AI-Powered Nutrition Analyser For Fitness Enthusiasts

1. INTRODUCTION

1. Project Overview

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

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. Purpose

- Know fundamental concepts and techniques of Convolutional Neural Network.
- Gain a broad understanding of image data.
- Knowhow to pre-process/clean the data using different data pre-processing techniques.
- Know how to build a web application using the Flask framework.

2. LITERATURE SURVEY

1. Existing Problem

2.1.1 AUTHOR : Jaroslaw Sak and Magdalena Suchodolska YEAR : 2021

ABSTRACT: Artificial intelligence (AI) as a branch of computer science, the purpose of which is to imitate thought processes, learning abilities and knowledge management, finds more and more applications in experimental and clinical medicine. In recent decades, there has been an expansion of AI applications in biomedical sciences. The possibilities of artificial intelligence in the field of medical diagnostics, risk prediction and support of therapeutic techniques are growing rapidly. The aim of the article is to analyze the current use of AI in nutrients science research. The literature review was conducted in PubMed. AI in biomedical nutrients research (20 studies), AI in clinical nutrients research (22 studies) and AI in nutritional epidemiology (13 studies). It was found that the artificial neural network (ANN) methodology was dominant in the group of research on food composition study and production of nutrients. However, machine learning (ML) algorithms were widely used in studies on the influence of nutrients on the functioning of the human body in health and disease and in studies on the gut microbiota. Deep learning (DL) algorithms prevailed in a group of research works on clinical nutrients intake. The development of dietary systems using AI technology may lead to the creation of a global network.

2.1.2

AUTHOR: Chang Liu, Yu Cao, Senior Member, IEEE, Yan Luo, Member, IEEE

YEAR: 2021

ABSTRACT: Literature has indicated that accurate dietary assessment is very important for assessing the effectiveness of weight loss interventions. However, most of the existing dietary assessment methods rely on memory. With the help of pervasive mobile devices and rich cloud services, it is now possible to develop new computer-aided food recognition system for accurate dietary assessment. However, enabling this future Internet of Things-based dietary assessment imposes several fundamental challenges on algorithm development and system design. In this paper, we set to address these issues from the following two aspects: (1) to develop novel deep learning-based visual food recognition algorithms to achieve the best-in-class recognition accuracy; (2) to design a food recognition system employing edge computing-based service computing paradigm to overcome some inherent problems of traditional mobile cloud computing paradigm, such as unacceptable system latency and low battery life of mobile devices. We have conducted extensive experiments with real-world data. Our results have shown that the proposed system achieved three objectives: (1) outperforming existing work in terms of food recognition accuracy; (2) reducing response time that is equivalent to the minimum of the existing approaches; and (3) lowering energy consumption which is close to the minimum of the state-of-the-art.

2.1.3

AUHTOR: Rijwan Khan, Santosh Kumar

In order to prevent foodborne illness and harm, food must be properly prepared, transported, and stored. Food products may encounter a variety of health risks from farm to factory and factory to fork. Food safety is therefore essential from a financial and moral standpoint. The consequences of not complying with food safety regulations are diverse.)e demand for precise, rapid, and neutral quality assessments of these qualities in food products is increasing as dietary requirements and high-quality standards are demanded more frequently. To accomplish these goals, computer vision offers an automated, non-destructive, and cost-effective method. Its usefulness for fruit and vegetable assessment and classification has been proven by a significant body of research. It highlights the key elements of image processing technology and provides an overview of the most recent developments in the food industry. Public health is consistently and significantly burdened by foodborne illnesses. After more than a century Large-scale changes in food production, distribution, and regulations were pushed and fed into macrosocial pressures like population growth, urbanisation, and globalisation . Compared to other economic sectors, the food industry and distribution network, in particular, have created huge amounts of data in recent years. To increase the safety of the food supply, several types of data were imaginatively examined at various points along the agricultural value chain . For instance, toxic contaminations on farmlands were forecasted in preharvest, field, and weather forecasts; in the retail setting, contactless audits and record-keeping were carried out for 1.4 million months; and observations of Hindawi Journal of Food Quality Volume

2.1.4

AUTHOR: Eileen R Gibney YEAR: 2019

ABSTRACT :Dietary assessment methods are important tools for nutrition research. Online dietary assessment tools have the potential to become invaluable methods of assessing dietary intake because, compared with traditional methods, they have many advantages including the automatic storage of input data and the immediate generation of nutritional outputs.The aim of this study was to develop an online food frequency questionnaire (FFQ) for dietary data collection in the "Food4Me" study and to compare this with the validated European Prospective Investigation of Cancer (EPIC) Norfolk printed FFQ.The Food4Me FFQ used in this analysis was developed to consist of 157 food items. Standardized color photographs were incorporated in the development of the Food4Me FFQ to facilitate accurate quantification of the portion size of each food item. Participants were recruited in two centers and each received the online Food4Me FFQ and the printed EPIC-Norfolk FFQ in random order. Participants completed the Food4Me FFO online and, for most food items, participants were requested to choose their usual serving size among seven possibilities from a range of portion size pictures. The level of agreement between the two methods was evaluated for both nutrient and food group intakes using the Bland and Altman method and classification into quartiles of daily intake. Correlations were calculated for nutrient and food group intakes.

2.1.5

AUTHOR: Raza Unus,Omar Arif YEAR: 2017

ABSTRACT:There has been a rapid increase in dietary ailments during last few decades, caused by unhealthy food routine. Mobile-based dietary assessment systems that can record real time images of meal and analyze it for nutritional content can be very handy and improve the dietary habits, and therefore, result in healthy lite. This paper proposes a novel system to automatically estimate food attributes such as ingredients and nutritional value by classifying the input image of food. Our method employs different deep learning models for accurate food identification. In addition to image analysis, attributes and ingredients are estimated by extracting semantically related words from a huge corpus of text, collected over the Internet.We performed experiments with a dataset comprising 100 classes, averaging 1000 images for each class to acquire top 1 classification rate of up to 85 percent. An extension of a benchmark dataset Food-101 is also created to include sub-continental foods. Results show that our proposed system is equally efficient on basic Food-101 dataset and its extension for sub-continental foods. The proposed system is implemented

2. REFERENCES

1. Nareen O. M. Salim, Study for Food Recognition System Using Deep Learning,2021
2. Yao Liu , Hongbin Pu, Efficient extraction of deep image features using convolutional neural network (CNN) for applications in detecting and analysing complex food matrices, 2021
3. Rijwan Khan, Santosh Kumar, The Use of Different Image Recognition Techniques in Food Safety: A Study, 2021
4. Laura Maria König, Barriers to and Facilitators for Using Nutrition Apps: Systematic Review and Conceptual Framework, 2021
5. Dhanamjayulu C, Identification of malnutrition and prediction of BMI from facial images using real-time image processing and machine learning, 2021

2.3 PROBLEM STATEMENT DEFINITION

Food is a necessity for human life and has been addressed in numerous medical conventions. Modern dietary evaluation and nutrition analysis technologies give consumers more possibilities to explore nutrition patterns, comprehend their daily eating habits, and keep up a balanced diet.

The biggest challenge for fitness lovers is keeping track of their daily nutrition intake, which is crucial for staying in shape. But with today's busy world and the abundance of internet fitness resources, keeping track of your nutrition will become increasingly difficult and inaccurate. Fitness fanatics typically stick to their diet programmers, but they have trouble keeping track of the food's nutritional value.

Fruits are easily digestible since they are high in vitamins, fiber, and minerals, but eating too much of them can cause weight gain and even diabetes because fruit contains natural sugar.

Fitness aficionados eat a diet high in fruits, vegetables, foods high in protein, and low in carbohydrates. However, it is difficult to identify and keep track of the nutritional components of unknown foods, such as fiber, protein, and nutrition.

I am (USER)

User has to upload the food (fruits and vegetables) image to know the healthy content.

I am Trying To

Instead of waiting for a diet expert, users may acquire dietary specifics through this application.

But

This might be the result of a human error, such as a lack of quality control, poor customer service, or even a lack to provide healthy suggestions.

Because

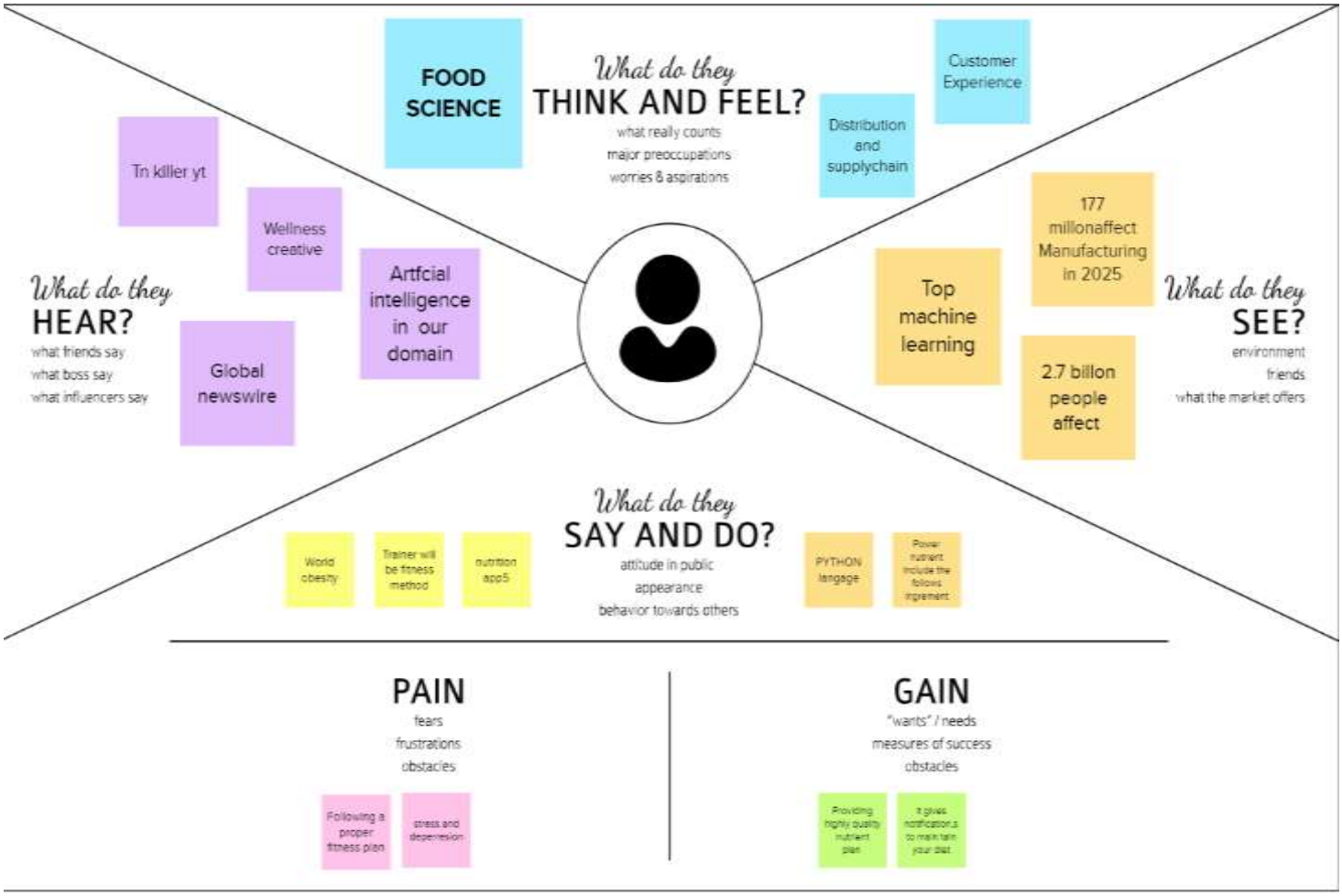
It is hard, and there is a delay to know about the food details and also awkward for providing our healthy facts.

