

.Project Report Format

1. INTRODUCTION

1.1 Project Overview

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.

1.2 Purpose

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

2. LITERATURE SURVEY

2.1 Existing problem

With diet and nutrition analyzer apps, you can analyze your client's current goal of diet and provide them with effective feedback on what to improve.

It is possible to create personalized meal plans and healthy recipes and generate nutrient analysis reports by using barcodes and other relevant nutrition tracking tools compared to the users' nutritional requirements.

2.2 References

1. UK Department of Health. Dietary reference values for food energy and nutrients in the United Kingdom (Report on health and social subjects; 41). London, UK: Her Majesty's Stationery Office; 1991. [March 4, 2018].
2. "Leading health and fitness," April 2017 [Online]. Available: <https://www.statista.com/statstics/650748/health-fitness-app-usage-usa>
3. Mukasine Angelique (2014). "Ontology-Based Personalized System to Support Patients at Home." Research paper [Online]. Available: <https://brage.bibsys.no/xmlui/bitstream/handle/11250/221227/IKT-590%20Spring%20Matster's%20thesis%20Angelique%20MUKASINE.pdf?sequence=1>
4. Nutritionix API, June 2107 [Online]. Available: <https://www.nutritionix.com/business/api>
5. Run Android Application, April 2017 [Online]. Available: <https://developer.android.com/training/basics/firstapp/running-app.html>

2.3 Problem Statement Definition

A program is needed to help users who want to manage their weight and calculate calories based on their daily activity and intake of food. Our team has the privilege of developing this analyzer program that can help the average person calculate their calories intake and how much weight they are expected to stay fit. Every day people are concerned about their health. Often times, reading the nutritional label of the food being consumed isn't enough to determine if a calorie balance goal is being met. It is also difficult to calculate what is expended based on the activities performed each day. Even now, there are those who are willing to record such information on paper to do the math themselves. This can prove to be quite trivial and tedious, especially in our modern technological world. Others will often be deterred from calculating calorie balance because of this as well

3. IDEATION & PROPOSED SOLUTION

3.1 Empathy Map Canvas



3.2 Ideation & Brainstorming

1. CUSTOMER SEGMENT(S) CS Nutrition analyzes designed to help health professionals, dietitians, coaches, trainers and gymnasiums manage client, create personalized meal plans.	4. CUSTOMER CONSTRAINTS Food allergies or sensitivities, religious, practices, and ideologies beliefs are some of the main reasons people rely on specific diets or follow Dietary restrictions.	7. AVAILABLE SOLUTIONS AS <ul style="list-style-type: none"> ➤ Information available on internet or web ➤ To prevent malnutrition is to Eat a healthy, balanced diet.
2. JOBS-TO-BE-DONE / PROBLEM J&P People are eating far more food Than is necessary for their health.	5. PROBLEM ROOT CAUSE RC People are not eating sufficient amount of food to provide the calories, vitamins and minerals they need optimal health. While in others, people are eating far more food than is Necessary for their health.	8. BEHAVIOUR BE <ul style="list-style-type: none"> ❖ To give the regular notification ❖ To give proper diet ❖ Don't beat yourself up if you miss a day ❖ Add to your existing habits
3. TRIGGERS Nutritional Analysis detects the exact nutritional value of any given food item. It determines the percentage of macro and micronutrients present in that food item as well as the presence of inhibitors, toxic chemicals, or any other new component.	6. EMOTIONS: BEFORE / AFTER BEFORE: Improper health maintenance. AFTER: proper health maintenance	9. YOUR SOLUTION Wellness and Mental health for end user who uses our application.

Proposed Solution

People are not eating sufficient amounts of food to provide the calories, vitamins and minerals they need optimal health, while in others, people are eating far more food than is necessary for their health.

Food Nutrition Analysis helps in the detailed and perfect determination of the component nutrients present in any food item.

It is web based program that allow anyone to analysis the foods they eat to determine if they are consuming all the different nutrients.

This review demonstrates that general food quality fundamentally influences customer satisfaction and behaviour intentions.

3.3 Problem Solution fit

People are not eating sufficient amount of food to provide the calories, vitamins and minerals they need optimal health. While in others, people are eating far more food than is necessary for their health.

