pass
Print Engine	5	0	0	5
Client Application	15	0	0	15

Security	2	0	0	2
Outsource shipping	3	0	0	3
Exception Reporting	1 5	0	0	15
Final Report Output	5	0	0	5
Version Control	2	0	0	2

9 RESULTS

PERFORMANCE METRICS

Confusion Matrix




```
print(metrics.classification_report(test_data['label'].values, test_data['model_preds'].values))
```

	precision	recall	f1-score	support
0	1.00	1.00	1.00	157
1	1.00	1.00	1.00	287
2	1.00	1.00	1.00	160
3	1.00	1.00	1.00	294
4	1.00	1.00	1.00	157
accuracy			1.00	1055
macro avg	1.00	1.00	1.00	1055
weighted avg	1.00	1.00	1.00	1055

1 Accuracy – 100 %

```
[8] print(f"the accuracy is {metrics.accuracy_score(test_data['label'].values, test_data['model_preds'].values)}")

    the accuracy is 1.0
```

2 Precision – 100 %

```
[11] print(f"the precision is {metrics.precision_score(test_data['label'].values, test_data['model_preds'].values, average = 'weighted')}")

     the precision is 1.0
```

3 Recall – 100 %

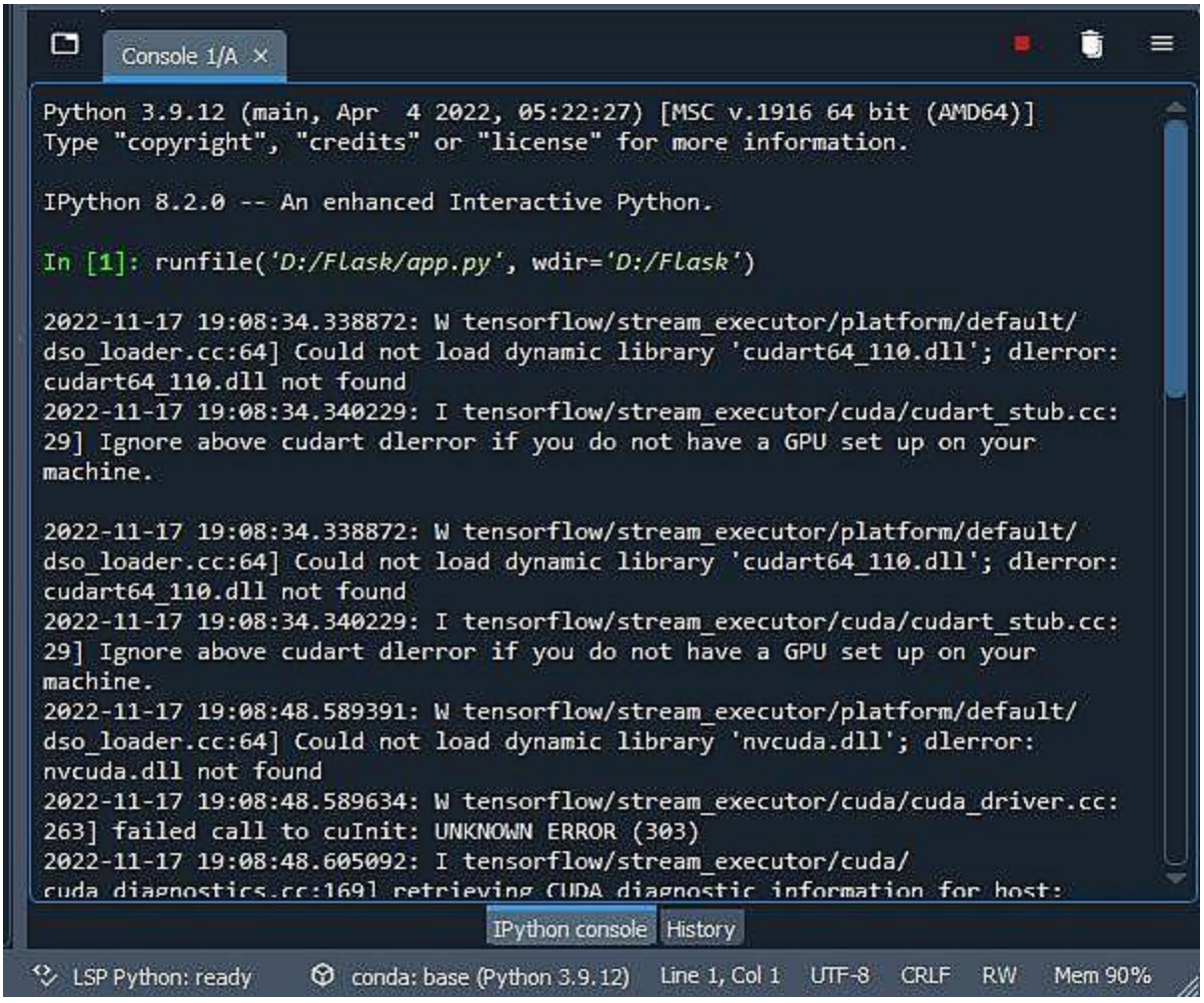
```
[12] print(f"the recall is {metrics.recall_score(test_data['label'].values, test_data['model_preds'].values, average = 'weighted')}")

     the recall is 1.0
```

