AI-POWERED NUTRITION ANALYZER FOR FITNESS ENTHUSIASTS

TEAM ID:PNT2022TMID11872

TEAM LEADER:KAVINAYA V(73771921148)

TEAM MEMBERS: KANIMOZHI. P(73771921139)

KANIMOZHI.S(73771921141)

GOWRISHREE(73771921130)

DEPARTMENT: INFORMATION TECHNOLOGY

COLLEGE NAME: K S RANGASAMY COLLEGE OF TECHNOLOGY

FACULTY MENTOR :

GAYATHRI.S

ABSTRACT:

AI Powered Nutrition Analyzer For Fitness Enthusiasts suggests using an intelligent-Powered create diet regimens depending on the user's inputs. The system generates a food plan using information about a person's lifestyle and health requirements. 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 and fitness issues, 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 people at the preferred time and via mobile phones without having to visit arises in order to facilitate them with a proper diet chart along with light exercises according lifestyle and cope up with their busy schedule.

The main goal of the software is to provide the use the a list of all feasiblediet 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.

TABLE OF CONTENTS

1.INTRODUCTION

1.1PROJECT 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, explore 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.

1.2PURPOSE

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

2.1EXISTING 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 144children 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 80street 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

i. Davenport.T and Kalakota R.2019.The potential for ArtificialIntelligence 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 AIare 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 inFitness: 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 technologicalissue, instead it invokes a host of medical, legal, ethical, and societal questions thatrequire thorough exploration.

2.3 PROBLEM STATEMENT DEFINITION

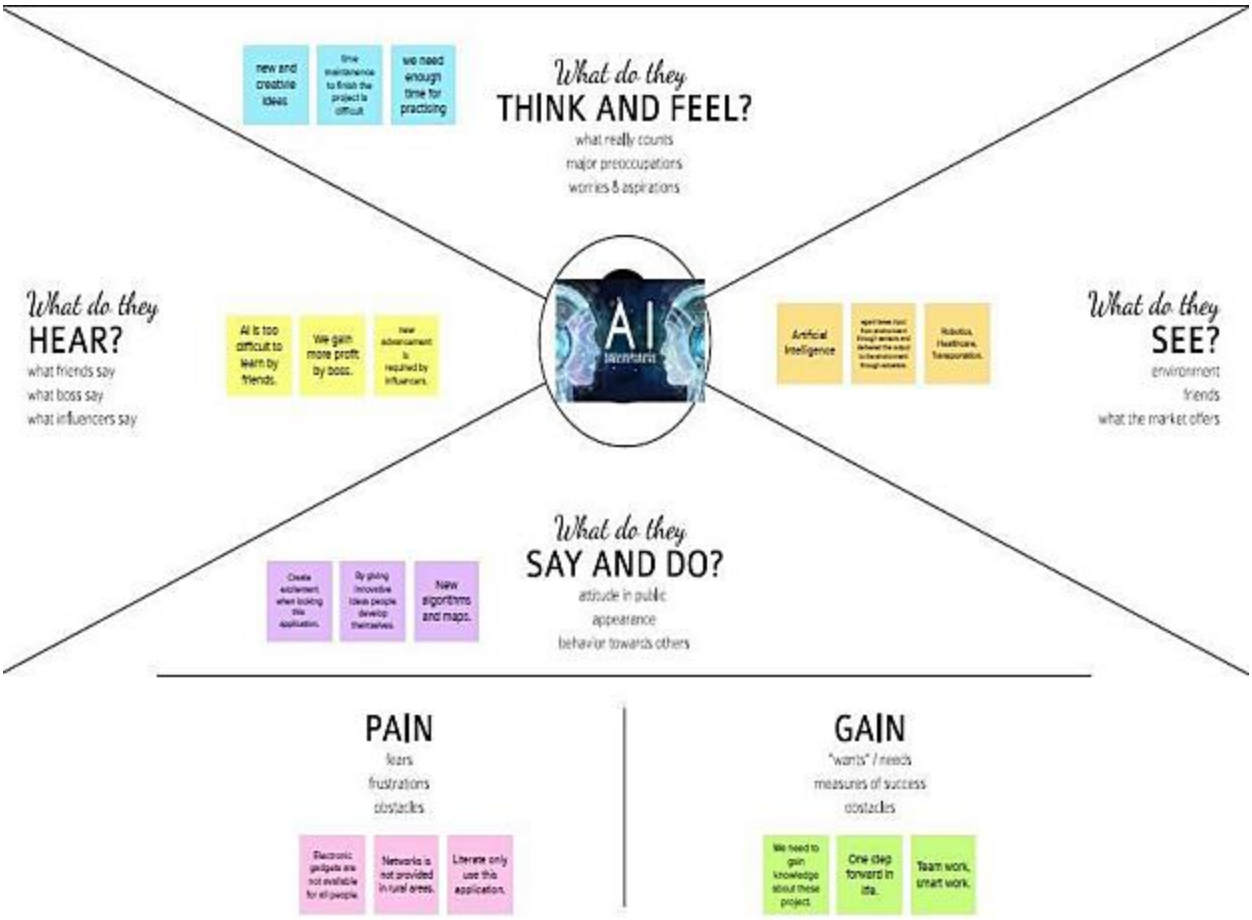
Due to change in food habits people do not get aware of food items. Our project is to get details about food nutritions, carbohydrate, protein and fat.

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.

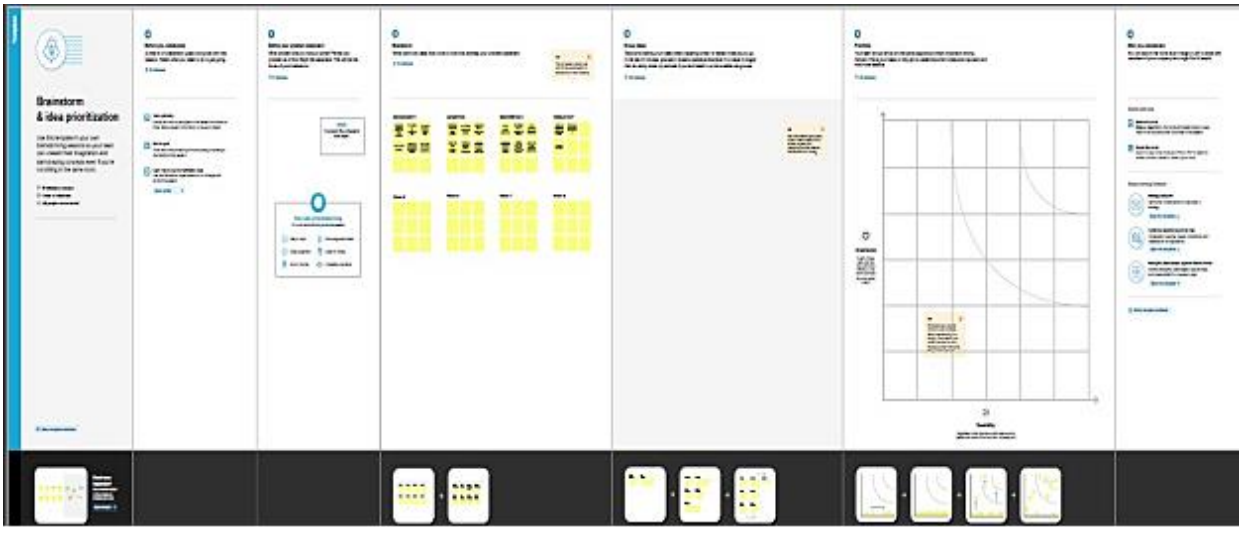
The World Health Organisation(WHO) data reveals that more than60% of world's population is not physically active enough to induce health benefits.

3. IDEATION & PROPOSED SOLUTION

3.1 EMPATHY MAP CANVAS



3.1 IDEATION &BRAINSTORMING



3.3 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 thefruits like (Sugar, Fibres, Protein, Calories, etc.).

3.4 PROBLEM SOLUTION FIT

4.REQUIREMENT ANALYSIS

4.1FUNCTIONAL REQUIREMENTS

FR NO.	FUNCTIONAL REQUIREMENTS(EPIC)	SUB REQUIREMENT (STORY/SUBTASK)			
FR-1	USER REGISTRATION	a. Registration through Gmail b. Registration through Mobile Number c. Registration through Face-book			
FR-2	USER CONFIRMATION	a. Confirmation via Email b. Confirmation via OTP			
FR-3	USER DETAILS		Age	Food	
			Height	Recipe	
			Weight	Added Ingredients	

			Diseases if any	Age	
			Conditions if any		
			Allergies if any		
FR-4	USER REQUIREMENTS	a. The user simply inputs your recipe ingredients and amounts. b. The software will instantly produce an accurate readout of your dish in terms of nutritional analysis in a readable format that consumers are familiar with. c. With already given details the system can alert the consumer if any content of their allergies ,it can alert the consumer			

4.2 NON FUNCTIONAL REQUIREMENTS

FR.NO	NON-FUNCTIONAL REQUIREMENTS	DESCIPTION
NFR-1	USABILITY	• No training is required to access the Nutrition Analyzer. • The results should be loaded within30 seconds. • It should be user friendly and comfortable. • It should be simple and easy to use. • The results should be self explanatory so that it can be understood by common people.

NFR-2	SECURITY	• AI powered nutrition analyzer for fitness shouldcontain more security in which our data which entered or maintained should be more security. • With the help of the username and password itprovides more security n which it can access more securable and the data areprivate. • It should be social-economic which should access to sufficient and safe to use.

		• It is Important that the AI Powered nutrition analyzer for fitness provides should Must reliable. • How a person can find it is reliable? It is easy to find that is he/she can compare the nutrition based food with other nutrition related application so, it can easily rectify whether it is reliable or not. • But it take too much time, to avoid this a reliable application should made in which it itself produces whether we can get correct solution or not. So, it is necessary that the AI powered nutrition analyzer for

NFR-3	RELIABILITY	fitness should have proper data and information in which we can get a correct information about it and also get a proper guidance about it. a. With the proper guideness and proper information in which we can get a nutrition properly and we can have geta proper fitness plan. b. It should also provides the information on nutrition and health which it should prevent from health information on diseases, health risks and prevention guidelines. It should also provides an extension a research based online learning network with several resource areas, so it provides more reliability in that area. For more reliable it can also contains the calorie information, balanced diet plans, what type food can consumed at what time etc..... So, by this way it is reliable.

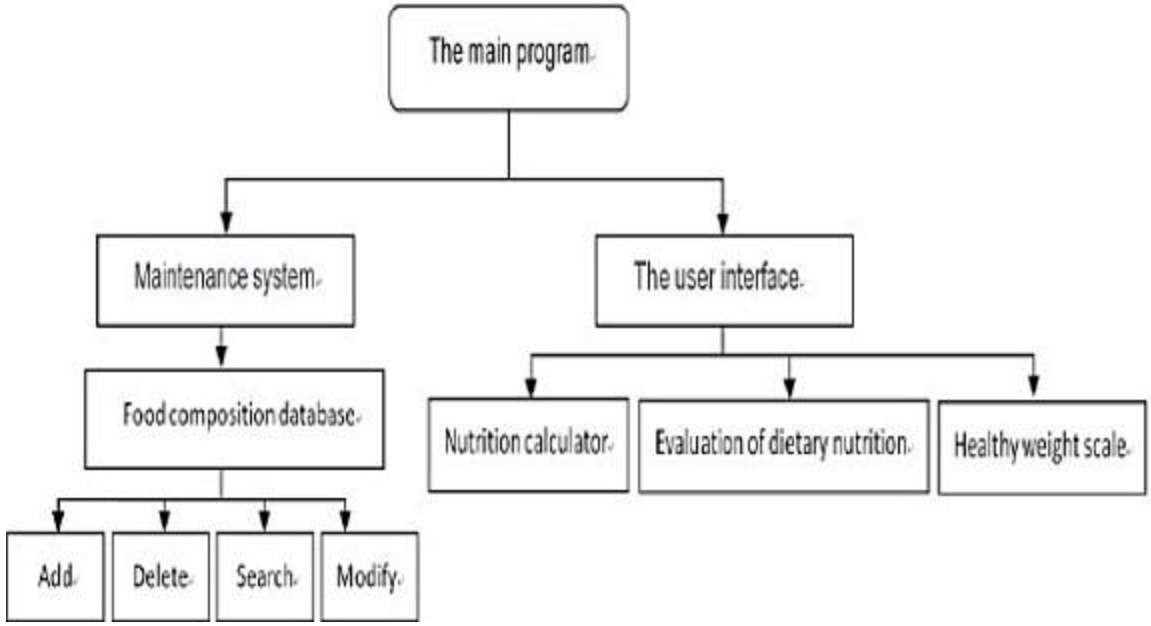
NFR-4	PERFORMANCE	a. It should provide more number of users to consume at any time and at any place. b. It should provide Reliability, Scalability, Security and Usability. c. It should contain minimum data while over-paging the websites or application and it is necessary that it should not exceed more than 20mb. d. While consuming the page it should provide the response as much as possible without any delayor time traffic. e. The connection should e properly maintained so that it can use while travelling orin remote places. f. The nutritious food to meet the irdietary needs and the food

