RMK ENGINEERING COLLEGE

(An Autonomous Institution)

**R.S.M. Nagar, Kavaraipettai-601 206, Gummidipoondi Taluk,
Thiruvallur District.**

PROJECT

AI-Powered Nutrition Analyzer For Fitness Enthusiasts

DONE BY

TEAM ID: PNT2022TMID15882

THIRISHAL S (111719104165)

YUVARAJ M (111719104181)

THARUN P (111719104164)

SANTHOSH S(111719104136)

INDEX

1. INTRODUCTION

1.1 Project Overview

People can follow a healthy lifestyle through eating healthy food. The food we eat must contain nutrients which are essential for proper nourishment, growth and immunity for the human body. In today's world, most of the people are living under uncertainty to decide on which foods are healthy and if healthy how much of it can be consumed. The nutritional facts label is printed on food products all over the world and they are represented using a similar structure but these labelling of nutritional information is difficult to understand by the common people. Another issue is that these labels are only for processed and manufactured foods which can be bought in the stores. To make this information accessible in an easier way by classifying these food products into five levels of healthiness ranging from very healthy to very dangerous is the aim of this project work. This is done by a sequential process of data retrieval, data cleaning, data labelling and supervised learning.

1.2 Purpose

The idea of this application is that the user can capture the images of different fruits and vegetables, and then the image will be sent to the trained model. The model analyses the image and detects the nutrition based on the fruits like (Sugar, Fibre, Protein, Calorie intake, etc.).

2. LITERATURE SURVEY

2.1 Existing problem

There are several major health problems in society today and Obesity is one of the main issues. It has increased for nearly three times as much as compared to the year 1975. In 2016, 39% of the adults who are aged 18 years old and above were overweight, and 13% of them were obese (WHO, 2018). It is associated with diseases like cardiovascular, hypertension and also menstrual problems.

Over time, people have been become more conscious about their diet and attempt to have calories control over the years. People have better self-conscious in taking care of these issues and eat healthier with a proper diet plan. Counting the Calorie is a common technique used to calculate their energy taken from one's food consumption. Many are used for the purpose of losing, gaining and maintaining weight. In the past, people have been using traditional ways in calorie counting where they estimate the portion of their meals and then estimating the number of calories in the calories listed book. But this method is pretty much inaccurate since people required to estimate their food portion and the process is very time consuming where they need to look for different listing in the book.

As of now, people have been using modern ways to estimate the calories. Many health-based applications have featured this function in it, where they provide a calories information database for the users and they can search manually through the food listed, then the application will summarize the counting. Nutritional information will also be listed for the user purpose. Moreover, some applications allow the users to capture a picture of their meal & algorithm which is used to process the image and automatically detect the objects, listing down all the nutritional facts and calories information related to the food in a generalized portion.

Although these applications are able to do the basic calories counting, but the result is based on a generalized portion and the amount can be customized by the users. This will lead to inaccurate result, since the portion itself is selected by the users. A user doesn't know how much amount of proper food they are consuming, so the calculations for the calories counting could be wrong. However, this can be improved and enhanced using object counting algorithm. The idea here is to classify each of the class into different kind of portion with an appropriate calorie's information, and then the applications should able to do

quantities counting from the image taken by the user. Therefore, the algorithm is able to segment the portion size of the food itself and then determine the correct quantities, and finally provide a more accurate and reliable calories information based on the values. This is aimed to further enhance calories counting by improving the accuracy of the result and avoid the hassle of having users to manually do a look up in the listing, the process could be shortened and assist them in planning a more reliable diet.

2.2 References

L. Jiang, B. Qiu, X. Liu, C. Huang and K. Lin, "DeepFood: Food Image Analysis and Dietary Assessment via Deep Model," in IEEE Access, vol. 8, pp. 47477-47489, 2020, doi: 10.1109/ACCESS.2020.2973625.

J. Aravind and J. D. Sweetlin, "Nutrient facts analysis using supervised learning approaches," 2017 Conference on Information and Communication Technology (CICT), 2017, pp. 1-6, doi: 10.1109/INFOCOMTECH.2017.8340604.

M. -L. Chiang, C. -A. Wu, J. -K. Feng, C. -Y. Fang and S. -W. Chen, "Food Calorie and Nutrition Analysis System based on Mask R-CNN," 2019 IEEE 5th International Conference on Computer and Communications (ICCC), 2019, pp. 1721-1728, doi: 10.1109/ICCC47050.2019.9064257.

Gerald F. CombsJrProfessor Emeritus, in The Vitamins (Fourth Edition), 2012

Mirjana Gurinović, ... Maria Glibetić, in Reference Module in Food Science, 2017

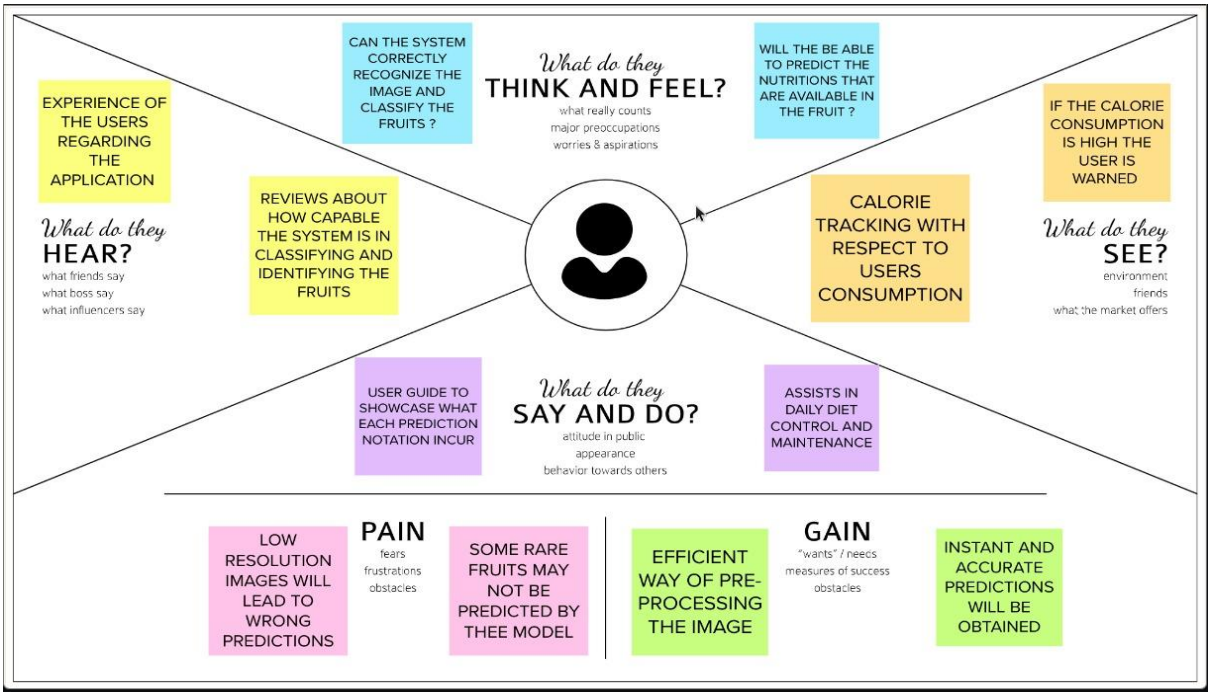
2.3 Problem Statement Definition

Question	**Description**
Who does the problem affect?	The users who want to maintain a healthy and fit body but no one to guide them on their dieting.
Why is it important?	It is important and easy for a user to use AI software rather than having a physical consultant.
What are the benefits?	● AI based technology to detect accurately

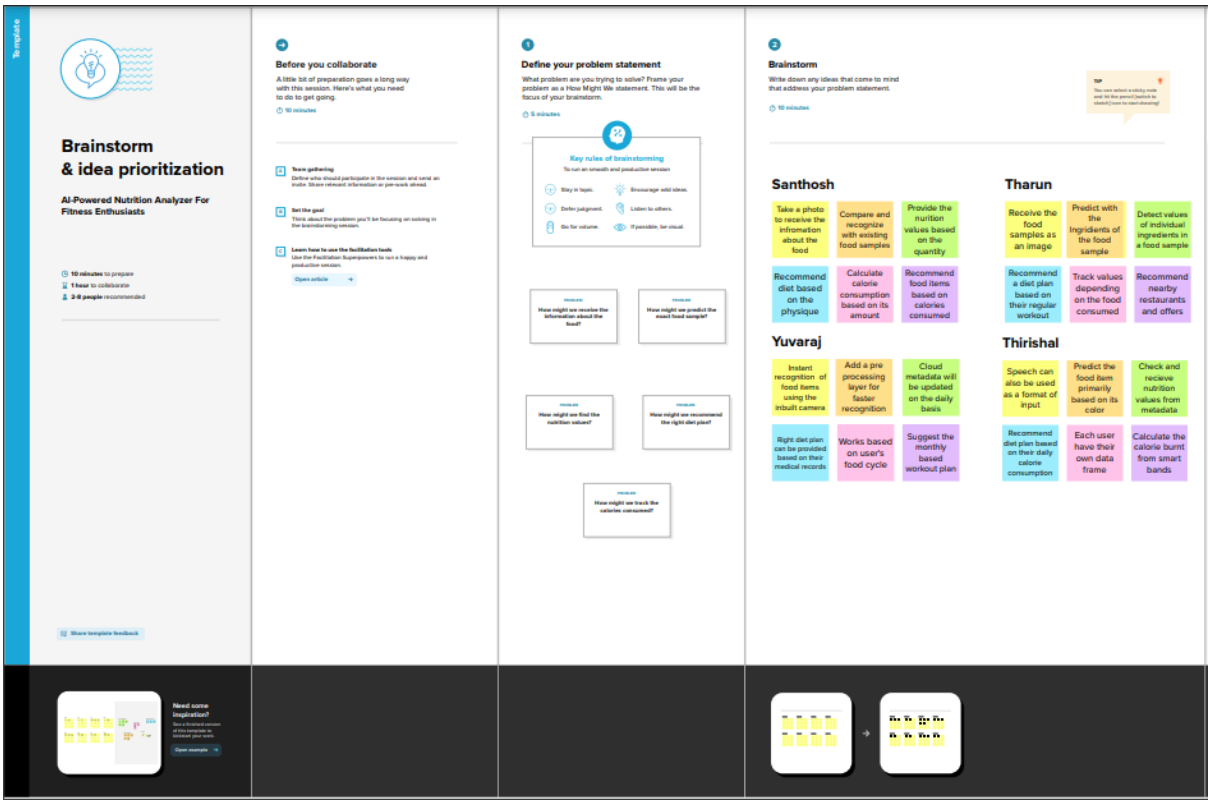
	• Faster processing of data • Better and interactive UI / UX
How is it better than the others?	Application with interactive UI / UX optimized model with higher accuracy
When to use?	In scenarios when we want to have a nutrition analyser.