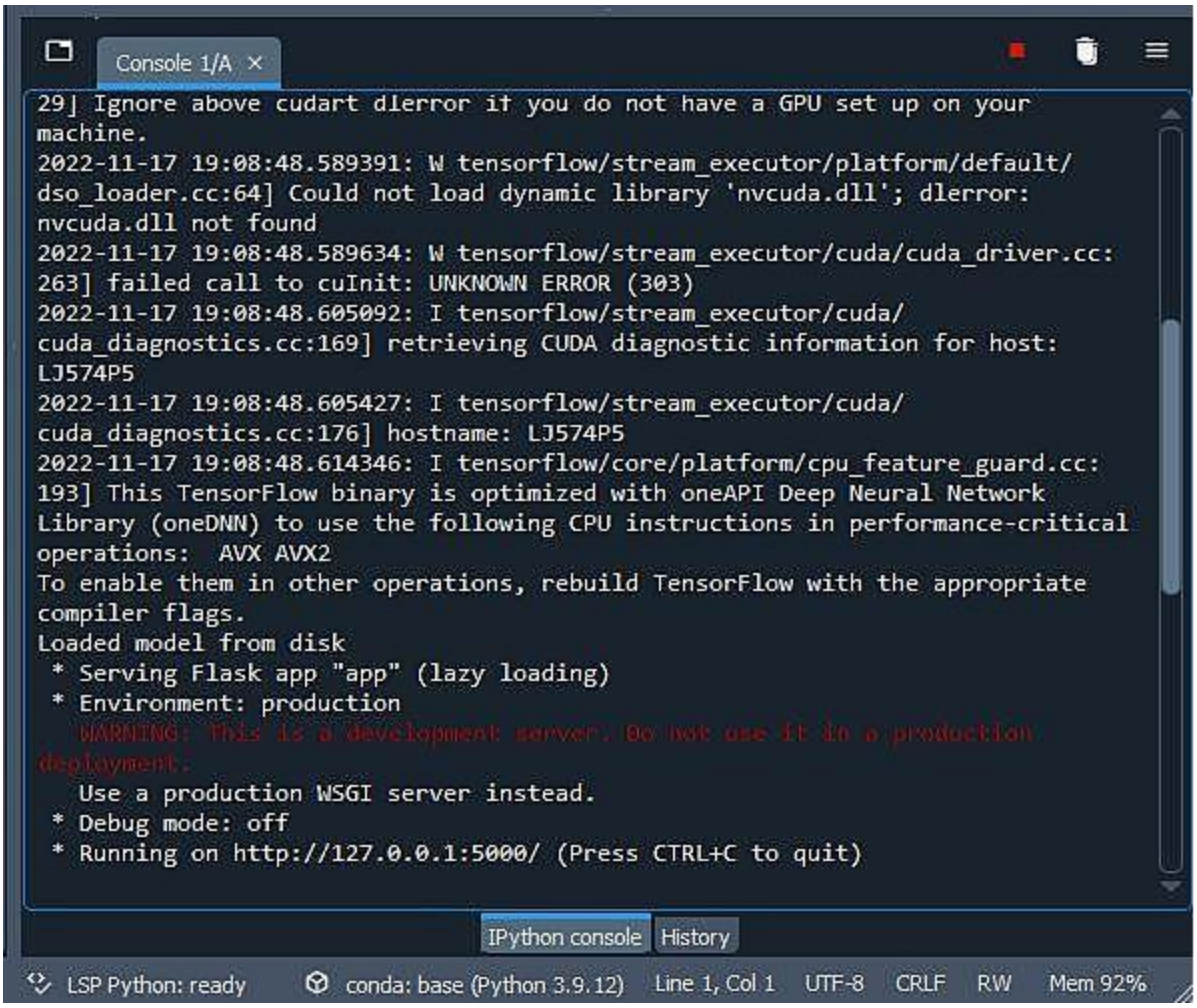
4 Specificity – 100 %

```
print(f"the specificity is {metrics.recall_score(test_data['label'].values, test_data['model_preds'].values, pos_label=0,average = 'weighted')}")

the specificity is 1.0
```

6. F1-Score – 100 %

```
[13] print(f"the f1 score is {metrics.f1_score(test_data['label'].values, test_data['model_preds'].values,average = 'weighted')}")

     the f1 score is 1.0
```