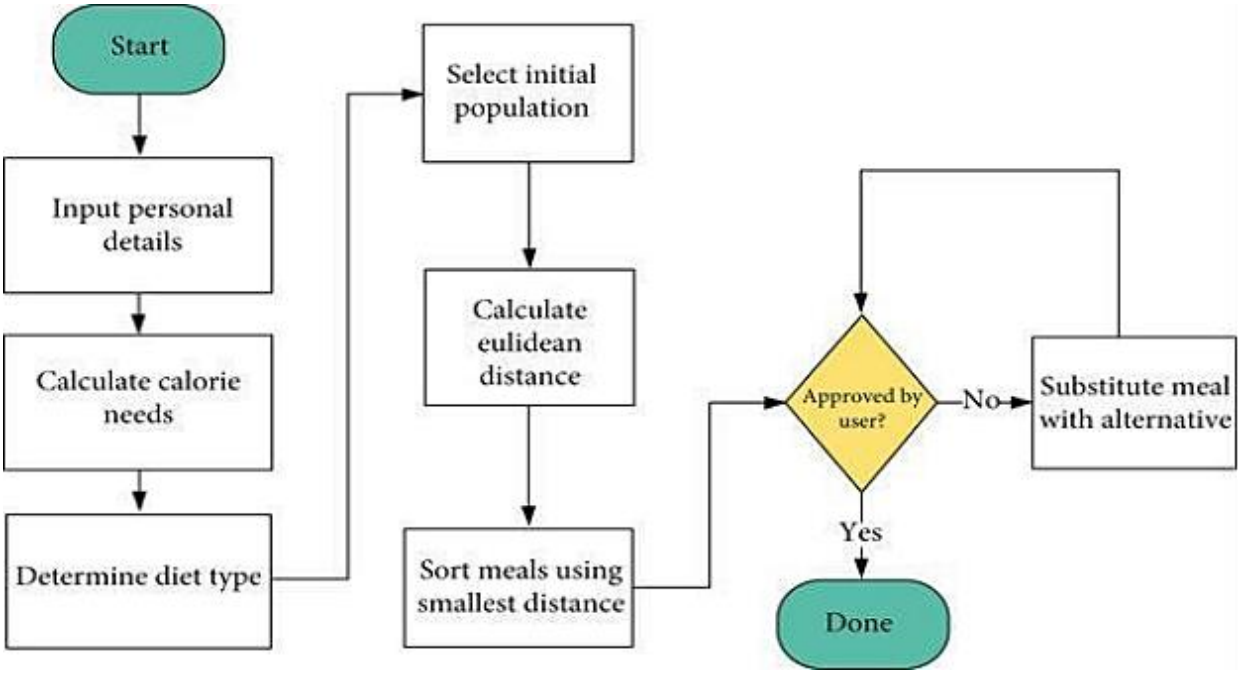
		preferences for an active andhealthy life. a. It should be consistently access, availability and affordability of foods and beverages that promotewell-being and prevent from diseases. b. It should suitable in all situationsthat exists to all people, at all times.
NFR-5	AVAILABILITY	a. Easy to access Data. b. Avoids Data redundancy and inconsistency. c. Fast and Efficient. d. User Friendly.

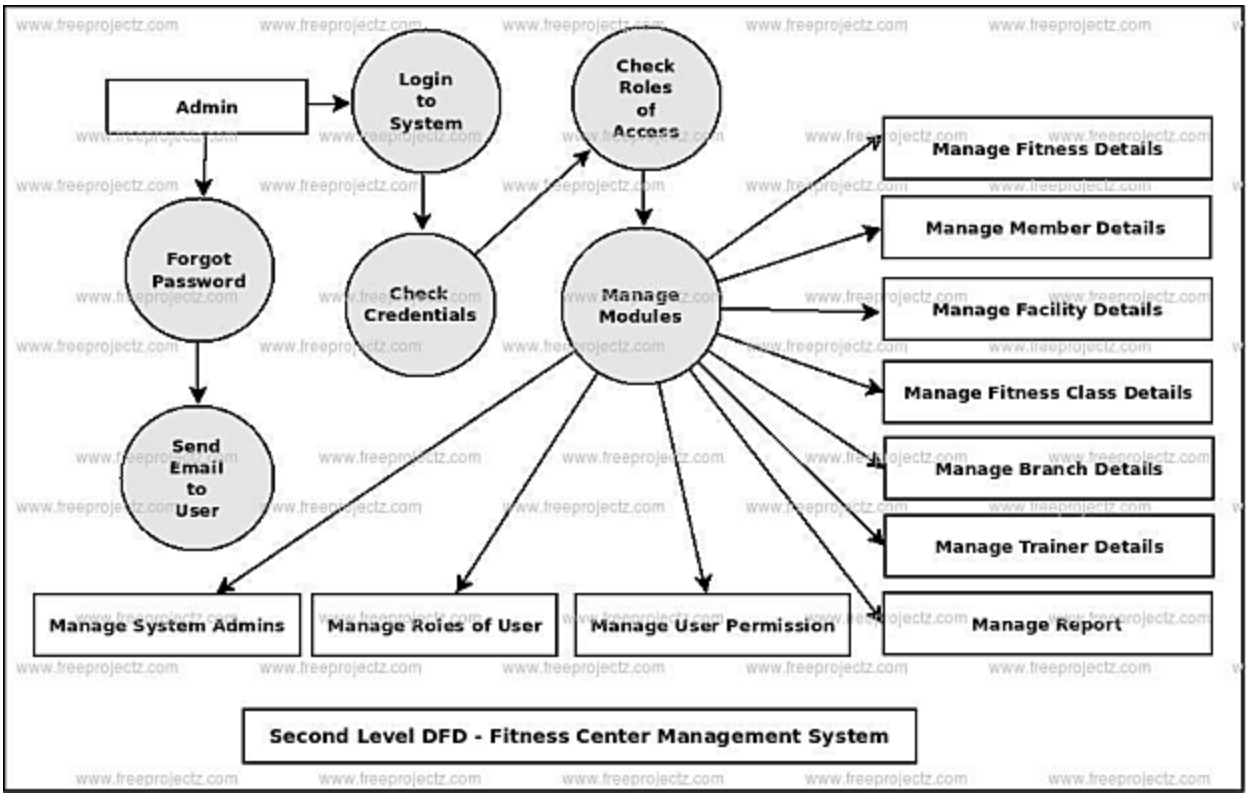
NFR-6	SCALABILITY	a. The architecture for AI powered Nutrition Analyzer for fitness provides the clear procedure daily consumption of food and helps theuser to maintain a healthy diet. b. According to their tracking systemimplemented in architecture provide the proper mechanism to the every individual of their nutrients intake which can be increased or decreased. c. The premium amount for analyzeris very much optimum.

