

AI-Powered Nutrition Analyzer For Fitness Enthusiasts

TEAM ID:PNT2022TMID45226

PROJECT REPORT

1.INTRODUCTION

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

1.1ProjectOverview:

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 to the trained model. The model analyses the image and detects the nutrition based on the fruits like (Sugar, Fibre, Protein, Calories, etc.). Food is comprised of macronutrients including protein, carbohydrate and fat that not only offer calories to fuel the body and give it energy but play specific roles in maintaining health.

1.2 Purpose:

This allows the users to keep track of their diet and exercise regime, take expert advice and connect to other fitness enthusiasts thus equipping them to maintain a healthy lifestyle. The system plans to offer its customer and fitness enthusiasts many beauty tips options that can help them reach their goals. The importance of functional food is already well established as it has become an integral part in treating various ailments, as well as improving overall health and well-being. The biologically active components in these foods are considered to be responsible for the therapeutic effects.

Nutrition is a critical part of health and development. Better nutrition is related to improved infant, child and maternal health. Food is comprised of macronutrients including protein, carbohydrate and fat that not only offer calories to fuel the body .

2.LITERATURE SURVEY

The system will analyze the nutrition ingredients based on the recognition result and generated a dietary assesment report by calculating the amount of calorie,fat,carbohידrate and protein.Nutrition is vital to the growth Nutritional analysis guarantees that the meal meets the appropriate vitamin and mineralrequirements, and the examination of nutrition in food aid inunderstanding the fat proportion, carbohydrate dilution, proteins, fiber,sugar, and so on. Another thing to keep in mind is not to exceed our dailycalorie requirements. If this limit is surpassed, we may become fat.

Artificial Intelligence Nutrition App. As the name implies, the app provides nutrition-based analytics and data to its customers and is quickly becoming a prominent platform for offering AI fitness services. It deploys predictive analysis for personalized data compilation using mathematical and natural language processing (NLP) models. Furthermore, it shares nutrition-related data with its partners via SDK and API integration toimprove its services and product offerings. It Is an Israel-based firm createdin 2011 that allows pregnant women.This software collaborated with IBM’s natural language capability to provide 24-hour assistance anddietaryrecommendations The Ultimate Workout at Home SolutionThis fitness AI software is designed with personalized training regimens foreach individual. It began as “gym only software,” but has now improved itssystem to satisfy “at home fitness” expectations. FitnessAI says that their algorithm has been trained on over 5.9 million exercises, allowing it to“outperform any human fitness teacher.” Furthermore, it analyzed almost10 million sets, weights, and reps from about 30000 expert gym-goers and weightlifters during a three-year period MyFitnessPal app creates a daily food diary for you by recogonizing the food from photos you shoot. It is supposed to be as simple as that. You take a picture.

2.1 Existing problem:

Neutrino delivers nutrition-based data services and analytics to its users and wants to turn into a leading source of the nutrition-related platform. The platform employs NLP and mathematical models from the optimization theory as well as predictive analysis to enable individualized data compilation. The application relies on Artificial Intelligence to produce custom data related to smart calorie counter powered by AI. Their artificial intelligence learns an individual's tastes, preferences, and body type. All of this is packaged in a comprehensive nutrition and activity tracker.



2.2 Problem Statement Definition:

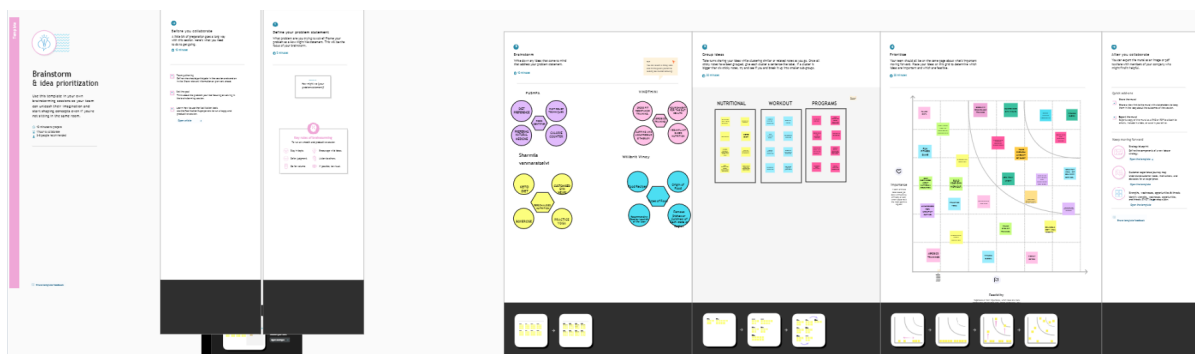
The main aim of the project is to building a model which is used for classifying the fruit depends on the different characteristics like colour, shape, texture etc. Here the user can capture the images of different fruits and then the image will be sent to the trained model. The model analyses the image and detects the nutrition based on the fruits like (Sugar, Fibre, Protein, Calories, etc.)

3. IDEATION & PROPOSED SOLUTION

3.1 Empathy Map Canvas:



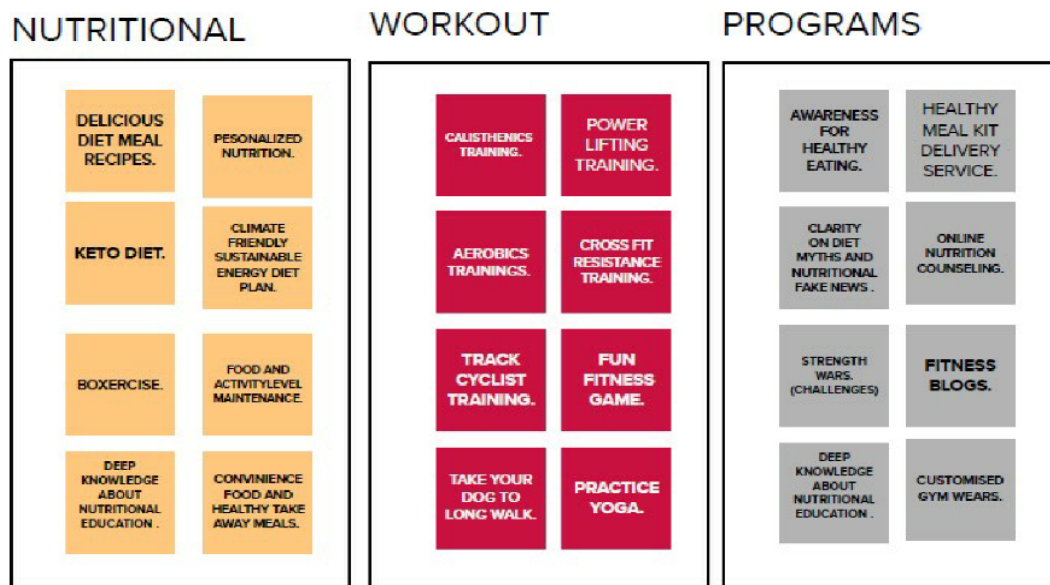
3.2 IDEATION AND BRAINSTORMING



3.3 PROPOSED SOLUTION:

S.NO	PARAMETER	DESCRIPTION
1.	Problem Statement (Problem to be solved)	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.
2.	Idea / Solution description	The idea of the project is to building a model which is used for classifying the fruit depends on the different characteristics like color, shape, texture etc.
3.	Novelty / Uniqueness	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.)
4.	Social Impact / Customer Satisfaction	This project is very helpful to People. Everyone Maintaining their own diet, to manage the time
5.	Business Model (Revenue Model)	By using this system, the users can predict and analyze the picture of the fruits and foods. In which it results to the visualizing the description of the foods taken as input.
6.	Scalability of the Solution	By implementing this system, the people

		can efficiently and effectively to gain knowledge about the fitness.They want and they wish to use at anytime. This system can also be integrated with the future technologies.
--	--	---