Which makes me feel?

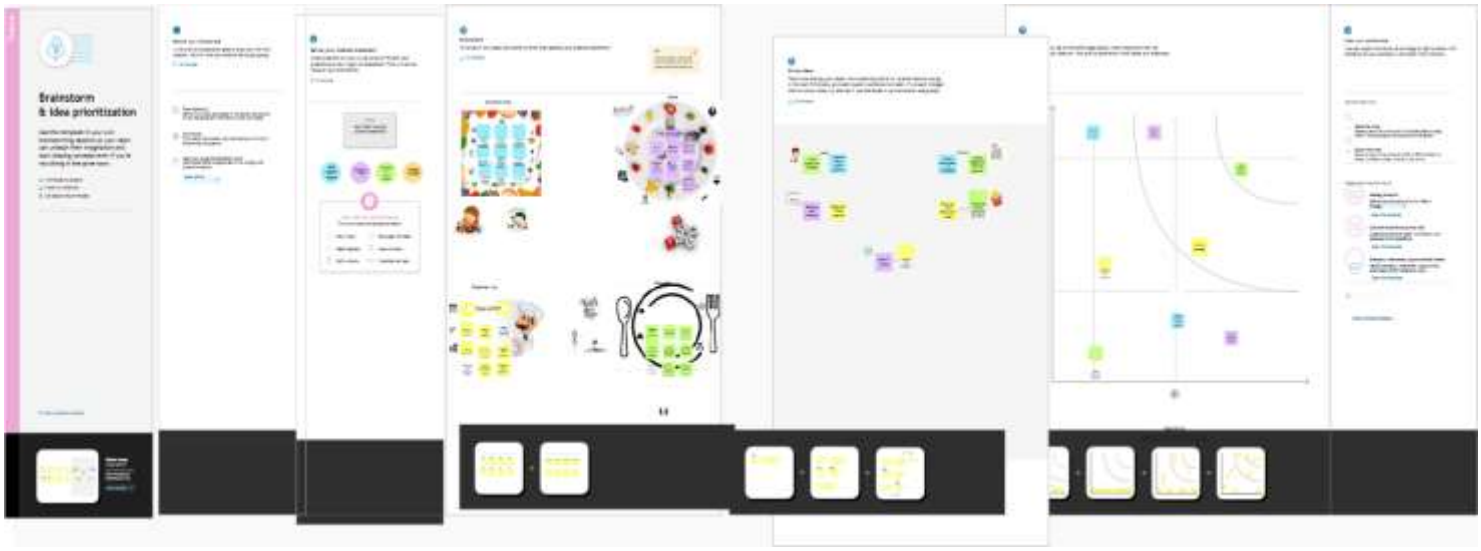
Deep learning algorithms may assist to address these challenges by automating nutrition content assessment. Finally, by analysing the nutritional components in the images, compute the calories, fat, carbs, and protein amounts to give a dietary evaluation report. The addition of more food kinds to the dataset will increase the system's efficiency and precision.

3. IDEATION & PROPOSED SOLUTION

1. Empathy Map Canvas



3.2 Ideation &Brainstorming



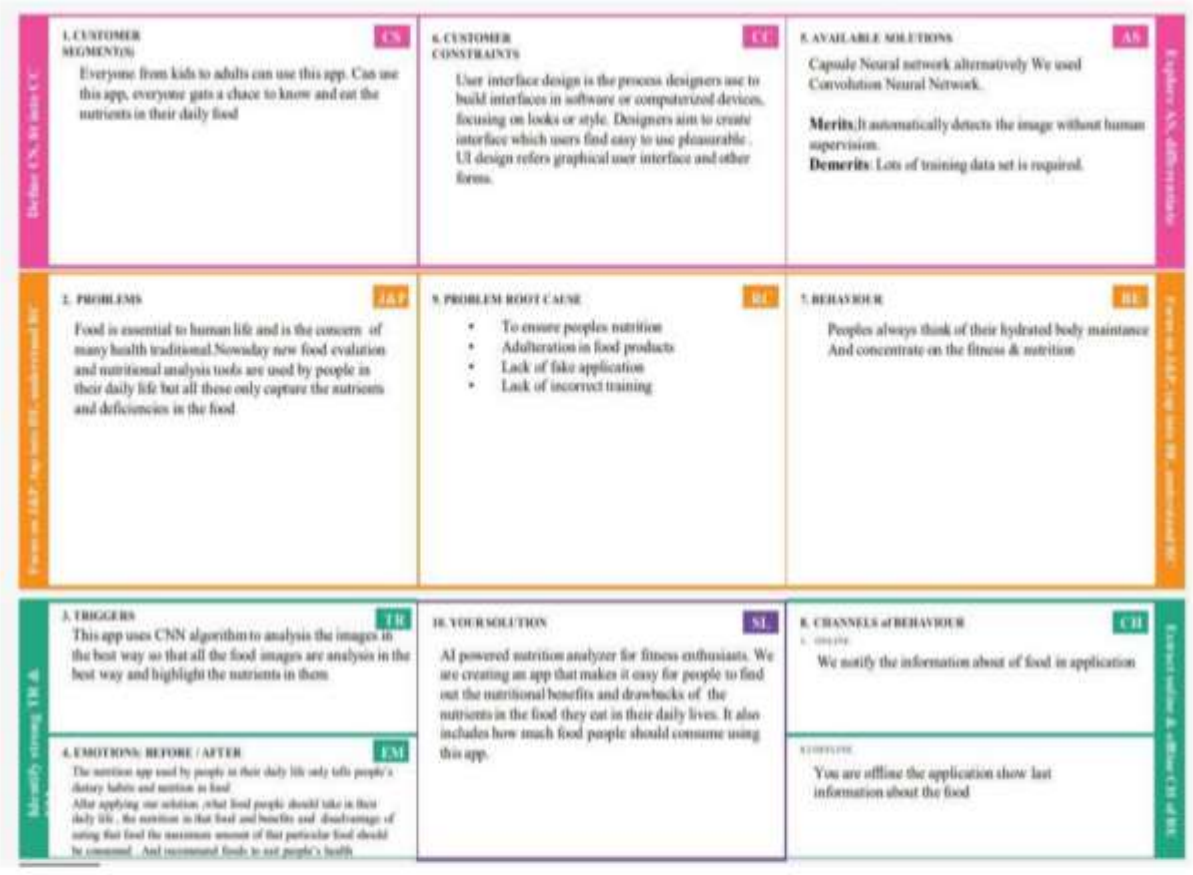
3.3 Proposed Solution

To recognise and find food items from the given photographs, develop an automated nutrition analysis system for the proposed system. By locating promising locations and classifying them with deep neural networks, you may create a three-step process only for detecting various cuisines in photographs. From the provided photos, the automatic algorithm initially generates a large number of suggestion regions. Then, it aggregates each region of ideas by placing them on feature maps, categorising them into different food groups, and identifying their locations in the original photos. Finally, by analysing the nutritional elements in the photographs, determine the quantities of calories, fat, carbs, and protein to generate a dietary evaluation report. The system's effectiveness and accuracy will also be increased by expanding the dataset to cover a larger variety of food kinds.

S. No.	Parameter	Description
1.	Problem Statement (Problem to be solved)	Peoples doesn’t aware about foods and its Nutrition facts.
2.	Idea / Solution description	Solving problem through CNN

3.	Novelty / Uniqueness	It helps user to get nutrition facts with the suggestion of the scanned food is suitable for user or not. The suggestion is based on user's data and BMI value.
4.	Social Impact / Customer Satisfaction	Users are aware of their food habits by getting a food facts.
5.	Business Model (Revenue Model)	By showing a supplements and food advertisements we can earn money through the project.
6.	Scalability of the Solution	Its plays a vital role in users life because it shows and give a suggestions to user so they can control their food habits and maintain their health and fitness.

3.4 Problem Solution Fit



4. REQUIREMENTANALYSIS

1. Functional Requirement

Functional Requirements

Upload Image

In this module, upload the nutrition datasets in the form of CSV file format. In addition, the data is saved in a database for future use. Fruits and vegetables calorie, protein, fat, carbohydrate, vitamin, and cholesterol values are included in the dataset. These values are taken from the Kaggle website and saved as integer values.

Filtering Noise

Filter techniques are used to remove noise in images in order to evaluate nutrients based on the fruits or vegetables. The filter's objective is to remove noise from photos. It is supported by a statistical methodology. The usual frequency response of a filter is built. Filtering is a nonlinear image processing technique used to minimise "salt and pepper" noise. When edge preservation and noise reduction are concerns, a median filter is superior to convolution.

Classification

The food image uploaded from the user end will be compared with the food items in the system database for the features obtained in the feature extraction step. The specific food item will be recognised when the perfect match is obtained based on the attributes matched. The name of the detected food item and the nutrition details will be displayed over the food.

Nutrition Detection

The request for an insurance claim can be viewed and approved by the insurance company. Once the damaged image has been uploaded and the degree of the damage has been determined, the user may receive insurance only if the firm accepts the damaged image and the condition is greater than 80%.

Non - Functional Requirements

Usability

Efficient for the frequent users, User can easily understand what the application does and feel satisfied with the system.

Availability

It must be available to user for any time. Users can access any time without any hesitation

Scalability

It construct with scalable so it can supports for many users at a single time.

Security

Assures all the data inside the system will be protected against malware attacks or unauthorized access. This application must remain resilient in the face of attacks. The behaviour of the application must be correct and predictable

Performance

This application supporting high amount of users per hour must provide 6 seconds or less response time in a desktop browser, including the rendering of text and images, over an good network connection

Reliability

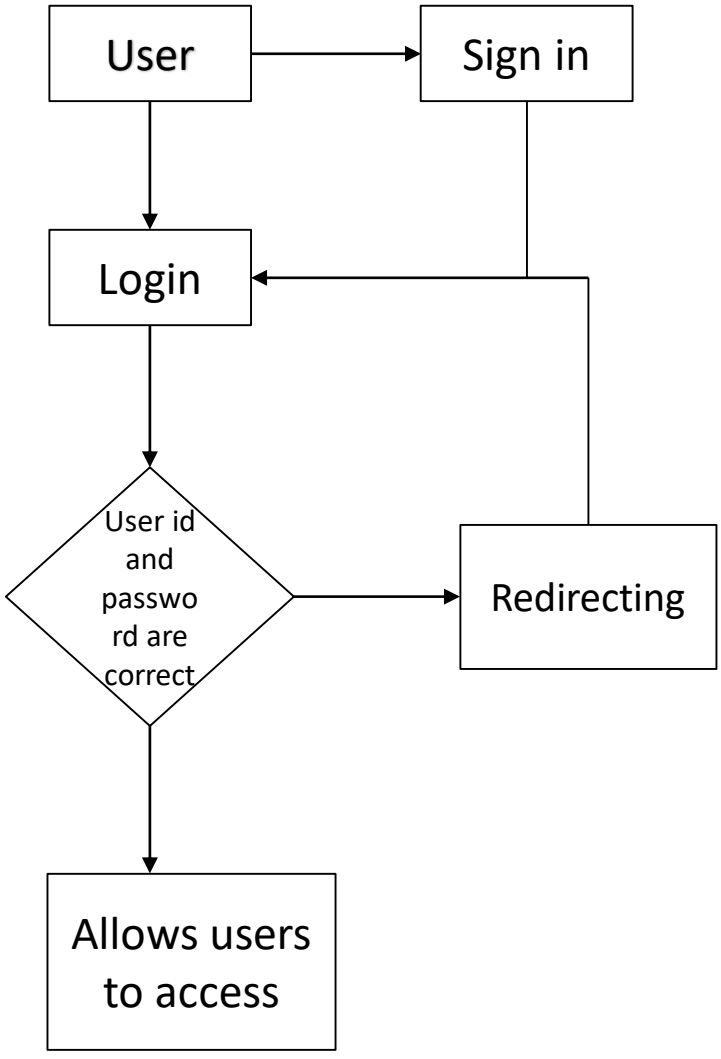
This application must perform without any failure