OUTPUTS

home.html

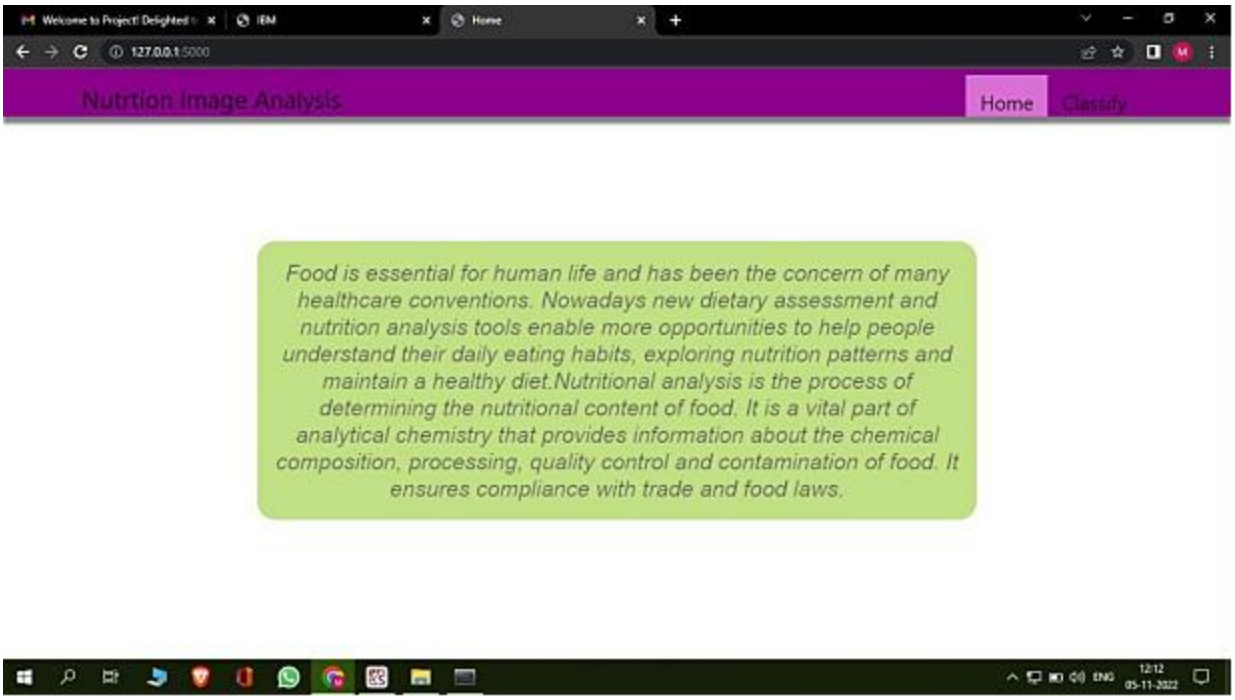

Nutrtion Image Analysis
Home
Classify
Food is essential for human life and has been the concern of many healthcare conventions. Nowadays new dietary assessment and nutrition analysis tools enable more opportunities to help people understand their daily eating habits, exploring nutrition patterns and 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.