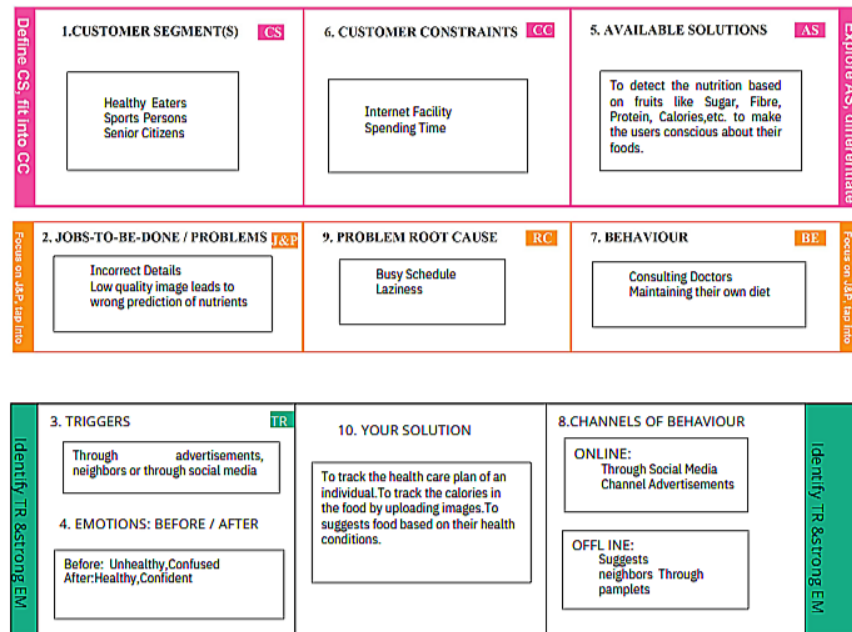


3.4 PROBLEM SOLUTION FIT:

The Problem-Solution Fit simply means that you have found a problem with your customer and that the solution you have realized for it actually solves the customer's problem. It helps entrepreneurs, marketers and corporate innovators identify behavioral patterns.

PURPOSE:

- Solve complex problems in a way that fits the state of your customers.
- Succeed faster and increase your solution adoption by tapping into existing mediums and channels of behavior.
- Sharpen your communication and marketing strategy with the right triggers and messaging.
- Increase touch-points with your company by finding the right problem-behavior fit.



4. REQUIREMENT ANALYSIS

4.1 Functional requirement :

- It will generate the diet plan as well as monitor the user's health to classify the category of the disease and to create the diet plan. It will also reduce the cost of consulting the person nutritionist.
- The task of food detection/classification is not easy as it seems. All possible options related to the given Image.
- Image classification, object detection, segmentation, face recognition.
- Classification of crystal structure using a convolutional neural network
- Nutrition is vital to the growth of the human body. Nutritional analysis guarantees that the meal meets the appropriate vitamin and mineral requirements, and the examination of nutrition in food aids in understanding the fat proportion, carbohydrate dilution, proteins, fiber, sugar, and so on. Another thing to keep in mind is not to exceed our daily calorie requirements.
- Here the user can capture the images of different fruits and then the image will be sent to the trained model. The model analyzes the image and detects the nutrition based on the fruits like (Sugar, Fiber, Protein, Calories, etc.) It began as "gym only software," but has now improved its system to satisfy "at home fitness" expectations
- You take a picture, dial in data such as whether you are eating breakfast or lunch and add a quick text label, and the app estimates the calorie content.
- This software collaborated with IBM's natural language capability to provide 24-hour

assistance and dietary recommendations

FR.NO	FUNCTIONAL REQUIRMENT	SUB REQUIREMENT(story/subtask)
Fr-1	User Registration	Registration through form,registration through Gmail
Fr-2	User confirmation	Confirmation via OTP,confirmation via Email
Fr-3	Capturing image	Capture the image of the food or fruits and check the parameter of the image.
Fr-4	Image processing	Upload the image for the prediction of the nutrition in food or fruits.
Fr-5	Image identification	Identify the fruit and predict the nutrition.
Fr-6	Image description	Suggesting the best fruit or food for fitness training days.

- The comparison of the proposed model with the conventional models shows that the results of this model are exceptionally good and promising to use in real-world applications.
- This sort of higher accuracy and precision will work to boost the machine's general efficiency in fruitrecognition more appropriately.
- Ageneric model for the dietary protein requirement(as with any nutrient) defines the requirement in terms of the needs of the organism,i.e. metabolic demands, and the dietary amount which will satisfy those needs, i.e. efficiency of utilization, thus: dietary requirement = metabolic demand/efficiency .

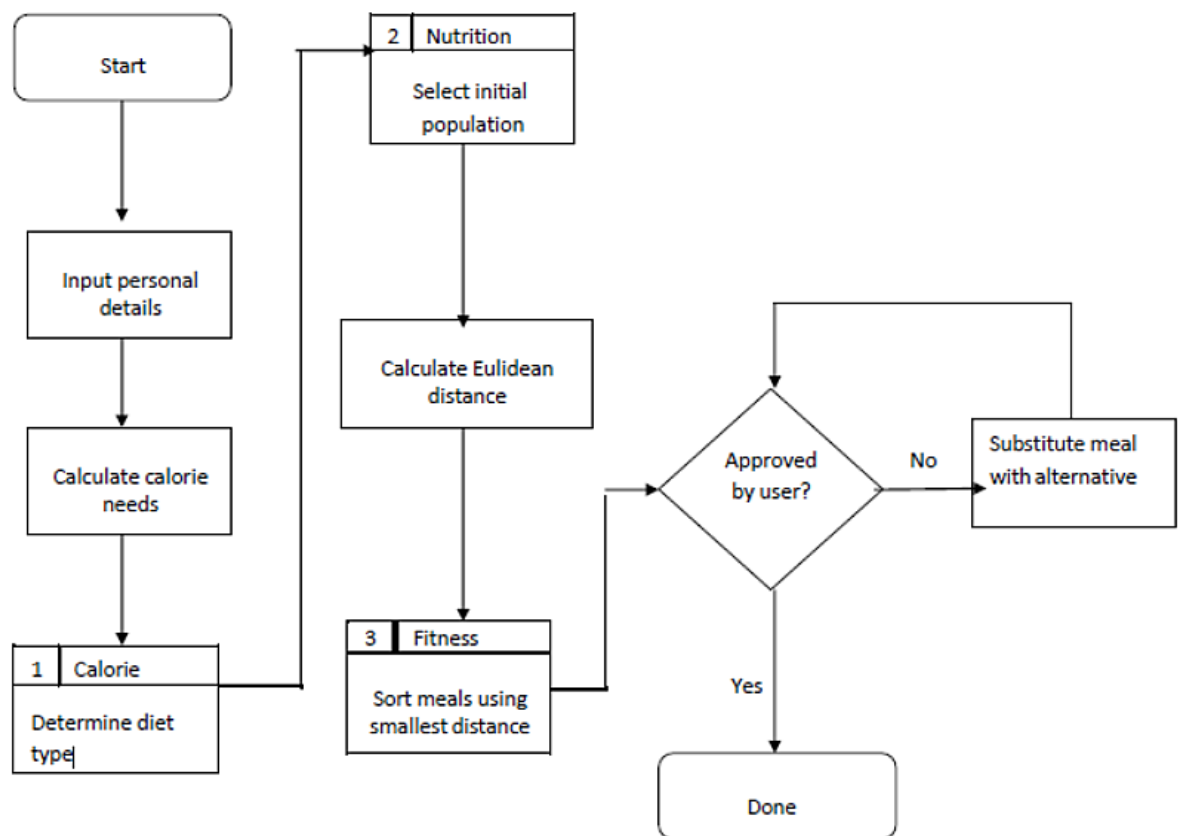
NON FUNCTIONAL REQUIREMENT

Nfr.no	Non functional requirement	description
nfr-1	Usability	Datasets of all the fruits is used to detecting the nutrtnon that present in the fruits.
nfr-2	Security	The information belongs to the user and fruits are secured highly.
nfr-3	Reliability	The fruit picture quality is important for the predicting the nutrition in the fruit .