5. PROJECT DESIGN

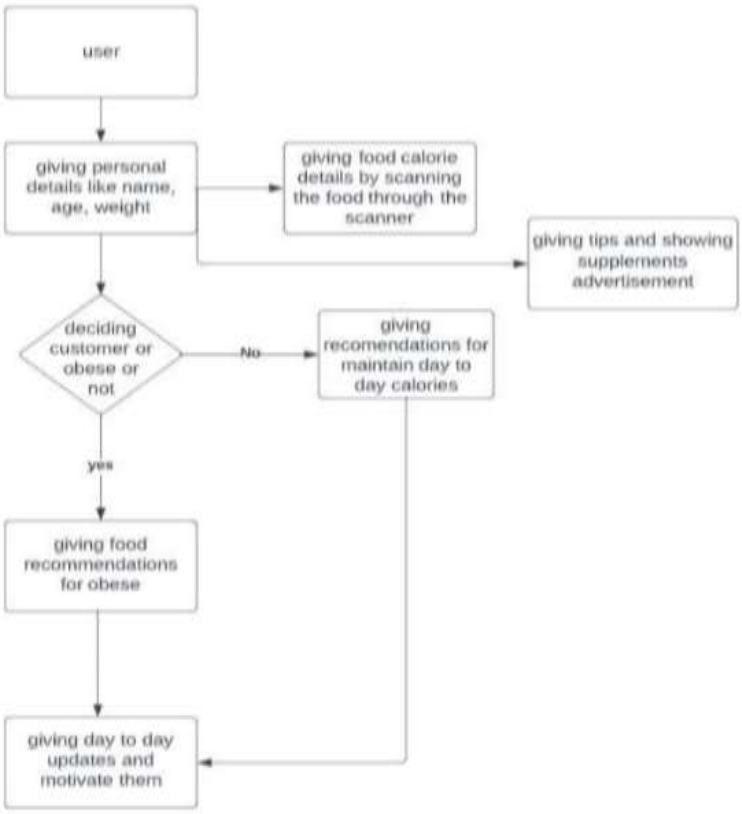
1. Data Flow Diagram

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.

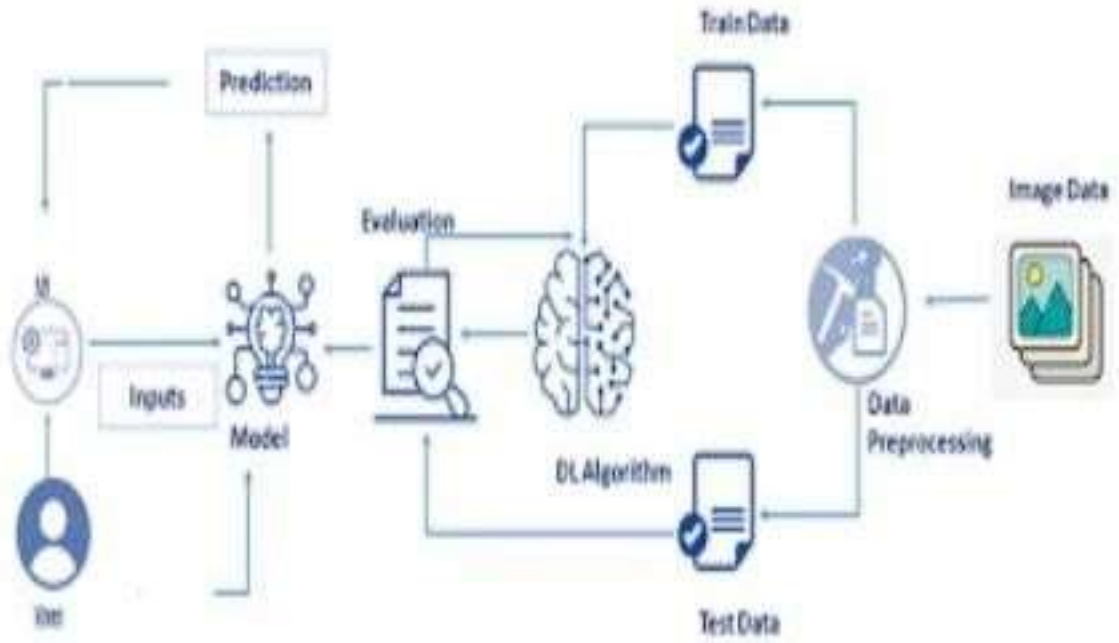
DFD LEVEL 0:



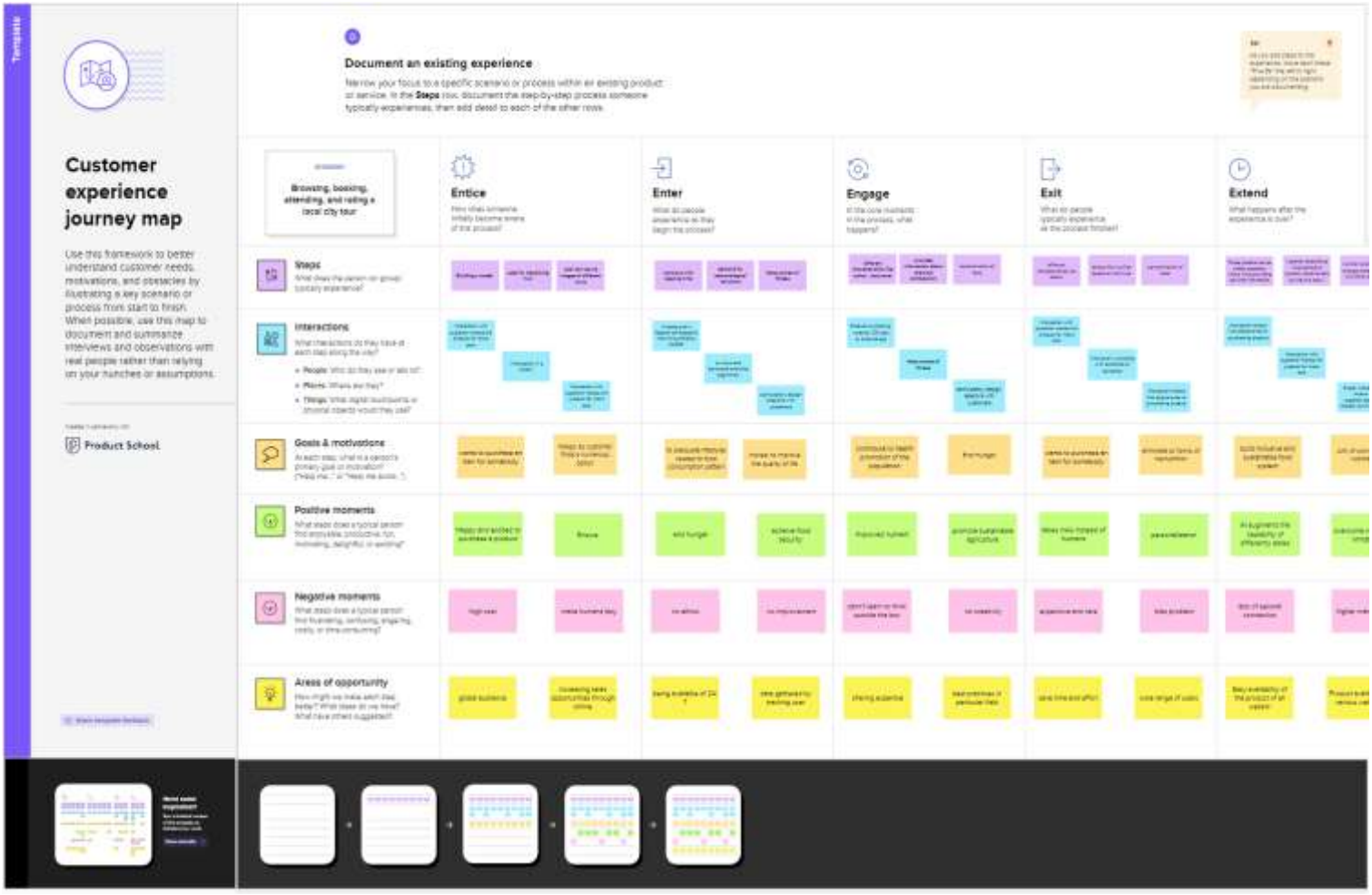
DFD LEVEL 1:



5.2 Solution & Technical Architecture



5.3User Stories



6. PROJECT PLANNING & SCHEDULING

1. Sprint Planning & Estimation

PRODUCT BACKLOG, SPRINT SCHEDULE, AND ESTIMATION

Activity number	Activity name	Activity description	Assigned to
1	Preparation Phase	• Access the resources(courses) in project dashboard • Access the guided project workspace • Create a GitHub account & collaboratewith Project Repository in the project workspace • Set up the Laptop / Computers based on the prerequisites for each technology track	Santhosh, Ashika, Sankarlal, K Praveen
2	Ideation Phase		
2.1	Literature survey	Literature survey on the selected project & Information Gathering	Santhosh, Ashika, Sankar Lal, K Praveen
2.2	Define a problem statement	Prepare the list of problem statements to understand the user's needs	Santhosh, Ashika, Sankar Lal, K Praveen
2.3	Empathy Map	Preparation of Empathy Map Canvas to capture the user's Pains & Gains	Sankarlal, Santhosh
2.4	Brainstorm &idea prioritization	List the ideas by organizing the brainstorming session and prioritize the top 3 ideas based on the feasibility & importance	Santhosh, Ashika, Sankar Lal, K Praveen
3	Project Design Phase I		
3.1	Proposed Solution	Preparation of proposed solution document, which includes the novelty, feasibility of the idea, business model, social impact, scalability of the solution	Santhosh, Ashika
3.2	Problem Solution Fit	Prepared problem is analyzing and making effective solutions for the problem	Sankarlal, K Praveen
3.3	Solution Architecture	Prepare an architecture for the solution	Santhosh, Ashika, Sankar Lal, K Praveen
4	Project Design Phase II		

4.1	Requirement Analysis	Prepare the Functional Requirement and Non- Functional Document	Sankarlal, Ashika
4.2	Customer Journey	Preparation of customer journey maps to understand the user interactions & experiences with the application (entry to exit)	Santhosh, K Praveen
4.3	Data Flow Diagrams	Prepare a Data Flow Diagram for Project use level 0 (Industry Standard)	Santhosh, Sankarlal
4.4	Technology Architecture	Prepare the Technology Architecture of thesolution	Ashika, K Praveen
5	Project Planning Phase		
5.1	Milestones & Tasks	Prepare Milestone & Activity List	Santhosh, Ashika
5.2	Sprint Schedules	Prepare Sprint Delivery Plan	Sankar Lal, K Praveen
6	Project Development Phase		
6.1	Coding & Solutioning	Sprint-1 Delivery: Develop the Code, Test, and push it to GitHub.	K Praveen, Santhosh
6.2	Acceptance Testing	Sprint-2 Delivery: Develop the Code, Test, and push it to GitHub. Sprint-3 Delivery: Develop the Code, Test and push it to GitHub	Sankarlal, Ashika Santhosh, Ashika, Sankar Lal, K Praveen
6.3	Performance Testing	Sprint-4 Delivery: Develop the Code, Test, and push it to GitHub.	Santhosh, Ashika, Sankar Lal, K Praveen

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	08	5 Days	29 Oct 2022	02 Nov 2022	20	3 Nov 2022
Sprint-2	15	5 Days	03 Oct 2022	07 Nov 2022	20	8 Nov 2022
Sprint-3	15	5 Days	08 Nov 2022	12 Nov 2022	20	11 Nov 2022
Sprint-4	25	5 Days	13 Nov 2022	17 Nov 2022	20	16 Nov 2022