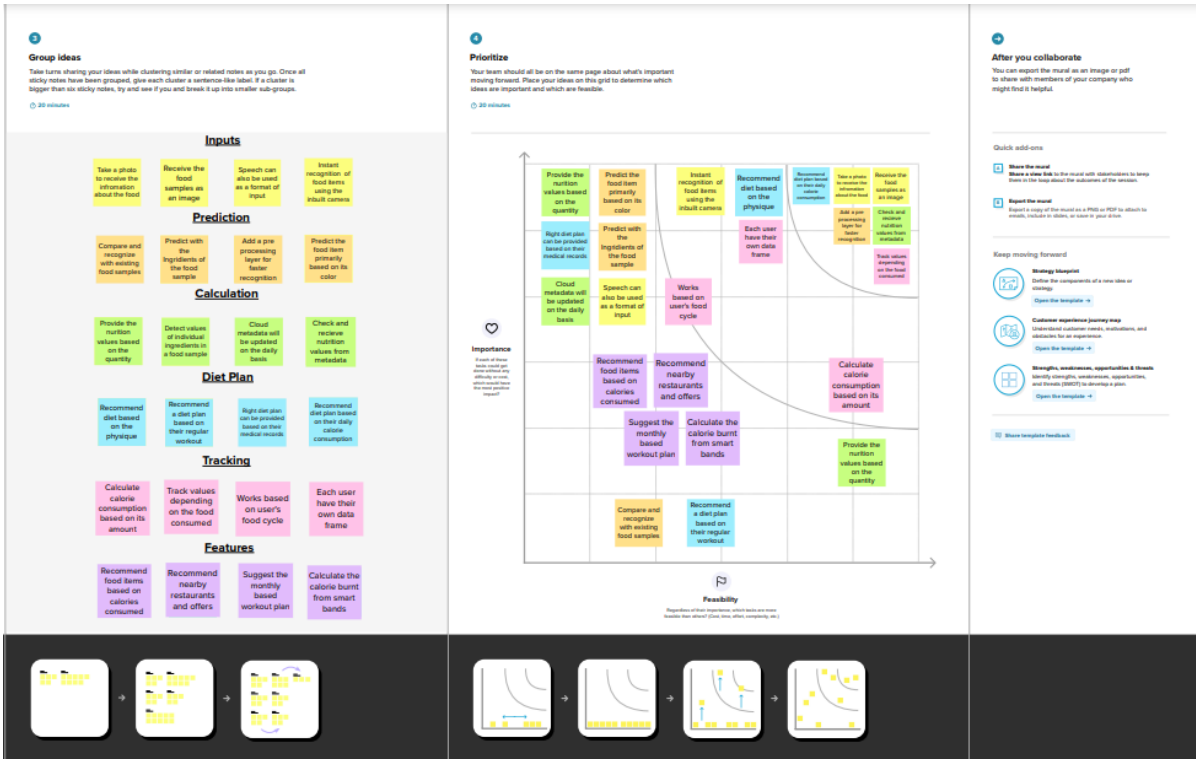
3. IDEATION & PROPOSED SOLUTION

3.1 Empathy Map Canvas



3.2 Ideation & Brainstorming





3.3 Proposed Solution

S. No.	Parameter	Description
1.	Problem Statement (Problem to be solved)	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.	Idea / Solution description	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.
3.	Novelty / Uniqueness	It provides a diet plans for a user based on various criteria to maintain user's physical as well as mental health. The trained system can able to provide suggestions regarding the choice of food and calorie intake to be followed based on fitness goals stating that how much weight should gain / lose according to the current height and weight of an individual.
4.	Social Impact / Customer Satisfaction	It has a major impact on the users who are health conscious and maintain the well being of their mental health
5.	Business Model (Revenue	Along with the application installation revenue the offers and restaurants recommendations for food yields more revenue

	Model)	
6.	Scalability of the Solution	The model is very scalable as each and every feature is built around the model. It is a vital part of analytical chemistry that provides information about the chemical composition, processing, quality control and contamination of food.

3.4 Problem Solution fit

1. Customer Segments:

- Old age people
- Willing to lose weight
- Maintain physique
- follow diet

2. Jobs to be done / Problems:

- Regular Completion of week task
- Keep calm and hold your patience
- Stability and confidentiality
- People need to understand the problem adequately.

3. Triggers

- Task triggers the persons who involved in various exercises and to be active.
- Good user experience
- Easy usable

4. Emotions: BEFORE / AFTER

- It boost up their confidence level
- It endured their health always healthy

5. Available Solutions

- For all problems one solution, for everything this app has remedy.
- Reduces diseases and not necessary to visit hospital.
- It identifies vitamins, protein, carbohydrates in our body.

6. Customer

- Customer problem must be clear.
- Low cost and easy accessible.
- It should not cause any side effects.

7. Behaviour

- It replicates their day to day improvement and they can see their change in regular basis.
- Chat options are provided for any queries.
- It replicates your health issue.

8. Channels of Behaviour

ONLINE

- You can afford this by both online and offline mode
- In online, chat with experts and video call and tell your mentors

OFFLINE

In offline, must follow diet chart, maintaining health properly, taking exercises regularly.

9. Problem Root Cause

- Persons should consider the sugar content, fat content on the food they are taking. They must consider about it very seriously
- This all leads to serious health problem issues
- If the maintain their healthy properly

10. Your Solutions

- It is an AI driven detection nutrition detection model. This app makes clear about your health and it takes care about the health in regular basis.
- Every age member can use this app under any condition.
- Overall a best one for being a healthy person.

4. REQUIREMENT ANALYSIS

4.1 Functional requirements

Following are the functional requirements of the proposed solution.

FR No.	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)
FR-1	**User Registration**	Registration through Form Registration through Gmail
FR-2	**User Confirmation**	Confirmation via Email Confirmation via OTP
FR-3	**User Login**	Login through Google Login through Email
FR-4	**Image Capturing and Processing**	Selection of desired package
FR-5	**Diet Plan Specification**	Daily plans will be generated by dietician
FR-6	**Storage of Data**	Gathering information from database and generating report
FR-7	**Calorie Over-Consumption Notification**	The user will receive notification if he/she consuming more calories within a short period of time
FR-8	**Diet Plan Specification**	Users can choose the type of diet plan they want to follow based on their goals, such as weight loss or muscle building. The application searches the internet for diet plans and food items that will help them achieve their goals.
FR-9	**Query**	The user can ask for changes in plan. Can have query regarding their food consumption..

4.2 Non-Functional requirements

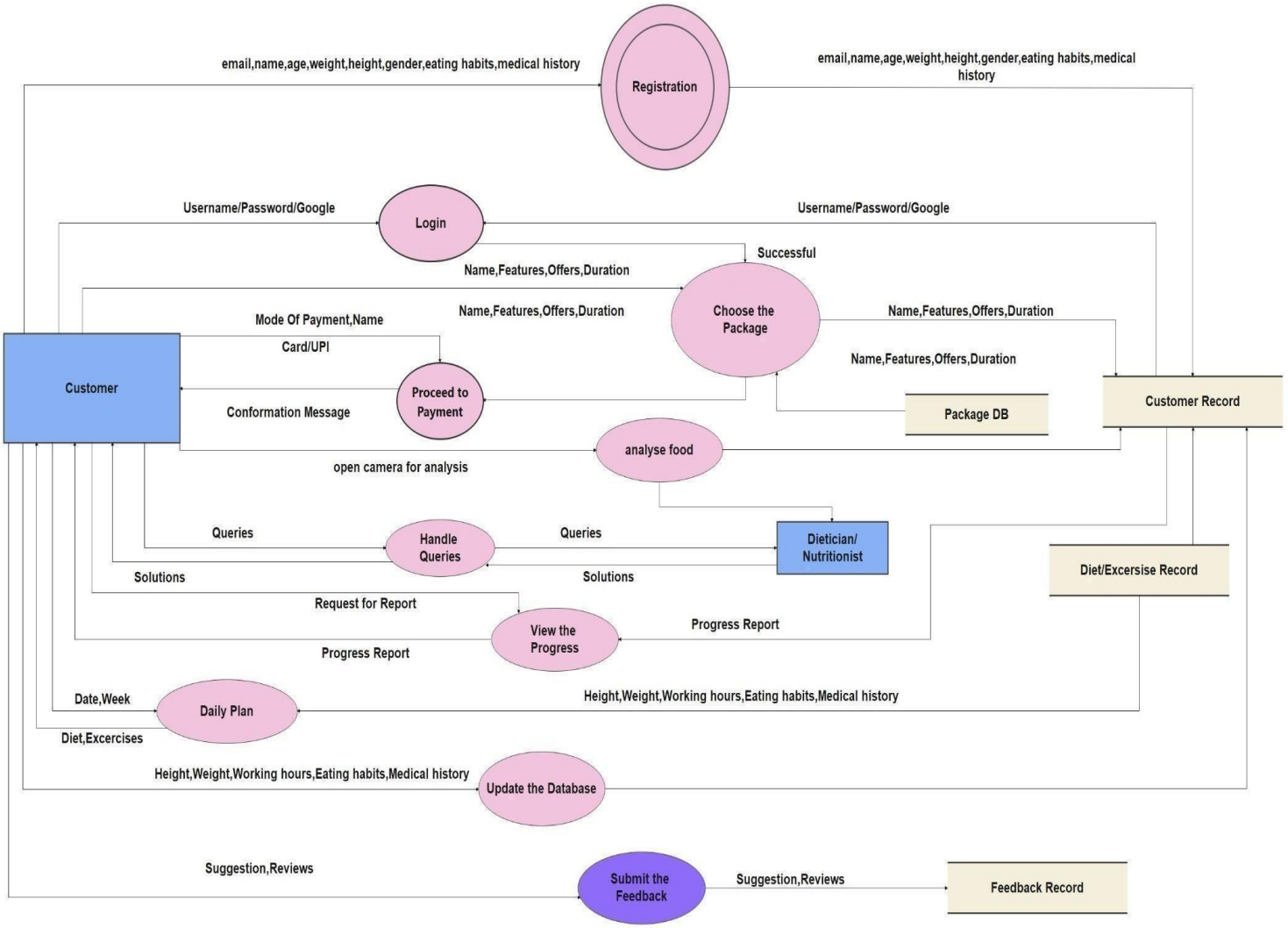
Following are the non-functional requirements of the proposed solution.