image.html

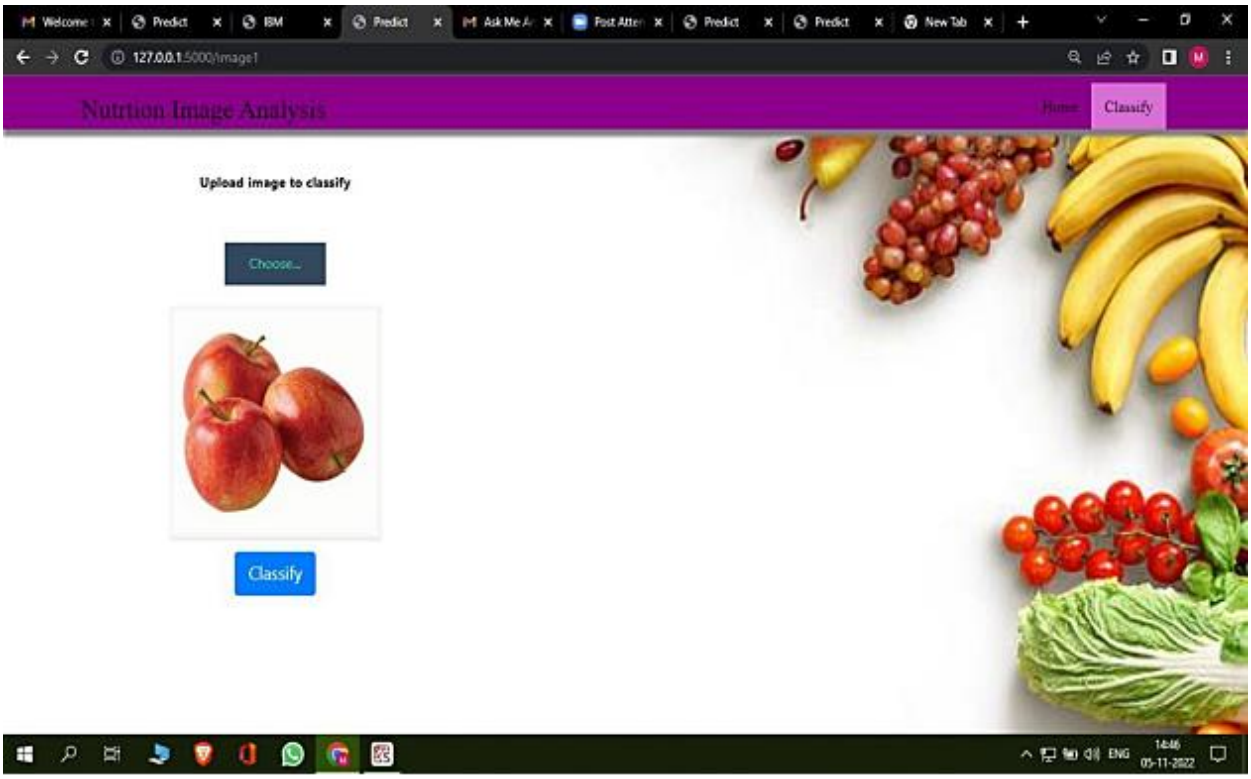
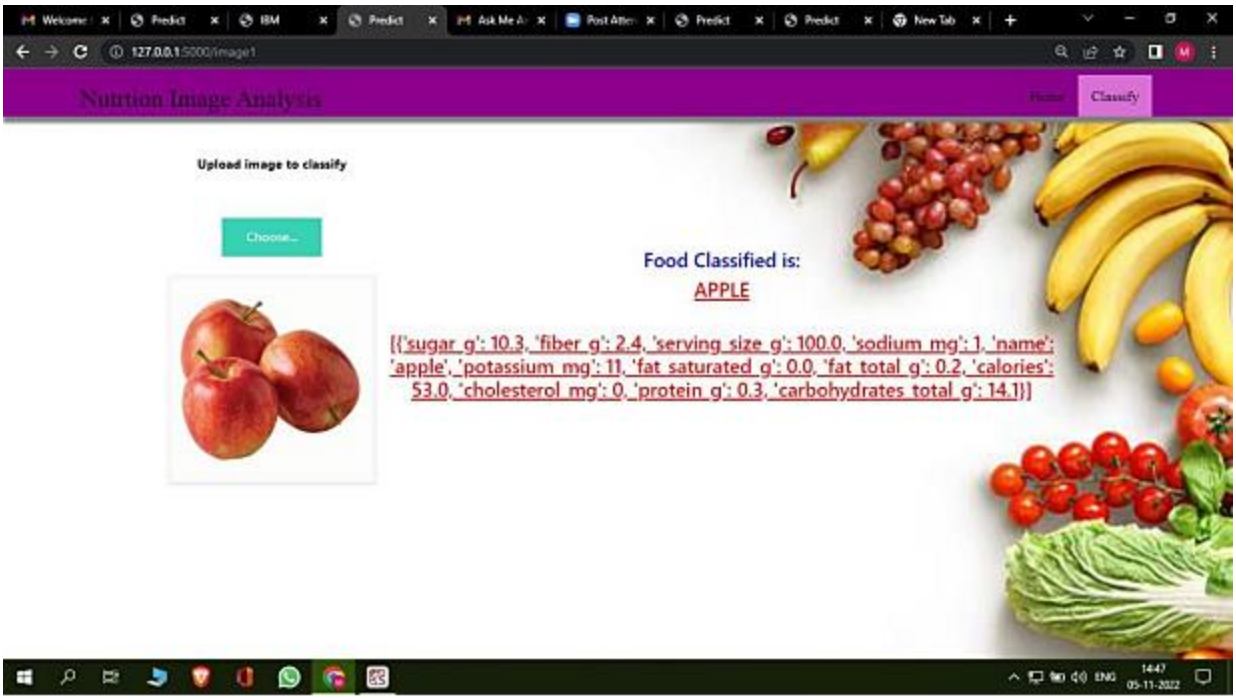