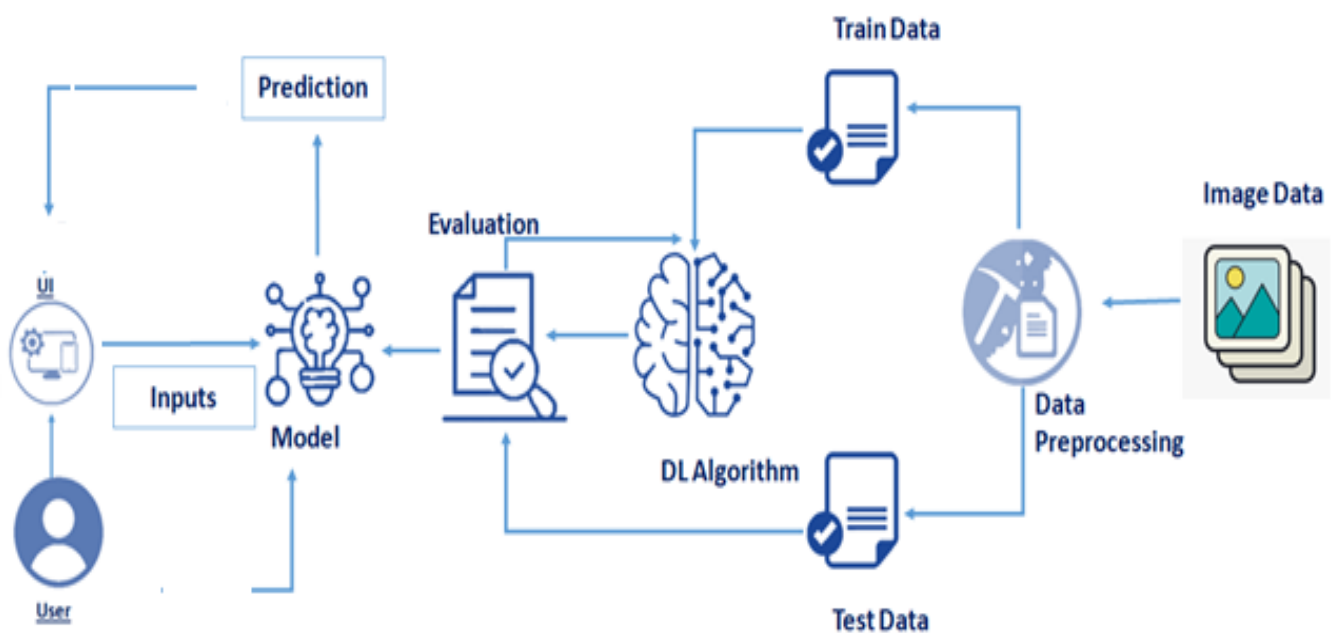
nfr-4	Performance	The performance is based on the quality of the fruit used in nutrition prediction.
nfr-5	Availability	It is available for all user to predict nutrition in the fruit.
nfr-6	Scalability	Increasing the prediction of nutrition in the leaf.

5.PROJECTDESIGN

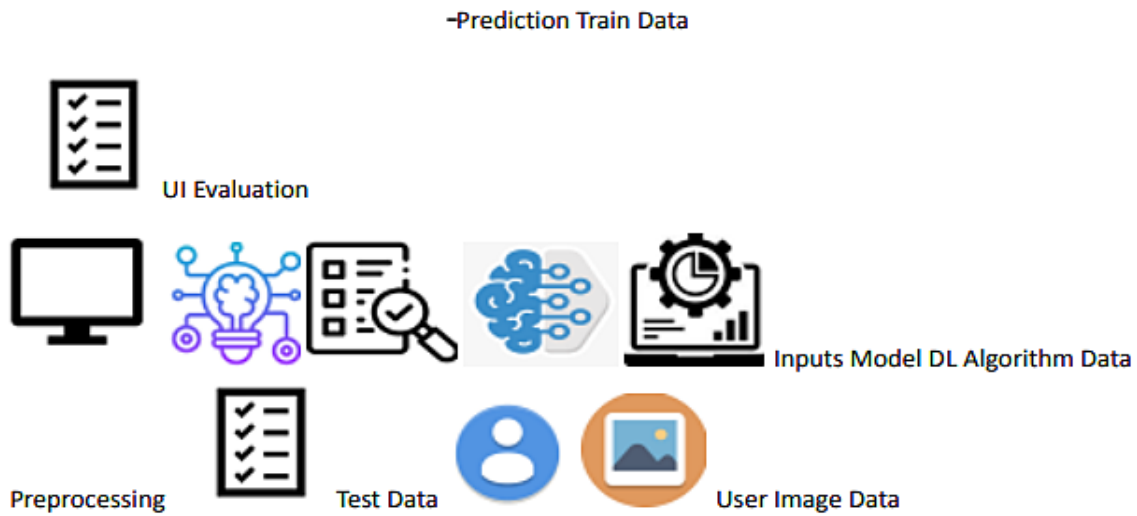
5.1 Data Flow Diagrams:



5.2 SOLUTION & TECHNICAL ARCHITECTURE



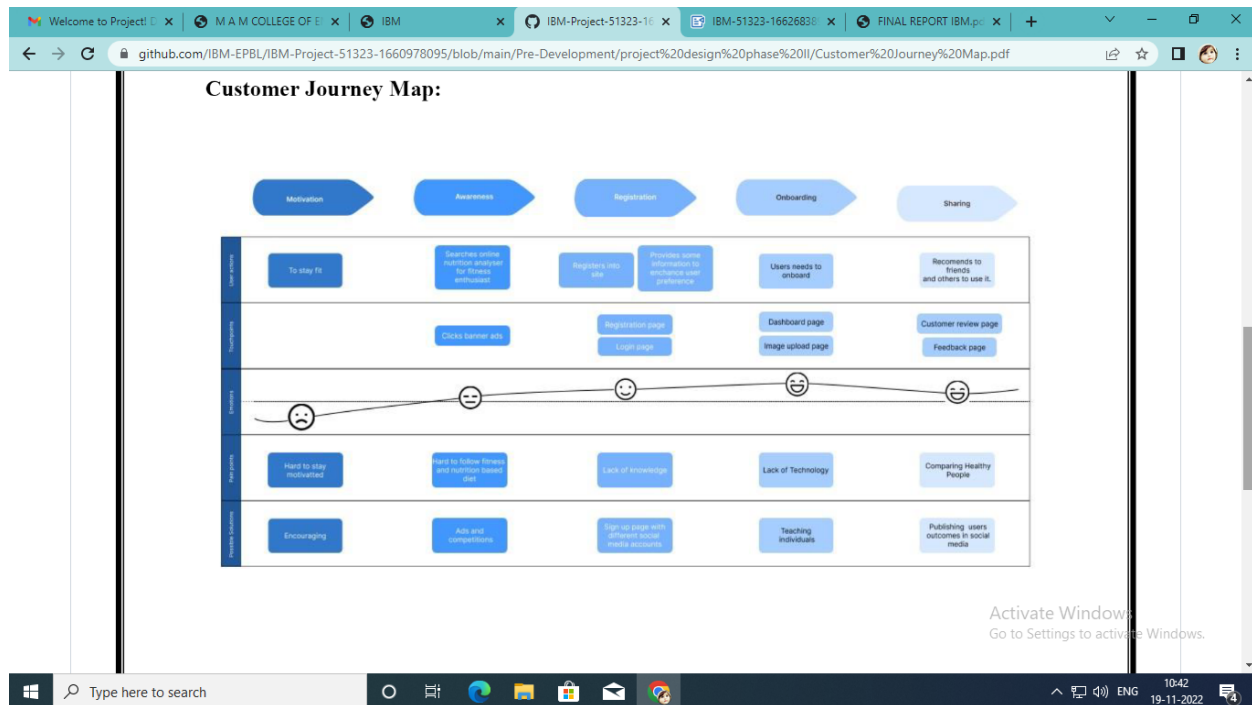
a) Technical architecture



b)Solution architecture

5.3 User stories

A user story is a small, self-contained unit of development work designed to accomplish a specific goal within a product. A user story is usually written from the user's perspective and follows the format: "As [a user persona], I want [to perform this action] so that [I can accomplish this goal]."