FR No.	Non-Functional Requirement	Description
NFR-1	**Usability**	Easy to use with interactive User Interface
NFR-2	**Security**	User can access only their personal information and not that of other users.
NFR-3	**Reliability**	The average time of failure shall be 7 days.
NFR-4	**Performance**	The results has to be shown within 10 sec
NFR-5	**Availability**	The dietician shall be available to users 24 hours a day, 7 days a week.
NFR-6	**Scalability**	Supports various food items

5. PROJECT DESIGN

5.1 Data Flow Diagrams

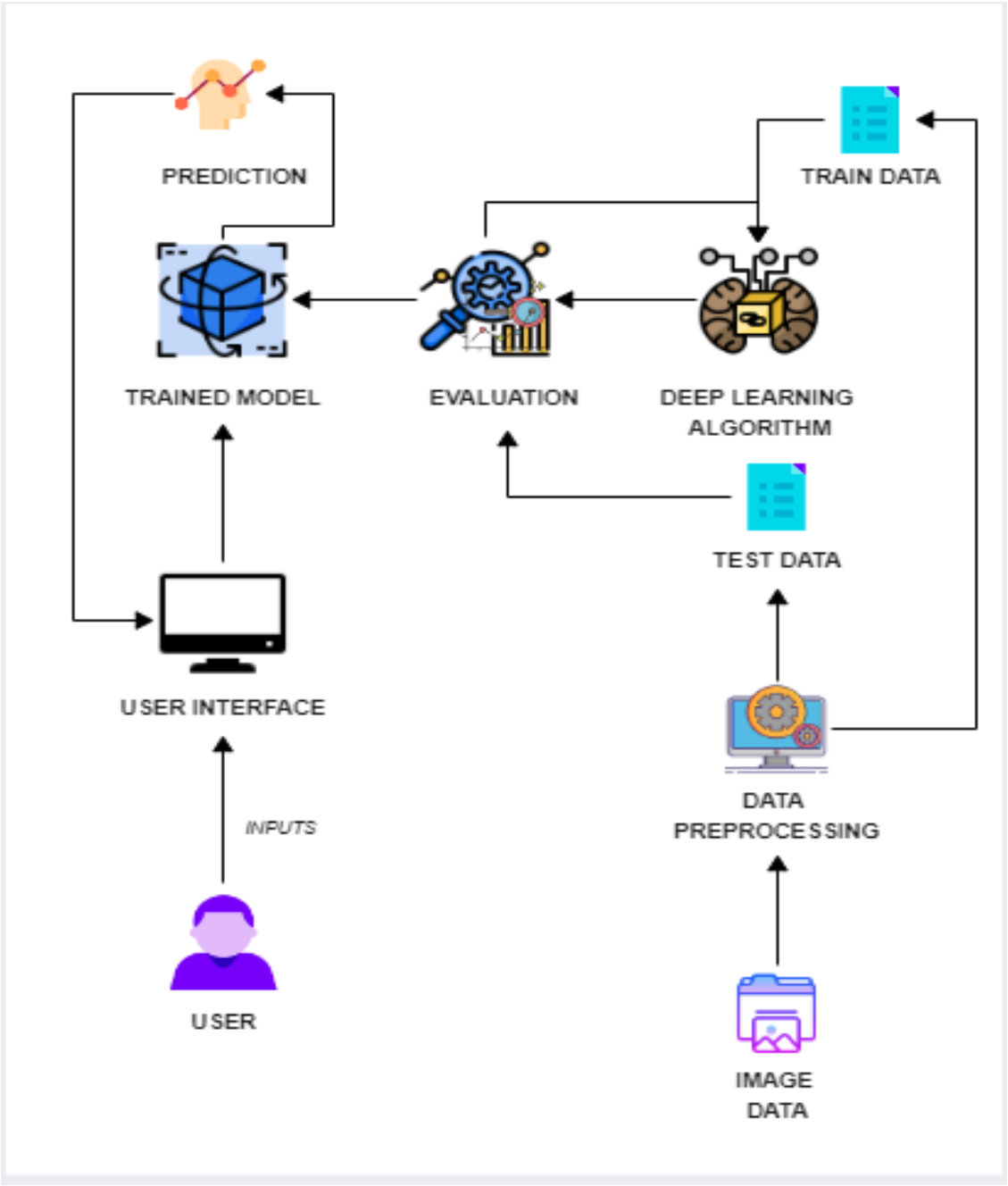
A Data Flow Diagram (DFD) is a traditional visual representation of the information flows within a system. A neat and clear DFD can depict the right amount of the system requirement graphically. It shows how data enters and leaves the system, what changes the information, and where data is stored.



5.2 Solution & Technical Architecture

Solution architecture (SA) is an architectural description of a specific solution. SAs combine guidance from different enterprise architecture viewpoints (business, information and technical), as well as from the enterprise solution architecture (ESA).

A technical solutions architect is somebody who helps companies design and delivers a range of solutions to their problems. Technical solutions architects need to have the skills and the knowledge to create solutions that fit in with company strategy.



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	As a user, I can register for the application by entering my email, password, and confirming my password.	I can access my account / dashboard	High	Sprint-1
		USN-2	As a user, I will receive confirmation email once I have registered for the application	I can receive confirmation email & click confirm	High	Sprint-1
		USN-3	As a user, I can register for the application through Facebook	I can register & access the dashboard with Facebook Login	Low	Sprint-2
		USN-4	As a user, I can register for the application through Gmail		Medium	Sprint-1
	Login	USN-5	As a user, I can log into the application by entering email & password		High	Sprint-1
	Dashboard	USN-6	Not Required		Medium	Sprint-2
Customer (Web user)	Data Setup	USN-7	As a User, I will enter my Data's which includes Height, Weight, Age & Gender ..	I can include the data contents in the application	High	Sprint-2
Customer Care Executive	Queries	USN-8	As a User, I can raise and submit their queries in the Q&A section	I can raise the queries how to use the application in Q&A.	Medium	Sprint-1
Administrator	Diet Chart	USN-9	User can customize the Diet chart based upon their Maintenance calorie.	I can customise the diet chart.	High	Sprint-2
	Liquid Intake	USN-10	User can add the amount water they consume and get the hourly remainder to consume water	I can get the hourly remainder to consume water .	Medium	Sprint-1
	Workout	USN-11	I can add the customized workout to do on the daily basis.	I can the daily workout session.	High	Sprint-1
	Community Chart	USN-12	User can create a community and chat with them accordingly	I can interact with other users.	Medium	Sprint-2

6. PROJECT PLANNING & SCHEDULING

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-1	Data Collection	USN-1	Download the food nutrition dataset	2	High	Tharun P,Yuvaraj M
Sprint-1	Data Preprocessing	USN-2	Importing the Dataset into Workspace	1	Medium	Thirishal S,Yuvaraj M
Sprint- 1		USN-3	Handling Missing data	3	Low	Yuvaraj M,Santhosh S
Sprint- 1		USN-4	Feature Scaling	3	Medium	Santhosh S
Sprint- 1		USN-5	Data Visualization	3	Low	Tharun P, Yuvaraj M

Sprint- 1		USN-6	Splitting Data into Train and set	4	High	Santhosh S
Sprint-1		USN-7	Creating A Dataset with Sliding Windows	4	Medium	Santhosh S
Sprint- 2	Model Building	USN-8	Importing The Model	1	High	Thirishal S
			Building Libraries			
Sprint-2		USN-9	Initializing The Model	1	Medium	Yuvaraj M
Sprint- 2		USN-10	Adding CNN Layers	2	High	Yuvaraj M
Sprint- 2		USN-11	Adding Dense Layers	3	Low	Yuvaraj M
Sprint- 2		USN-12	Configure The Learning Process	4	Medium	Thirishal S
Sprint- 2		USN-13	Train the model	2	Medium	Thirishal S
Sprint- 2		USN-14	Save the model	2	Medium	Thirishal S
Sprint- 2		USN-15	Test the model	3	High	Tharun P
Sprint- 3	Application Building	USN-16	Create an HTML file	4	Medium	Tharun P

Sprint- 3		USN-17	Build Python code	4	High	Santhosh S
Sprint- 3		USN-18	Creating our flask application & loading our model using local model method	4	Medium	Santhosh S

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint- 3		USN-19	Run the application	4	High	S,Yuvaraj M, Thirishal S
Sprint- 4	Train the model on IBM	USN-20	Register for IBM Cloud	4	Medium	Tharun P, Santhosh S, Yuvaraj M, Thirishal S
Sprint- 4		USN-21	Train the ML Model on IBM	4	High	Tharun P, Santhosh S, Yuvaraj M, Thirishal S
Sprint- 4		USN-22	Integrate Flask with scoring End Point	8	High	Thirishal S

Project Tracker, Velocity & Burndown Chart:

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	6 Days	24 Oct 2022	29 Oct 2022	20	29 Oct 2022
Sprint- 2	20	6 Days	31 Oct 2022	05 Nov 2022	20	05 Nov 2022
Sprint- 3	20	6 Days	07 Nov 2022	12 Nov 2022	20	12 Nov 2022

Sprint- 4	20	6 Days	14 Nov 2022	19 Nov 2022	20	19 Nov 2022

Velocity:

Imagine we have a 10-day sprint duration, and the velocity of the team is 20 (points per sprint). Let's calculate the team's average velocity (AV) per iteration unit (story points per day)

$$AV = \frac{sprint\ duration}{velocity} = \frac{20}{10} = 2$$

6.2 Sprint Delivery Schedule

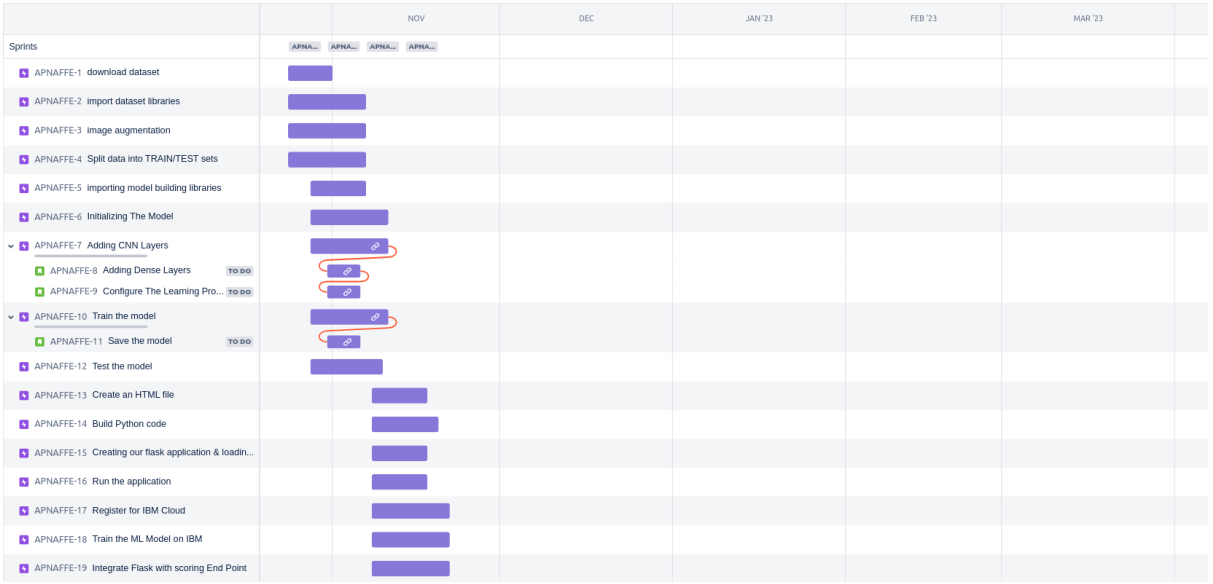
Sprint	Total Story Points	Duration	Sprint Start Date	Sprint End Date (Planned)	Story Points Completed (as on Planned End Date)	Sprint Release Date (Actual)
Sprint-1	20	6 Days	24 Oct 2022	29 Oct 2022	20	29 Oct 2022
Sprint-2	20	6 Days	31 Oct 2022	05 Nov 2022	20	05 Nov 2022
Sprint-3	20	6 Days	07 Nov 2022	12 Nov 2022	20	12 Nov 2022
Sprint-4	20	6 Days	14 Nov 2022	19 Nov 2022	20	19 Nov 2022