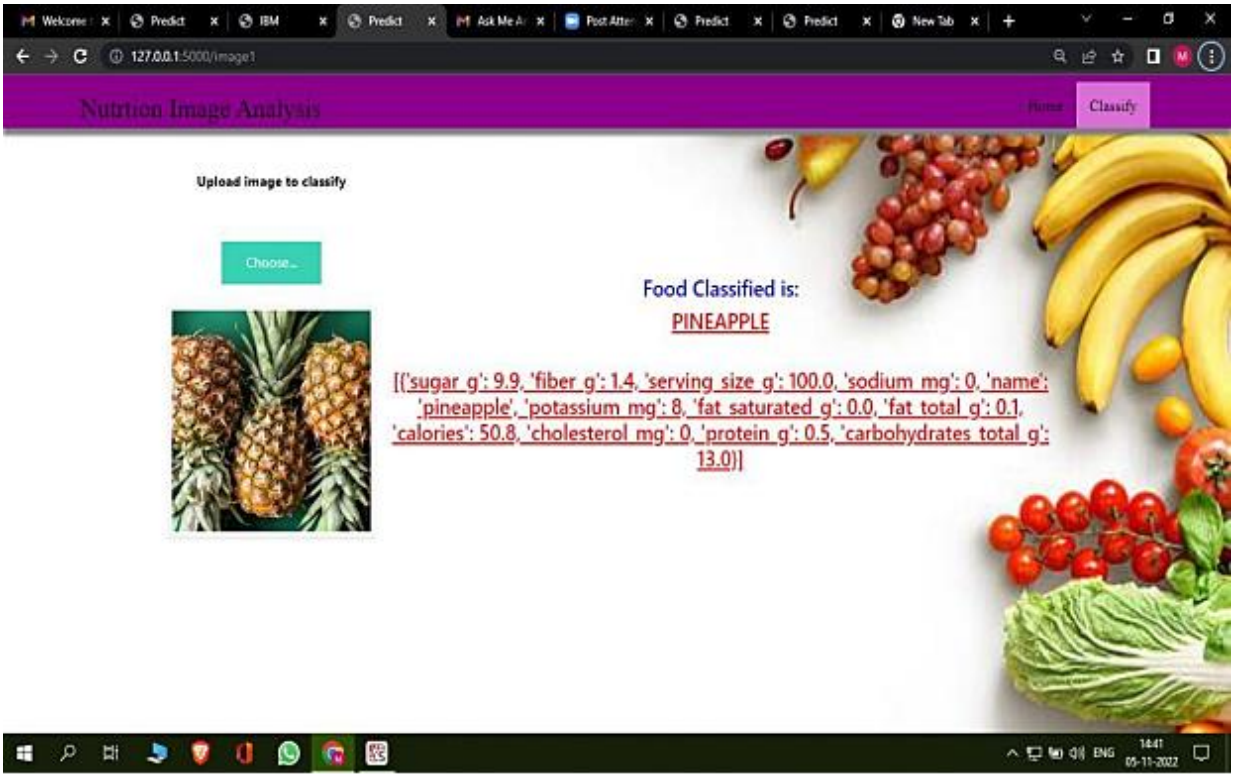
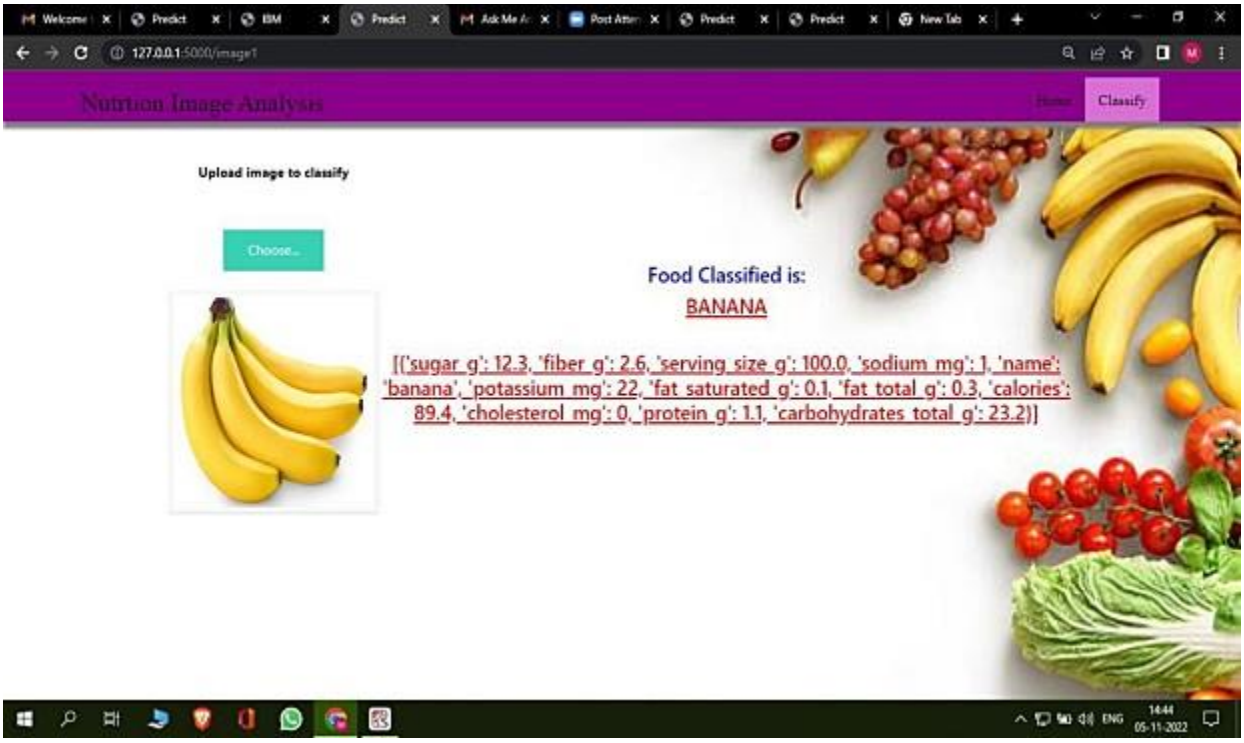
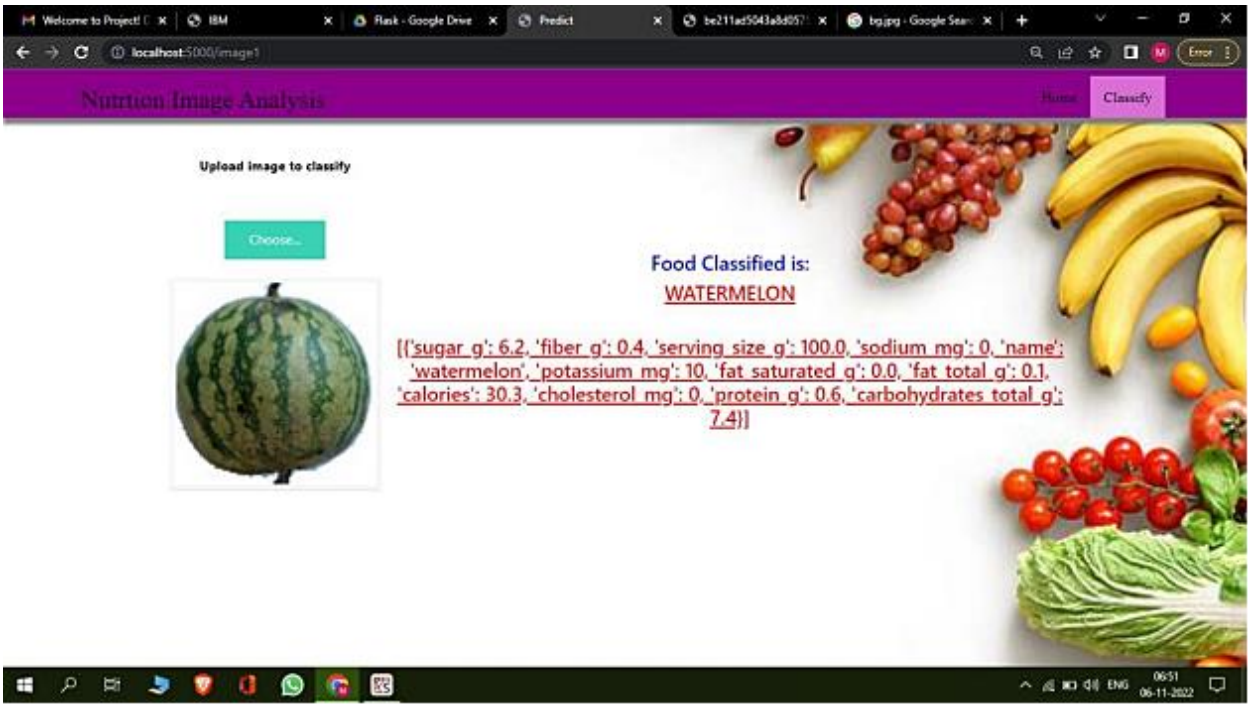