Velocity:

Average Velocity= 12/4 =3

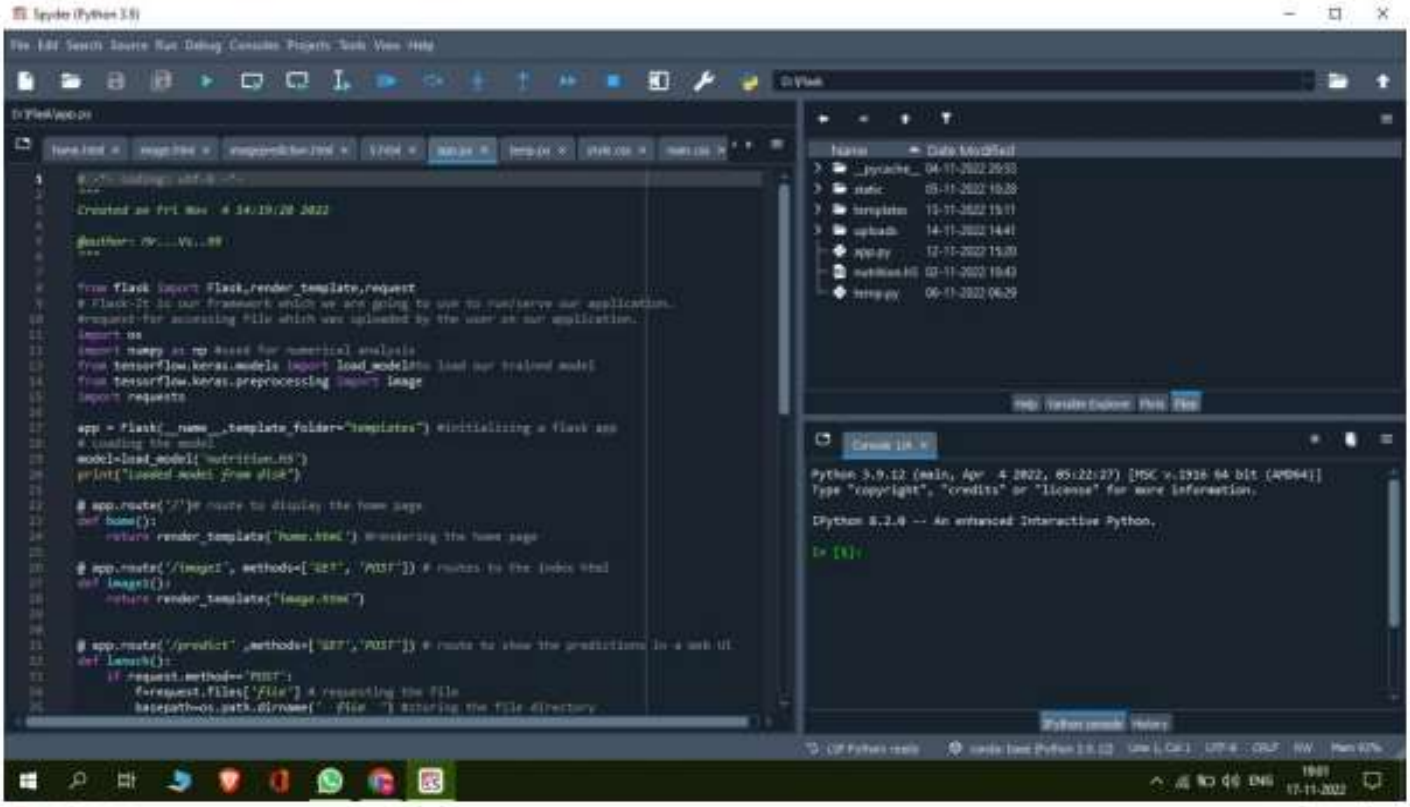
7. CODING & SOLUTIONING

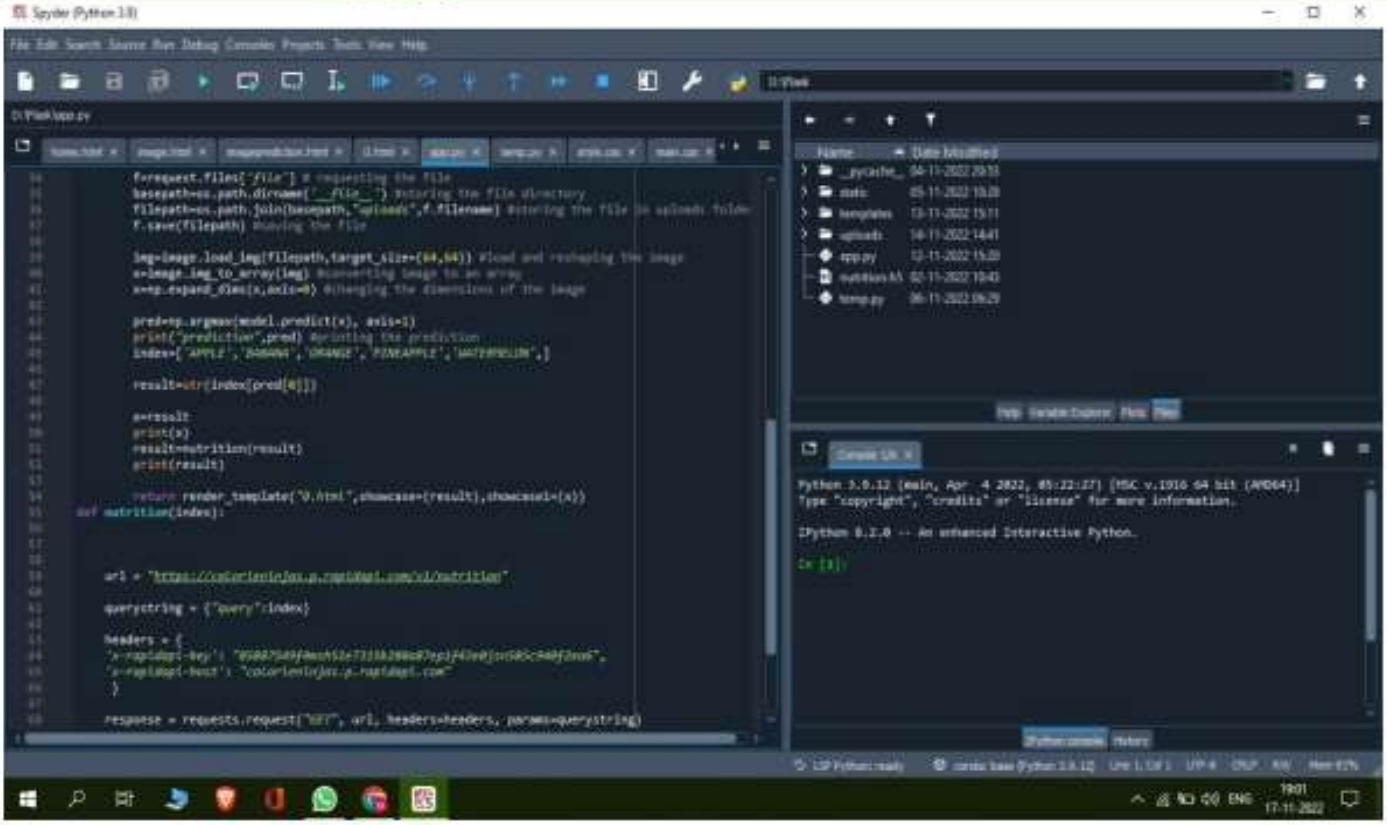
1. Features 1

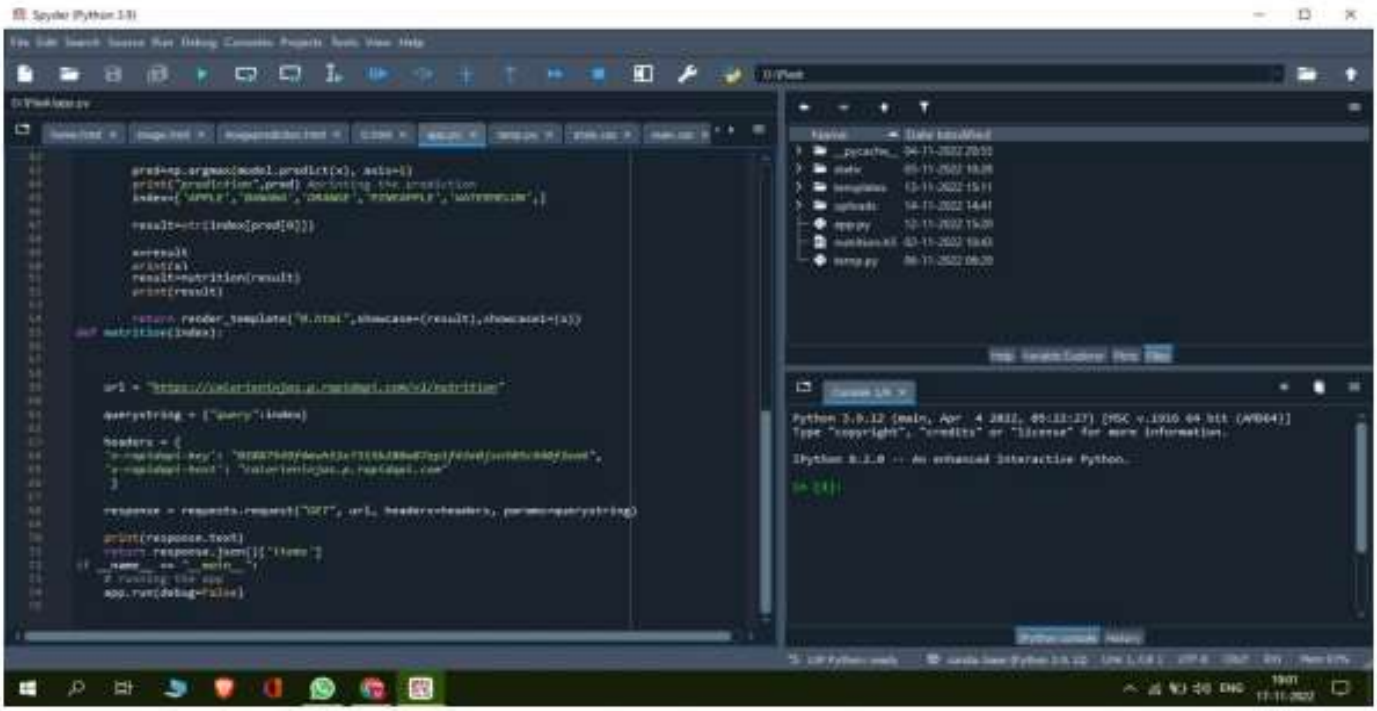
```
from flask import Flask,render_template,request
# Flask-It is our framework which we are going to use to run/serve our application.

#request-for accessing file which was uploaded by the user on our application.
import os
import numpy as np #used for numerical analysis
from tensorflow.keras.models import load_model#to load our trained model
from tensorflow.keras.preprocessing import image
import requests

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

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

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


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

```
    f=request.files['file'] # requesting the file
    basepath=os.path.dirname('_file_') #storing the file directory
    filepath=os.path.join(basepath,"uploads",f.filename) #storing the file in uploads folder
    f.save(filepath) #saving the file

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

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

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

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

    return render_template("0.html",showcase=(result),showcase1=(x))
def nutrition(index):
```

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

    querystring = {"query":index}

    headers = {
        'x-rapidapi-key': "85887549f4msh51e7315b280a87ep1f43e0jsn585c940f2ea6",
        '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)
```