6.3 Reports from JIRA

Velocity:

Imagine we have 10-day sprint duration, and the velocity of the team is 20 (points per sprint). Let's calculate the team's average velocity (AV) per iteration unit (story points per day).

$$\text{AV} = \frac{sprint\ duration}{velocity} = \frac{20}{10} = 2$$



7. CODING & SOLUTIONING

nutritionTracker/main.py

```
from flask import Flask,render_template,request,redirect
import os
import numpy as np
from tensorflow.keras.models import load_model
from tensorflow.keras.preprocessing import image
import requests


app=Flask(__name__,template_folder="templates")


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


@app.route('/')
def index():
```

```
    return redirect("/home",code=200)


@app.route('/home')
def home():
    return render_template("home.html")


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


@app.route('/predict',methods=['GET','POST'])
def predict():
    if request.method=='POST':
        f=request.files['file']
        basepath=os.path.dirname('__file__')
        filepath=os.path.join(basepath,'static',f.filename)
        f.save(filepath)



        img=image.load_img(filepath,target_size=(64,64))
        x=image.img_to_array(img)
        x=np.expand_dims(x,axis=0)

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

```
        print("prediction",pred)


index=["APPLES","BANANA","ORANGE","PINEAPPLE","WATERMELON
"]


        result=str(index[pred[0]])
        x=result
        result=nutrition(result)
        print(result)


                                                                    return
render_template("0.html",scase=(index[pred[0]]),showcase=(result),showcase1
=(f.filename))



def nutrition(index):


    url="https://calorieninjas.p.rapidapi.com/v1/nutrition"
    querystring={"query":index}


    headers={

"X-RapidAPI-Key":"228bc54e2bmsh125425366c0edcdp11af24jsn5f87cef4e48
e",
            "X-RapidAPI-Host": "calorieninjas.p.rapidapi.com"
            }
```

```
    response=requests.request("GET",url,headers=headers,params=querystring)
    print(response.json())
    return response.json()['items']
```

nutritionTracker/templates/base.html

```
<html>
    <head>
        <title>Nutrition Tracker</title>
        <style>
            a{
                text-decoration:none;
            }

            body{
                                                    background-image:
url("https://wallpapercave.com/wp/wp4696898.png") ;
            }
            .homepage{
                display: flex;
```

```
    flex-direction: column;
    font-family: 'Segoe UI', Tahoma, Geneva, Verdana, sans-serif;
}
.navbar{
    height:150px;
    display: flex;
    justify-content: space-between;
    align-items: center;
    margin:10px 10px 100px 10px;
    border-bottom: 1px solid black;
    background:#3e3e3e;
    color:white;
}

.nav-left{
    padding-left:20px;
    color:white;
}

.nav-left h1{
    color:white;
}

.nav-right{
```

```
    display: flex;
    width: 300px;
    justify-content: space-evenly;
    text-decoration: none;

}

.nav-right a{
    color: white;
}

.homeview{
    font-size: x-large;
    margin: 0 350px 0 350px;
    display: flex;
    flex-direction: column;
    justify-items: center;
}



.uploadview{
    flex-direction: column;
    padding: 0 200px 0 200px;
```

```
    justify-content: center;
    justify-items: center;
}

.formView{
    display: flex;

    flex-direction: column;
    justify-content: center;
    justify-items: center;
    align-items: center;
}


.homeview p{
    color: black;
    text-align: center;
}

.submit{
    border:none;
    color: #a0d2eb;
    background:#3e3e3e;
    border-radius:50px;
```

```
    height:40px;
    color:white;
    width:130px;
    box-shadow: 2px 0px 0px grey;
    margin-top: 20px;
}
.submit:hover{
    background-color: #a0d2eb ;
    color: #3e3e3e;
}


.inpFile{
    width: 0.1px;
    height: 0.1px;
    opacity: 0;
    overflow: hidden;
    z-index: -1;
    font-size: 10px;
}

.inpFile+label{
    border-radius: 10px;
    color: white;
```

```
        padding: 5px;
        background-color: black;
        display: inline-block;
        cursor: pointer;
    }

.ansView{
    padding: 0 100px 0 100px;
    display: flex;
    justify-content: space-evenly;
    color: white;

}

.image-cont{
    height: 500px;width: 500px;
}

.flex{
    display: flex;
    align-items: center;
}

.pr5{
```

```
            margin: 10px;
        }



        </style>
    </head>

    <body>
        <div class="homepage">
        <div class="navbar">
            <div class="nav-left">
                                <a  href={{  url_for('home')  }}><h1>Nutrition  Tracker
Demo</h1></a>
            </div>
            <div class="nav-right">
                <a href={{ url_for('home') }}><h3>Home</h3></a>
                <a href={{ url_for('image1') }}><h3>Classify</h3></a>
            </div>
        </div>
        {% block content %}
        {% endblock %}
        </div>
    </body>


</html>
```

nutritionTracker/templates/home.html

```
{% extends 'base.html' %}

{% block content %}


<div class="homeview">

    <p>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.</p>

</div>


{% endblock %}


{% extends 'base.html' %}
```

nutritionTracker/templates/image.html

```
{% block content %}
<div class="uploadview">
                <form class="formView" method="post" action="/predict" enctype="multipart/form-data">
      <h2>Upload your image</h2>
      <input id="choose-file" type="file" name="file" required>
      <input class="submit" type="Submit" value="Check and tell!">
   </form>
</div>
{% endblock %}
```

nutritionTracker/templates/0.html

```
{% extends 'base.html' %}

{% block content %}
<div class="ansView">
   <div class="ans-left">
      <h1>Image Uploaded:</h1>
       <img class="image-cont" src={{ url_for('static', filename=showcase1  ) }}
>
```

```
    </div>

    <div class="ans-right">
        <h1 style="color: black;">Nutritional Values: </h1>
        <div class="flex" style="color: black;">
            <h3>Food detected:</h3>
            <p class="pr5">{{ scase }}</p>
        </div>
        {% for k in showcase[0] %}
        <p><b style="color: black;">{{ k }}</b> : {{ showcase[0].get(k) }}</p>
        {% endfor %}
    </div>
</div>
{% endblock %}
```

TRAINING THE CNN MODEL:

```
import numpy as np
import tensorflow
from tensorflow.keras.models import Sequential
from tensorflow.keras import layers
from tensorflow.keras.layers import Dense,Flatten
```

```
from tensorflow.keras.layers import Conv2D,MaxPooling2D

from keras.preprocessing.image import ImageDataGenerator


train_datagen=ImageDataGenerator(rescale=1./255,shear_range=0.2,zoom_range=0.2
,horizontal_flip=True)

test_datagen=ImageDataGenerator(rescale=1./255)

x_train = train_datagen.flow_from_directory(r'TRAIN_SET',
                                            target_size=(64, 64),
                                            batch_size=5,
                                            color_mode='rgb',
                                            class_mode='sparse')


x_test = test_datagen.flow_from_directory(r'TEST_SET',
                                          target_size=(64, 64),
                                          batch_size=5,
                                          color_mode='rgb',
                                          class_mode='sparse')


print(x_train.class_indices)

from collections import Counter as ct

ct(x_train.labels)

classifier=Sequential()

classifier.add(Conv2D(32,(3,3),input_shape=(64,64,3),activation='relu'))

classifier.add(MaxPooling2D(pool_size=(2,2)))


classifier.add(Conv2D(32,(3,3),activation='relu'))
```

```
classifier.add(MaxPooling2D(pool_size=(2,2)))


classifier.add(Flatten())


classifier.add(Dense(units=128,activation='relu'))

classifier.add(Dense(units=5,activation='softmax'))

classifier.summary()

classifier.compile(optimizer='adam',loss='sparse_categorical_crossentropy',metrics=['a
ccuracy'])


classifier.fit_generator(x_train,
                    steps_per_epoch = len(x_train) ,
                    epochs = 20,
                    validation_data = x_test,
                    validation_steps = len(x_test) )
classifier.save('Nutrition-Analysis.h5')
```

TESTING THE CNN MODEL:

```
from tensorflow.keras.models import load_model

# from keras.preprocessing import image

import keras

import numpy as np
```

```
import glob
import matplotlib.pyplot as plt
model=load_model("Nutrition-Analysis.h5")
img=keras.utils.load_img(r"test_16.jpg",grayscale=False,target_size=(64,64))

x=keras.utils.img_to_array(img)
x=np.expand_dims(x,axis=0)

pred=model.predict(x)
pred
np.argmax(pred)
index=["APPLES","BANANAS","ORANGE","PINEAPPLE","WATERMELO
N"]
result=str(index[np.argmax(pred)])
result
fig=plt.figure(figsize=(5,5))
count=0
testing_imgs=glob.glob("*.jpg")
for i in testing_imgs:
    count+=1
    img=keras.utils.load_img(i,grayscale=False,target_size=(64,64))