Nutrtion Image Analysis
Home
Classify
Upload image to classify
Choose...
Classify

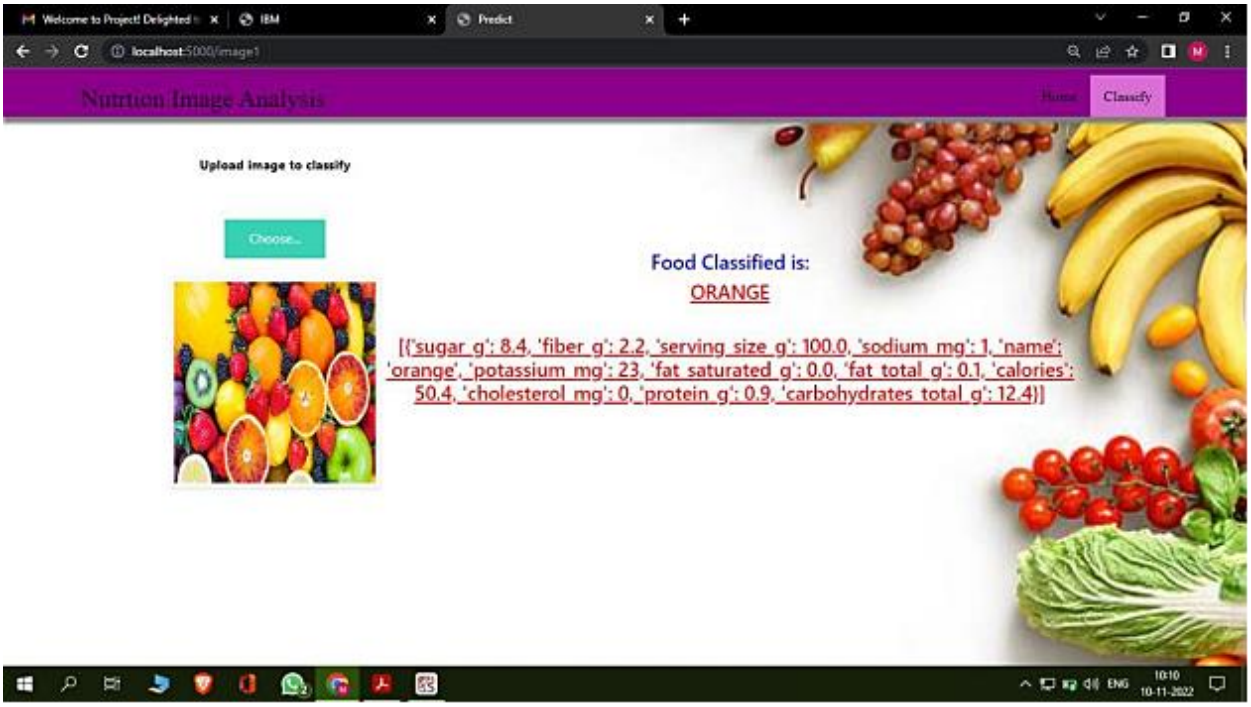
image prediction.html






127.0.0.1:5000/image1
Nutrition 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}]
ENG
14:44
05-11-2022


localhost:5000/image1
Nutrition 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}]
ENG
06:51
06-11-2022