6.PROJECT PLANNING & SCHEDULING

6.1 SPRINT PLANNING & ESTIMATION

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-1	Data collection	USN-1	Dataset - Collecting images of food itemsapples , banana, orange, pineapple, watermelon for analysis	5	High	Sharmila Pushpa Vanmaraiselvi Vinothini Willbrit Vincy
Sprint-1	Image Preprocessing	USN-2	Image data augmentation - Increasing theamount of data by generating new data points from existing data	4	Medium	Sharmila Pushpa Vanmaraiselvi Vinothini Willbrit Vincy

Sprint-1	Image Preprocessing	USN-3	Image Data Generator Class - Used for getting the input of the original data	4	Medium	Sharmila Pushpa Vanmaraiselvi Vinothini Willbrit Vincy
Sprint-1	Image Preprocessing	USN-4	Applying image data generator functionality to train set and test set	4	Medium	Sharmila Pushpa Vanmaraiselvi Vinothini Willbrit Vincy
Sprint-2	Modeling Phase	USN-5	Defining the model architecture - Building the model using deep learning approach and adding CNN layers	4	High	Sharmila Pushpa Vanmaraiselvi Vinothini Willbrit Vincy
Sprint-2	Modeling Phase	USN-6	Training , saving, testing and Initializing the model	5	High	Sharmila Pushpa Vanmaraiselvi Vinothini Willbrit Vincy
Sprint-2	Modeling Phase	USN-7	Adding Dense layer of model	4	High	Sharmila Pushpa Vanmaraiselvi Vinothini Willbrit Vincy
Sprint-2	Modeling Phase	USN-8	Configure the model	4	High	Sharmila Pushpa Vanmaraiselvi Vinothini Willbrit Vincy
Sprint-3	Development phase	USN-9	Project development delivery of sprint 1	3	Medium	Sharmila Pushpa Vanmaraiselvi Vinothini Willbrit Vincy
Sprint-3	Development	USN-10	Project	2	Low	Sharmila

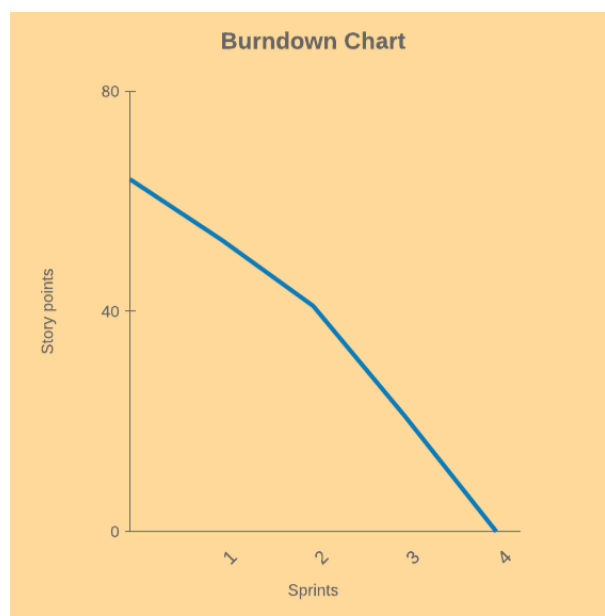
	phase		developmentdelivery of sprint 2			Pushpa Vanmaraiselvi Vinothini Willbrit Vincy
Sprint-3	Development phase	USN-11	Project developmentdelivery of sprint 3	2	Low	Sharmila Pushpa Vanmaraiselvi Vinothini Willbrit Vincy
Sprint-3	Development phase	USN-12	Project developmentdelivery of sprint 4	2	Low	Sharmila Pushpa Vanmaraiselvi Vinothini Willbrit Vincy
Sprint-4	Model on IBM	USN-12	Cloud deployment- Deployment of application by using Register for IBM cloud	4	High	Sharmila Pushpa Vanmaraiselvi Vinothini Willbrit Vincy
Sprint-4	Testing Phase	USN-13	Functional testing – Checking usability and accessibility	3	Medi um	Sharmila Pushpa Vanmaraiselvi Vinothini Willbrit Vincy
Sprint-4	Testing Phase	USN-14	Non-Functional testing – Checking scalability and performance of the application	3	Medi um	Sharmila Pushpa Vanmaraiselvi Vinothini Willbrit Vincy

6.2 SPRINT DELIVERY SCHEDULE

SPRINT	TOTAL STORY POINTS	DURATION	SPRINT START DATE	SPRINT END DATE	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= $12/4 = 3$

6.3 REPORTS FROM JIRA



7.CODING AND SOLUTIONING:

7.1 FEATURES 1:

```
from flask
import Flask, render_template, flash, request, session, send_file
from flask
import render_template, redirect, url_for, request
import warnings
import datetime
import cv2
app = Flask( name )
app.config['DEBUG']
app.config['SECRET_KEY'] = '7d441f27d441f27567d441f2b6176a'
@app.route("/")
def homepage():
return render_template('index.html')
@app.route("/Test")
def Test():
return render_template('NewUser.html')
@app.route("/testimage", methods=['GET', 'POST'])
def testimage():
if request.method == 'POST':
file=request.files['fileupload']
file.save('static/Out/Test.jpg')
img = cv2.imread('static/Out/Test.jpg')
if img is None: print('no data')
img1 = cv2.imread('static/Out/Test.jpg')
print(img.shape)
img = cv2.resize(img, ((int)(img.shape[1] / 5), (int)(img.shape[0] / 5))) o
riginal = img.copy()
neworiginal = img.copy()
cv2.imshow('original', img1)
gray = cv2.cvtColor(img1, cv2.COLOR_BGR2GRAY)
img1S = cv2.resize(img1, (960, 540))
cv2.imshow('Original image', img1S)
grayS = cv2.resize(gray, (960, 540))
cv2.imshow('Gray image', grayS)
gry = 'static/Out/gry.jpg'
```



```

cv2.imwrite(gry, grayS)
from PIL
import ImageOps,
Image im = Image.open(file)
im_invert = ImageOps.invert(im)
inv = 'static/Out/inv.jpg'
im_invert.save(inv, quality=95)
dst = cv2.fastNlMeansDenoisingColored(img1, None, 10, 10, 7, 21)
cv2.imshow("Nosie Removal", dst)
noi = 'static/Out/noi.jpg'
cv2.imwrite(noi, dst)
import warnings
warnings.filterwarnings('ignore')
import tensorflow as tf
classifierLoad = tf.keras.models.load_model('model.h5')
import numpy as np
from keras.preprocessing import image
test_image = image.load_img('static/Out/Test.jpg', target_size=(200, 200))
img1 = cv2.imread('static/Out/Test.jpg')
# test_image = image.img_to_array(test_image)
test_image = np.expand_dims(test_image, axis=0)
result = classifierLoad.predict(test_image)
print(result)
out = "
fer = "
if result[0][0] == 1:
out = "APPLES"
fer = '52 calories/1per' \
'Potassium 107mg' \
'Sodium 1mg' \
'Sugar 10g '
elif result[0][1] == 1:
out = "BANANA"
fer = '100 Calories/1per'
elif result[0][2] == 1:
out = "ORANGE"
fer = '50 calories/1per'

```

```

elif result[0][3] == 1:
    out = "PINEAPPLE"
    fer = '60 calories/1per'
elif result[0][4] == 1:
    out = "WATERMELON"
    fer = '400 calories/1per '
    '99% of Water'
    org = 'static/Out/Test.jpg'
    return render_template('NewUser.html',fer=fer,result=out,org=org)
if name == ' main ':
    app.run(debug=True, use_reloader=True

```

7.2 FEATURE 2

```

# Part 1 - Building the CNN
# Importing the Keras libraries and packages
from keras.models import Sequential
from keras.layers import Convolution2D
from keras.layers import MaxPooling2D
from keras.layers import Flatten
from keras.layers import Dense
from keras.models import model_from_json
import matplotlib.pyplot as plt
import warnings
warnings.filterwarnings('ignore')
batch_size = 32
from tensorflow.keras.preprocessing.image import ImageDataGenerator
# All images will be rescaled by 1./255
train_datagen = ImageDataGenerator(rescale=1/255)
# Flow training images in batches of 128 using train_datagen generator
train_generator = train_datagen.flow_from_directory(
'Data', # This is the source directory for training images
target_size=(200, 200), # All images will be resized to 200 x 200
batch_size=batch_size,

#Specify the classes explicitly classes =
['APPLES','BANANA','ORANGE','PINEAPPLE','WATERMELON'],