    x=keras.utils.img_to_array(img)
    x=np.expand_dims(x,axis=0)
```

```
pred=model.predict(x)
print(pred)
# plt.imshow(img)
result=str(index[np.argmax(pred)])
print(result)
fig=plt.figure(figsize=(5,5))
plt.title(result)
plt.axis("off")
plt.imshow(img)
```

8. TESTING

8.1 Test Cases

S.No	Test Cases	Passed/ Failed
1.	Input image data	Passed
2.	Upload data	Passed
3.	Model integration	Passed
4.	Prediction	Passed
5.	Nutrition values	Passed

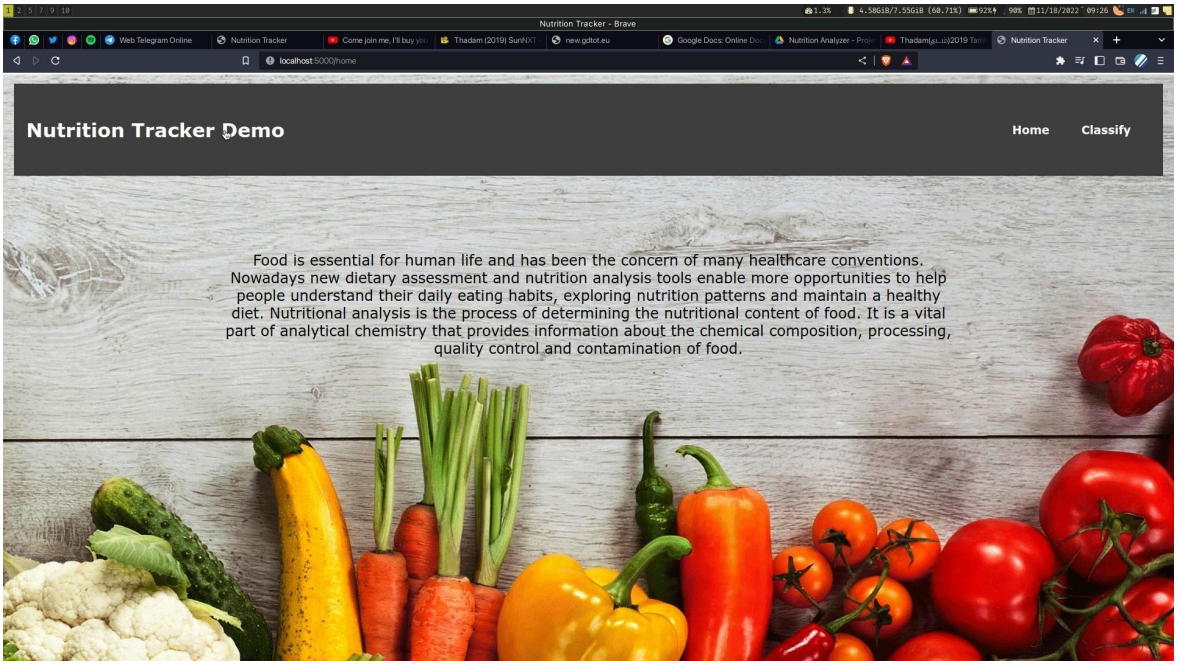
8.2 User Acceptance Testing

S.No	Test Cases	Yes/ No
1.	Keyword driven	Yes
2.	Responds in manually drafted rules	Yes
3.	Conversational Paradigm	Yes
4.	Learns from real interactions	No
5.	Training via historical data	No
6.	Has decision-making skills	No

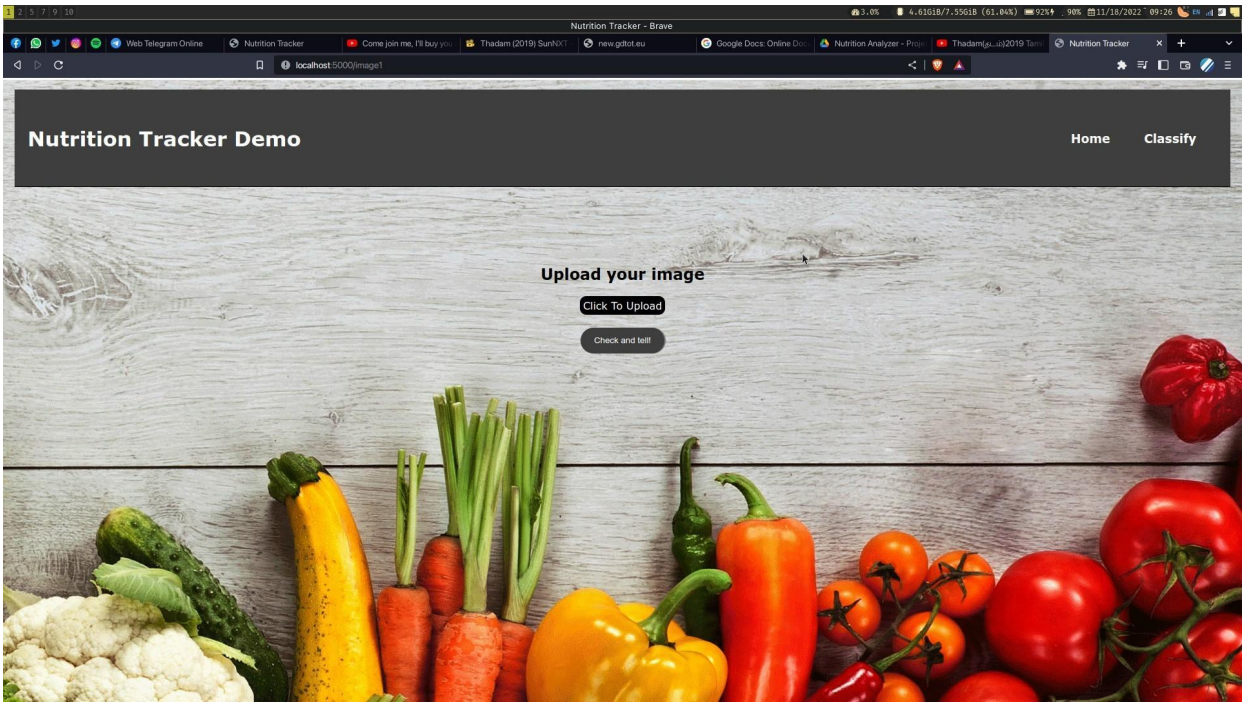
9. RESULTS

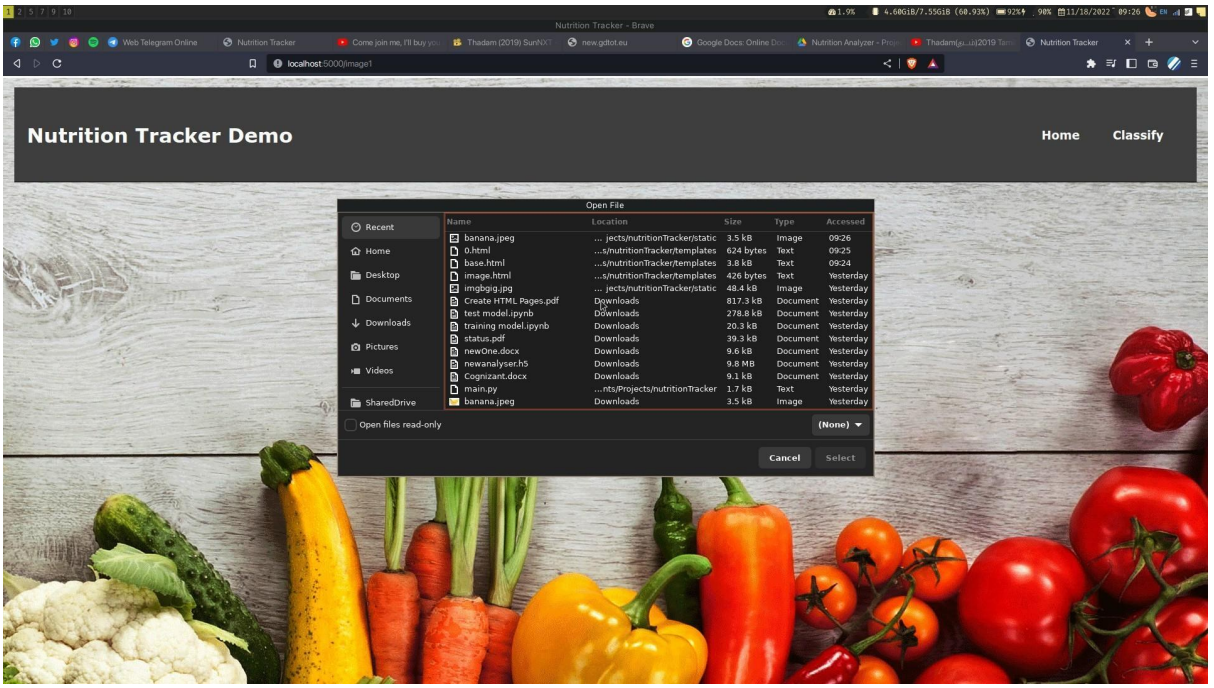
Screenshots:

- **Home page**

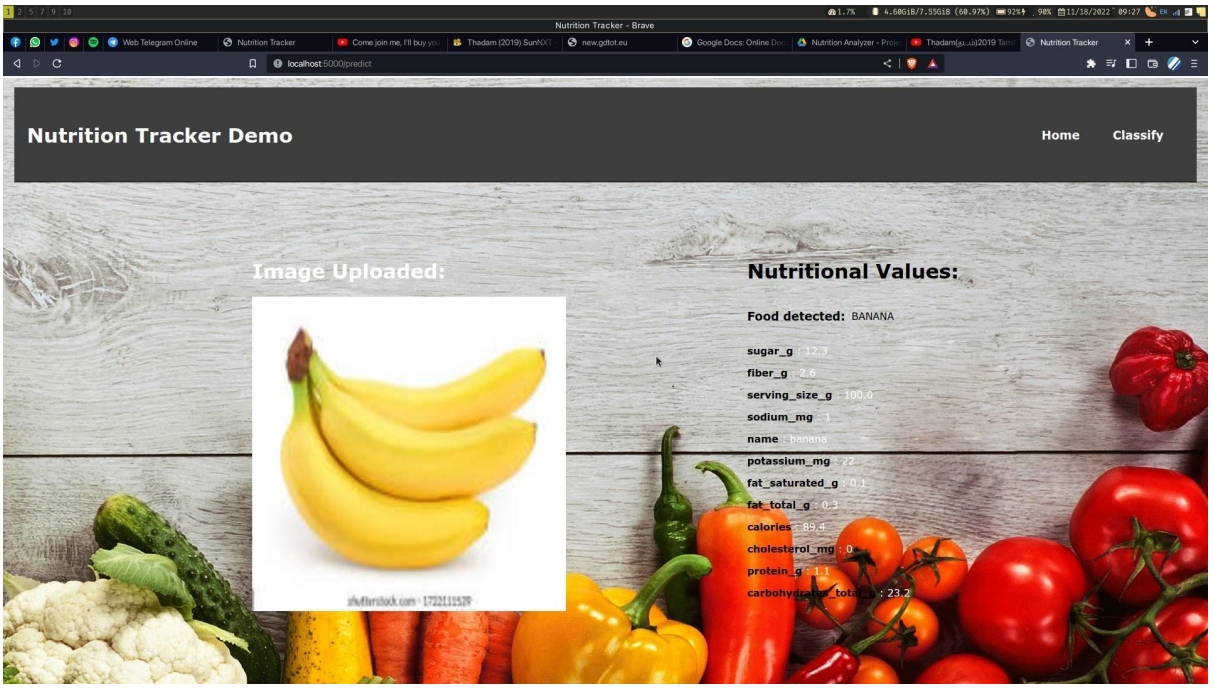


- **Uploading page**

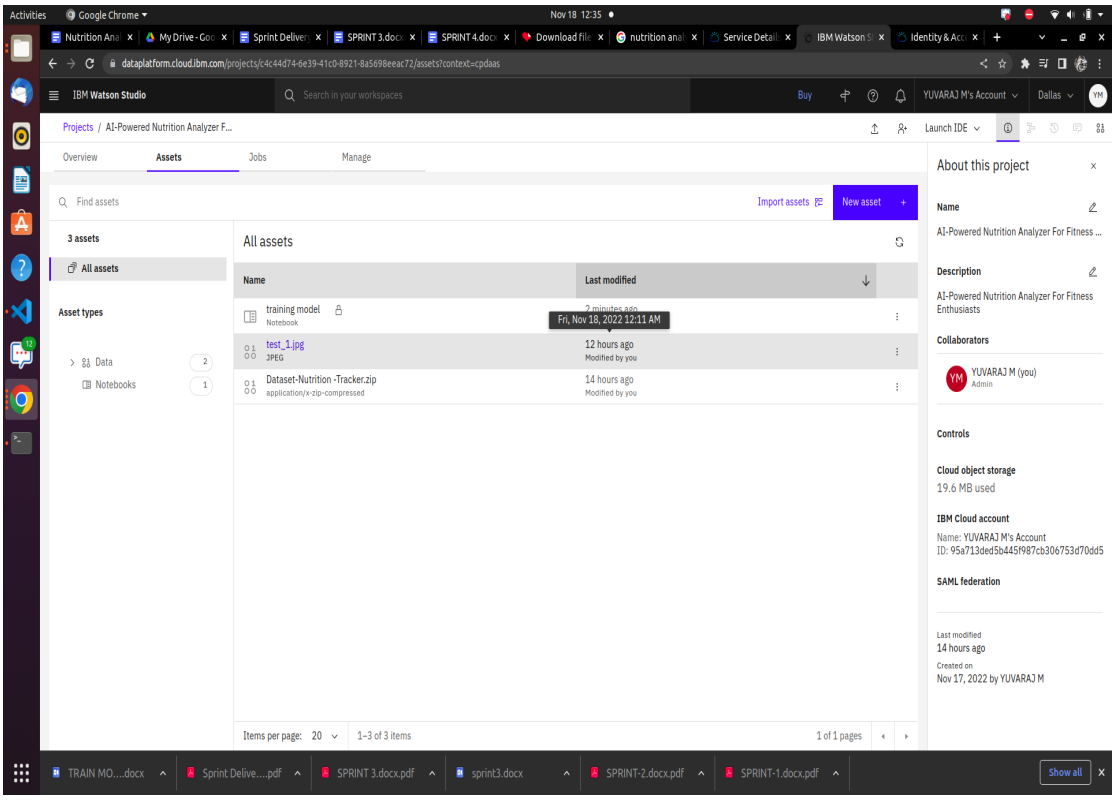


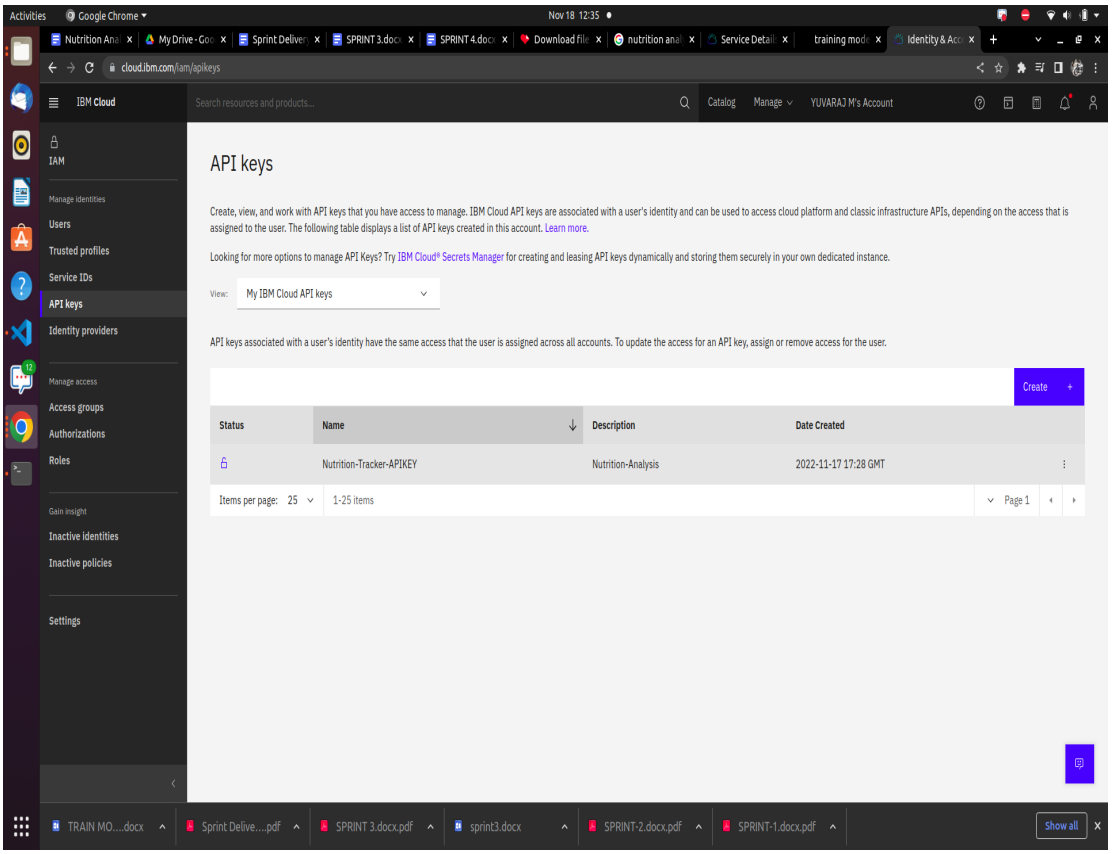


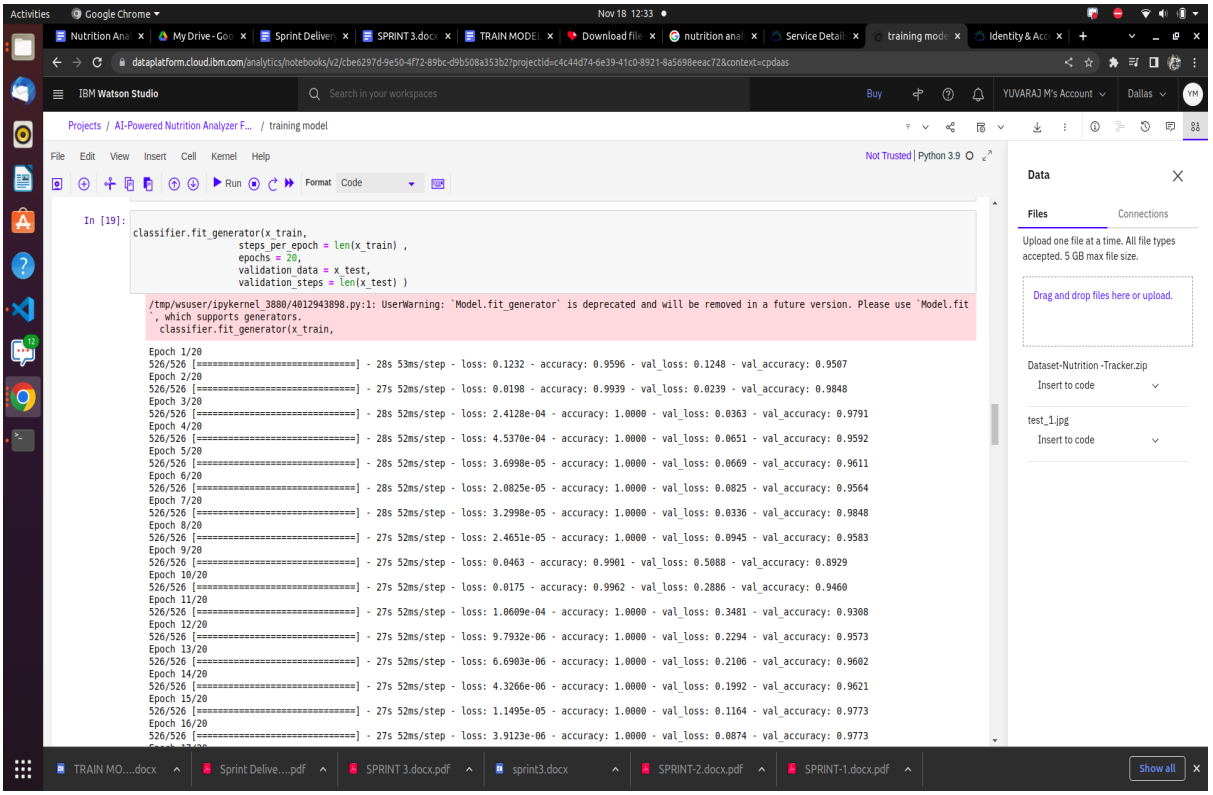
- **Model Prediction**



SOURCE CODE/IBM CLOUD :







Activities Google Chrome Nov 18 12:33

dataplatform.cloud.ibm.com/analytics/notebooks/v2/cbe6297d-9e50-4f72-89bc-d9b508a353b2?projectid=c4c44d74-6e39-41c0-8921-8a5698eeac72&context=cpdaas

IBM Watson Studio | Search in your workspaces | Buy | YUVARAJ M's Account | Dallas

Projects / AI-Powered Nutrition Analyzer F... / training model

File Edit View Insert Cell Kernel Help | Not Trusted | Python 3.9

Run | Format Code

In [19]:
```
classifier.fit_generator(x_train,
                         steps_per_epoch = len(x_train) ,
                         epochs = 20,
                         validation_data = x_test,
                         validation_steps = len(x_test) )
```