4. REQUIREMENT ANALYSIS

4.1 Functional requirement

FR No.	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)
FR-1	User Registration	Registration through Web application Form Registration through Gmail
FR-2	User Confirmation	Confirmation via Email Confirmation via OTP

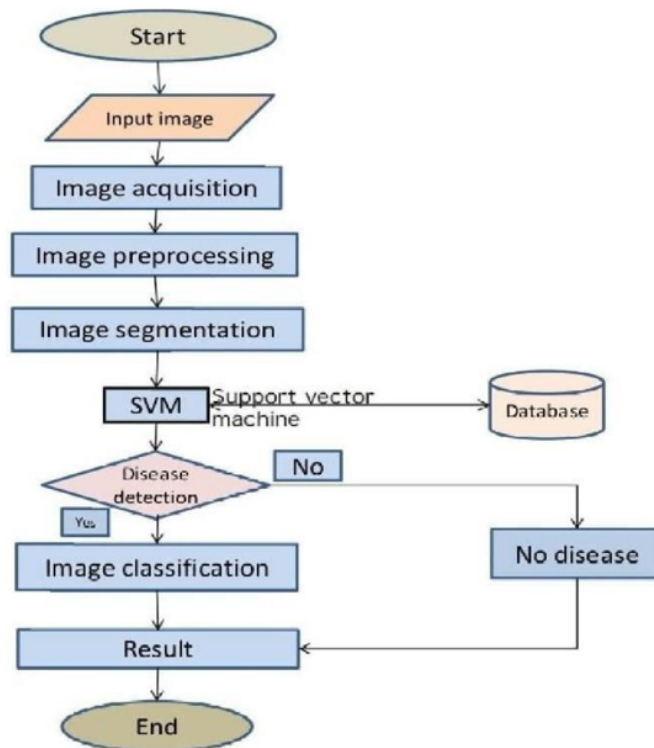
4.2 Non-Functional requirements

FR No.	Non-Functional Requirement	Description
NF R-1	Usability	Efficient for the frequent users. Users can easily understand what the application does and feel satisfied with the system.
NF R-2	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.
NF R-3	Reliability	This application must perform without failure in 95 percent of use cases during a month.
NF R-4	Performance	This application supporting 1,000 users per hour must provide 6 seconds or less response time in a desktop browser, including the rendering of text and images, over an LTE connection.
NF R-5	Availability	The web dashboard must be available to user's 99.9 percent of the time every month during business hours EST. Users can access every time.

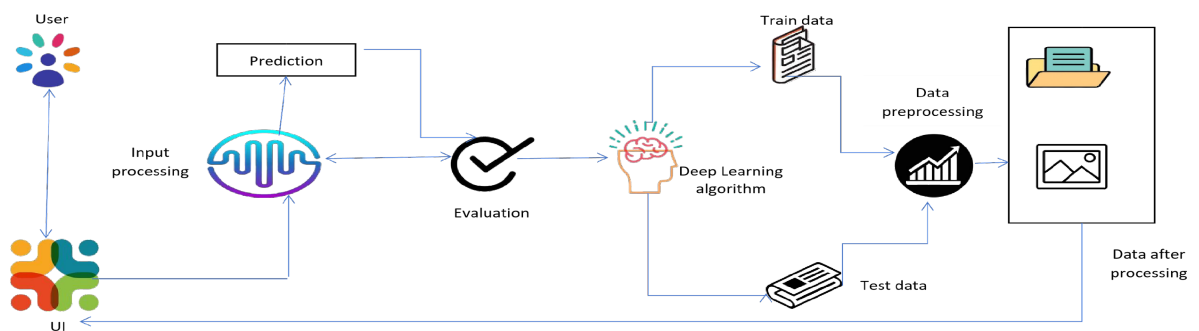
NF R-6	Scalability	The application must be scalable enough to support 10,000 visits at the same time while maintaining optimal performance.
-----------	--------------------	--