```

```

# Since we use categorical_crossentropy loss, we need categorical labels
class_mode='categorical')
import tensorflow as tf
model = tf.keras.models.Sequential([
# Note the input shape is the desired size of the image 200x 200 with 3 bytes color
# The first convolution
tf.keras.layers.Conv2D(16, (3,3), activation='relu', input_shape=(200, 200, 3)),
tf.keras.layers.MaxPooling2D(2, 2),
# The second convolution
tf.keras.layers.Conv2D(32, (3,3), activation='relu'),
tf.keras.layers.MaxPooling2D(2,2),
# The third convolution
tf.keras.layers.Conv2D(64, (3,3), activation='relu'),
tf.keras.layers.MaxPooling2D(2,2),
# The fourth convolution
tf.keras.layers.Conv2D(64, (3,3), activation='relu')
tf.keras.layers.MaxPooling2D(2,2),
# The fifth convolution
tf.keras.layers.Conv2D(64, (3,3), activation='relu'),
tf.keras.layers.MaxPooling2D(2,2),
# Flatten the results to feed into a dense layer
tf.keras.layers.Flatten(),
# 128 neuron in the fully-connected layer
tf.keras.layers.Dense(128, activation='relu')
# 5 output neurons for 5 classes with the softmax activation
tf.keras.layers.Dense(5, activation='softmax') ])
model.summary()
from tensorflow.keras.optimizers import RMSprop
early = tf.keras.callbacks.EarlyStopping(monitor='val_loss',patience=5)
model.compile(loss='categorical_crossentropy',
optimizer=RMSprop(lr=0.001),
metrics=['accuracy'])
total_sample=train_generator.n
n_epochs = 10
history = model.fit_generator(
train_generator,
steps_per_epoch=int(total_sample/batch_size),
epochs=n_epochs,

```

```
verbose=1)
model.save('model.h5')
acc = history.history['accuracy']
loss = history.history['loss']
epochs = range(1, len(acc) + 1)
# Train and validation accuracy
plt.plot(epochs, acc, 'b', label=' accuracy')
plt.title(' accuracy')
plt.legend()
plt.figure()
# Train and validation loss
plt.plot(epochs, loss, 'b', label=' loss')
plt.title(' loss')
plt.legend()
plt.show()
```

8.TESTING

8.1 TEST CASES

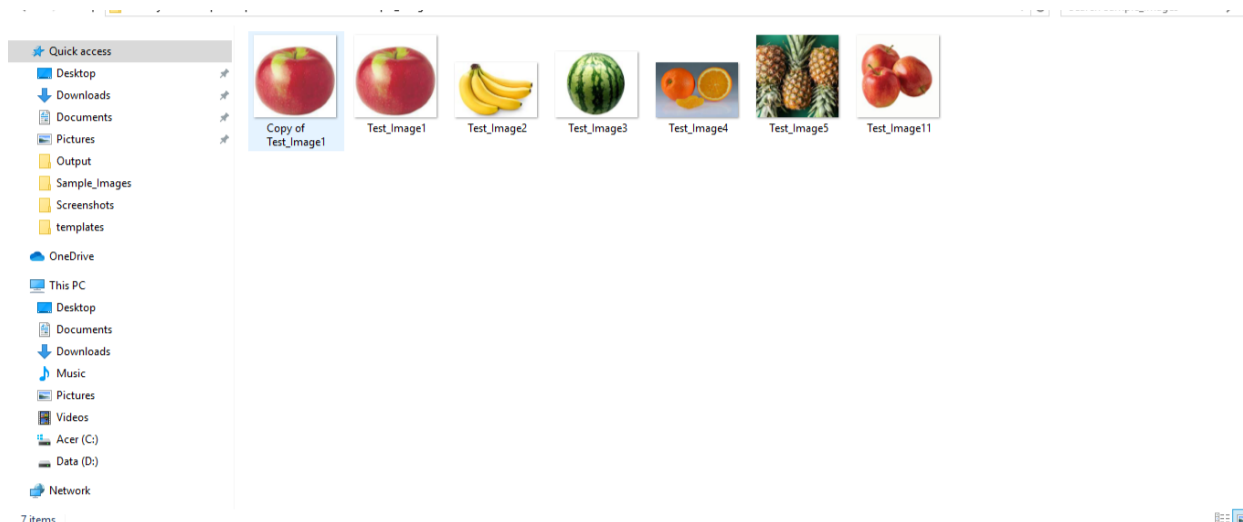
A test case has components that describe input, action and an expected response, in order to determine if a feature of an application is working correctly. A test case is a set of instructions on “HOW” to validate a particular test objective/target, which when followed will tell us if the expected behaviour of the system is satisfied or not. Characteristics of a good test case:

- Accurate: Exacts the purpose.
- Economical: No unnecessary steps or words.
- Traceable: Capable of being traced to requirements.
- Repeatable: Can be used to perform the test over and over.
- Reusable: Can be reused if necessary.

S.NO	SCENARIO	INPUT	EXPECTED OUTPUT	ACTUAL OUTPUT
1.	User login	User name and password	Login	Login sucess.
2.	Upload image	upload input image(fruits and vegetables)	Predicting calorie,fat,carbs and food content of given image	Details are stored in a database.

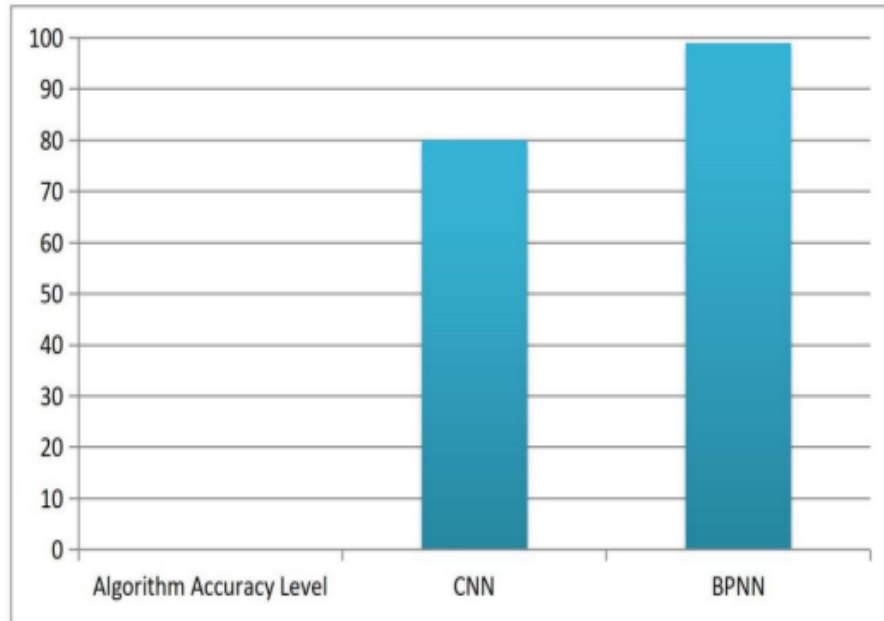
8.2 USER ACCEPTANCE TESTING

This sort of testing is carried out by users, clients, or other authorised bodies to identify the requirements and operational procedures of an application or piece of software. The most crucial stage of testing is acceptance testing since it determines whether or not the customer will accept the application or programme. It could entail the application's U.I., performance, usability, and usefulness. It is also referred to as end-user testing, operational acceptance testing, and user acceptance testing (UAT).



9.RESULTS

9.1 PERFORMANCE METRICS



10. ADVANTAGES & DISADVANTAGES

ADVANTAGES

- 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

DISADVANTAGES

- 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.