```
/tmp/wsuser/ipykernel_3880/4012943898.py:1: UserWarning: `Model.fit_generator` is deprecated and will be removed in a future version. Please use `Model.fit
`, which supports generators.
  classifier.fit_generator(x_train,
Epoch 1/20
526/526 [==============================] - 28s 53ms/step - loss: 0.1232 - accuracy: 0.9596 - val_loss: 0.1248 - val_accuracy: 0.9507
Epoch 2/20
526/526 [==============================] - 27s 52ms/step - loss: 0.0198 - accuracy: 0.9939 - val_loss: 0.0239 - val_accuracy: 0.9848
Epoch 3/20
526/526 [==============================] - 28s 52ms/step - loss: 2.4128e-04 - accuracy: 1.0000 - val_loss: 0.0363 - val_accuracy: 0.9791
Epoch 4/20
526/526 [==============================] - 28s 52ms/step - loss: 4.5370e-04 - accuracy: 1.0000 - val_loss: 0.0651 - val_accuracy: 0.9592
Epoch 5/20
526/526 [==============================] - 28s 52ms/step - loss: 3.6998e-05 - accuracy: 1.0000 - val_loss: 0.0669 - val_accuracy: 0.9611
Epoch 6/20
526/526 [==============================] - 28s 52ms/step - loss: 2.0825e-05 - accuracy: 1.0000 - val_loss: 0.0825 - val_accuracy: 0.9564
Epoch 7/20
526/526 [==============================] - 28s 52ms/step - loss: 3.2998e-05 - accuracy: 1.0000 - val_loss: 0.0336 - val_accuracy: 0.9848
Epoch 8/20
526/526 [==============================] - 27s 52ms/step - loss: 2.4651e-05 - accuracy: 1.0000 - val_loss: 0.0945 - val_accuracy: 0.9583
Epoch 9/20
526/526 [==============================] - 27s 52ms/step - loss: 0.0463 - accuracy: 0.9901 - val_loss: 0.5088 - val_accuracy: 0.8929
Epoch 10/20
526/526 [==============================] - 27s 52ms/step - loss: 0.0175 - accuracy: 0.9962 - val_loss: 0.2886 - val_accuracy: 0.9460
Epoch 11/20
526/526 [==============================] - 27s 52ms/step - loss: 1.0609e-04 - accuracy: 1.0000 - val_loss: 0.3481 - val_accuracy: 0.9308
Epoch 12/20
526/526 [==============================] - 27s 52ms/step - loss: 9.7932e-06 - accuracy: 1.0000 - val_loss: 0.2294 - val_accuracy: 0.9573
Epoch 13/20
526/526 [==============================] - 27s 52ms/step - loss: 6.6903e-06 - accuracy: 1.0000 - val_loss: 0.2106 - val_accuracy: 0.9602
Epoch 14/20
526/526 [==============================] - 27s 52ms/step - loss: 4.3266e-06 - accuracy: 1.0000 - val_loss: 0.1992 - val_accuracy: 0.9621
Epoch 15/20
526/526 [==============================] - 27s 52ms/step - loss: 1.1495e-05 - accuracy: 1.0000 - val_loss: 0.1164 - val_accuracy: 0.9773
Epoch 16/20
526/526 [==============================] - 27s 52ms/step - loss: 3.9123e-06 - accuracy: 1.0000 - val_loss: 0.0874 - val_accuracy: 0.9773
```

Data | Files | Connections

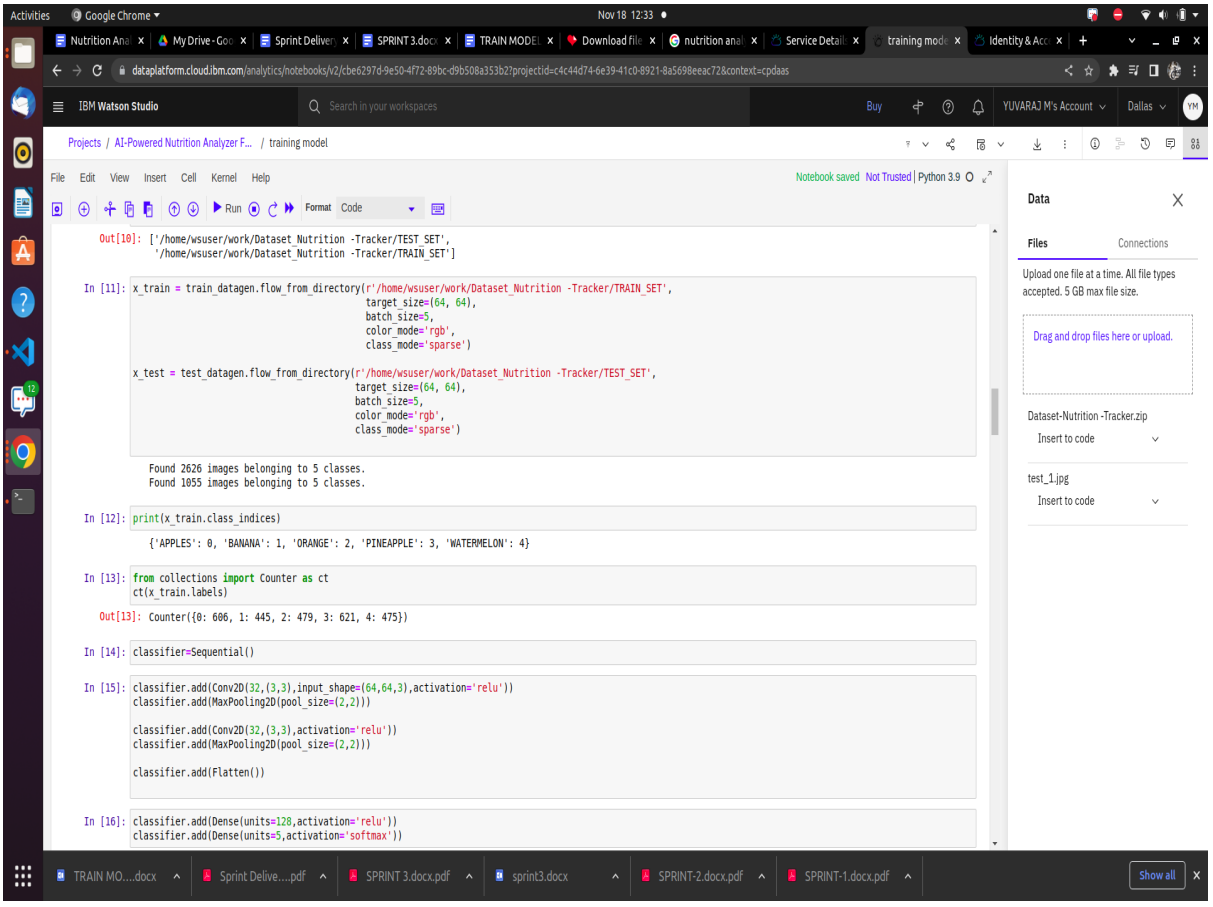
Upload one file at a time. All file types accepted. 5 GB max file size.

Drag and drop files here or upload.

Dataset-Nutrition -Tracker.zip — Insert to code

test_1.jpg — Insert to code

TRAIN MO....docx | Sprint Delive....pdf | SPRINT 3.docx.pdf | sprint3.docx | SPRINT-2.docx.pdf | SPRINT-1.docx.pdf | Show all



Activities Google Chrome Nov 18 12:33

dataplatform.cloud.ibm.com/analytics/notebooks/v2/cbe6297d-9e50-4f72-89bc-d9b508a353b2?projectid=c4c44d74-6e39-41c0-8921-8a5698eeac72&context=cpdaas

IBM Watson Studio | Search in your workspaces | Buy | YUVARAJ M's Account | Dallas

Projects / AI-Powered Nutrition Analyzer F... / training model

File Edit View Insert Cell Kernel Help | Notebook saved Not Trusted | Python 3.9

Run | Format Code

Out[10]:
```
['/home/wsuser/work/Dataset_Nutrition -Tracker/TEST_SET',
 '/home/wsuser/work/Dataset_Nutrition -Tracker/TRAIN_SET']
```

In [11]:
```
x_train = train_datagen.flow_from_directory(r'/home/wsuser/work/Dataset_Nutrition -Tracker/TRAIN_SET',
                                            target_size=(64, 64),
                                            batch_size=5,
                                            color_mode='rgb',
                                            class_mode='sparse')