5. PROJECT DESIGN

5.1 Data Flow Diagrams



5.2 Solution & Technical Architecture



5.3 User Stories

6. PROJECT PLANNING & SCHEDULING

6.1 Sprint Planning & Estimation

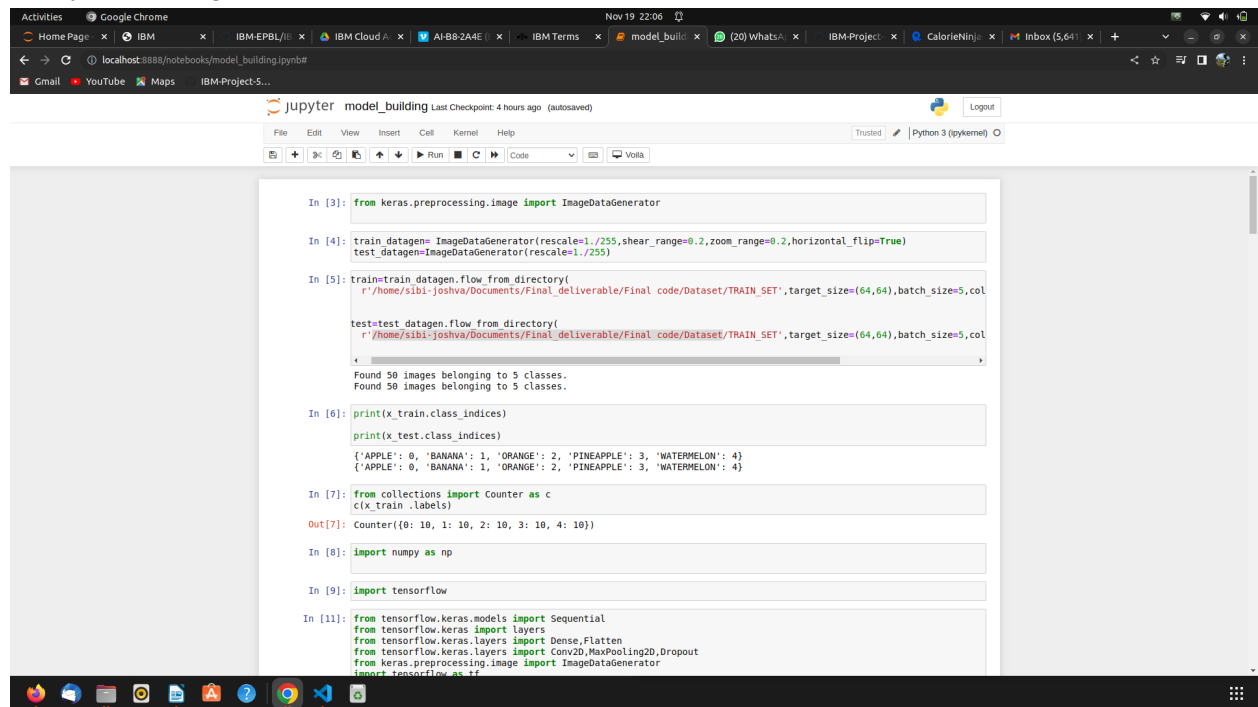
Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint 1	Registration	USN-1	As a analyzer , I can register for the application by entering my email, password, and confirming my password	2	High	SIBI M
Sprint 1		USN-2	As a analyzer, I can register for the application through Gmail	1	Medium	SANJAY J
Sprint 1	User Confirmation	USN-3	As a analyzer, I will receive confirmation email once I have registered for the application	1	High	SHIVAM
Sprint1	Login	USN-4	As a analyzer, I can log into the application by entering email & password	2	High	SUNDARAKRISHNAN

6.2 Sprint Delivery Schedule

Sprint	Total Story Points	Total Story Points	Sprint Start Date	Sprint End Date (Planned)	Story Points Completed (as on Planned End Date)	Sprint Release Date (Actual)
Sprint-1	20	4 Days	24 Oct 2022	27 Oct 2022	20	29 Oct 2022
Sprint-2	20	5 Days	28 Oct 2022	01 Nov 2022	20	04 Nov 2022
Sprint-3	20	8 Days	02 Nov 2022	09 Nov 2022	20	11 Nov 2022
Sprint-4	20	9 Days	10 Nov 2022	18 Nov 2022	20	19 Nov 2022