7.2 FEATURE-2

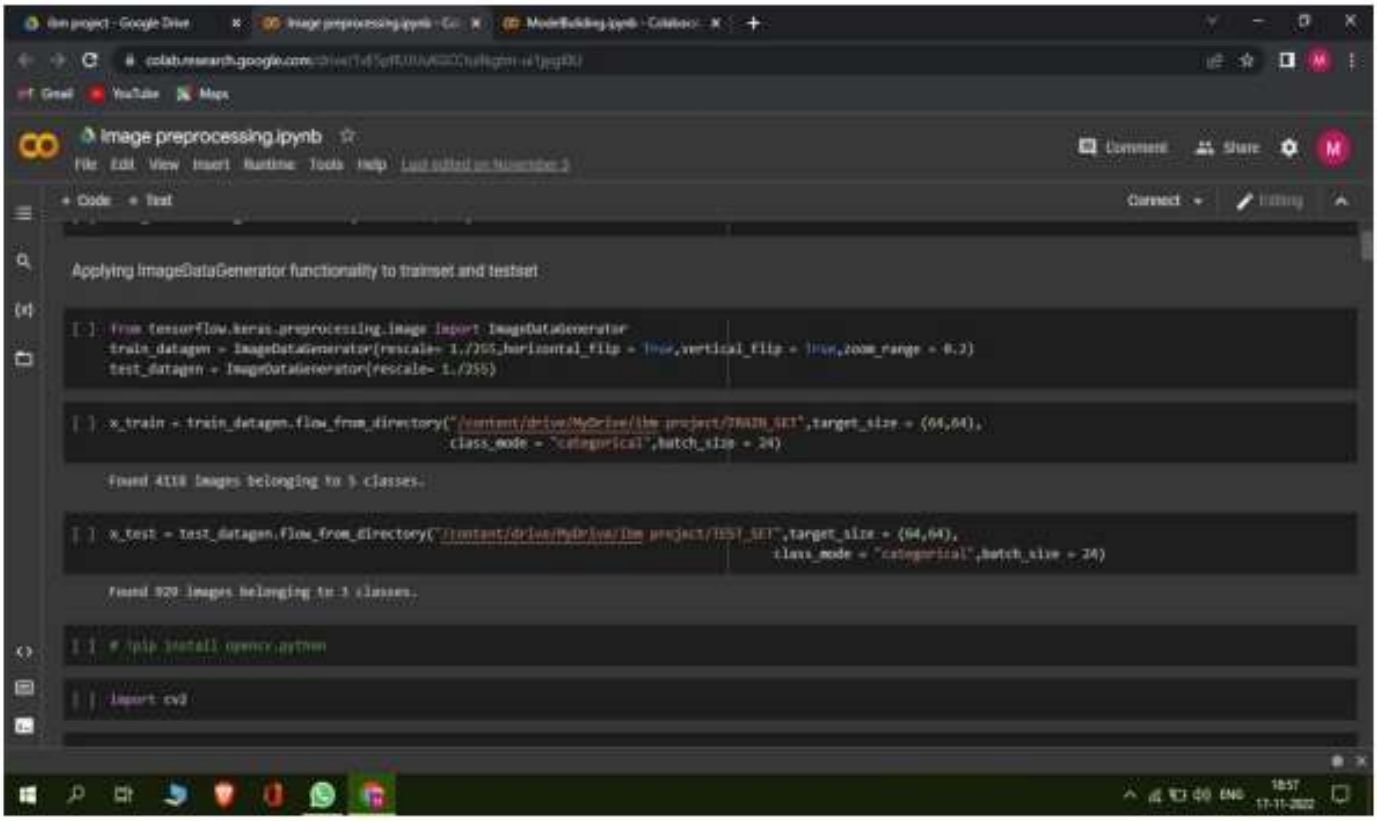
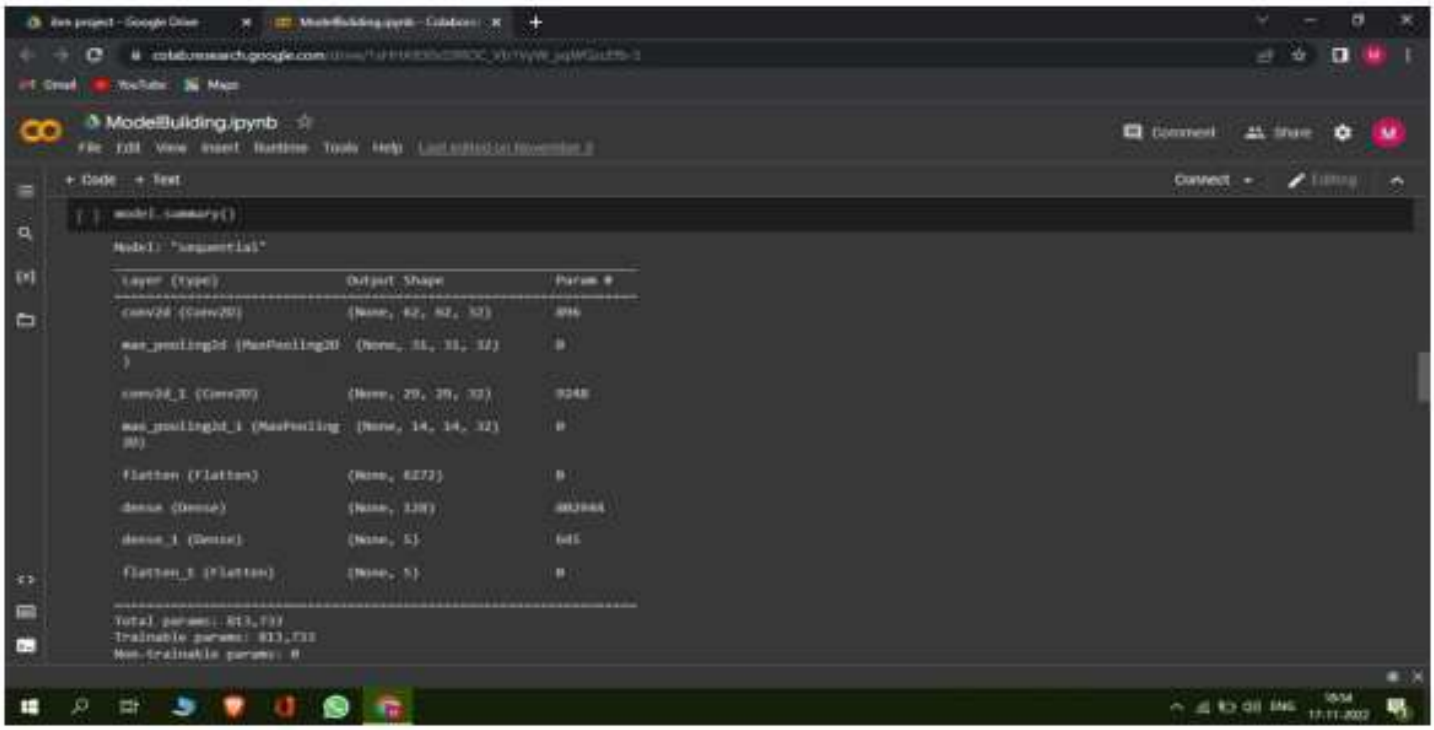


image preprocessing.ipynb
Applying ImageDataGenerator functionality to trainset and testset
Found 4118 images belonging to 5 classes.
Found 929 images belonging to 5 classes.


ModelBuilding.ipynb
File Edit View Insert Runtime Tools Help
+ Code + Text
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 (MaxPooling2D) (None, 14, 14, 32) 0
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