x_test = test_datagen.flow_from_directory(r'/home/wsuser/work/Dataset_Nutrition -Tracker/TEST_SET',
                                          target_size=(64, 64),
                                          batch_size=5,
                                          color_mode='rgb',
                                          class_mode='sparse')
```

```
Found 2626 images belonging to 5 classes.
Found 1055 images belonging to 5 classes.
```

In [12]:
```
print(x_train.class_indices)
```

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

In [13]:
```
from collections import Counter as ct
ct(x_train.labels)
```

Out[13]:
```
Counter({0: 606, 1: 445, 2: 479, 3: 621, 4: 475})
```

In [14]:
```
classifier=Sequential()
```

In [15]:
```
classifier.add(Conv2D(32,(3,3),input_shape=(64,64,3),activation='relu'))
classifier.add(MaxPooling2D(pool_size=(2,2)))

classifier.add(Conv2D(32,(3,3),activation='relu'))
classifier.add(MaxPooling2D(pool_size=(2,2)))

classifier.add(Flatten())
```

In [16]:
```
classifier.add(Dense(units=128,activation='relu'))
classifier.add(Dense(units=5,activation='softmax'))
```

Data | Files | Connections

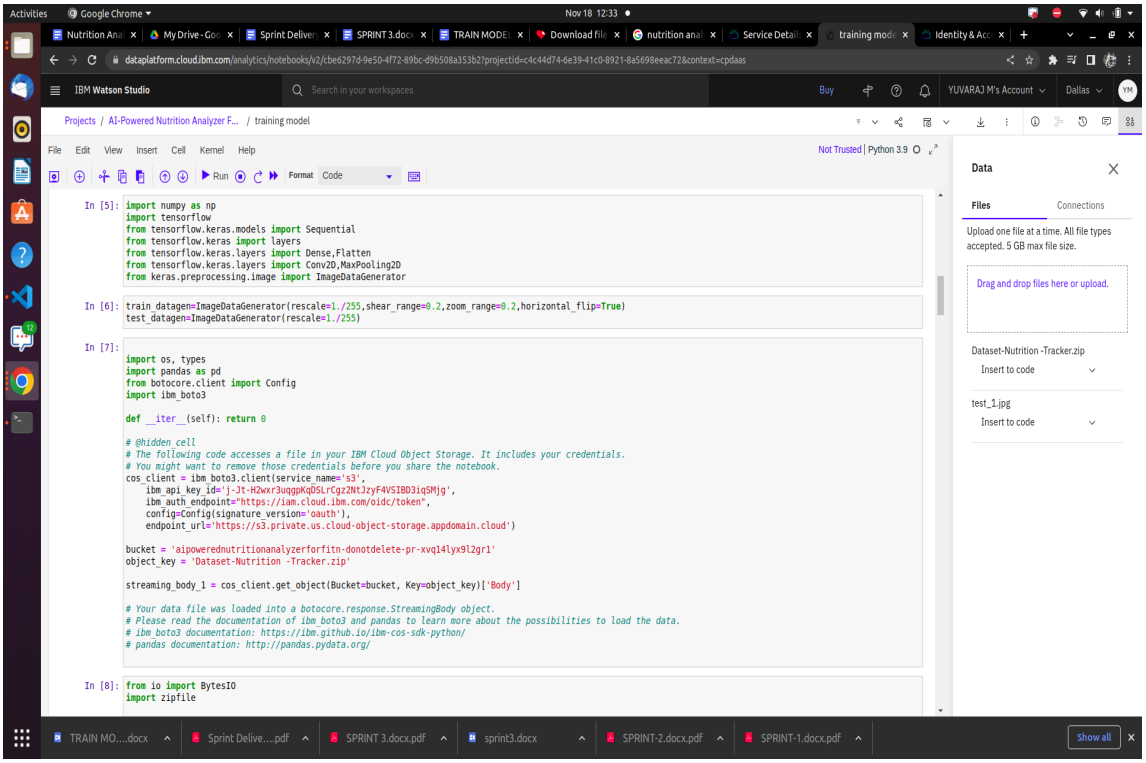
Upload one file at a time. All file types accepted. 5 GB max file size.

Drag and drop files here or upload.

Dataset-Nutrition -Tracker.zip — Insert to code

test_1.jpg — Insert to code

TRAIN MO....docx | Sprint Delive....pdf | SPRINT 3.docx.pdf | sprint3.docx | SPRINT-2.docx.pdf | SPRINT-1.docx.pdf | Show all

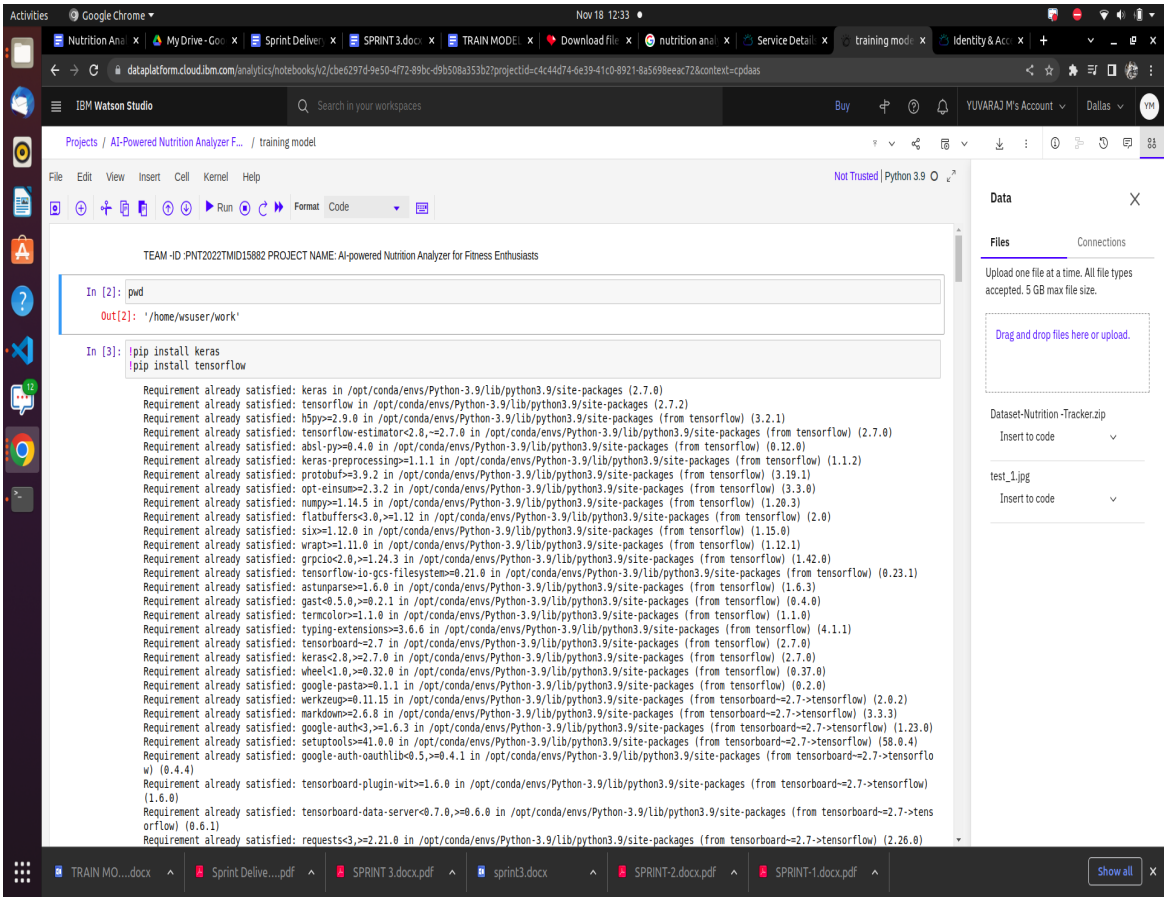


In [5]:
```
import numpy as np
import tensorflow
from tensorflow.keras.models import Sequential
from tensorflow.keras import layers
from tensorflow.keras.layers import Dense,Flatten
from tensorflow.keras.layers import Conv2D,MaxPooling2D
from keras.preprocessing.image import ImageDataGenerator
```

In [6]:
```
train_datagen=ImageDataGenerator(rescale=1./255,shear_range=0.2,zoom_range=0.2,horizontal_flip=True)
test_datagen=ImageDataGenerator(rescale=1./255)
```

In [7]:
```
import os, types
import pandas as pd
from botocore.client import Config
import ibm_boto3