7. CODING & SOLUTIONING (Explain the features added in the project along with code)

7.1 Python coding



```
In [3]: from keras.preprocessing.image import ImageDataGenerator

In [4]: train_datagen= ImageDataGenerator(rescale=1./255,shear_range=0.2,zoom_range=0.2,horizontal_flip=True)
test_datagen=ImageDataGenerator(rescale=1./255)

In [5]: train=train_datagen.flow_from_directory(
r'/home/sibi-joshva/Documents/Final_deliverable/Final_code/Dataset/TRAIN_SET',target_size=(64,64),batch_size=5,col

test=test_datagen.flow_from_directory(
r'/home/sibi-joshva/Documents/Final_deliverable/Final_code/Dataset/TRAIN_SET',target_size=(64,64),batch_size=5,col

Found 50 images belonging to 5 classes.
Found 50 images belonging to 5 classes.

In [6]: print(x_train.class_indices)
print(x_test.class_indices)

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

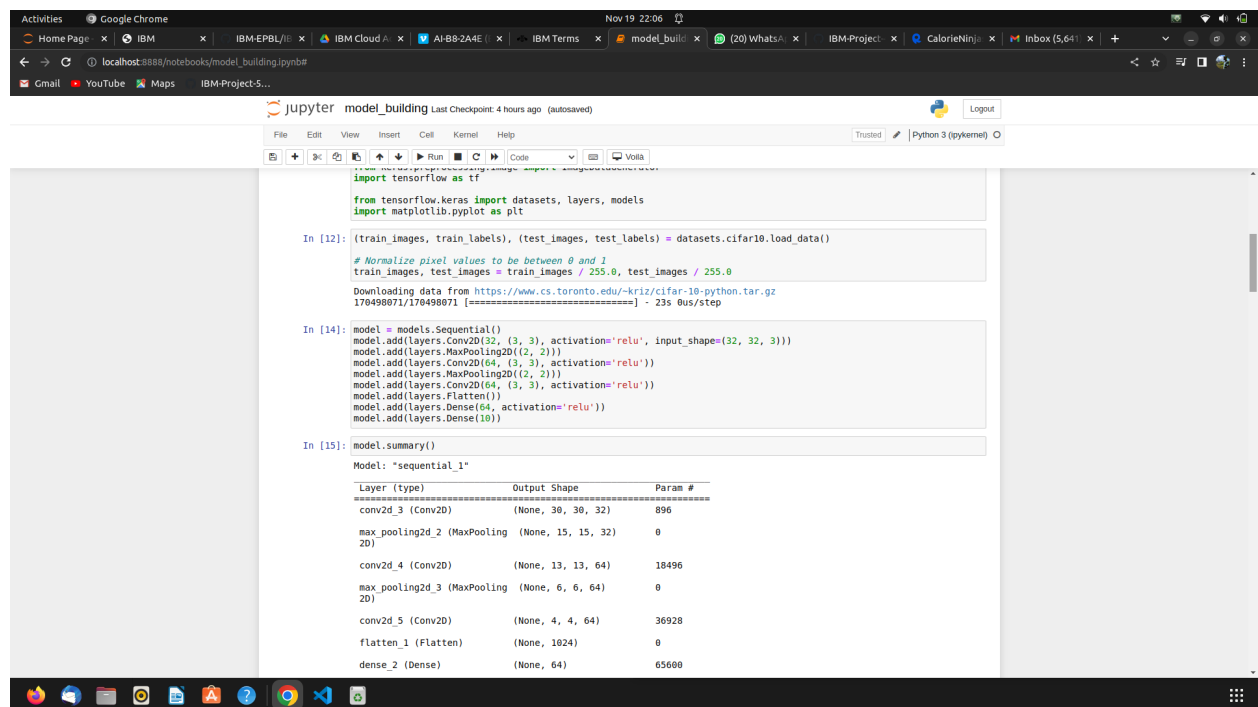
In [7]: from collections import Counter as c
c(x_train.labels)

Out[7]: Counter({0: 10, 1: 10, 2: 10, 3: 10, 4: 10})

In [8]: import numpy as np

In [9]: import tensorflow

In [11]: from tensorflow.keras.models import Sequential
from tensorflow.keras import layers
from tensorflow.keras.layers import Dense,Flatten
from tensorflow.keras.layers import Conv2D,MaxPool2D,Dropout
from keras.preprocessing.image import ImageDataGenerator
import tensorflow as tf
```



```
from tensorflow.keras import layers, models
import matplotlib.pyplot as plt

In [12]: (train_images, train_labels), (test_images, test_labels) = datasets.cifar10.load_data()

# Normalize pixel values to be between 0 and 1
train_images, test_images = train_images / 255.0, test_images / 255.0

Downloading data from https://www.cs.toronto.edu/~kriz/cifar-10-python.tar.gz
170498071/170498071 [=====] - 23s 0us/step

In [14]: model = models.Sequential()
model.add(layers.Conv2D(32, (3, 3), activation='relu', input_shape=(32, 32, 3)))
model.add(layers.MaxPooling2D((2, 2)))
model.add(layers.Conv2D(64, (3, 3), activation='relu'))
model.add(layers.MaxPooling2D((2, 2)))
model.add(layers.Conv2D(64, (3, 3), activation='relu'))
model.add(layers.Flatten())
model.add(layers.Dense(64, activation='relu'))
model.add(layers.Dense(10))

In [15]: model.summary()

Model: "sequential_1"
Layer (type) Output Shape Param #
-----
conv2d_3 (Conv2D) (None, 30, 30, 32) 896
max_pooling2d_2 (MaxPooling (None, 15, 15, 32) 0
2D)
conv2d_4 (Conv2D) (None, 13, 13, 64) 18496
max_pooling2d_3 (MaxPooling (None, 6, 6, 64) 0
2D)
conv2d_5 (Conv2D) (None, 4, 4, 64) 36928
flatten_1 (Flatten) (None, 1024) 0
dense_2 (Dense) (None, 64) 65600
```