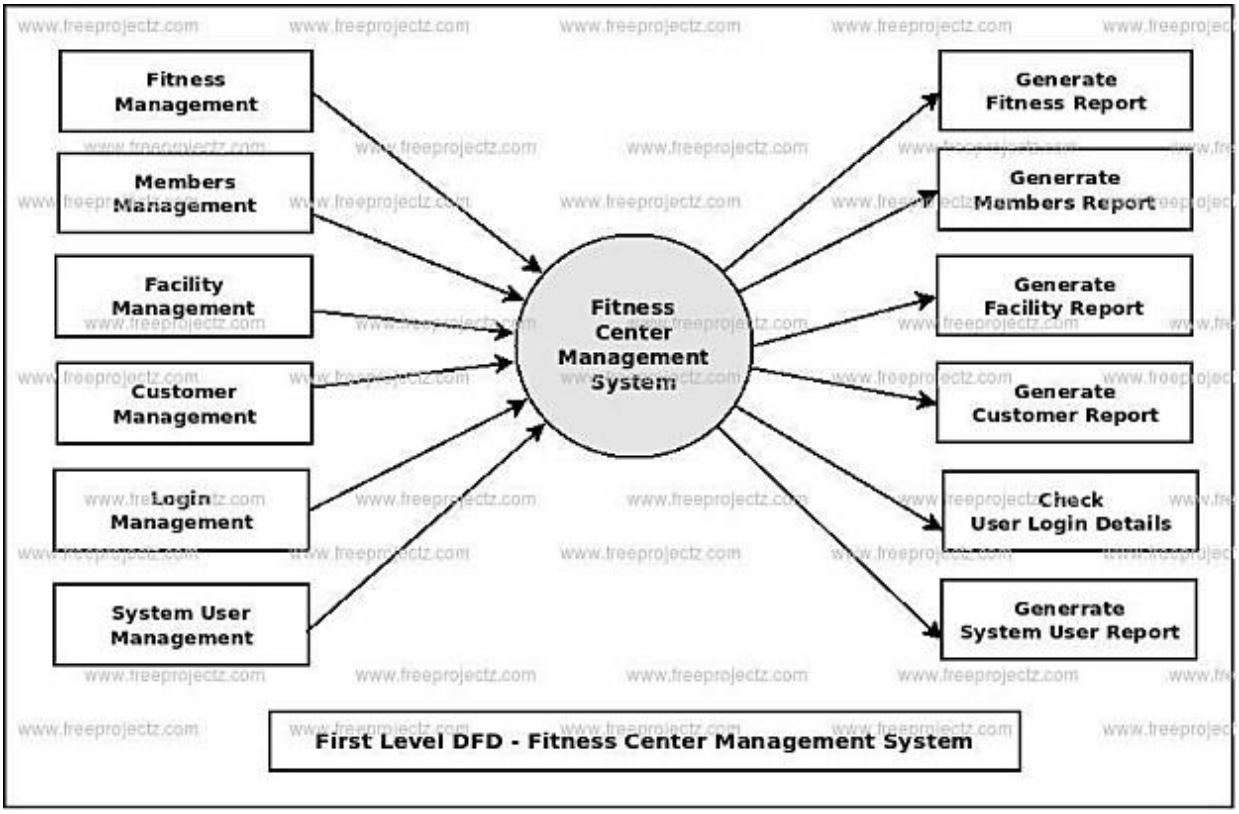
5. PROJECT DESIGN

5.1 DATA FLOW DIAGRAMS

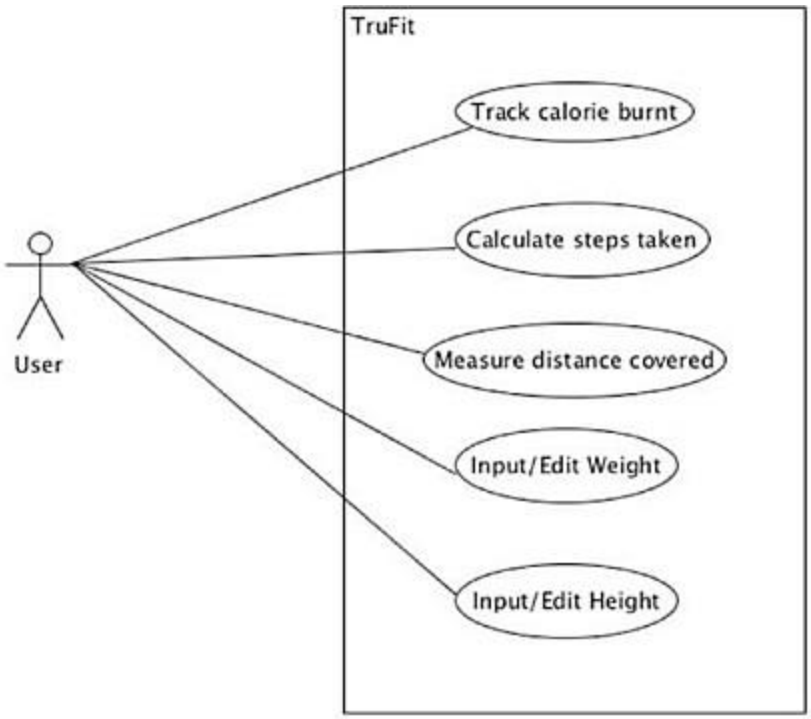


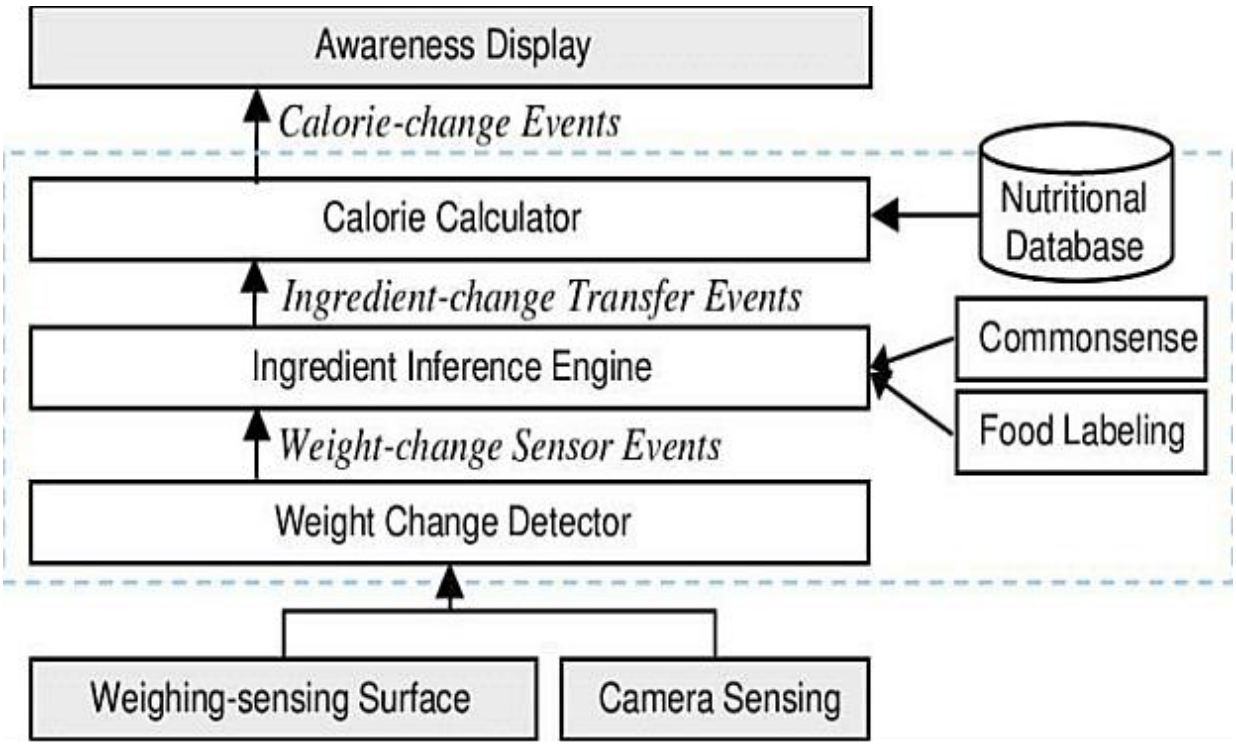


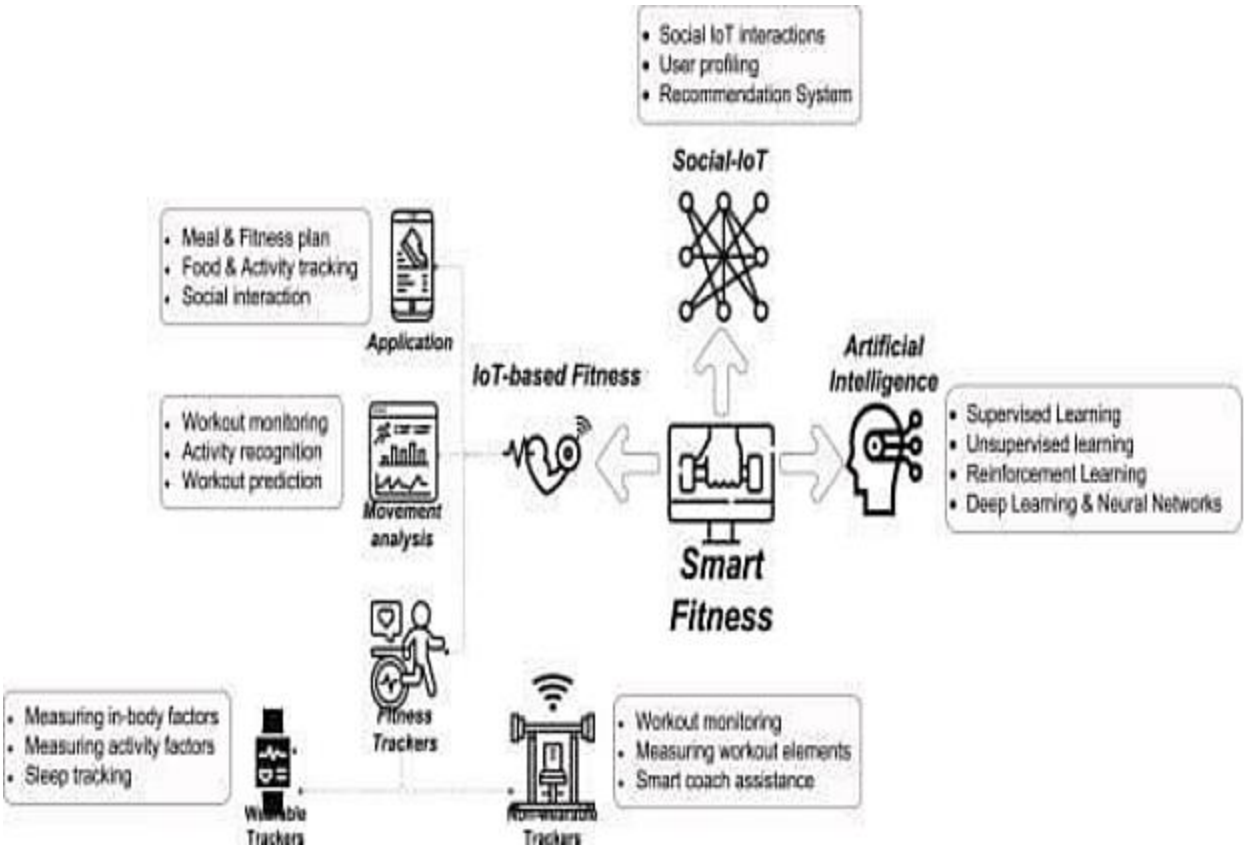

Admin
Login to System
Check Roles of Access
Forgot Password
Check Credentials
Manage Modules
Send Email to User
Manage Fitness Details
Manage Member Details
Manage Facility Details
Manage Fitness Class Details
Manage Branch Details
Manage Trainer Details
Manage Report
Manage System Admins
Manage Roles of User
Manage User Permission
Second Level DFD - Fitness Center Management System

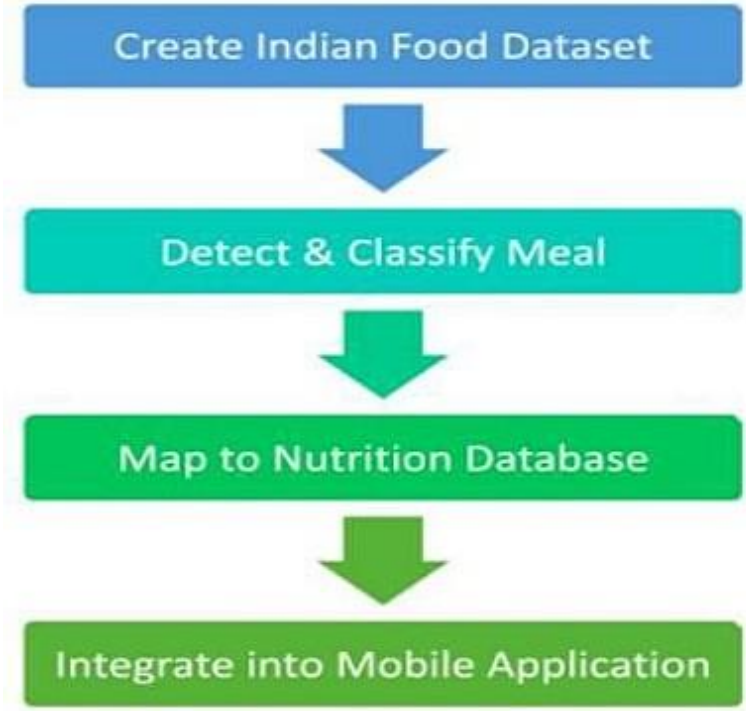

Fitness Management
Members Management
Facility Management
Customer Management
Login Management
System User Management
Fitness Center Management System
Generate Fitness Report
Generrate Members Report
Generate Facility Report
Generate Customer Report
Check User Login Details
Generrate System User Report
First Level DFD - Fitness Center Management System

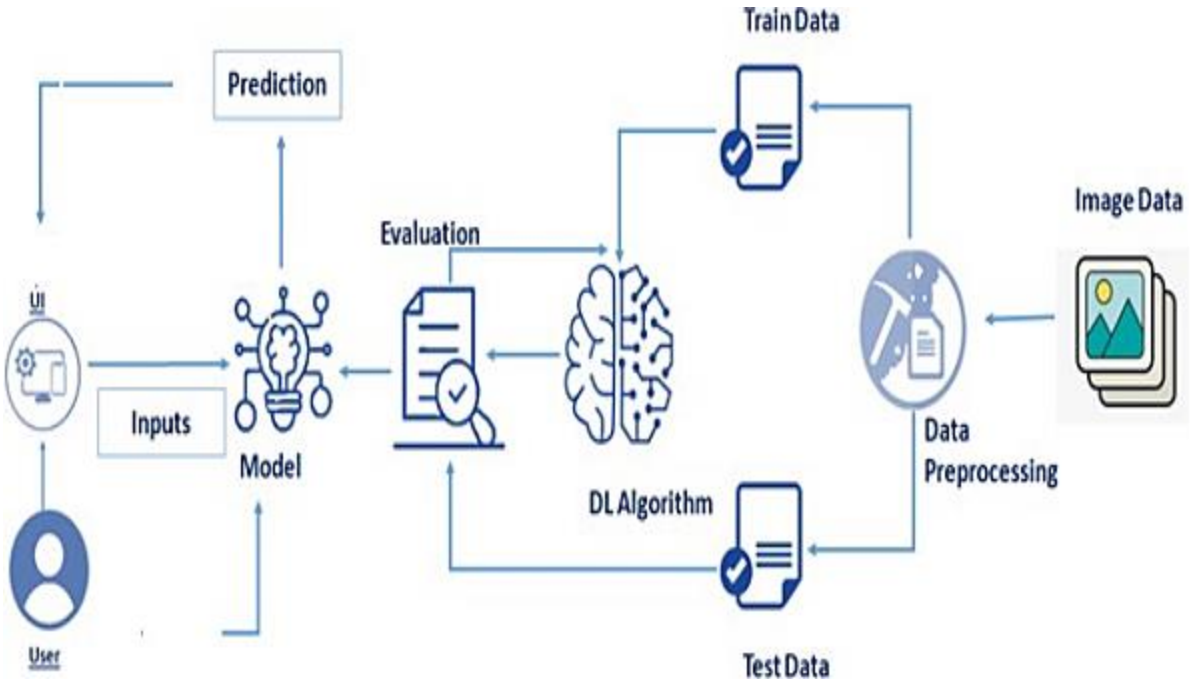
5.2 SOLUTION & TECHNICAL ARCHITECTURE






Social IoT interactions
User profiling
Recommendation System
Social-IoT
Meal & Fitness plan
Food & Activity tracking
Social interaction
Application
IoT-based Fitness
Artificial Intelligence
Workout monitoring
Activity recognition
Workout prediction
Movement analysis
Supervised Learning
Unsupervised learning
Reinforcement Learning
Deep Learning & Neural Networks
Smart Fitness
Fitness Trackers
Measuring in-body factors
Measuring activity factors
Sleep tracking
Workout monitoring
Measuring workout elements
Smart coach assistance


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



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 foranalysing the nutrient	Python
3.	Application Logic-2	IBM Watson Health is adigital tool that helps thehealthcare services through AI	IBM Watson STTservice
4.	Database	Datatype, Configurations,Data, etc.,	MSSQL

5.	Cloud Database	Cloud Database Service	IBM DB2, IBM Cloudant
6.	Notification	Nutrition notification willbe Sent from the server	Grid
7.	File Storage	File storage requirements	IBM Block Storageor Other Storage Services
8.	External API	External API is used inthe Application	IBM Weather API,Aadhar API
9.	Machine LearningModel	Detect and identify theimage 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 authentication, Security controls ,etc.,	Encryption, firewalls
3.	Scalable Architecture	Supports high workloads	Artificial Intelligence
4.	Availability	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 or 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 tothe 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 five daily camera picture captures.	Every day, my food consumption is calculated and analysed.	Medium	Sprint-2
	Diet Plan	USN-9	I, as a user, 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 ordashboard	High	Sprint-3
Customer Care Executive	Solving customerqueries	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

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	5	High	Kavinaya.V Kanimozhi.P Akalya M Gowrishree.B

			supporti ngthe model			
Sprint-1		USN-1	As a user, I can register for the application by entering my email, password, and confirming my password	5	High	Kanimozhi. P

Sprint-1		USN-2	As a user, I will receive confirmation email once I have registered forthe application	5	High	Gowrishree.B
Sprint-2		USN-3	As a user, I will receive confirmation email once I have registered forthe application	3	Low	Kanimozhi.S
Sprint-1		USN-4	As a user, I can register	3	Medium	Kavinaya.V
			for the applicati on through Gmail			

Sprint-1	Login	USN-5	As a user, I can log intothe application by entering email & password	5	High	Kanimozhi.P
Sprint-2	Model Building	USN-6	As a user, I can log intothe application by entering email & password	5	High	Kanimozhi.S
Sprint-2	Main Interface	USN-7	As a user I can view my calorie intakeby clicking photo of the food I eat	5	High	Gowrishree.B
Sprint-2	Package, Dashboard	USN-8	As a user I can choose variety of packages based on my requirement	4	Medium	Kanimozhi.P

Sprint-3	Diet planfor free users	USN-9	As a dietitianI provide daily plans for the betterment ofthe user	5	High	Gowrishree.B
Sprint-3	Personalized food habit-based diet plan for premium users	USN-10	As a Premium User, I can choose to follow diet plan based onmy food habits or the generalized one	3	Medium	Kavinaya.V
Sprint-2	User image analysis	USN-11	As a user I can track mycalorie intake, and know about my food in detail	5	High	Kanimozhi.S

Sprint-3	Improve efficiency of AI model	-	As a developer I have to givea better model that will analyse food precisely and	3	Medium	Kanimozhi.P
			provide accurate results			
Sprint-2	User Analysisrecord	US N-12	As a user, I can check theprevious records and I can analyse my food habits	4	Medium	Kavinaya.V