def __iter__(self): return 0

# @hidden_cell
# The following code accesses a file in your IBM Cloud Object Storage. It includes your credentials.
# You might want to remove those credentials before you share the notebook.
cos_client = ibm_boto3.client(service_name='s3',
    ibm_api_key_id='j-Jt-H2wxr3uqgpKqDSLrCgz2NtJzyF4VSIBD3iqSMjg',
    ibm_auth_endpoint="https://iam.cloud.ibm.com/oidc/token",
    config=Config(signature_version='oauth'),
    endpoint_url='https://s3.private.us.cloud-object-storage.appdomain.cloud')

bucket = 'aipoweredrutritionanalyzerforfitn-donotdelete-pr-xvq14lyx9l2gr1'
object_key = 'Dataset-Nutrition -Tracker.zip'

streaming_body_1 = cos_client.get_object(Bucket=bucket, Key=object_key)['Body']

# Your data file was loaded into a botocore.response.StreamingBody object.
# Please read the documentation of ibm_boto3 and pandas to learn more about the possibilities to load the data.
# ibm_boto3 documentation: https://ibm.github.io/ibm-cos-sdk-python/
# pandas documentation: http://pandas.pydata.org/
```

In [8]:
```
from io import BytesIO
import zipfile
```

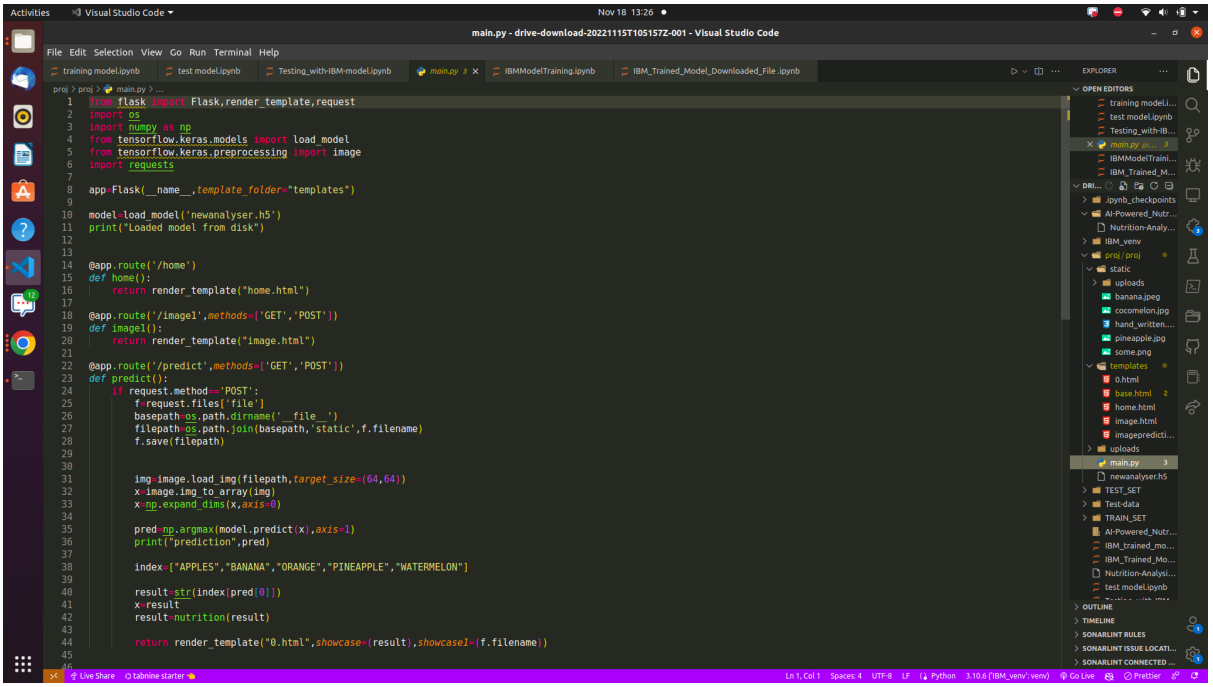


TEAM -ID :PNT2022TMID15882 PROJECT NAME: AI-powered Nutrition Analyzer for Fitness Enthusiasts

In [2]:
```
pwd
```
Out[2]: '/home/wsuser/work'

In [3]:
```
!pip install keras
!pip install tensorflow
```

```
Requirement already satisfied: keras in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (2.7.0)
Requirement already satisfied: tensorflow in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (2.7.2)
Requirement already satisfied: h5py>=2.9.0 in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from tensorflow) (3.2.1)
Requirement already satisfied: tensorflow-estimator<2.8,~=2.7.0 in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from tensorflow) (2.7.0)
Requirement already satisfied: absl-py>=0.4.0 in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from tensorflow) (0.12.0)
Requirement already satisfied: keras-preprocessing>=1.1.1 in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from tensorflow) (1.1.2)
Requirement already satisfied: protobuf>=3.9.2 in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from tensorflow) (3.19.1)
Requirement already satisfied: opt-einsum>=2.3.2 in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from tensorflow) (3.3.0)
Requirement already satisfied: numpy>=1.14.5 in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from tensorflow) (1.20.3)
Requirement already satisfied: flatbuffers<3.0,>=1.12 in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from tensorflow) (2.0)
Requirement already satisfied: six>=1.12.0 in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from tensorflow) (1.15.0)
Requirement already satisfied: wrapt>=1.11.0 in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from tensorflow) (1.12.1)
Requirement already satisfied: grpcio<2.0,>=1.24.3 in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from tensorflow) (1.42.0)
Requirement already satisfied: tensorflow-io-gcs-filesystem>=0.21.0 in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from tensorflow) (0.23.1)
Requirement already satisfied: astunparse>=1.6.0 in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from tensorflow) (1.6.3)
Requirement already satisfied: gast<0.5.0,>=0.2.1 in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from tensorflow) (0.4.0)
Requirement already satisfied: termcolor>=1.1.0 in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from tensorflow) (1.1.0)
Requirement already satisfied: typing-extensions>=3.6.6 in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from tensorflow) (4.1.1)
Requirement already satisfied: tensorboard~=2.7 in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from tensorflow) (2.7.0)
Requirement already satisfied: keras<2.8,>=2.7.0 in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from tensorflow) (2.7.0)
Requirement already satisfied: wheel<1.0,>=0.32.0 in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from tensorflow) (0.37.0)
Requirement already satisfied: google-pasta>=0.1.1 in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from tensorflow) (0.2.0)
Requirement already satisfied: werkzeug>=0.11.15 in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from tensorboard~=2.7->tensorflow) (2.0.2)
Requirement already satisfied: markdown>=2.6.8 in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from tensorboard~=2.7->tensorflow) (3.3.3)
Requirement already satisfied: google-auth<3,>=1.6.3 in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from tensorboard~=2.7->tensorflow) (1.23.0)
Requirement already satisfied: setuptools>=41.0.0 in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from tensorboard~=2.7->tensorflow) (58.0.4)
Requirement already satisfied: google-auth-oauthlib<0.5,>=0.4.1 in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from tensorboard~=2.7->tensorflo
w) (0.4.4)
Requirement already satisfied: tensorboard-plugin-wit>=1.6.0 in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from tensorboard~=2.7->tensorflow)
(1.6.0)
Requirement already satisfied: tensorboard-data-server<0.7.0,>=0.6.0 in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from tensorboard~=2.7->tens
orflow) (0.6.1)
Requirement already satisfied: requests<3,>=2.21.0 in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from tensorboard~=2.7->tensorflow) (2.26.0)
```



10.ADVANTAGES & DISADVANTAGES

Advantages:

- Early detection of health problems.
- Easy to know about the nutrition values.
- User friendly and gives accurate suggestions.

Disadvantages:

- Requires training the system with large dataset.
- Works only on the pretrained Images.
- User may not have time to upload the image before eating.

11. CONCLUSION

Hence a system that takes in images as user input, analyses those and identifies the Nutritional content , and gives all the ingredients present in the image with its nutritional content.

12. FUTURE SCOPE

The system must be trained with numerous images of food and suggest some healthy recipes for the same . Help users to connect with other users and share their feedback .

13. APPENDIX

15.GitHub & Project Demo

GitHub : https://github.com/IBM-EPBL/IBM-Project-26035-1659981448

Project Demo Link :