Activities Google Chrome Nov 19 22:06

Home Page x IBM x IBM-EPBL/ll x IBM Cloud A x AI-B8-2A4E x IBM Terms x model_build x (20) WhatsA x IBM-Project x CalorieNin x Inbox (5,64 x +

localhost:8888/notebooks/model_building.ipynb

Gmail x YouTube x Maps x IBM-Project-5...

jupyter model_building Last Checkpoint: 4 hours ago (autosaved)

File Edit View Insert Cell Kernel Help Trusted Python 3 (pykernel)

dense_3 (Dense) (None, 10) 658

=====

Total params: 122,570

Trainable params: 122,570

Non-trainable params: 0

In [16]: #Compiling the model

```
model.compile(optimizer='adam',
              loss=tf.keras.losses.SparseCategoricalCrossentropy(from_logits=True),
              metrics=['accuracy'])
#Fitting the model
history = model.fit(train_images, train_labels, epochs=5,
                  validation_data=(test_images, test_labels))
```

Epoch 1/5

2022-11-19 12:39:08.097157: W tensorflow/core/framework/cpu_allocator_impl.cc:82] Allocation of 614400000 exceeds 10% of free system memory.

1563/1563 [=====] - ETA: 0s - loss: 1.5119 - accuracy: 0.4478

2022-11-19 12:39:25.825959: W tensorflow/core/framework/cpu_allocator_impl.cc:82] Allocation of 122880000 exceeds 10% of free system memory.

1563/1563 [=====] - 18s 11ms/step - loss: 1.5119 - accuracy: 0.4478 - val_loss: 1.2970 - val accuracy: 0.5473

Epoch 2/5

1563/1563 [=====] - 16s 10ms/step - loss: 1.1821 - accuracy: 0.5796 - val_loss: 1.1222 - val accuracy: 0.6668

Epoch 3/5

1563/1563 [=====] - 16s 10ms/step - loss: 1.0302 - accuracy: 0.6379 - val_loss: 1.0108 - val accuracy: 0.6505

Epoch 4/5

1563/1563 [=====] - 15s 10ms/step - loss: 0.9227 - accuracy: 0.6771 - val_loss: 0.9277 - val accuracy: 0.6743

Epoch 5/5

1563/1563 [=====] - 16s 10ms/step - loss: 0.8510 - accuracy: 0.7008 - val_loss: 0.8977 - val accuracy: 0.6872

In [25]: # Initializing the CNN classifier = Sequential()

First convolution layer and pooling

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

Activities Google Chrome Nov 19 22:07

Home Page x IBM x IBM-EPBL/ll x IBM Cloud A x AI-B8-2A4E x IBM Terms x model_build x (20) WhatsA x IBM-Project x CalorieNin x Inbox (5,64 x +

localhost:8888/notebooks/model_building.ipynb

Gmail x YouTube x Maps x IBM-Project-5...

jupyter model_building Last Checkpoint: 4 hours ago (autosaved)

File Edit View Insert Cell Kernel Help Trusted Python 3 (pykernel)

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

# Second convolution layer and pooling
classifier.add(Conv2D(32, (3, 3), activation='relu'))

# input shape is going to be the pooled feature maps from the previous convolution layer
classifier.add(MaxPooling2D(pool_size=(2, 2)))