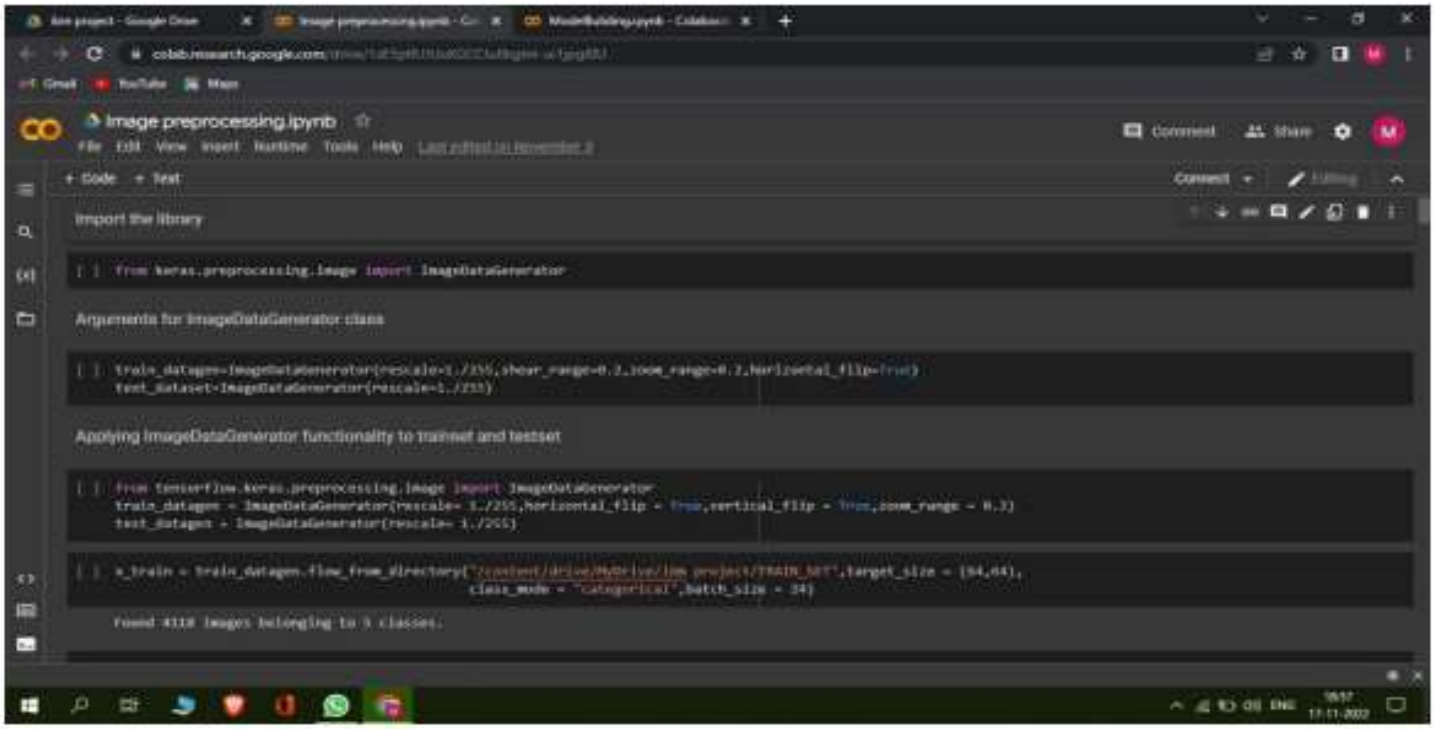
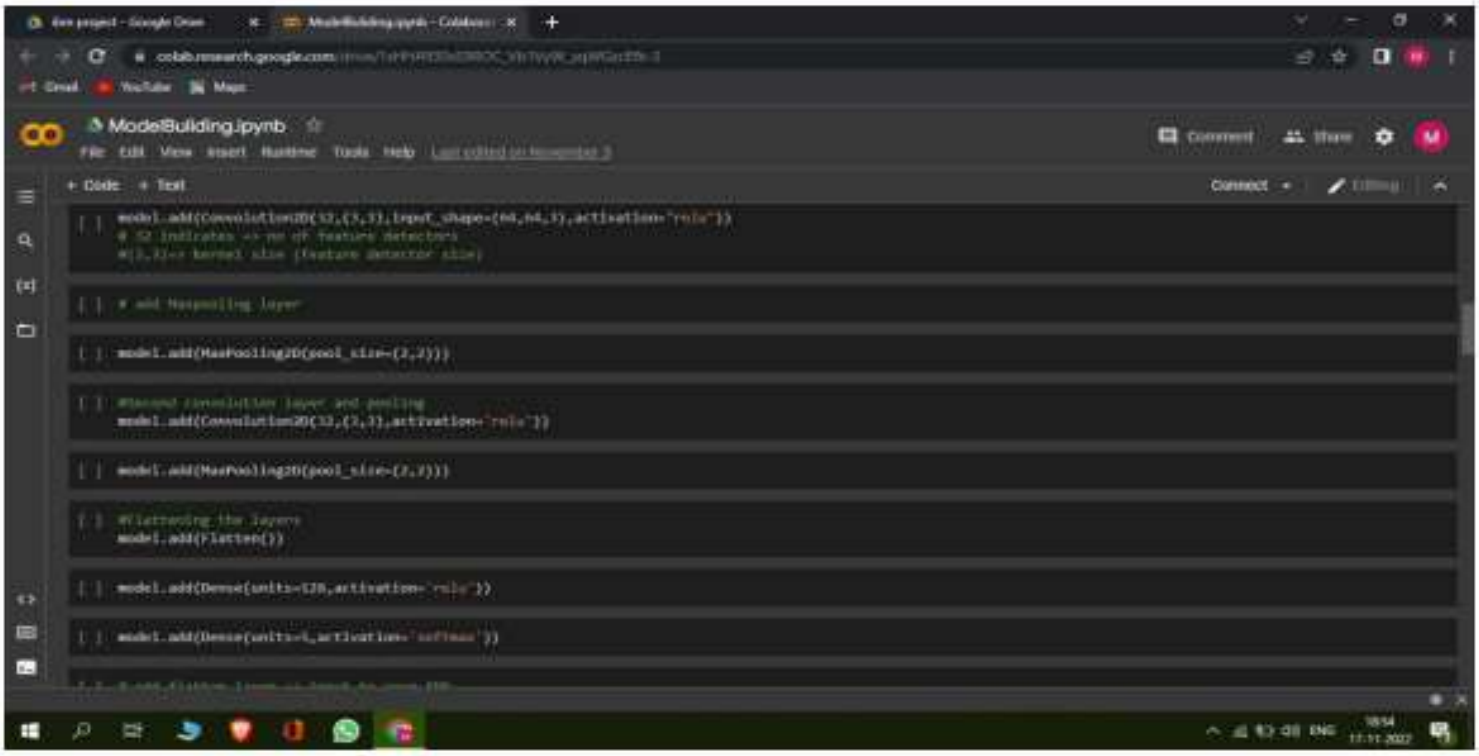
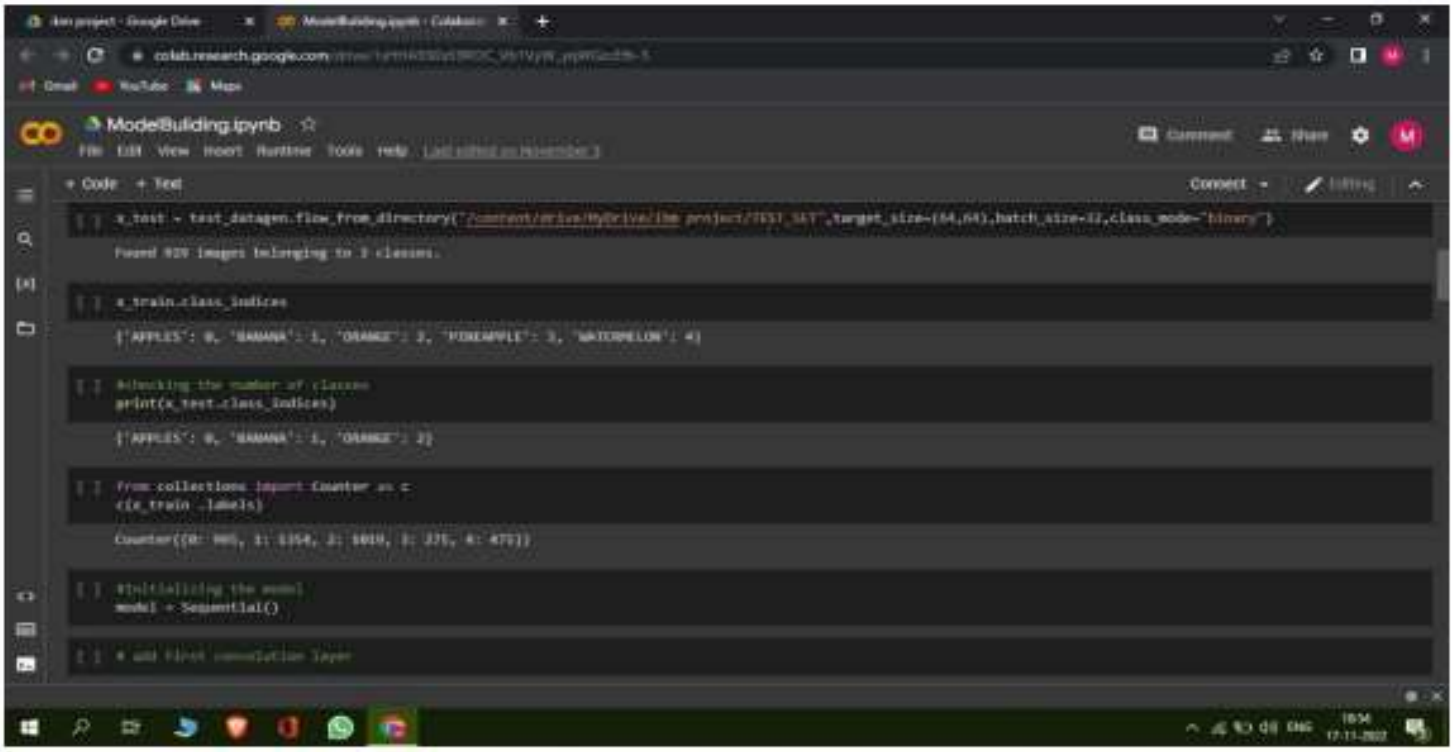
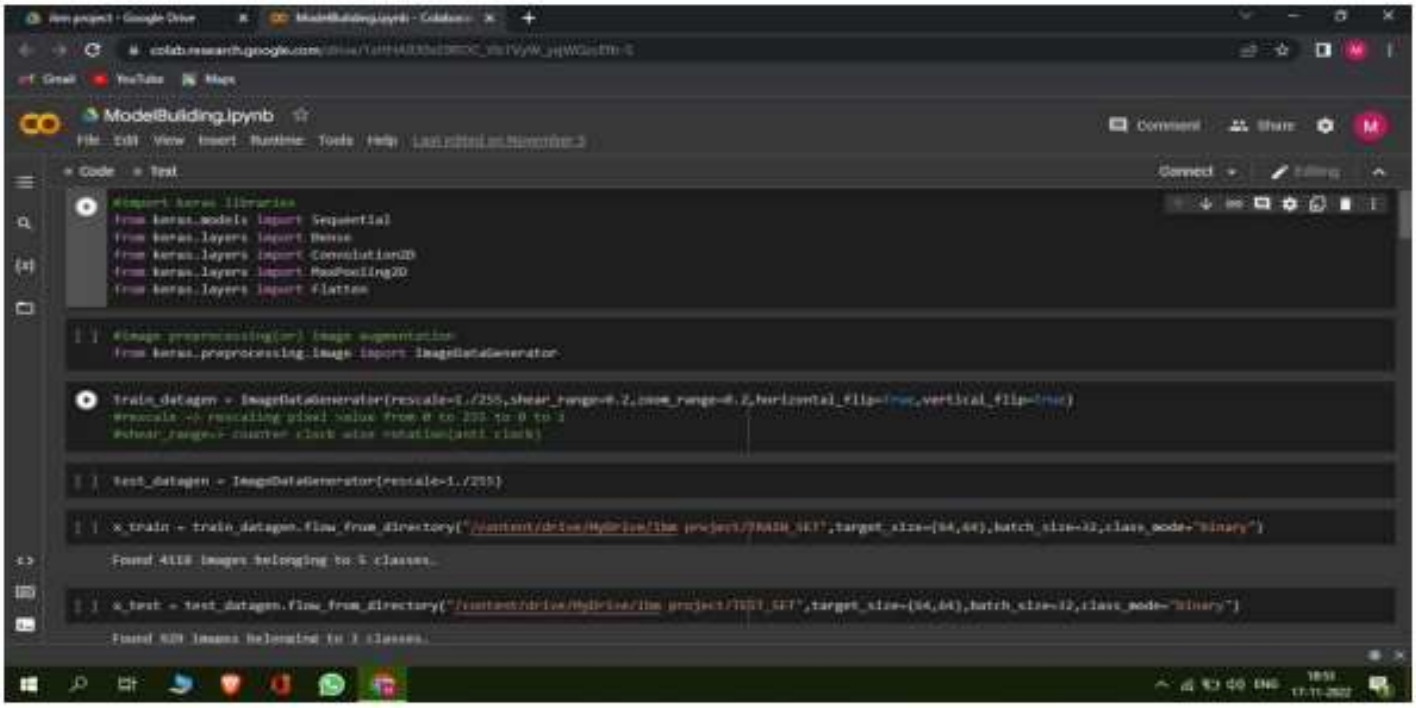


Image preprocessing.ipynb
File Edit View Insert Runtime Tools Help
+ Code + Text
Import the library
from keras.preprocessing.image import ImageDataGenerator
Arguements for ImageDataGenerator class
train_datagen=ImageDataGenerator(rescale=1./255,shear_range=0.2,zoom_range=0.2,horizontal_flip=True)
test_dataset=ImageDataGenerator(rescale=1./255)
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.

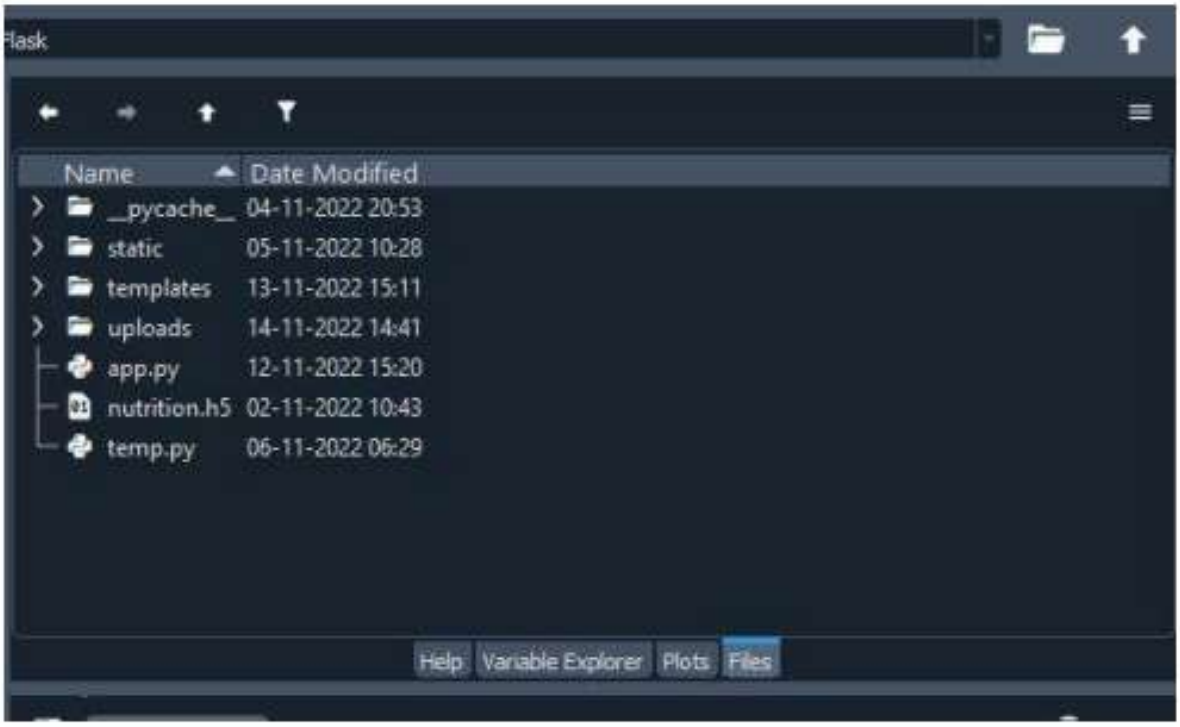

ModelBuilding.ipynb
File Edit View Insert Runtime Tools Help
+ Code + Text
Connect
model.add(Convolution2D(32,(3,3),input_shape=(64,64,3),activation='relu'))
# 32 indicates no of feature detectors
#(3,3)-> kernel size (feature detector size)
# add Maxpooling layer
model.add(MaxPooling2D(pool_size=(2,2)))
#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=4,activation='softmax'))


ModelBuilding.ipynb
File Edit View Insert Runtime Tools Help
+ Code + Text
Connect
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 5 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: 235, 4: 472})
#Initialising the model
model = Sequential()
# add first convolution layer



8. TESTING

8.1 TEST CASE



8.2 USER ACCEPTANCE TESTING

1. PURPOSE OF DOCUMENT The purpose of this document is to briefly explain the test coverage and open issues of the [AI-Powered Nutrition Analyzer For Fitness Euthusiasts] project at the time of the release to User Acceptance Testing (UAT).