10. ADVANTAGESD & DISADVANTAGES:

Advantages:

- Enables more opportunities to help people understand their daily eating habits exploring nutrition patterns and maintain a healthy diet.
- No more taking appointments from the dietician
- Less expensive with greater efficiency and correct results.
- AI-supported automated nutrition intervention utilizing mobile technology on glycemic control in patients with type 2 diabetes mellitus.
- The study was designed to fill the information gap between the demand for mobile technologies promoting behavior change in eating habits and the scientific evidence supporting its efficacy compared to the conventional in-person interventions by humans .

- Personalization of goal-settings and feedbacks are two additional strengths of AI-supported nutritional intervention and key components for the easy use of healthcare apps .

Disadvantages:

- Participants are limited to those who own and use a mobile phone, the results may not be generalized to generations with relatively lower information and communication technology literacy.
- People who adhere to the healthcare apps are likely to have a deliberative style rather than intuitive style when making health-related decisions.
- The patients in the AI arm would know that they are getting the experimental intervention, which may result in better outcomes in the AI intervention than the control.
- Independent contribution of nutritional intervention is to be evaluated by accounting for the effects of other wearable devices monitoring activities.
- The specific identified pattern is moderately unexpected, since it does not agree with previously described food intake curves that point either to a steady or a decelerated eating rate across meals .

11.Conclusion:

Exercise in the heat is associated with varying levels of thermal stress and potential effects on the health and performance of the athlete. Nutritional strategies before, during, and after exercise can address different aspects of exertional heat stress. These strategies need to be implemented using protocols that are individualized and made practical for the specific needs of the athlete and their event.

12 FUTURE SCOPE:

1. Mindful Eating and Food as Medicine:
    a. The distinction between food and supplements blur as functionalities, such as immune support or gut health, become apriority for consumers.
2. Plant-Based Eating and Alternative Proteins:
    a. Plant-based products accelerated this past year due to demand forhealthy food options during the pandemic

3. From Farm to Fork: Food Tech, Origins and Security:

Demand for sourcing transparency combined with unprecedentedinvestment in tech is advancing the ability to trace food from production to consumption.

12 APPENDIX

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
modelfrom 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 pagedef home():
    return render_template('home.html') #rendering the home page


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



@ app.route('/predict' ,methods=['GET','POST']) # route to show the
predictions ina 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 inuploads folder
        f.save(filepath) #saving the file


        img=image.load_img(filepath,target_size=(64,64)) #load and
reshaping theimage
        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(x)
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)
 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"></scr ipt>

<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"></scri pt>

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

background-image: url("https://www.livingproofnyc.com/wp-

```
    content/themes/livingproof/assets/img/hero-
    background.jpg");
        background-size: cover;
    }
    .bar
    {
margin: 0px;
padding:20px;-
color:white;

opacity:0.6;
color:black;
font-family:'Roboto';

font-style: italic;
border
radius:20px;
font-size:25px;
    }
    h3
    {
    margin:0px;
```

```
padding:0px;
background-
color:#9ACD32;
width: 800px;
opacity:0.6
font-family:'Roboto';
font-style: italic;
border-
radius:20;
font-
size:25px;
}
a
{

text-
colur:grey;
floating;
decoration:none;
font-style:normal;
padding-right:20 px;
}
```

```
a:hover{
background-
color:black;
color:white;
border radius:50px;
font_height:30px;
padding-left:10px;
}
.div1{
 background
 colour:white
 grey:
 border: 10px
 solid peach;
 padding: 20px;

margin:50px;
height:500px;
}
.header {position: relative;
                top:0
                margin:0
```

```
                z-index:1;

                right:0;

                left:0;

                background-color:
                #8B008B ;color:
                white;

                box-shadow: 0px 8px
                4px grey;overflow:
                hidden;
                padding-left:20px;
                width:100%
                height:8%
                text-align: center;
            }
            .topnav {
    overflow: hidden;
    background-color:
    #FCAD98;
   }
text-align:center;
```

```
padding:14px1px;
text-
decoration:none;
font-size: 22px;
    }

    .topnav-right a:hover {
    # Loading the model
    model=load_model('nutrition.h5')
    print("Loaded model from disk")

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

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

```
@ app.route('/predict' ,methods=['GET','POST']) # route to show the predictions ina 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 inuploads folder

        f.save(filepath) #saving the file



        img=image.load_img(filepath,target_size=(64,64)) #load and reshaping theimage

        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]])



        result(x);
        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)
print(response.text)
return response.json()
['items']if _name_ == "_main_":
# running the app

app.run(debug)