# Flattening the layers
classifier.add(Flatten())
```

In [26]: classifier.add(Dense(units=128, activation='relu'))

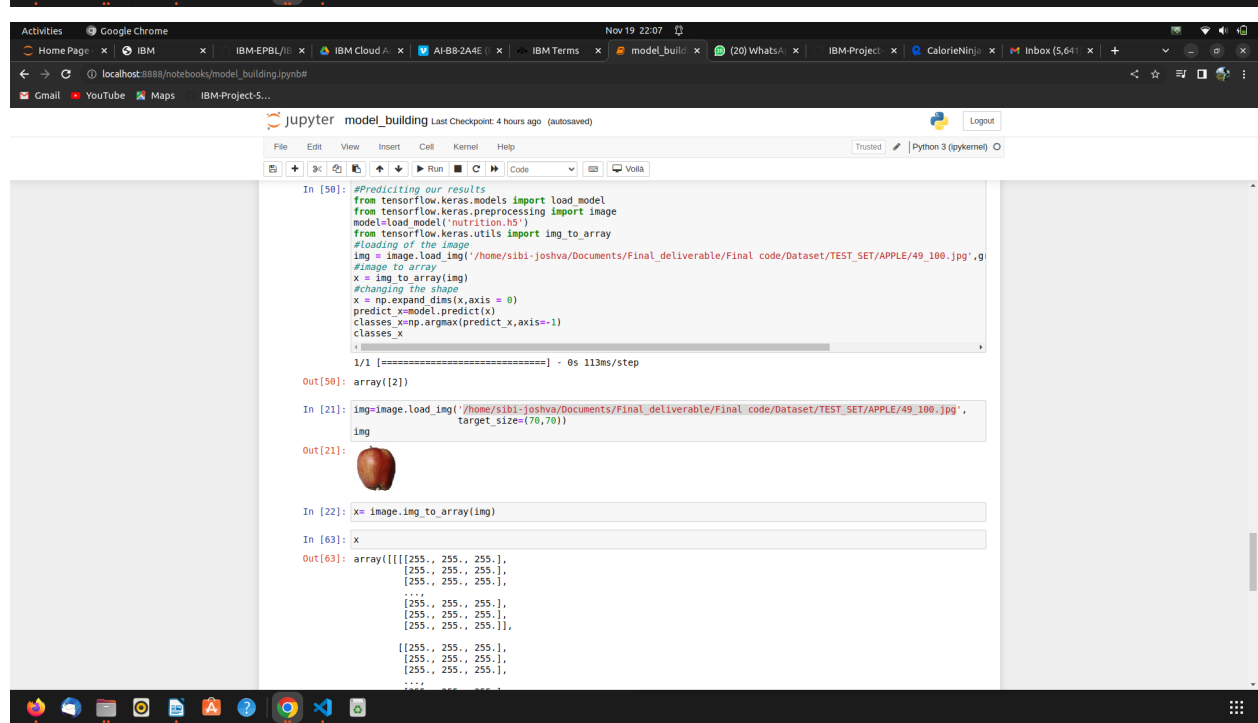
```
classifier.add(Dense(units=5, activation='softmax'))
#summary of our model
classifier.summary()
```

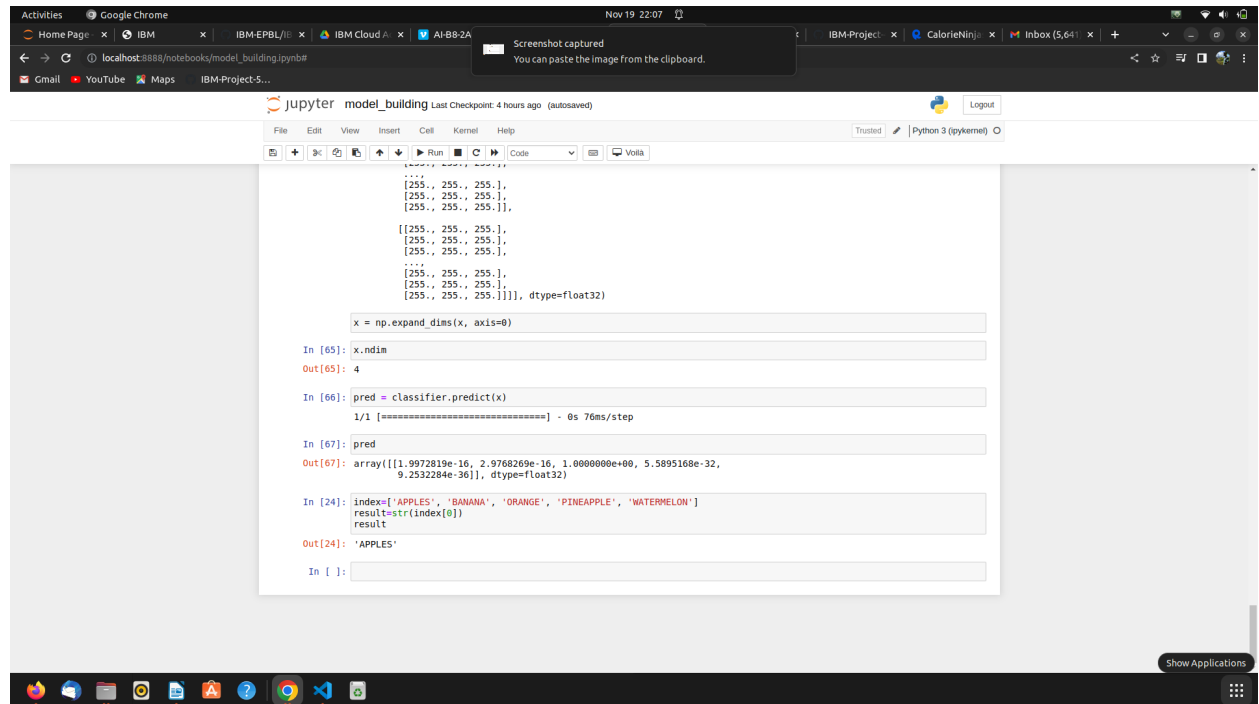
Model: "sequential_2"

Layer (type)	Output Shape	Param #
=====		
conv2d_6 (Conv2D)	(None, 62, 62, 32)	896
max_pooling2d_4 (MaxPooling2D)	(None, 31, 31, 32)	0
conv2d_7 (Conv2D)	(None, 29, 29, 32)	9248
max_pooling2d_5 (MaxPooling2D)	(None, 14, 14, 32)	0
flatten_2 (Flatten)	(None, 6272)	0
dense_4 (Dense)	(None, 128)	802944
dense_5 (Dense)	(None, 5)	645
=====		
Total params:	813,733	
Trainable params:	813,733	
Non-trainable params:	0	

In [17]: #Saving our model