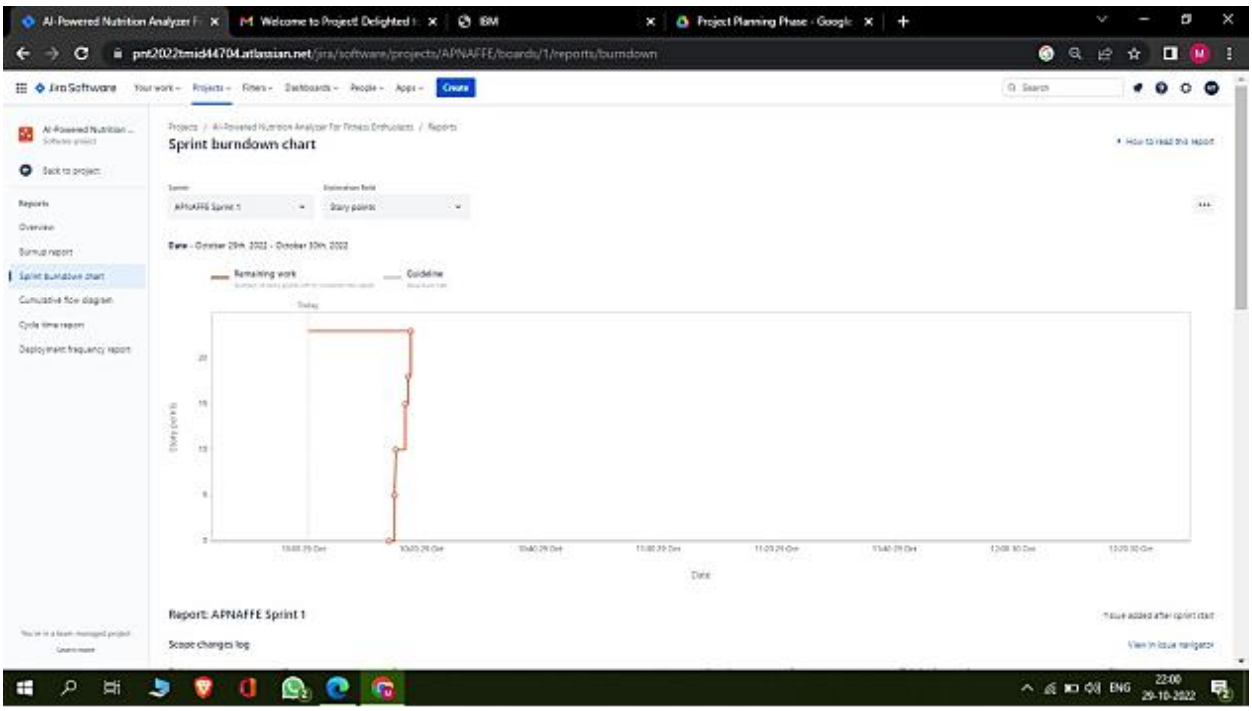
Sprint-4	Fitness tips and basic exercises	US N-13	As a user I can follow some fitness tips and I canmaintain weight as required	5	Medium	Kanimozhi.S
Sprint-4	Home remedies	US N-14	As a user I can follow some natural home remedies for common diseases like (cold, cough,fever) and treat myself	5	High	Gowrishree.B
Sprint-4	Optimize the user experien		As a developer Ihave to	5	High	Kanimozhi.P

	ce with the app		provide cleanand smooth interface to my user			
Sprint-4	Payment Gatew ay for purchasi ng package		As a developer I have to create a environment which makesuser feel easeto complete his/her Payments with various Payment options	3	Medi um	Gowrishree .B

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	29-Oct-2022
Sprint-2	20	6Days	31-Oct-2022	05-Nov-2022	26	05-Nov-2022
Sprint-3	20	6Days	07-Nov-2022	12-Nov-2022	11	12-Nov-2022
Sprint-4	20	6Days	14-Nov-2022	19-Nov-2022	18	18-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 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")

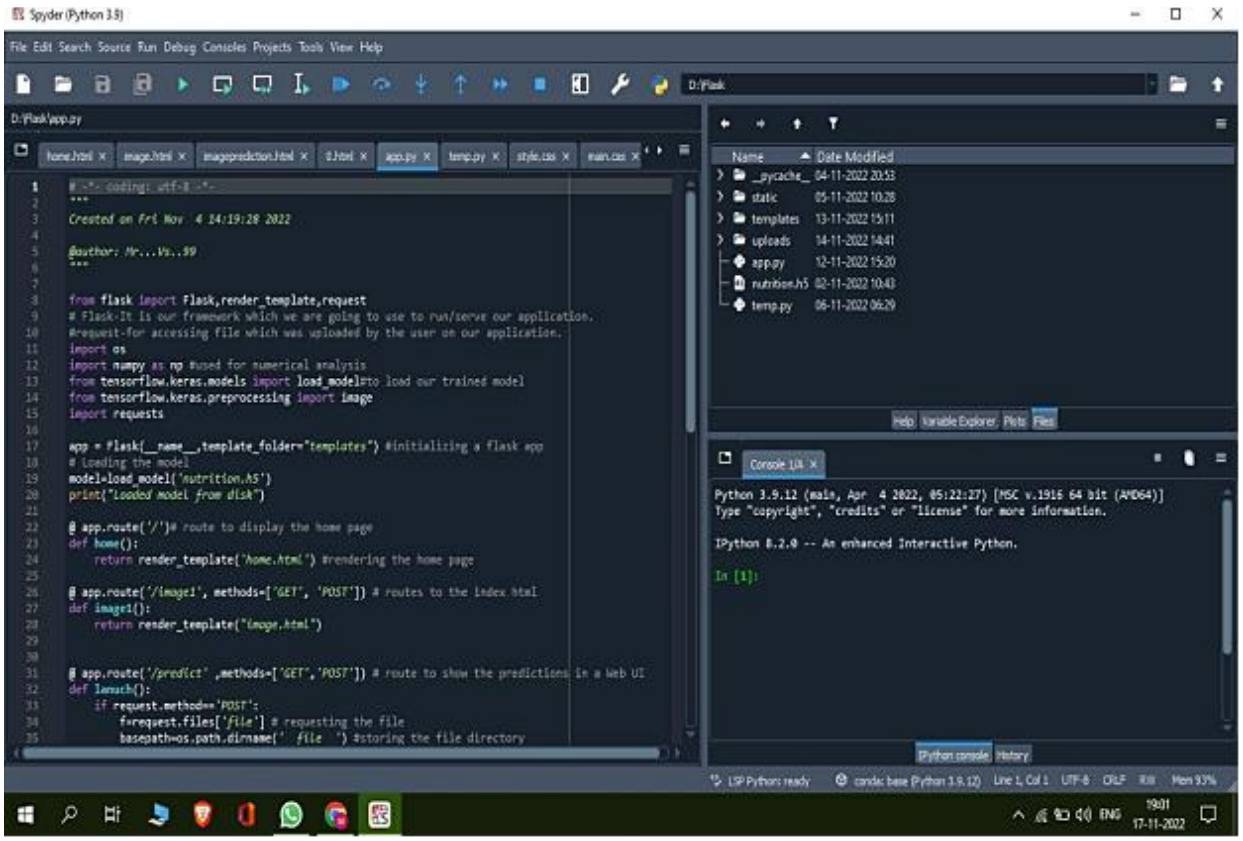
@ app.route('/')# route to display the home pagedef home():

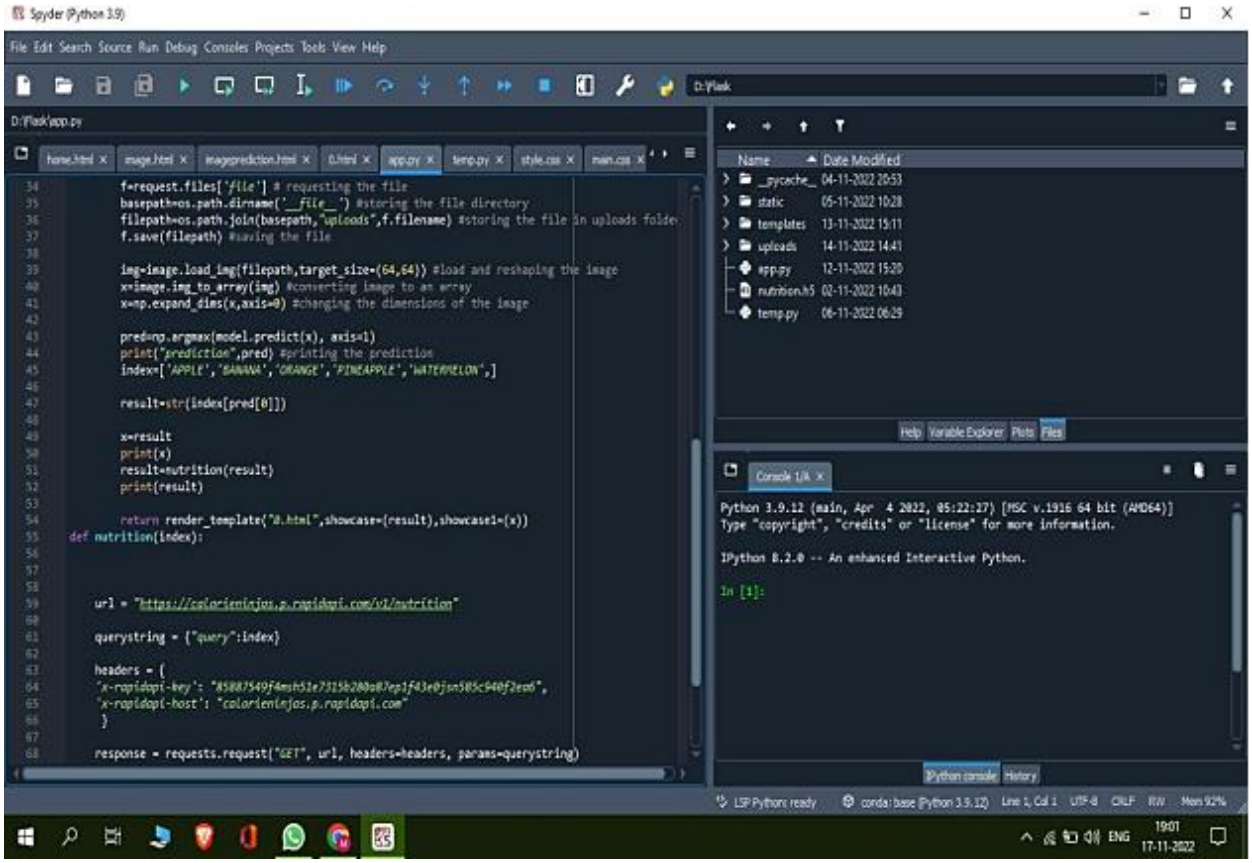
    return render_template('home.html') #rendering the home page

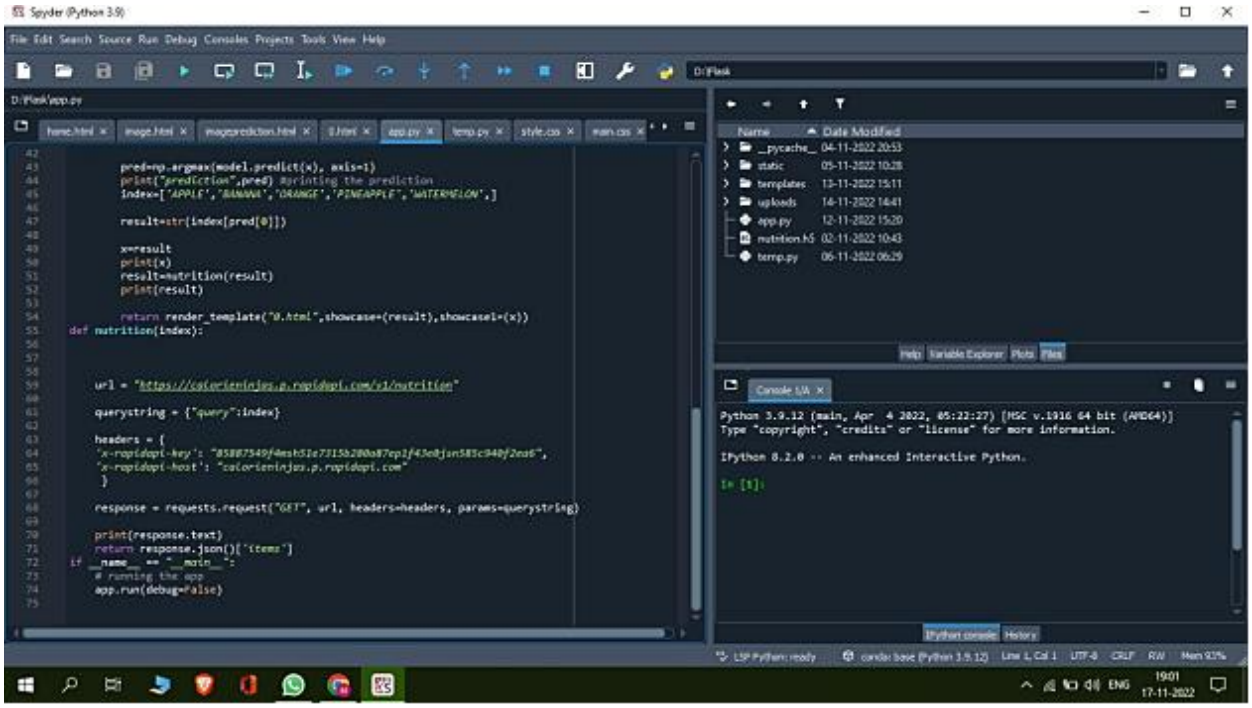
```
@ app.route('/image1', methods=['GET', 'POST']) # routes to the

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"
```

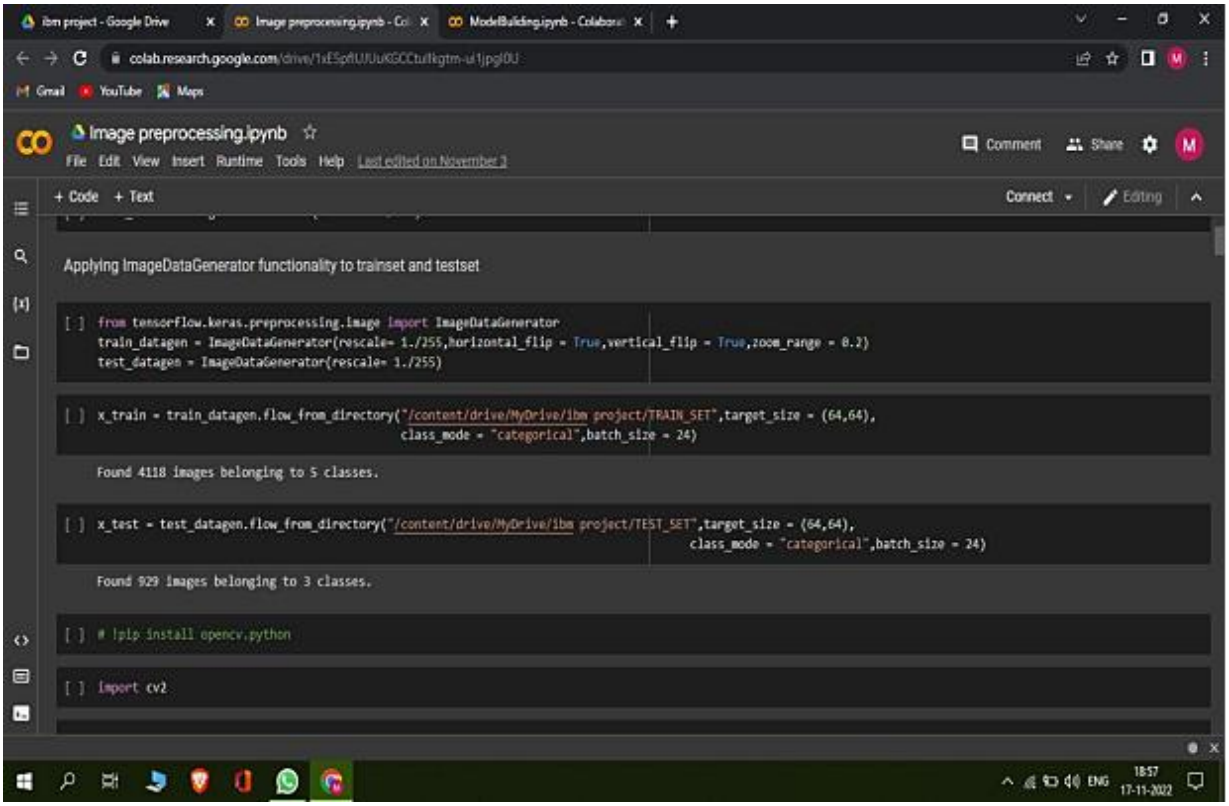
```
    }
    response = requests.request("GET", url, headers=headers,
    params=querystring)



    print(response.text)

    return

response.json()['items']

if _name_ ==

"_main_":

    # running the

    app

    app.run(debu

    g=False)
```