2. DEFECT ANALYSIS This report shows the number of resolved or closed bugs at each severity level, and how they were resolved

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

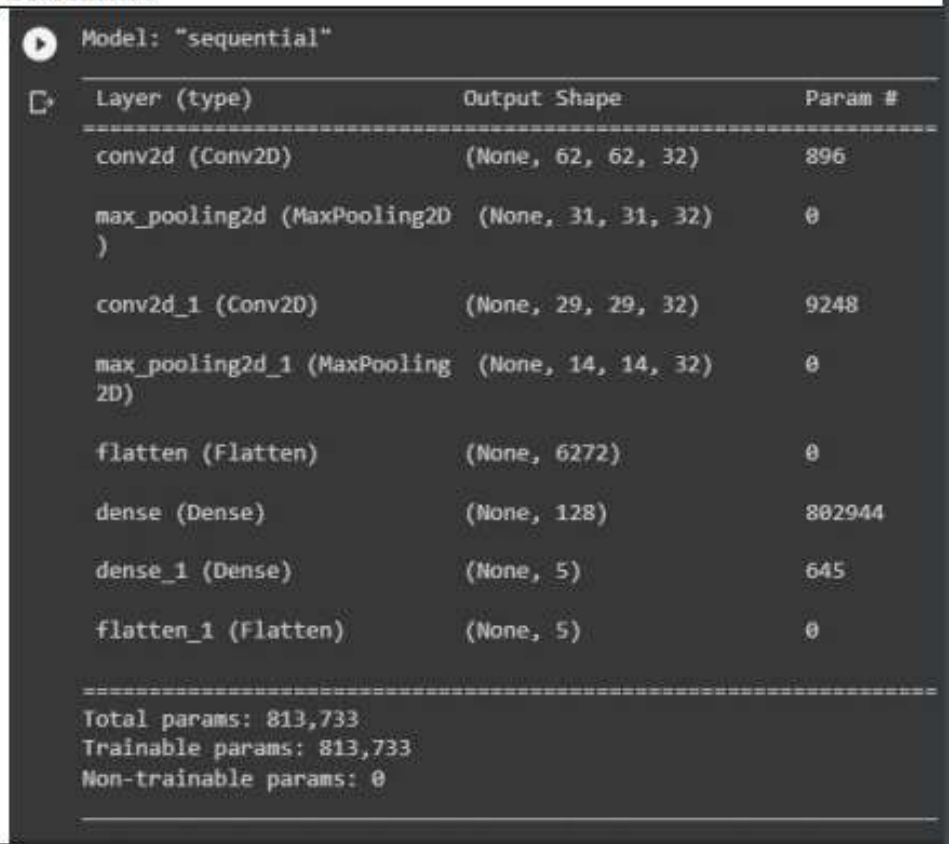
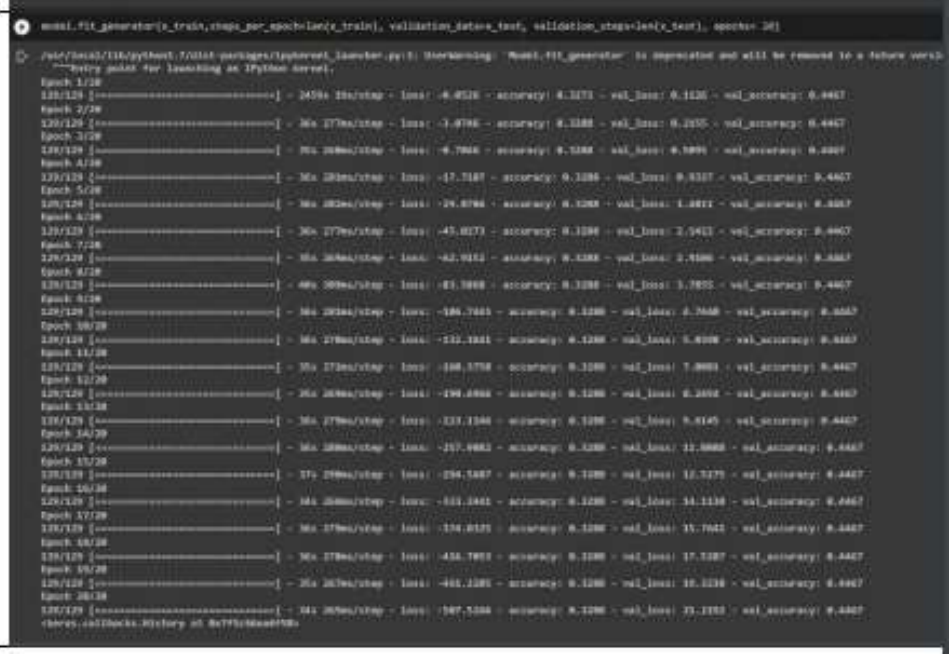
3. TEST CASE ANALYSIS

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

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

9. RESULTS

9.1 PERFORMANCE METRICS

S.NO	Parameter	values	Screenshot
1	Model summary	Total params : 813,733 Trainable params: 813,733 Non-trainable params: 0	
2	Accuracy	Training Accuracy – 96.55 Validation Accuracy- 97.45	

```
Model: "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
_________________________________________________________________
```

10. ADVANTAGES & DISADVANTAGES

ADVANTAGE

- Provide the nutrition content of Multifoods
- Helps for fitness people to maintain and know the proteins and calories of the food
- Gives accurate results in real-time application

DISADVANTAGE

- Hard to know the details of nutrition and calories of food
- Doesn’t ask to provide the users health condition
- Required more time to know the Multifoods

11. CONCLUSION

The approach for an automated food nutrition detection system that can determine the amount of nutrients in food is proposed in this project work. The machine has so far been able to place the meal into one of the many categories listed in the dataset. The well-known food dataset was used for the categorization. The classification of the food photos into their appropriate classifications using a deep learning approach. By reducing noise from the dataset, the classification process may be made better. The same research may be done with a larger dataset, more classes, and more photos in each class since a larger dataset increases accuracy by teaching the algorithm additional features and lowers the loss rate. The model's weights may be saved and used to create designs for food categorization, calorie extraction, and picture classification.

12. FUTURE SCOPE

The food photographs in this research study are categorised into the appropriate groups using a deep learning approach. In terms of future improvement, the classification task may be made better by reducing noise from the dataset. The same research may be done with a larger dataset, more classes, and more photos in each class since a larger dataset increases accuracy by teaching the algorithm additional features and lowers the loss rate. The model's weights may be saved and utilised to create a web or mobile application that classifies images and also extracts the calories from the food that has been identified.

1. APPENDIX

SOURCE CODE

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

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

```
print("Loaded model from disk")


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


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



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

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

        pred=np.argmax(model.predict(x), axis=1)
```

```
    print("prediction",pred) #printing the prediction
    index=['APPLE','BANANA','ORANGE','PINEAPPLE','WATERMELON',]

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

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

    return render_template("0.html",showcase=(result),showcase1=(x))
def nutrition(index):



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

  querystring = {"query":index}

  headers = {
      'x-rapidapi-key': "85887549f4msh51e7315b280a87ep1f43e0jsn585c940f2ea6",
      '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"></script>
   <script src="https://cdn.bootcss.com/jquery/3.3.1/jquery.min.js"></script>
   <script src="https://cdn.bootcss.com/bootstrap/4.0.0/js/bootstrap.min.js"></script>
   <link href="{{ url_for('static', filename='css/main.css') }}" rel="stylesheet">
<style>
body
{
```

```
   background-image: url("https://www.livingproofnyc.com/wp-
content/themes/livingproof/assets/img/hero-background.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;
}
h3
{
margin: 0px;
padding:20px;
background-color:#9ACD32;
width: 800px;
opacity:0.6;
color:#000000;
font-family:'Roboto',sans-serif;
```

```
font-style: italic;
border-radius:20px;
font-size:25px;
}
a
{
color:grey;
float:right;
text-decoration:none;
font-style:normal;
padding-right:20px;
}
a:hover{
background-color:black;
color:white;
border-radius:15px;0
font-size:30px;
padding-left:10px;
}
.div1{
 background-color: lightgrey;
 width: 500px;
 border: 10px solid peach;
 padding: 20px;
```

```
}

.topnav-right {
 float: right;
 padding-right:100px;
}
</style>
</head>
<body>

<!--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>
   <a  href="{{ url_for('image1')}}">Classify</a>
 </div>
</div>
</div>
<br>
<br>
<br>
```


<h1>

<h3>Food is essential for human life and has been the concern of many healthcare conventions. Nowadays new dietary assessment and nutrition analysis tools enable more opportunities to help people understand their daily eating habits, exploring nutrition patterns and maintain a healthy diet.Nutritional analysis is the process of determining the nutritional content of food. It is a vital part of analytical chemistry that provides information about the chemical composition, processing, quality control and contamination of food. It ensures compliance with trade and food laws.</h3>

</h1>

MAIN.JS

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

    // Upload Preview
    function readURL(input) {
        if (input.files && input.files[0]) {
            var reader = new FileReader();
            reader.onload = function (e) {
                $('#imagePreview').css('background-image', 'url(' + e.target.result + ')');
                $('#imagePreview').hide();
                $('#imagePreview').fadeIn(650);
            }
            reader.readAsDataURL(input.files[0]);
        }
```

```
}
$("#imageUpload").change(function () {
    $('.image-section').show();
    $('#btn-predict').show();
    $('#result').text('');
    $('#result').hide();
    readURL(this);
});

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

    // Show loading animation
    $(this).hide();
    $('.loader').show();

    // Make prediction by calling api /predict
    $.ajax({
        type: 'POST',
        url: '/predict',
        data: form_data,
```

```
            contentType: false,
            cache: false,
            processData: false,
            async: true,
            success: function (data) {
                // Get and display the result
                $('.loader').hide();
                $('#result').fadeIn(600);
                $('#result').html(data);
                console.log('Success!');
            },
        });
    });