```
model.save('nutrition.h5')
```



7.2 Feature 2

```
<!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.jsdelivr.net/npm/bootstrap@4.7.0/dist/css/bootstrap.min.css" rel="stylesheet">
```

```
<link href="https://cdn.jsdelivr.net/npm/@popperjs/core@2.9.2/dist/umd/popper.min.js" rel="stylesheet">
```

```
<script src="https://cdn.jsdelivr.net/npm/@popperjs/core@2.9.2/dist/umd/popper.min.js"></script>
```

```
<script src="https://cdn.jsdelivr.net/npm/jquery@3.3.1/dist/jquery.min.js"></script>
```

```
<script src="https://cdn.jsdelivr.net/npm/bootstrap@4.0.0/dist/js/bootstrap.min.js"></script>
```

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

```
<style>
```

```
.card1 {
```

```
box-shadow: 0 4px 8px 0 rgba(0, 0, 0, 0.2);
```

```
max-width: 300px;
```

```
margin: auto;
```

```
text-align: center;
```

```
font-family: arial;
```

```
}
```

```
.title {
```

```
color: grey;
font-size: 18px;
}
```

```
button {
border: none;
outline: 0;
display: inline-block;
padding: 8px;
color: white;
background-color: #000;
text-align: center;
cursor: pointer;
width: 100%;
font-size: 18px;
}
```

```
a {
text-decoration: none;
font-size: 22px;
color: black;
}
```

```
button:hover, a:hover {
opacity: 0.7;
}
.navbarScroll.navbarDark {
background-color: black;
}
body
{
```

```
background-image:
url("https://www.livingproofnyc.com/wp-content/themes/livingproof/assets/img/hero-background
.jpg");
background-size: cover;
}
.bar
{
margin: 0px;
padding: 30px;
background-color: black;
opacity: 0.6;
color: red;
font-family: 'Roboto', sans-serif;
```

```
font-style: italic;
border-radius:30px;
font-size:10px;
}
```

```
.header {      position: relative;
                top:0;
                margin:0px;
                z-index: 1;
                left: 0px;
                right: 0px;
                position: fixed;
                background-color: #01FF70 ;
                color: white;
                box-shadow: 0px 8px 4px grey;
                overflow: hidden;
                padding-left:10px;
                font-family: 'Josefin Sans'
                font-size: 1.5vw;
                width: 100%;
                height:10%;
            }
            .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: #FFDC00;
color: black;
}
```