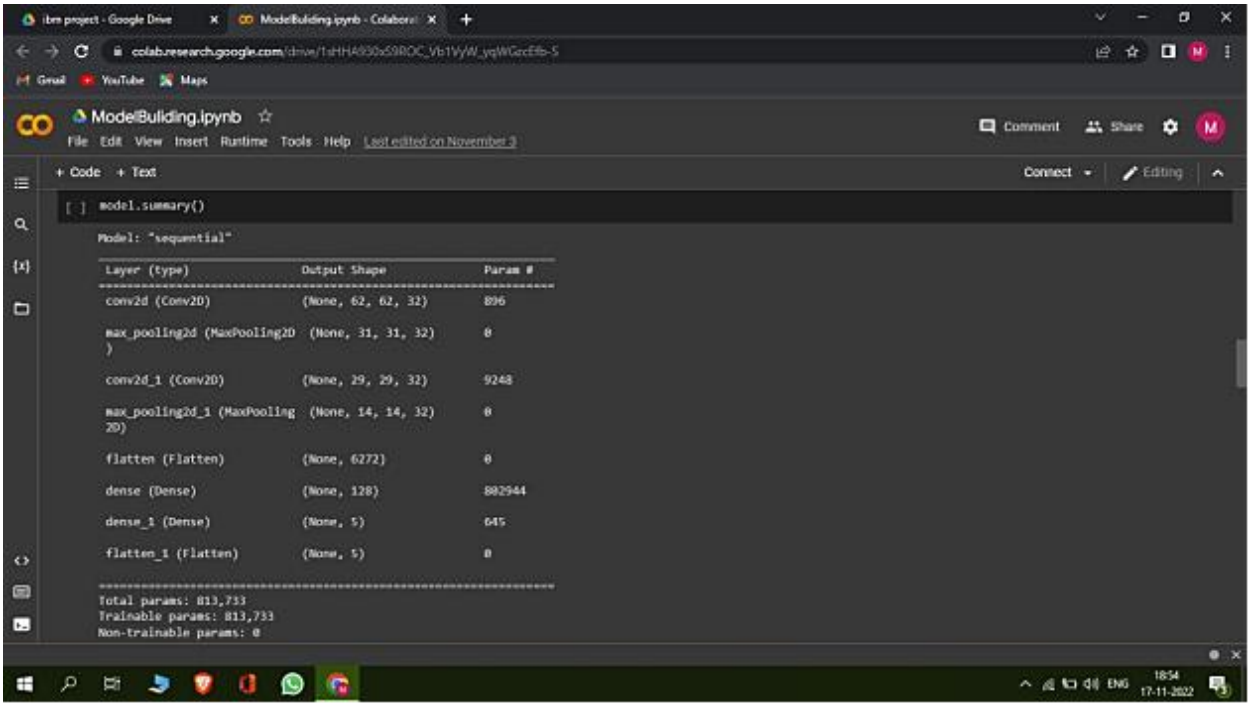

Spyder (Python 3.9)
File Edit Search Source Run Debug Consoles Projects Tools View Help
D:\Flask\app.py
home.html
image.html
imageprediction.html
0.html
app.py
temp.py
style.css
main.css
-*- coding: utf-8 -*-
"""
Created on Fri Nov 4 14:19:28 2022
@author: Mr...Vs..99
"""
from flask import Flask,render_template,request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

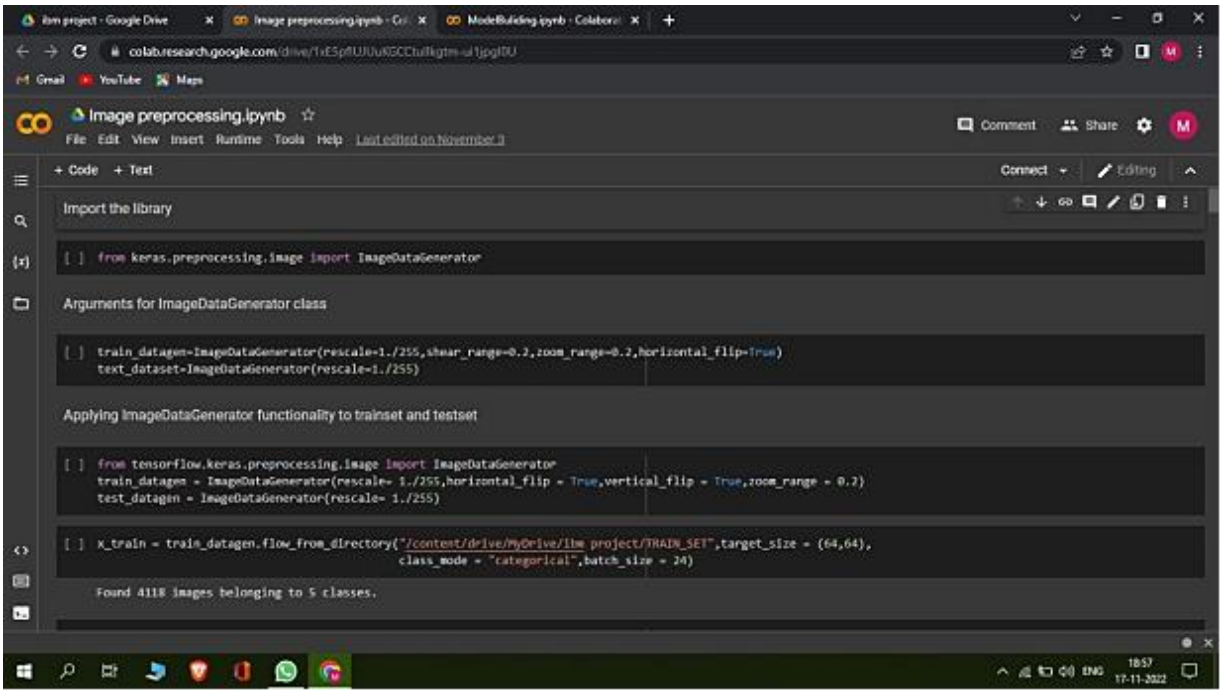

Spyder (Python 3.9)
File Edit Search Source Run Debug Consoles Projects Tools View Help
D:\Flask\app.py
f=request.files['file'] # requesting the file
basepath=os.path.dirname('__file__') #storing the file directory
filepath=os.path.join(basepath,"uploads",f.filename) #storing the file in uploads folder
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': "85087549f4msh52e7315b200a87ep1f43e0jsn585c940f2ea6",
'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

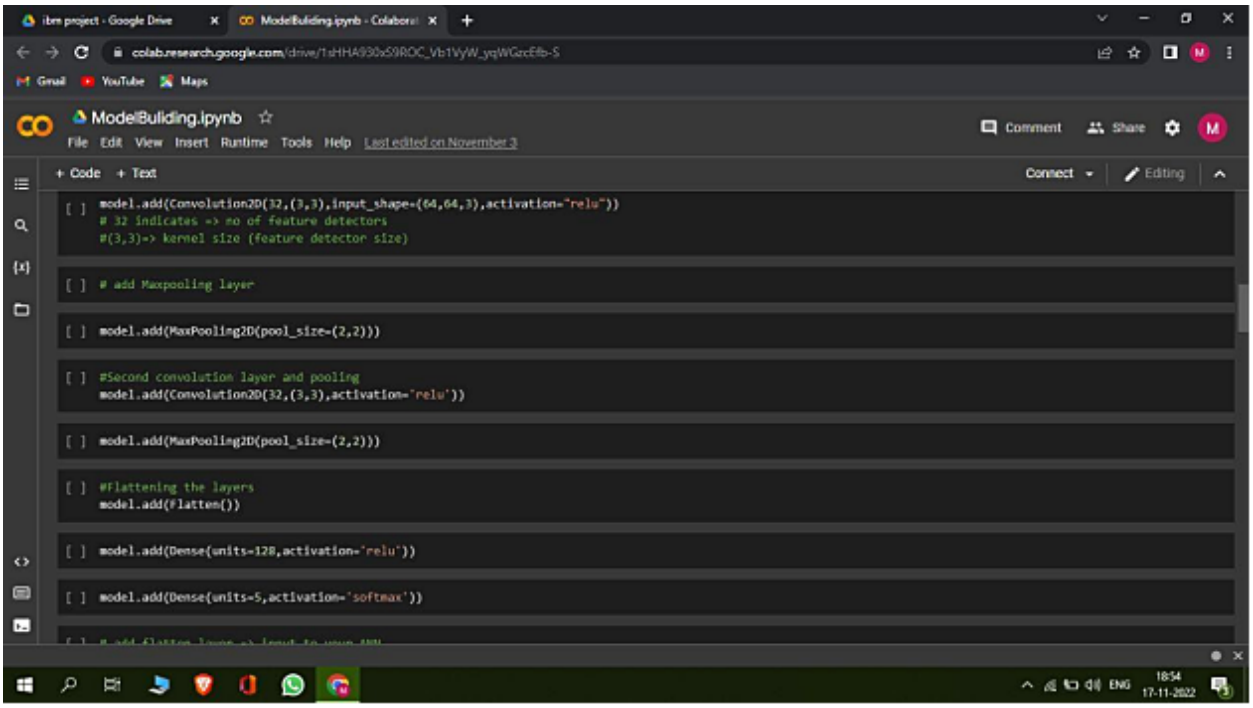

Spyder (Python 3.9)
File Edit Search Source Run Debug Consoles Projects Tools View Help
D:\Flask
D:\Flask\app.py
home.html
image.html
imagepredction.html
0.html
app.py
temp.py
style.css
main.css
pred=np.argmax(model.predict(x), axis=1)
print("prediction",pred) #printing the prediction
index=['APPLE','BANANA','ORANGE','PINEAPPLE','WATERMELON',]
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': "85887549f4eshd1e7315b200a07ep1f43e0jsn583c940f2ea6",
'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 93%
ENG
19:01
17-11-2022