});
```

HOME PAGE

Home.html

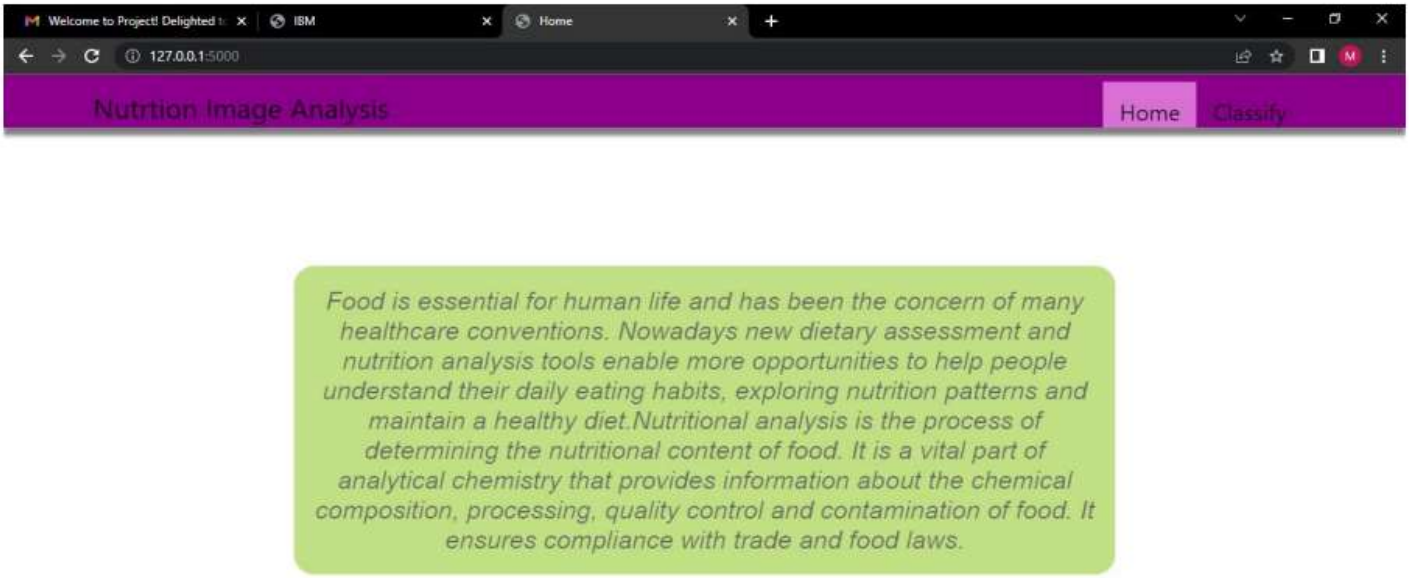
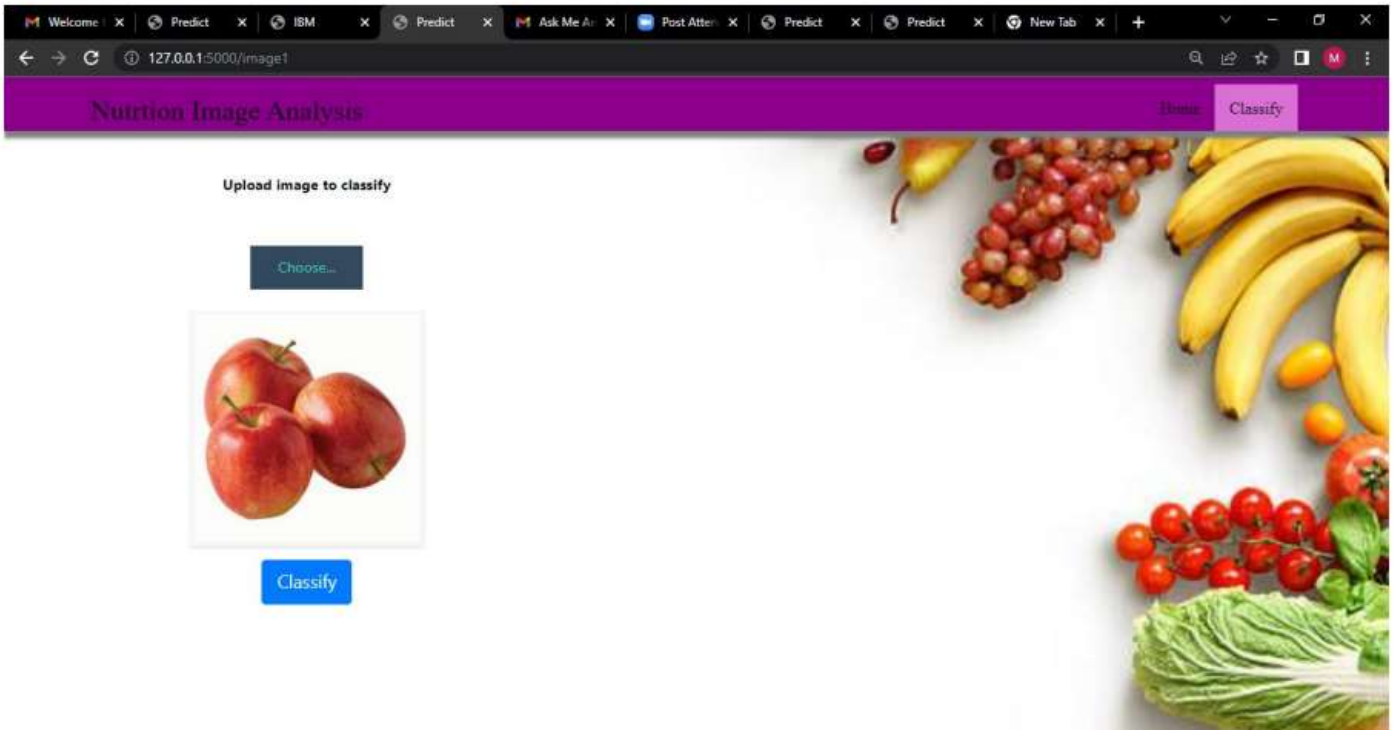
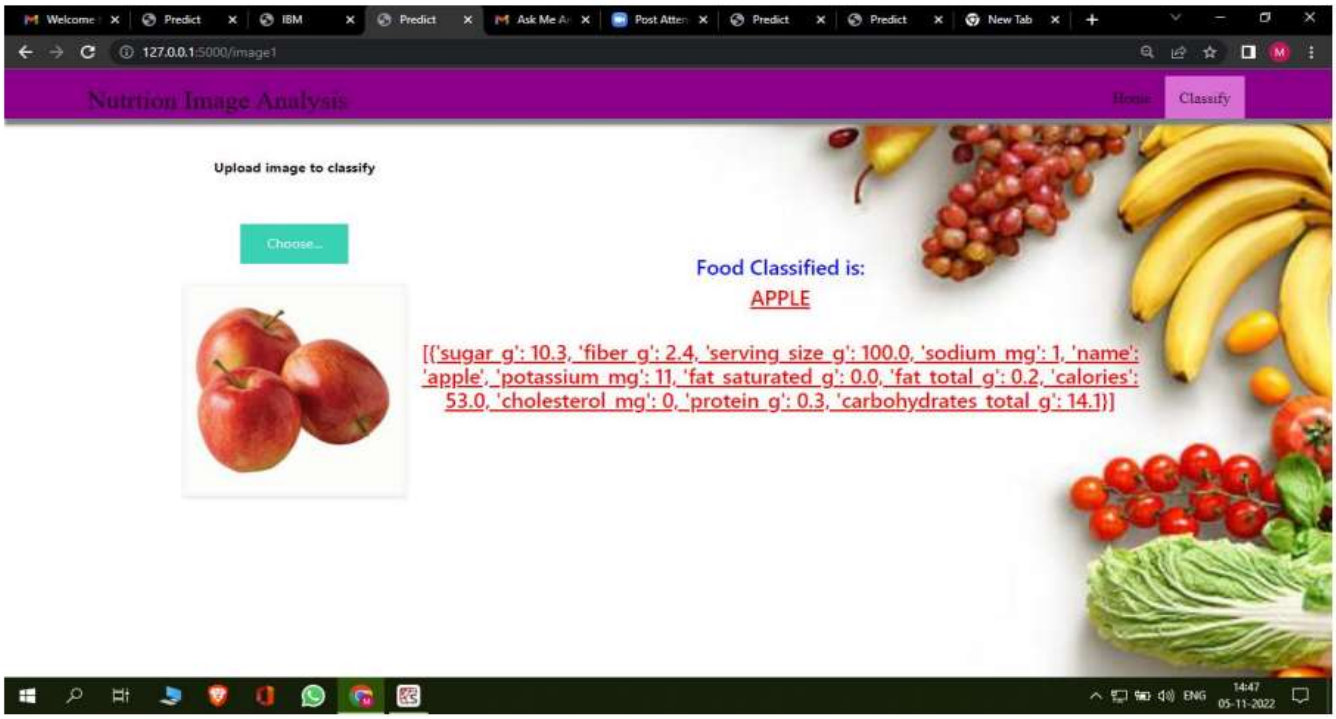
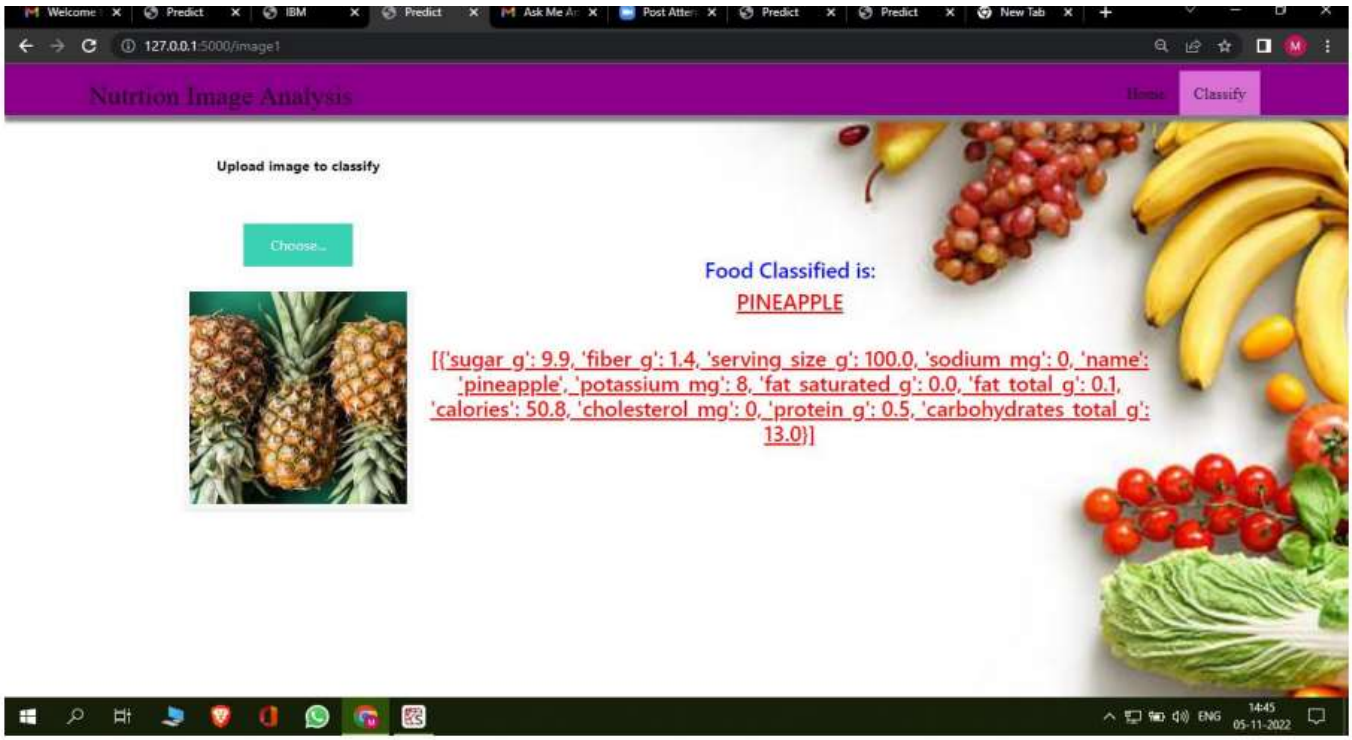


Image.html



Prediction.html





PROJECT DEMO LINK :

https://drive.google.com/drive/folders/1o2Be9IPgoFs9yTSGQ-sczLEDVVYRo7qp