10.APPENDIX

10.1 SOURCE CODE

```
from flask import Flask, render_template, flash, request, session, send_file
from flask import render_template, redirect, url_for, request
import warnings
import datetime
import cv2
app = Flask( name )
app.config['DEBUG']
app.config['SECRET_KEY'] = '7d441f27d441f27567d441f2b6176a'
@app.route("/")
def homepage():
    return render_template('index.html')
@app.route("/Test")
def Test():
    return render_template('NewUser.html')
@app.route("/testimage", methods=['GET', 'POST'])
def testimage():
    if request.method == 'POST':
        file = request.files['fileupload']
        file.save('static/Out/Test.jpg')
        img = cv2.imread('static/Out/Test.jpg')
        if img is None:
            print('no data')
        img1 = cv2.imread('static/Out/Test.jpg')
        print(img.shape)
        img = cv2.resize(img, ((int)(img.shape[1] / 5),
                               (int)(img.shape[0] / 5)))
        original = img.copy()
        neworiginal = img.copy()
        cv2.imshow('original', img1)
        gray = cv2.cvtColor(img1, cv2.COLOR_BGR2GRAY)
        img1S = cv2.resize(img1, (960, 540))
        cv2.imshow('Original image', img1S)
        grayS = cv2.resize(gray, (960, 540))
        cv2.imshow('Gray image', grayS)
        gry = 'static/Out/gry.jpg'
        cv2.imwrite(gry, grayS)
```

```

from PIL import ImageOps,
image im = Image.open(file)
im_invert = ImageOps.invert(im)
inv = 'static/Out/inv.jpg'
im_invert.save(inv, quality=95)
dst = cv2.fastNlMeansDenoisingColored(img1, None, 10, 10, 7, 21)
cv2.imshow("Noise Removal", dst)
noi = 'static/Out/noi.jpg' cv2.imwrite(noi, dst)
import warnings
warnings.filterwarnings('ignore')
import tensorflow as tf
classifierLoad = tf.keras.models.load_model('model.h5')
import numpy as np
from keras.preprocessing
import image
test_image=image.load_img('static/Out/Test.jpg',
target_size=(200, 200))
img1 = cv2.imread('static/Out/Test.jpg')
# test_image = image.img_to_array(test_image)
test_image = np.expand_dims(test_image, axis=0)
result = classifierLoad.predict(test_image)
print(result)
out = ""
fer = ""
if result[0][0] == 1:
out = "APPLES"
fer = '52 calories/1per'
elif result[0][1] == 1:
out = "BANANA"
fer = '100 calories/1per'
elif result[0][2] == 1:
out = "ORANGE"
fer = '50 calories/1per'
elif result[0][3] == 1:
out = "PINEAPPLE"
fer = '60 calories/1per'
elif result[0][4] == 1:
out = "WATERMELON"
fer = '400 calories/1per'
org = 'static/Out/Test.jpg'

```

```
return render_template('NewUser.html'  
fer=fer,result=out,org=org)  
if name == ' main ':  
app.run(debug=True, use_reloader=True)
```

HTML CODE:

```
<!DOCTYPE html>  
<html>  
<head>  
<meta name="viewport" content="width=device-width, initial-scale=1">  
<style>  
.container {  
  position: relative;  
  text-align: center;  
  color: black;  
}  
.bottom-left {  
<!DOCTYPE html>  
<html>  
<head>  
<meta name="viewport" content="width=device-width, initial-scale=1">  
<style>  
.container {  
  position: relative;  
  text-align: center;  
  color: black;  
}  
.bottom-left {  
  position: absolute;  
  bottom: 8px;  
  left: 16px;  
}  
.top-left {  
  position: absolute;  
  top: 8px;  
  left: 16px;  
}  
.centered {
```

```
position: absolute;
top: 50%;
left: 50%;
transform: translate(-50%, -50%);
}
```

```
</style>
```

```
</head>
```

```
<body>
```

```
<h2>IBM NALAIYATHIRAN</h2>
```

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 to the trained model. The model analyses the image and detect the nutrition based on the fruits like (Sugar, Fibre, Protein, Calories, etc.).

```
<div class="container">
```

```

```

```
<div class="bottom-left">HEALTHY FOOD, HEALTHY LIFE</div>
```

```
<div class="top-left"><i>WELCOME Move to the page center for uploading your image file.
</i></div>
```

```
<div class="centered"><button><a href = "image.html">UPLOAD</a></button></div>
</div>
```

```
</body>
```

```
</html>
```

IMAGE PREDICTION:

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

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

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

```
<meta http-equiv="X-UA-Compatible" content="ie=edge">
```

```
<title>Predict</title>
<link href="https://cdn.bootcss.com/bootstrap/4.0.0/css/bootstrap.min.css" rel="stylesheet">
<script src="https://cdn.bootcss.com/popper.js/1.12.9/umd/popper.min.js"></script>
<script src="https://cdn.bootcss.com/jquery/3.3.1/jquery.min.js"></script>
<script src="https://cdn.bootcss.com/bootstrap/4.0.0/js/bootstrap.min.js"></script>
<link href="{ { url_for('static', filename='css/main.css') } }" rel="stylesheet">
<style>
body
{
background-image;
url("https://i.pinimg.com/originals/be/21/1a/be211ad5043a8d05757a3538bdd8f450.jpg");
background-size: cover;
}
.bar
{
margin: 0px;
padding:20px;
background-color:white;
opacity:0.6;
color:black;
font-family:'Roboto',sans-serif;
font-style: italic;
border-radius:20px;
font-size:25px;
}
a
{
color:grey;
float:right;
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;
margin: 20px;
height: 500px;
}
.header { position: relative;
top:0;
margin:0px;
z-index: 1;
left: 0px;
right: 0px;
position: fixed;
background-color: #8B008B ;
color: white;
box-shadow: 0px 8px 4px grey;
overflow: hidden;
padding-left:20px;
font-family: 'Josefin Sans';
font-size: 2vw;
width: 100%;
height:8%;
text-align: center;
}
.topnav {
overflow: hidden;
background-color: #FCAD98;
}
.topnav-right a {
float: left;
color: black;
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;
}
.topnav-right {
float: right;
padding-right: 100px;
}
</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%;">Nutrition Image Analysis</div>
<div class="topnav-right" style="padding-top: 0.5%;">
<a href="{{ url_for('home') }}">Home</a>
<a class="active" href="{{ url_for('image1') }}">Classify</a>
</div>
</div>
<br>
</div>
<div class="container">
<center>
<div id="content" style="margin-top: 2em">{% block content %}{% endblock
%}</div></center>
</div>
</body>
<footer>
<script src="{{ url_for('static', filename='js/main.js') }}" type="text/javascript"></script>

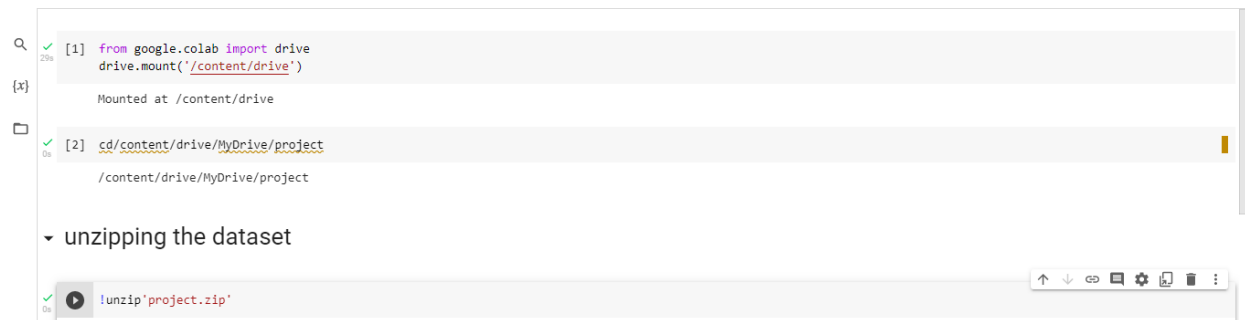
```


</footer>

</html>

SCREENSHOTS:

DATA COLLECTION



```
[1] from google.colab import drive
drive.mount('/content/drive')

Mounted at /content/drive

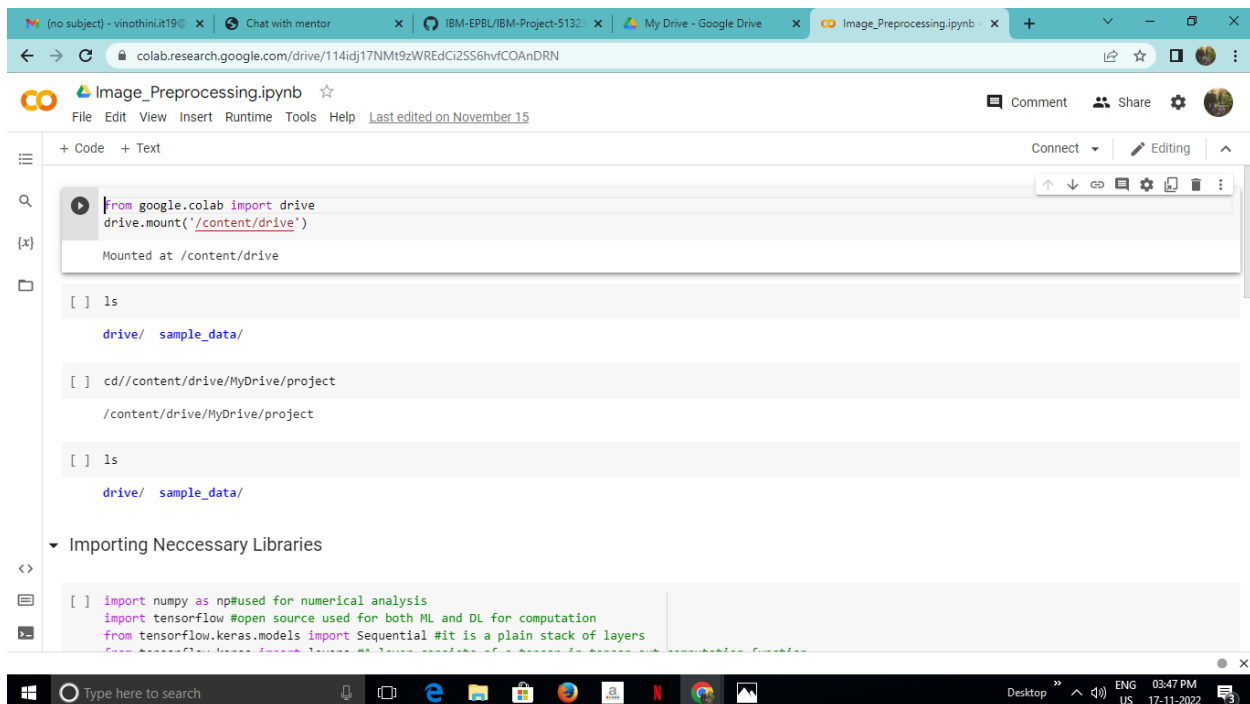
[2] cd /content/drive/MyDrive/project

/content/drive/MyDrive/project

unzipping the dataset

!unzip 'project.zip'
```

IMAGE PREPROCESSING



```
from google.colab import drive
drive.mount('/content/drive')

Mounted at /content/drive

[ ] ls

drive/ sample_data/

[ ] cd //content/drive/MyDrive/project

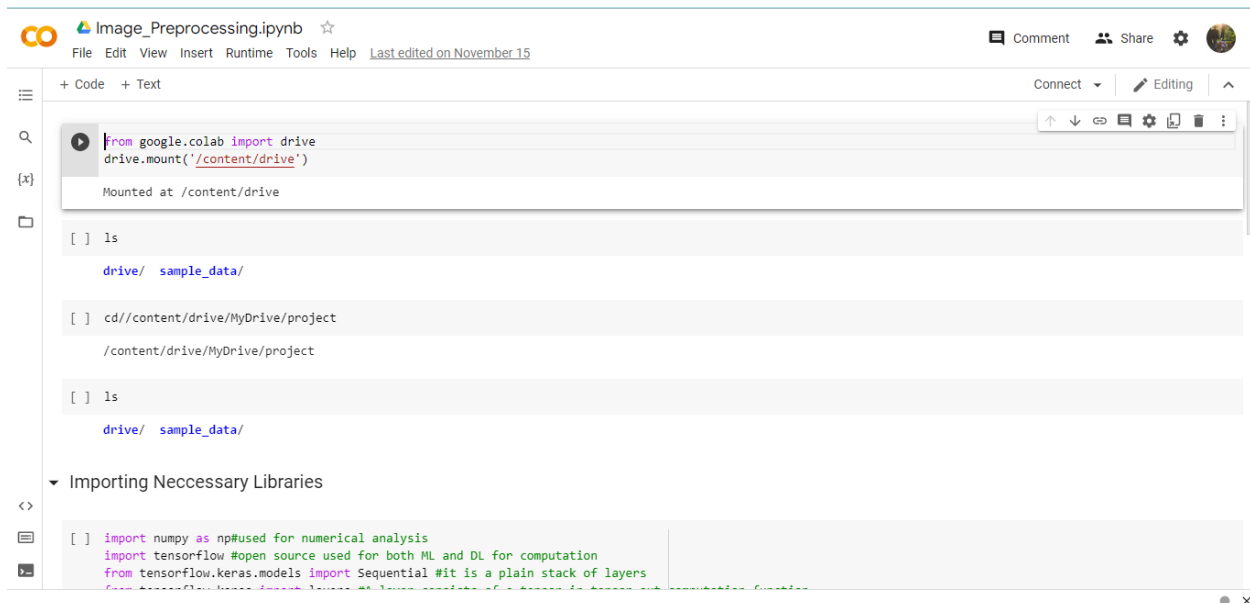
/content/drive/MyDrive/project

[ ] ls

drive/ sample_data/

Importing Neccessary Libraries

[ ] import numpy as np#used for numerical analysis
import tensorflow #open source used for both ML and DL for computation
from tensorflow.keras.models import Sequential #it is a plain stack of layers
```



```
from google.colab import drive
drive.mount('/content/drive')

Mounted at /content/drive

[ ] ls

drive/ sample_data/

[ ] cd//content/drive/MyDrive/project

/content/drive/MyDrive/project

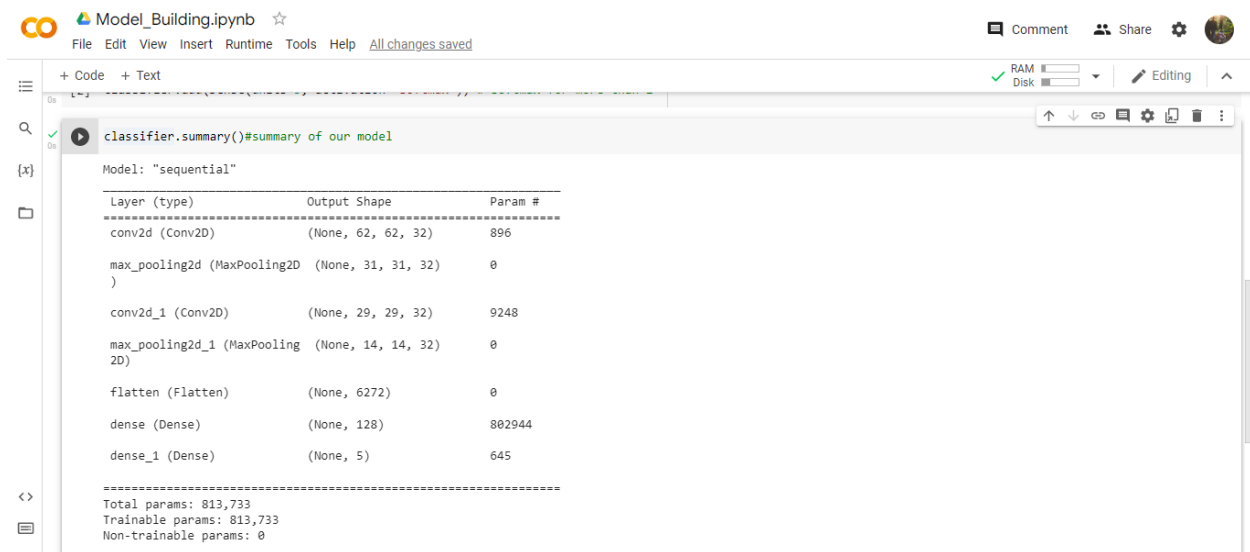
[ ] ls

drive/ sample_data/

▼ Importing Neccessary Libraries

[ ] import numpy as np#used for numerical analysis
import tensorflow #open source used for both ML and DL for computation
from tensorflow.keras.models import Sequential #it is a plain stack of layers
```