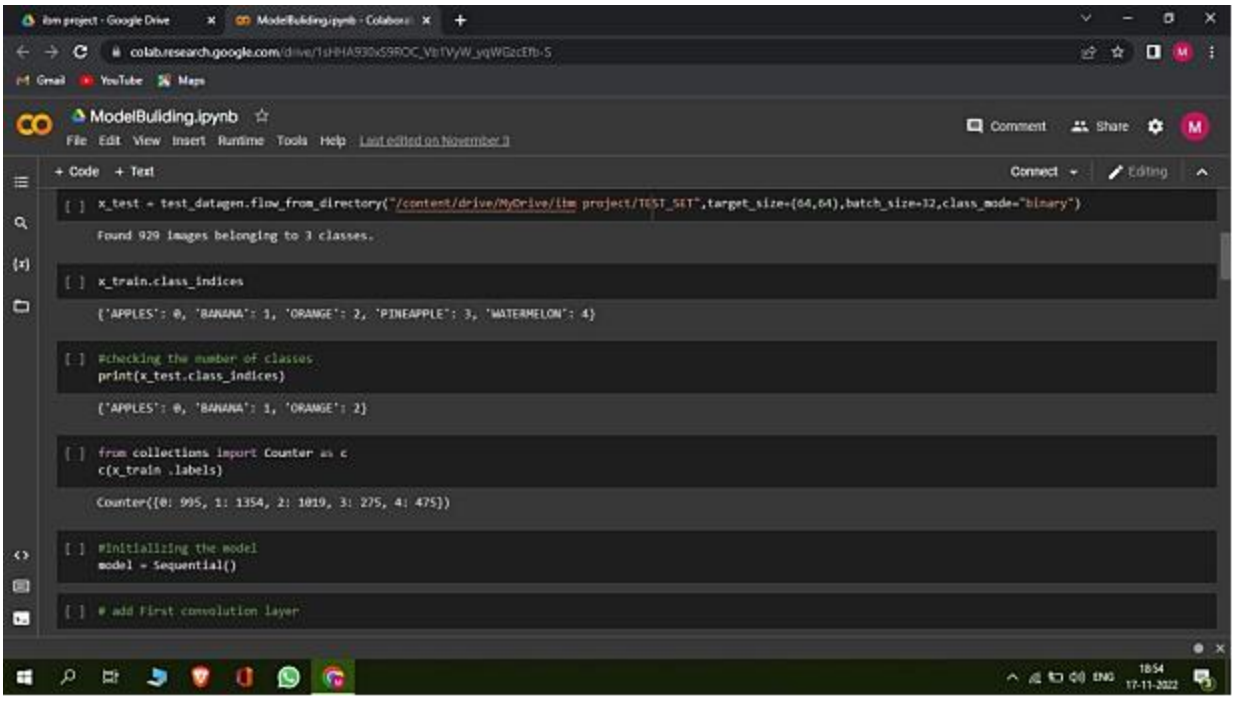
7.2 FEATURE-2

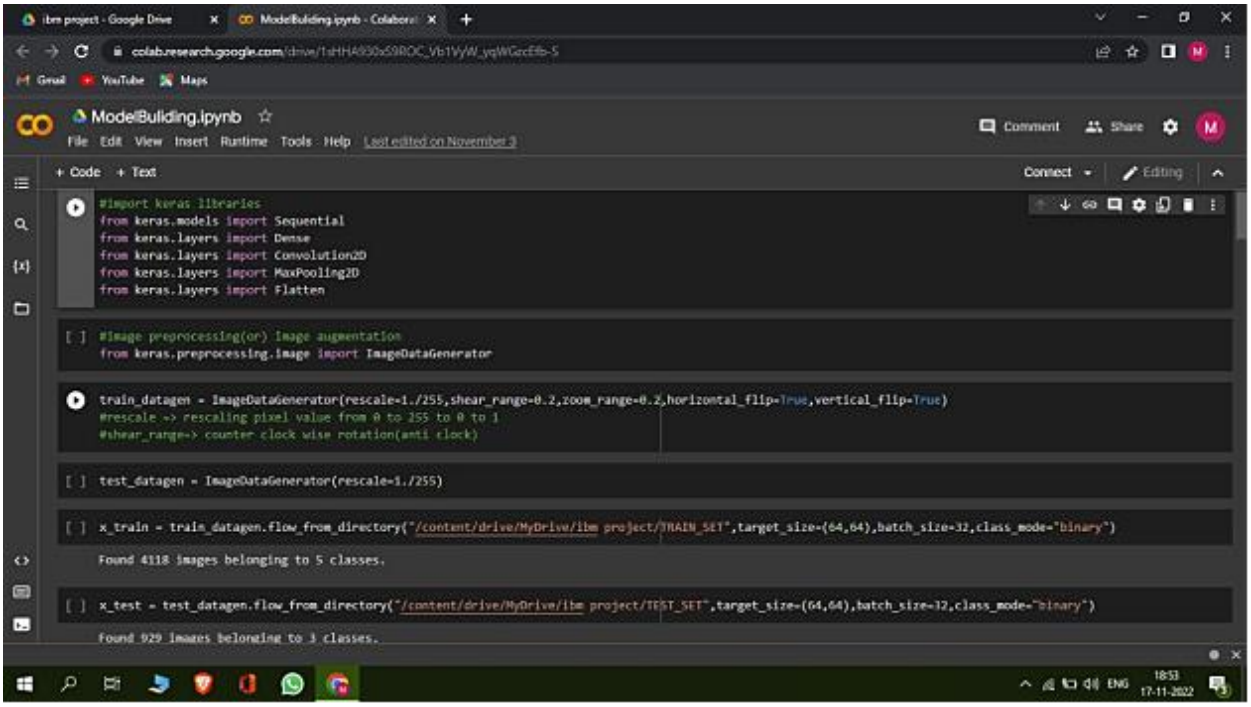

ibm project - Google Drive
Image preprocessing.ipynb - Co
ModelBuilding.ipynb - Colaborat
colab.research.google.com/drive/1xESpfUUuXGCCtuIkgtm-ul1jpgl0U
Gmail
YouTube
Maps
Image preprocessing.ipynb
File Edit View Insert Runtime Tools Help Last edited on November 3
Comment
Share
+ Code + Text
Connect
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


ibm project - Google Drive
ModelBuilding.ipynb - Colabora
colab.research.google.com/drive/1sHHA930o59ROC_Vb1VyW_yqWGzcE8b-S
Gmail
YouTube
Maps
ModelBuilding.ipynb
File Edit View Insert Runtime Tools Help Last edited on November 3
Comment
Share
+ 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
ENG
18:54
17-11-2022


ibm project - Google Drive
Image preprocessing.ipynb - Co
ModelBuilding.ipynb - Colabora
colab.research.google.com/drive/1xESpfIUIUuKGCCtuBkgtm-ul1jpgIDU
Gmail
YouTube
Maps
Image preprocessing.ipynb
File Edit View Insert Runtime Tools Help Last edited on November 3
Comment
Share
+ 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.
ENG
18:57
17-11-2022

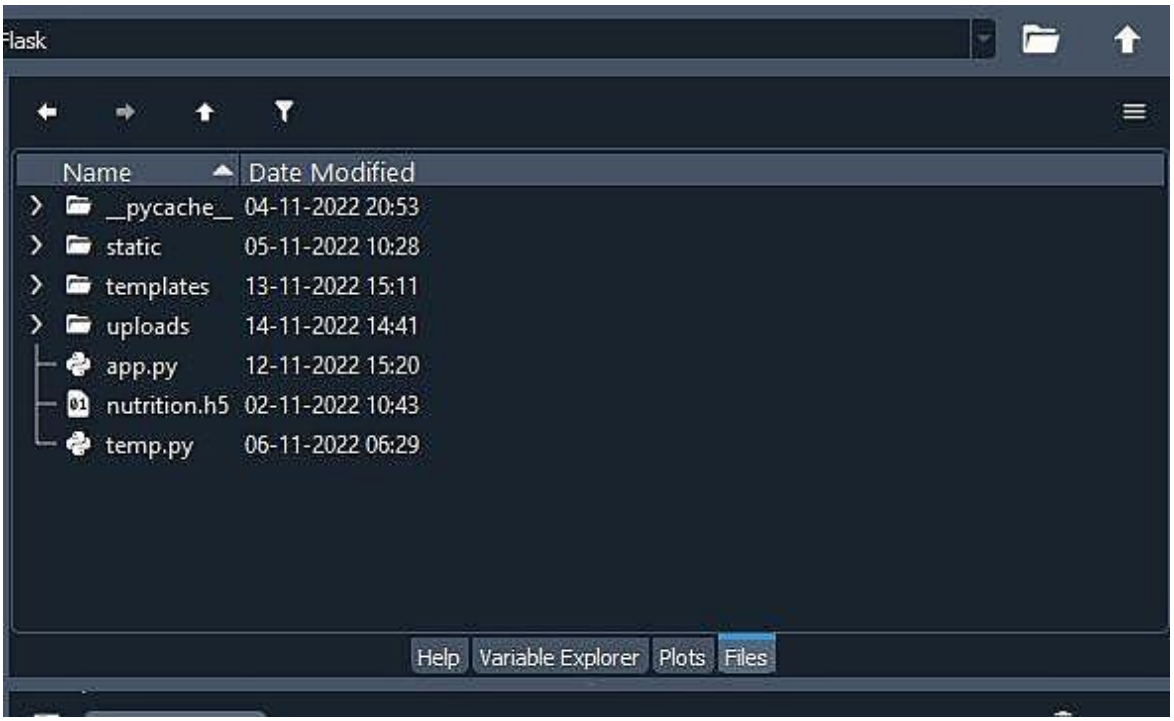

ibm project - Google Drive
ModelBuilding.ipynb - Colaborat
colab.research.google.com/drive/1sHHA930xS9ROC_Vb1VyW_yqWGzcEfb-S
Gmail
YouTube
Maps
ModelBuilding.ipynb
File Edit View Insert Runtime Tools Help Last edited on November 3
Comment
Share
+ Code + Text
Connect
Editing
[ ] model.add(Convolution2D(32,(3,3),input_shape=(64,64,3),activation="relu"))
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
ModelBuilding.ipynb - Colaborat
colab.research.google.com/drive/1sHHA930xS9ROC_Vb1VyW_yqWGzcEfb-S
Gmail
YouTube
Maps
ModelBuilding.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



8. TESTING

8.1 TEST CASE



8.2 USER ACCEPTENCE TESTING

1. PURPOSE OF DOCUMENT

    a. 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 User Acceptance Testing (UAT).

2. DEFECT ANALYSIS

    a. This report shows the number of resolved or closed bugs at each severitylevel, 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

2. TEST CASE ANALYSIS

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

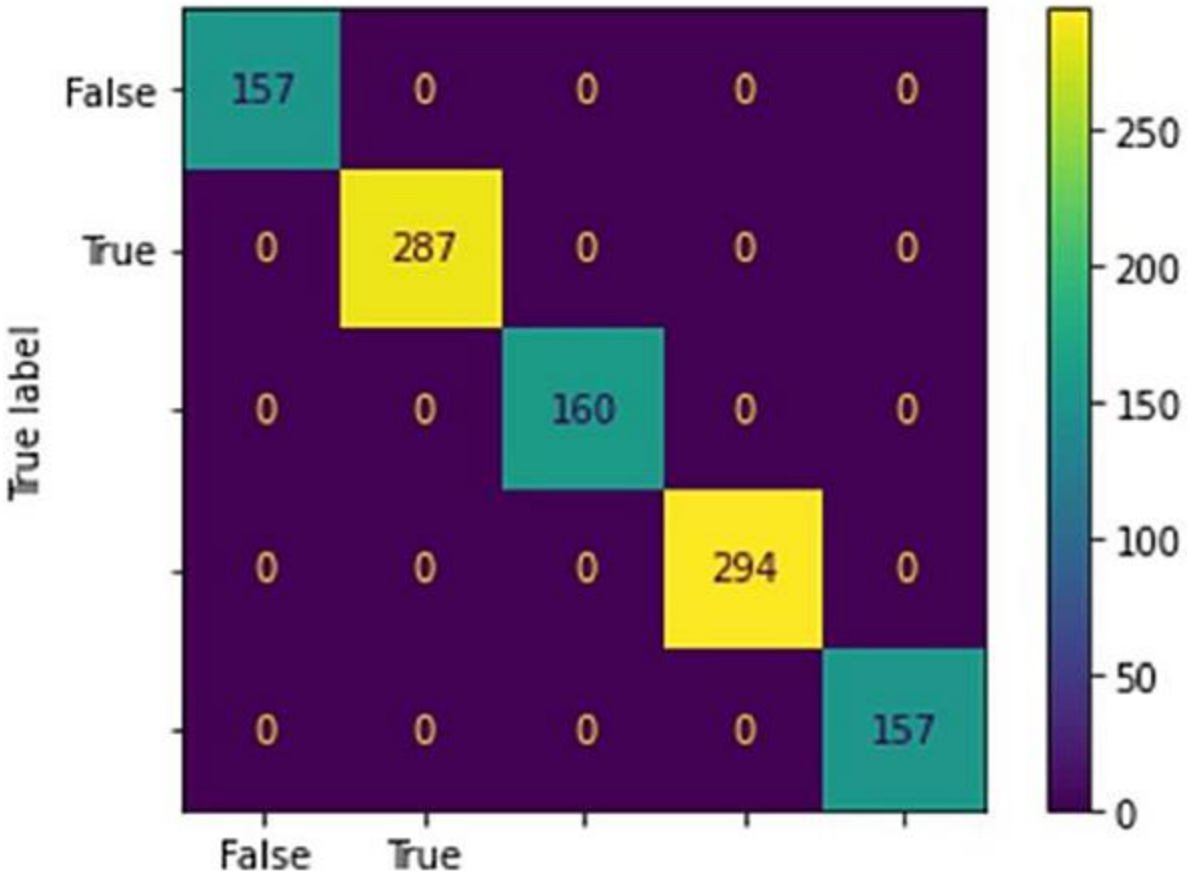
Section	Total Cases	Not Tested	Fail	Pass
Print Engine	5	0	0	5
Client Application	15	0	0	15

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

9.1 PERFORMANCE METRICS

1 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

2 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
```

3 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
```

4 Recall – 100 %

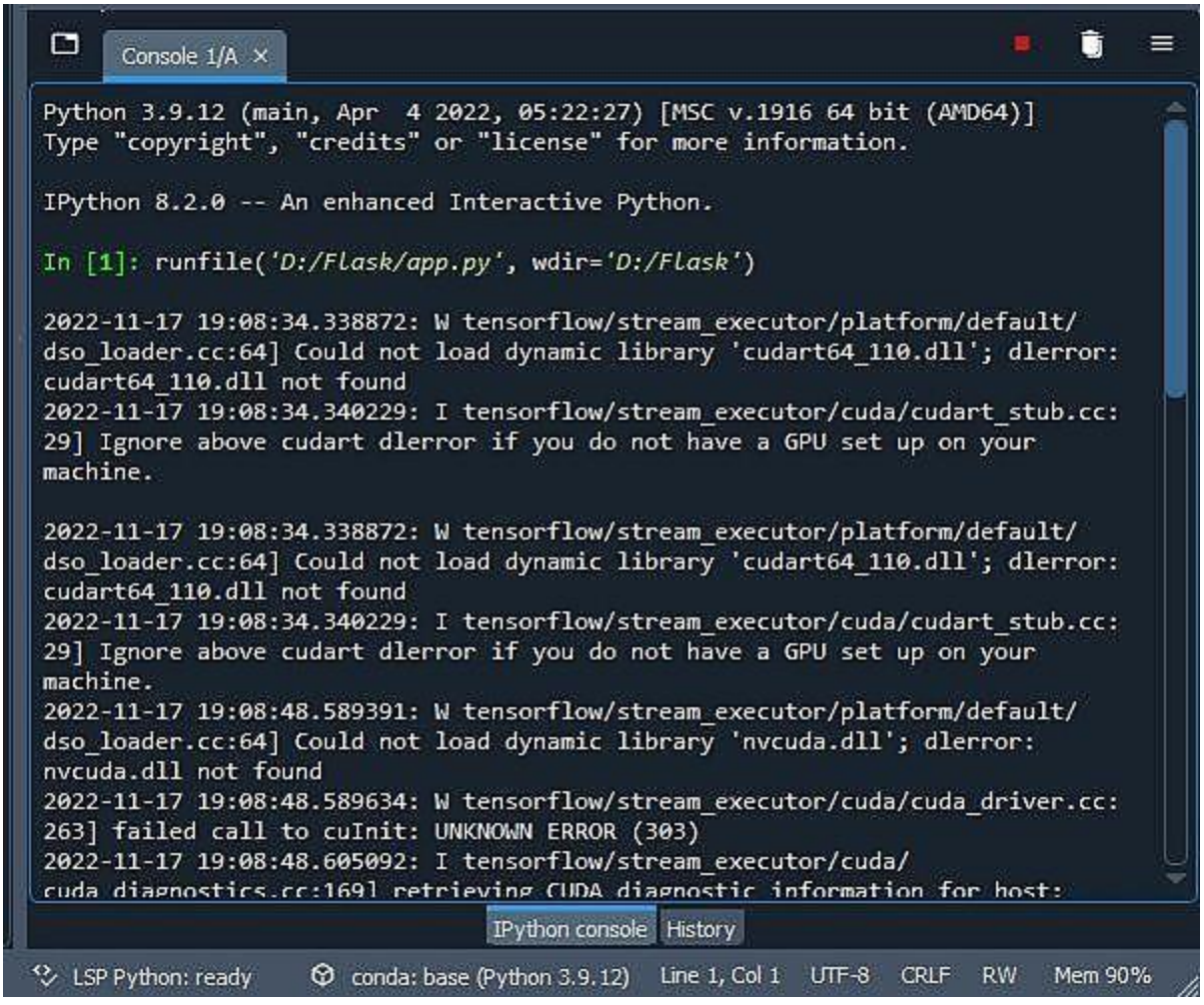
```
[12] print(f"the recall is {metrics.recall_score(test_data['label'].values, test_data['model_preds'].values, average = 'weighted')}")

     the recall is 1.0
```