<!--Brian Tracy-->



    <div class="header">
```

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


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

    <ahref="{{ url_for('image1')}}">Classify</a>
  </div>
</div>
</div>
<br>
<br>
<br>
<br>
<br>
<br>
<br>
<br>
<h1>
```

<h3>Food is essential for human life and has been the concern ofmany healthcare conventions. Nowadays new dietary assessmentand 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 contaminationof food. It ensures compliance with trade and food laws.</h3>

</h1>

</body>

</html>

IMAGE.HTML

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

<h5>Upload image toclassify</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="btn-predict">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>
```

```
</div>

</div>

{% 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"></scr
ipt>

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

<script="https://cdn.bootcss.com/bootstrap/4.0.0/js/bootstrap.min.js"></script>

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

```
    background-image:
    url("https://i.pinimg.com/originals/be/21/1a/be211ad5043a8d05757a3538bdd8f
    450
    .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
{
colour:grey;
floating:right;
text-
decoration:none;
font-
style:normal;
padding-
right:20px;
}
a:hover{
background-
color:black;
color:white;

border-
radius:15px
;0font-
size:30px;
padding-
```

```
    left:10px;
    }
    .div1{
      background-color:
      lightgrey;width:
      500px;
      border: 10px
      solid peach;
      padding: 20px;
.header {position: relative;
                        top:0px;
                        margin: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;

    background-color:

    #FCAD98;

   }

text-align: center;
padding: 14px 16px;
text-decoration: none;
font-size: 18px;
   }

   .topnav-right a:hover

    { background-color:

    #FF69B4;color:

    black;

   }
```

```
.topnav-right a.active
  { background-color:
  #DA70D6;color:
  black;
}
}
</style>
</head>
<body>
<div class="header">
<div style="width:50%;float:left;font-size:2vw;text-align:left;color:black;
padding-top: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>

<script src="{{ url_for('static', filename='js/main.js') }}"
type="text/javascript"></script>
 </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;
font-size: 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 */border-radius: 50%;

          animation: spin 1s linear infinite;

      }
```

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

STYLE.CSS

```
body{
background-image:url(bg.jpg);
background-size: 400% auto;
background-repeat: no-repeat;
background-position:center;
background 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
{
text-align:justify;
text-justify: auto;
      }
.parallax {

        /* The image used */

           background-image: url("doc.jpg");
 /* Set a specific height */min-height: 750px;

 /* Create the parallax scrolling effect */
background-attachment: fixed;
background-position: center;
background-repeat: no-repeat;
background-size: cover;
      }
```

```
    html {
      scroll-behavior: smooth;
    }
    }
    div.background
     {
    background: url("static/bgg2.jpg");
min-height: 5px;
background-attachment: fixed;
background-position: center;
background-repeat: no-repeat;
background-size: cover;
    }
 #navbar{
background-color:#fff;
color:#333;
    }
 #navbar ul{
            padding:0;
            list-style: none;
    }
 #navbar line{
```

```
        display:inline;
    }

    #navbar a{
        color:#fff;
        text-
        decoration:
        none;font-
        size:18px;

        padding-right:15px;
    }
#showcase{
min-height:300px;
margin-bottom:30px
    }

    }
#main{
        box-sizing: border-box;
    }
    #sidebar
```

```
    {
floating-background-color: #ffcccc;color:#000;
padding-left:10px;
padding-right:10px;
padding-top:1px;
box-sizing: border-box;
    }
      image preview(width);
height: 10px;
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: 10%;
        height: 10%;
        background-size:100px,10px;
        background-repeat: no-repeat;
        background-position: center;
    }
 input[type="file"]
{
display: none;
    }
```

```
.upload-label{
display: inline-block;
padding: 12px 30px;
background: #39D2B4;
colour:#fff;
transition: all .4s;
cursor: pointer;
      }
 .upload-label:hover
{
 background: #34495E;
color: #39D2B4;
      }
text-align: center;
cursor: pointer;
text-transform: uppercase;
outline: none;
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;
z-index: 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%;
```

```
animation: spin 1s linear infinite;
      }
 @keyframes spin {
0%
{
 transform: rotate(0deg);
 }
 100%
{
 transform: rotate(360deg);
 }
      }
main-footer
{
color:#fff;
text-align: center;
padding:1px;
margin-top:0px;
      }
```

```
 @media(max-width:600px)
{
#main{
            width:0;
            floating;none;
            }
 #sidebar{
            width:10%
            floating: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,
success: function (data)
```

```
{
                // Get and display the result
                $('.loader').hide();
                $('#result').fadeIn(600);
                $('#result').html(data);
                console.log('Success!');
            },
        });
    });

});
```

GITHUB

IBM-EPBL/**IBM-Project-40237-1660626501**

PROJECT DEMO LINK

https://drive.google.com/file/d/1V1l-hyJ8SUmAy6N6ITXh6xmDcuzOpZlr/view?usp=drivesdk