MODEL BUILDING



```
classifier.summary()#summary of our model

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

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

CLOUD REGISTRATION

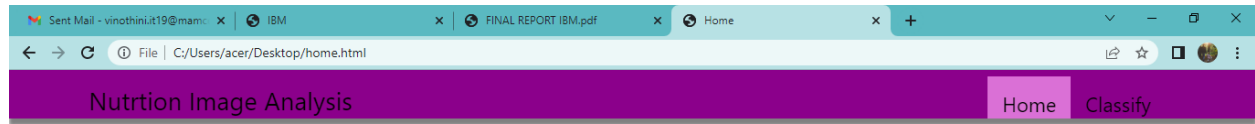
The screenshot displays the IBM Cloud Pak for Data web interface. The browser address bar shows the URL: `dataplatfom.cloud.ibm.com/projects/52dd451c-46b9-4b36-b8f3-408c66181d47/assets?context=cpdaas`. The page header includes the IBM Cloud Pak for Data logo, a search bar, and user account information for 'pushpa A's Account' in 'Dallas'. The main navigation bar shows 'Overview', 'Assets', 'Jobs', and 'Manage' tabs, with 'Assets' currently selected. On the left sidebar, under '1 assets', the 'All assets' option is highlighted. Below this, 'Asset types' are listed, with 'Notebooks' having a count of 1. The main content area, titled 'All assets', contains a table with the following data:

Name	Last modified
Build python code Notebook	17 hours ago Modified by you

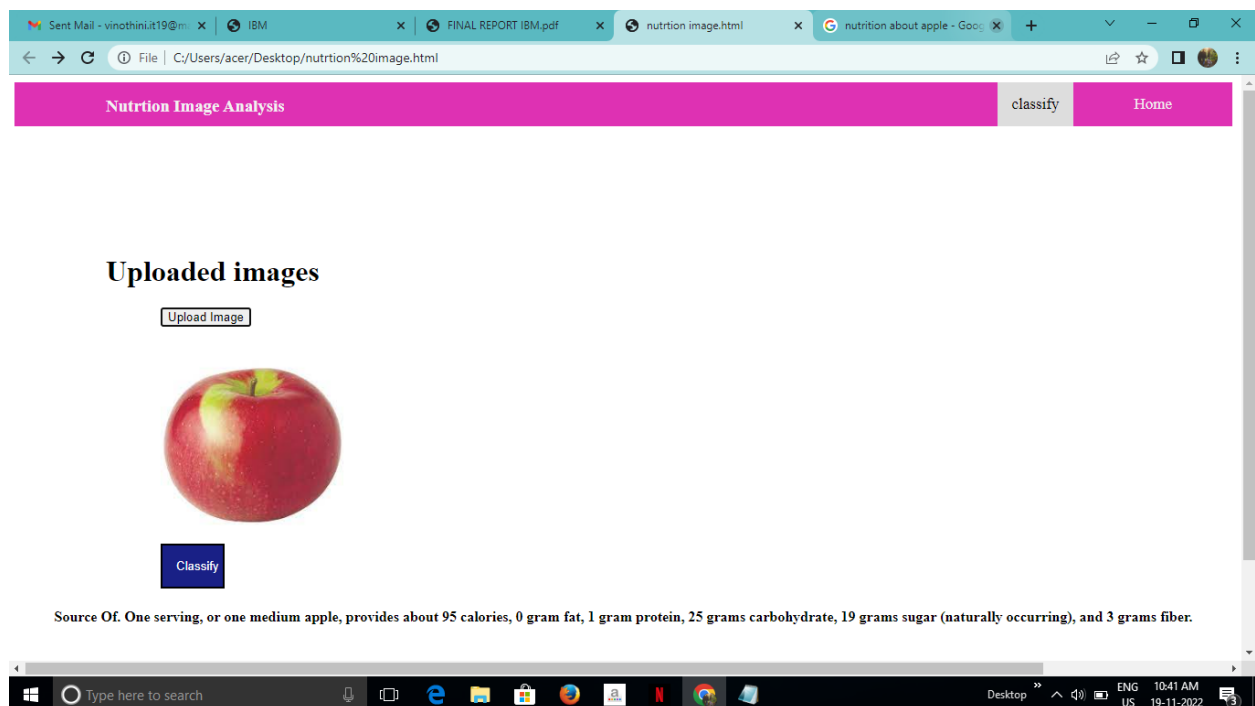
At the bottom of the table, it indicates 'Items per page: 20' and '1-1 of 1 items'.

On the right side of the interface, a panel titled 'Data in this project' contains a message: 'Drop data files here or browse for files to upload'. At the bottom of the screen, a Windows taskbar is visible with the search bar, taskbar icons, and system clock showing '04:25 AM 17/11/2022'. An 'Activate Windows' watermark is also present in the bottom right corner.

OUTPUT SCREENSHOTS



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.




Sent Mail - vinothini.it19@n x IBM x FINAL REPORT IBM.pdf x nutrition image.html x nutrition about banana - Go x + v - x

File | C:/Users/acer/Desktop/nutrition%20image.html

Nutrition Image Analysis classify Home

Uploaded images

Upload Image



Classify

One serving, or one medium ripe banana, provides about 110 calories, 0 gram fat, 1 gram protein, 28 grams carbohydrate, 15 grams sugar (naturally occurring), 3 grams fiber, and 450 mg potassium

Type here to search Desktop ENG US 10:38 AM 19-11-2022


Sent Mail - vinothini.it19@n x IBM x FINAL REPORT IBM.pdf x nutrition image.html x nutrition about watermelon x + v - x

File | C:/Users/acer/Desktop/nutrition%20image.html

Nutrition Image Analysis classify Home

Uploaded images

Upload Image



Classify

Calories: 46. Carbs: 11.5 grams. Fiber: 0.6 grams. Sugar: 9.4 grams. Protein: 0.9 grams. Fat: 0.2 grams. Vitamin A: 5% of the Daily Value (DV) Vitamin C: 14% of the DV.

Type here to search Desktop ENG US 10:40 AM 19-11-2022


Sent Mail - vinothini.it19@n... x IBM x FINAL REPORT IBM.pdf x nutrition image.html x nutrition about pineapple - G... x

File | C:/Users/acer/Desktop/nutrition%20image.html

Nutrition Image Analysis [classify](#) [Home](#)

Uploaded images

[Upload Image](#)



[Classify](#)

Calories 50 % Daily Value* Total Fat 0.1 g Saturated fat 0 g Cholesterol 0 mg Sodium 1 mg Potassium 109 mg Total Carbohydrate 13 g Dietary fiber 1.4 g Sugar 10 g

Type here to search Desktop ENG US 10:45 AM 19-11-2022


Sent Mail - vinothini.it19@n... x IBM x FINAL REPORT IBM.pdf x nutrition image.html x nutrition about oranges - G... x

File | C:/Users/acer/Desktop/nutrition%20image.html

Nutrition Image Analysis [classify](#) [Home](#)

Uploaded images

[Upload Image](#)



[Classify](#)

60 calories. No fat or sodium. 3 grams of fiber. 12 grams of sugar. 1 gram of protein. 14 micrograms of vitamin A. 70 milligrams of vitamin C. 6% of your daily recommended amount of calcium.

Type here to search Desktop ENG US 10:47 AM 19-11-2022

Github Link:

TEAM ID: PNT2022TMID45226

[**https://github.com/IBM-EPBL/IBM-Project-51323-1660978095**](https://github.com/IBM-EPBL/IBM-Project-51323-1660978095)

Project Demo Link:

[**https://drive.google.com/drive/folders/1Cank6PxaYakJjm2RIJHCCCMCYsS6hzBm**](https://drive.google.com/drive/folders/1Cank6PxaYakJjm2RIJHCCCMCYsS6hzBm)