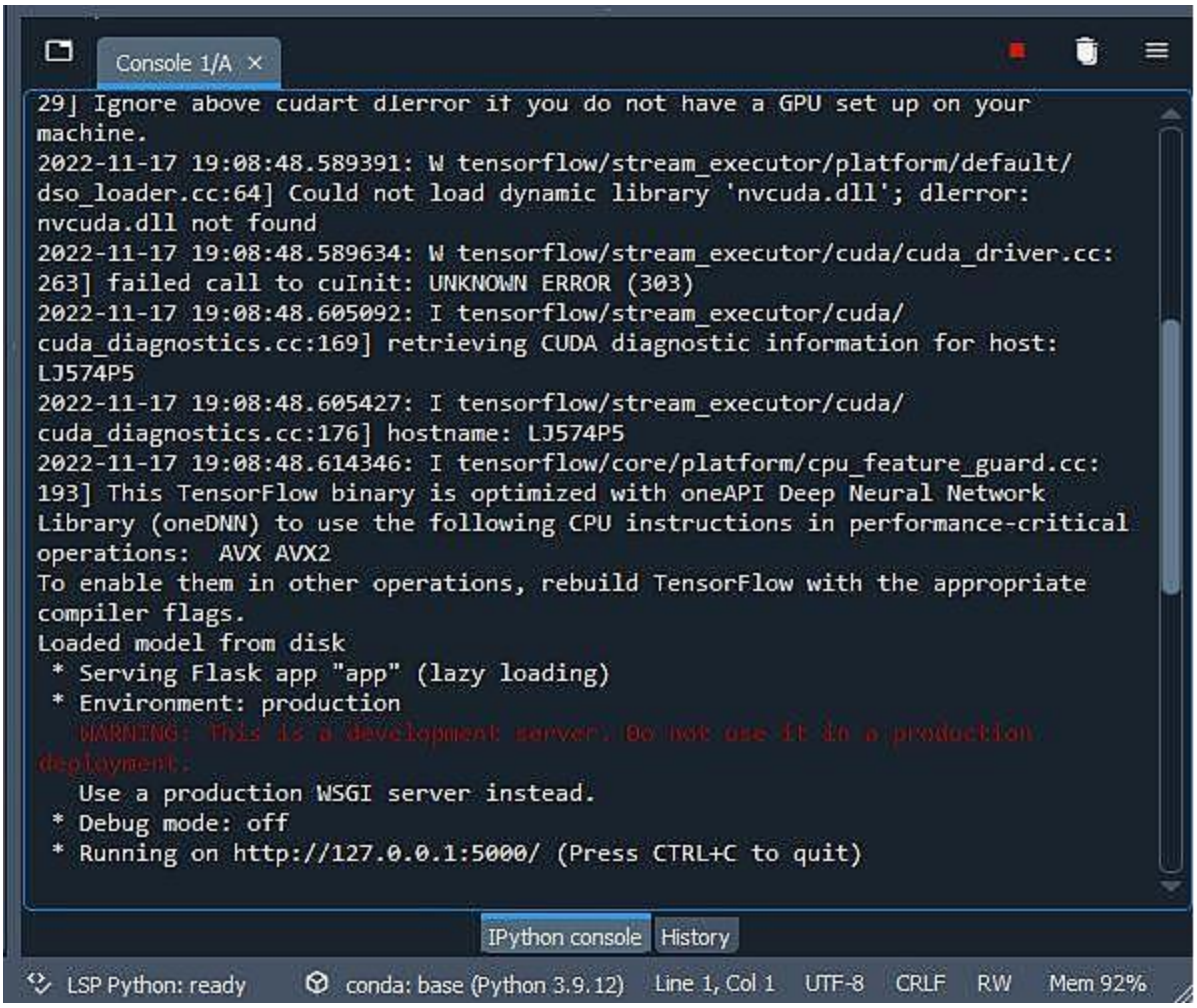
5 Specificity – 100 %

```
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 %

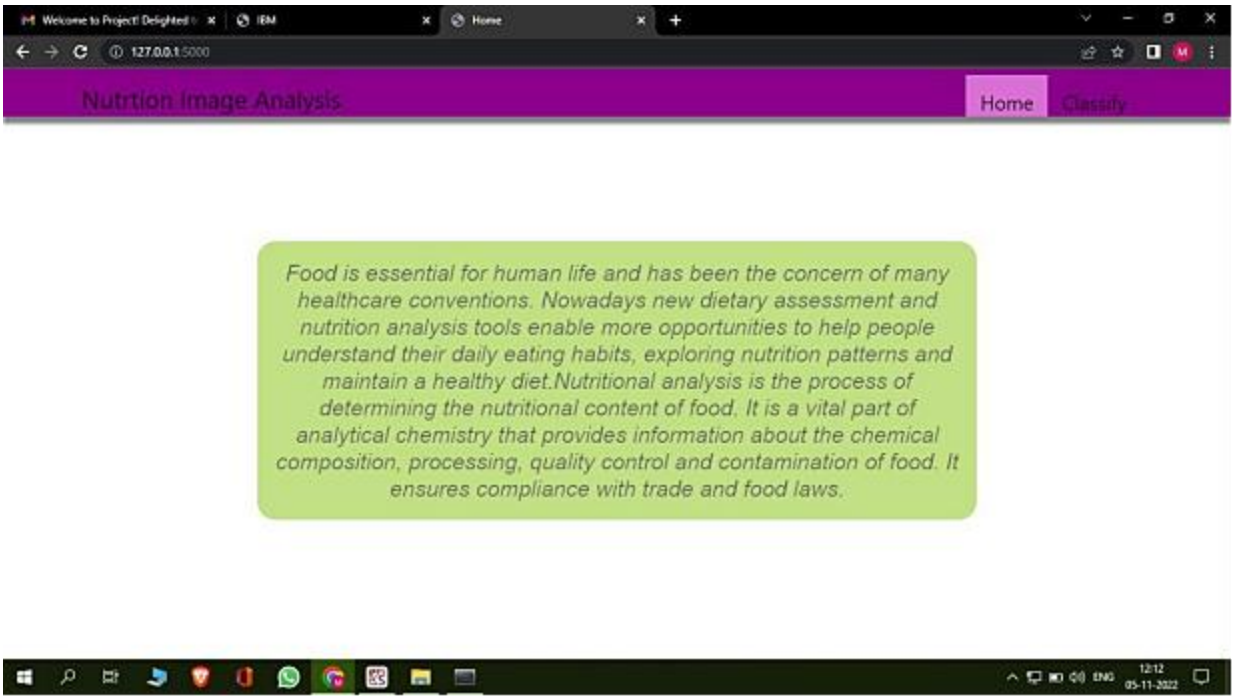
```
[13] print(f"the f1 score is {metrics.f1_score(test_data['label'].values, test_data['model_preds'].values,average = 'weighted')}")

     the f1 score is 1.0
```



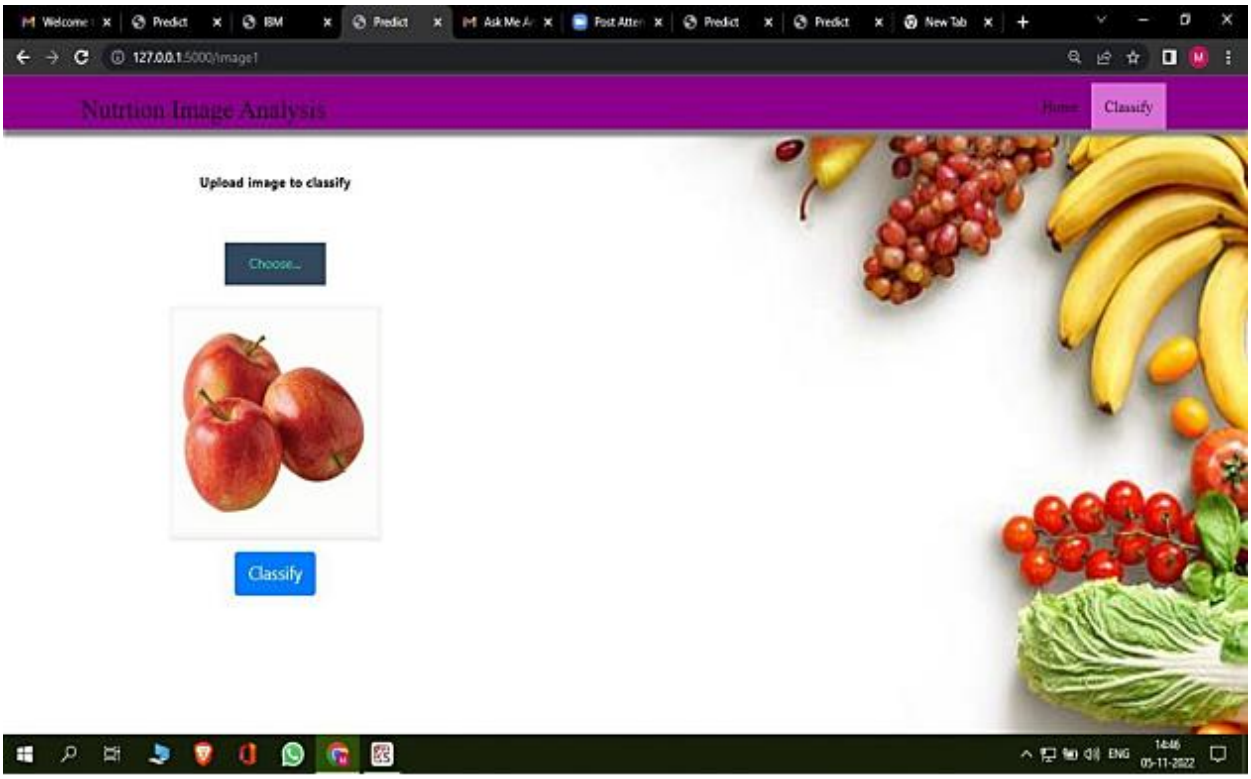


9.2 OUTPUTS

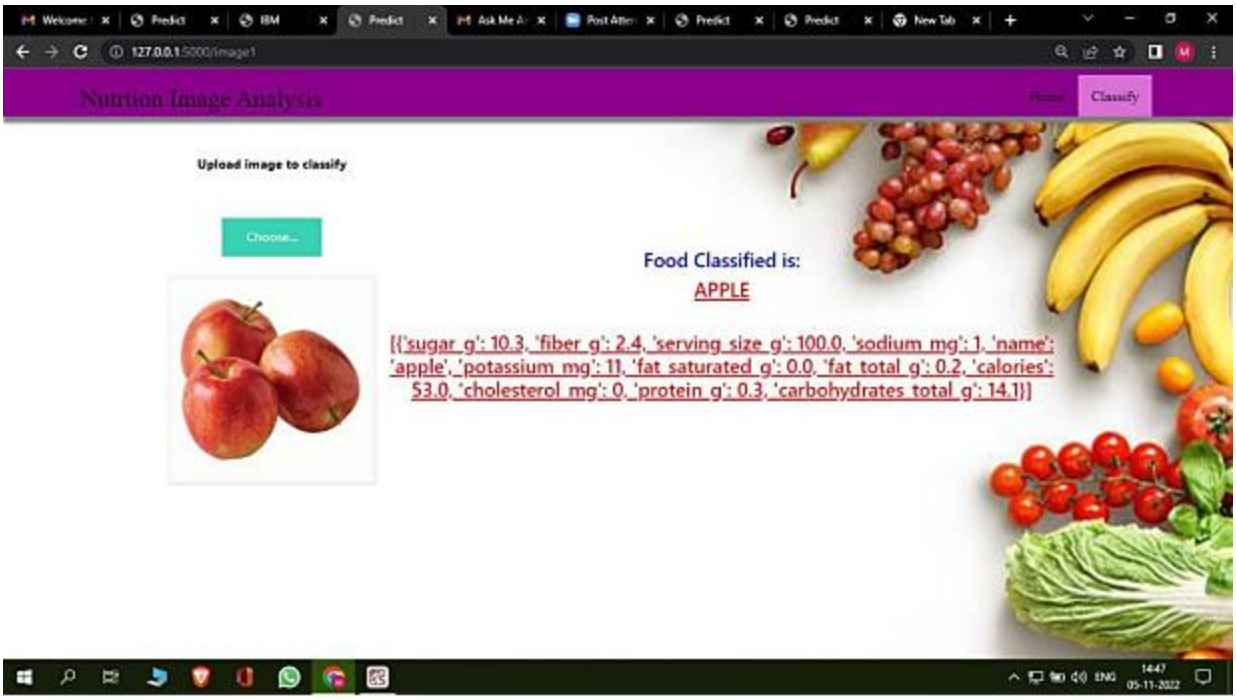
9.2.1 home.html

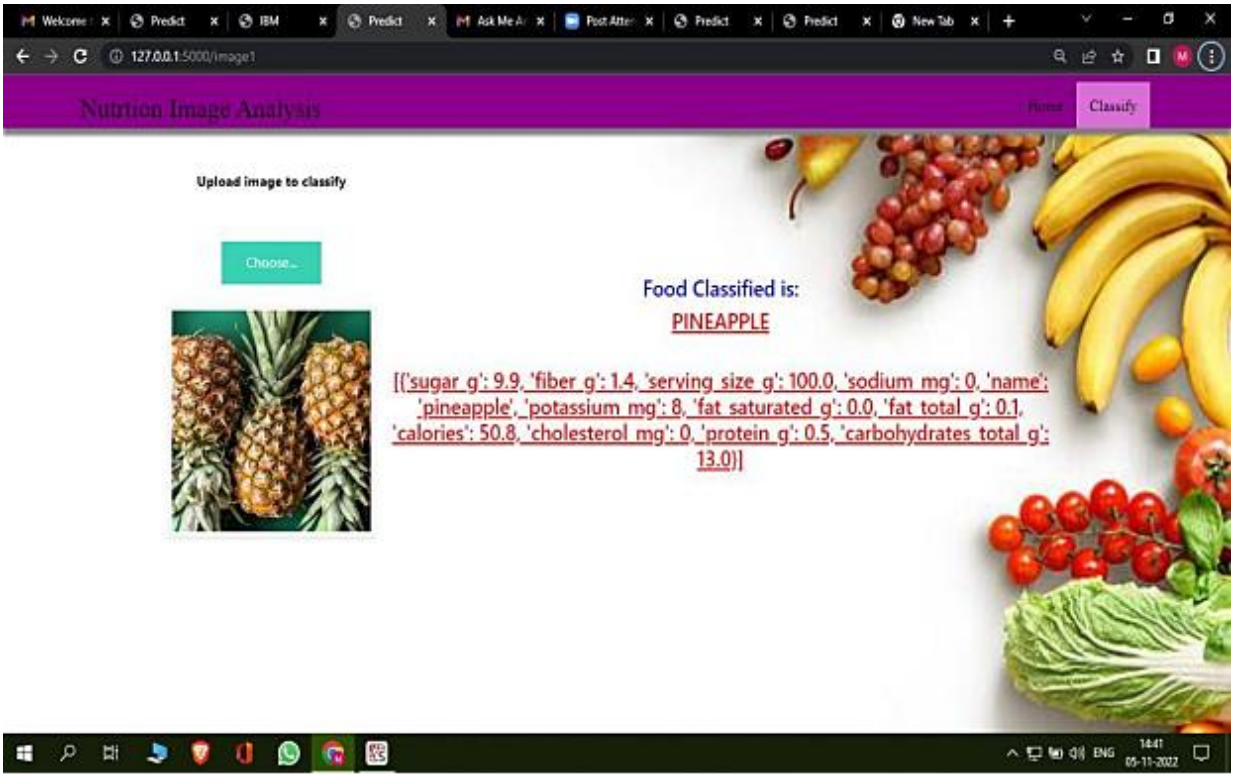


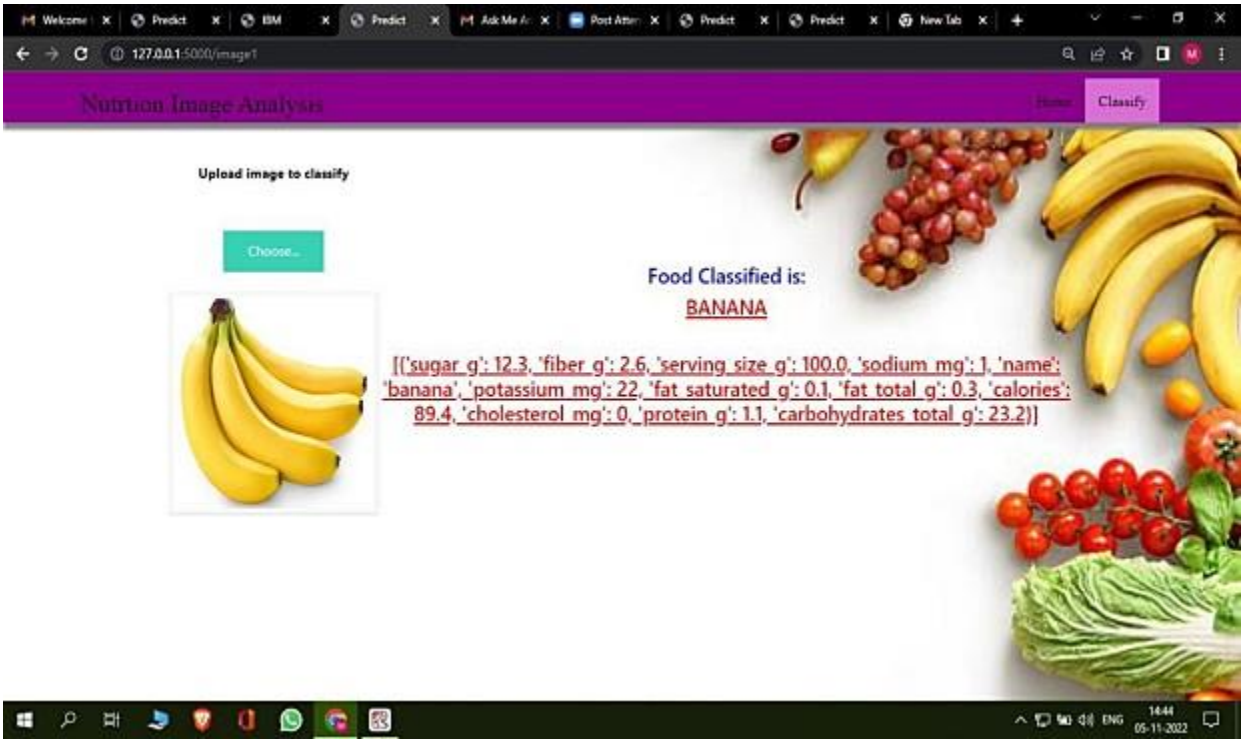
9.2.2 image.html

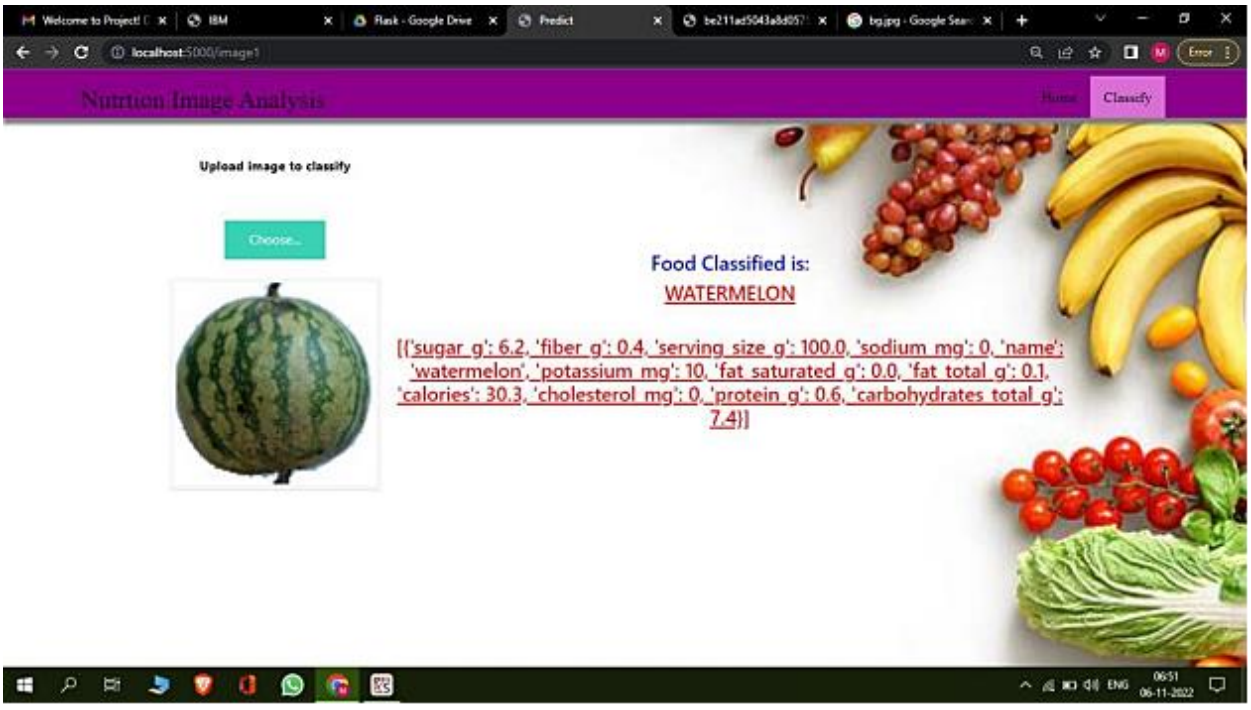


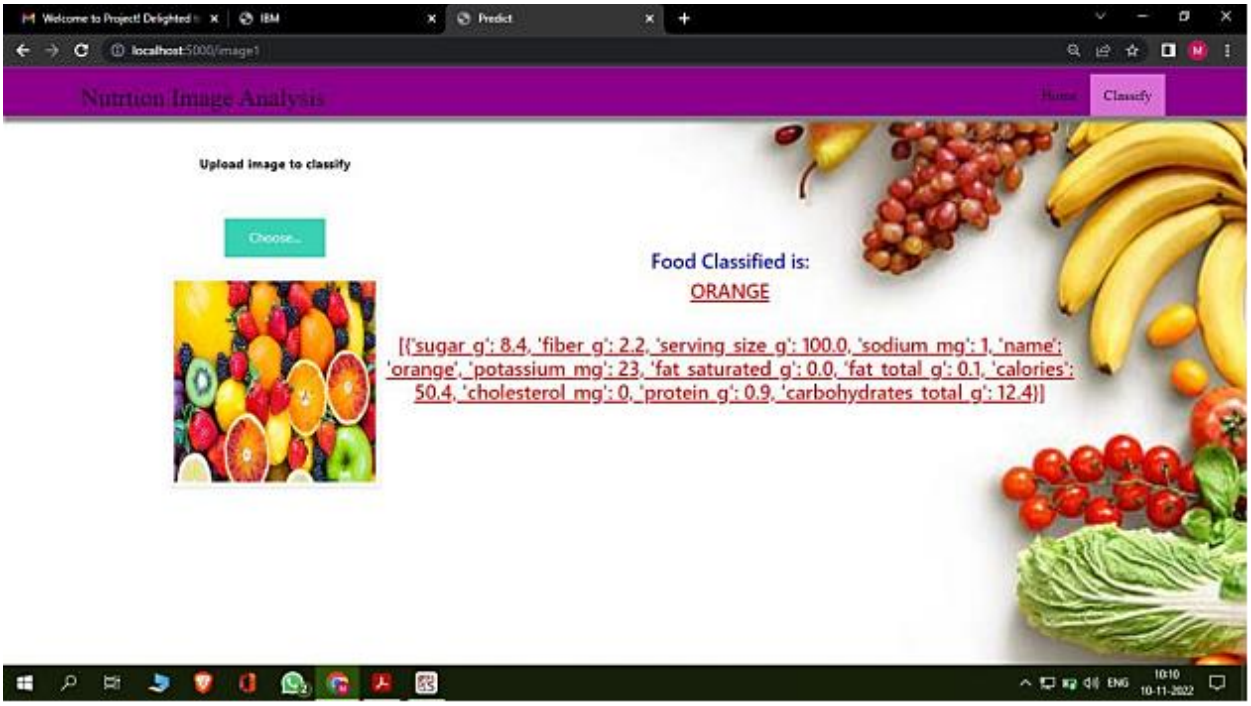
9.2.3 image prediction.html






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

10.1 ADVANTAGES

1. Picture of body identifying benefits of healthy eating for adults.

    a. May help you live longer.
    b. Keeps skin, teeth, and eyes healthy.
    c. Supports muscles.
    d. Boosts immunity.
    e. Strengthens bones.
    f. Lowers risk of heart disease, type 2 diabetes, and some cancers.
    g. Supports healthy pregnancies and breastfeeding.

10.2 DISADVANTAGES

1. These unhealthy eating habits can affect our nutrient intake, including energy (or kilojoules) protein, carbohydrates, essential fatty acids,

vitaminsand minerals as well as fibre and fluid.

a. Being overweight
b. Tooth decay
c. High blood pressure
d. Highcholesterol
e. Heart disease and stroke
f. Type-2 diabetes

11 CONCLUSION

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

12 FUTURE SCOPE

1. Mindful Eating and Food as Medicine:
    a. The distinction between food and supplements blurs as functionalities, such as immune support or gut health, become a priority for consumers.
2. Plant-Based Eating and Alternative Proteins:
    a. Plant-based products accelerated this past year due to the demandfor healthy food options during the pandemic
3. From Farm to Fork: Food Tech, Origins, and Security:

a. Demand for sourcing transparency combined with unprecedented investment in tech is advancing the ability totrace food from production to consumption.

13 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

https://github.com/IBM-EPBL/IBM-Project-15312-1659597033