```
.topnav-right a.active {
background-color: #FFDC00;
```

```
color: black;
}
```

```
.topnav-right {
  float: right;
  padding-right: 100px;
}
```

```
.navbarScroll.navbarDark {
  background-color: black;
}
```

```
.ct-socials {
  position: fixed;
  top: 25%;
  right: 0;
```

```
background-color: yellow;
padding-left: 20;
```

```
margin: 50;
```

```
padding: 10px;
font-size: 10px;
width: 40px;
text-align: center;
```

```
border: 80px;
```

```
}
```

```
.section.triad-section {
  margin-top: 10px;
}
```

```
section.section h2 {
  font-size: 20px;
  line-height: 46px;
  margin-bottom: 20px;
  text-align: center;
  margin-top: 0;
}
```

```
h2 {
```

```
    color: #000;
}
h1, h2, h3, h4, h5, h6 {
    font-weight: 200;
    letter-spacing: -1px;
    font-size: 30px;
}
section.section p.sub-heading {
    font-size: 16px;
    font-family: "Gotham SSm A", "Gotham SSm B";
    font-weight: 300;
    text-align: center;
    margin-bottom: 40px;
}
section.triad-section .triad-sub-section {
    padding-right: 60px;
}
section p.detail-paragraph:first-child {
    margin-top: 0;
}
section p.detail-paragraph {
    font-family: 'Open Sans Condensed', sans-serif;
    margin-top: 40px;
    font-size: 18px;
    color: #000;
}
b, strong {
    font-weight: 700;
}
.bgimage {
    height: 100vh;
    background: url('images/heroImage.jpg');
    background-size: cover;
    position: relative;
}

.hero_title {
    font-size: 4.5rem;
}
.hero_desc {
    font-size: 2rem;
}
.hero-text {
```

```
text-align: center;
position: absolute;
top: 50%;
left: 50%;
transform: translate(-50%, -50%);
color: white;
}
```

```
.imageAboutPage {
width: 100%;
}
```

```
#services .services {
flex-direction: column;
text-align: center;
max-width: 1500px;
margin: 0 auto;
padding: 100px 0;
}
#services .service-top {
max-width: 500px;
margin: 0 auto;
}
#services .service-bottom {
display: flex;
align-items: center;
justify-content: center;
flex-wrap: wrap;
color: red;
margin-top: 50px;
}
#services .service-item {
flex-basis: 80%;
display: flex;
align-items: flex-start;
justify-content: center;
flex-direction: column;
color: red;
padding: 30px;
```

```

        border-radius: 10px;
        background-image: url(/img/img-1.png);
        background-size: cover;
        margin: 10px 5%;
        position: relative;
        z-index: 1;
        overflow: hidden;
    }
    #services .service-item::after {
        content: "";
        position: absolute;
        left: 0;
        top: 0;
        height: 100%;
        width: 100%;
        background-image: linear-gradient(60deg, #29323c 0%, #485563 100%);
        opacity: 0.9;
        z-index: -1;
    }
    #services .service-bottom .icon {
        height: 80px;
        width: 80px;
        margin-bottom: 20px;
    }
    #services .service-item h2 {
        font-size: 2rem;
        color: red;
        margin-bottom: 10px;
        text-transform: uppercase;
        text-align: left;
    }
    #services .service-item p {
        color: white;
        text-align: left;
    }
    #services .service-item a {
        color: white;
        text-align: center;
    }

    .section-title {
        font-size: 4rem;
        font-weight: 300;
    }

```



```
        color: black;
        margin-bottom: 10px;
        text-transform: uppercase;
        letter-spacing: 0.2rem;
        text-align: center;
    }
    .section-title span {
        color: crimson;
    }
```

```
.cta:hover {
    color: white;
    background-color: crimson;
}
.brand h1 {
    font-size: 3rem;
    text-transform: uppercase;
    color: white;
}
.brand h1 span {
    color: crimson;
}
.brand a {
    font-size: 3rem;
    text-transform: uppercase;
    color: Tomato;
}
.brand a span {
    color: crimson;
}
.brand p{
```

```
    text-transform: uppercase;
    color: Tomato;
        font-size: 4rem;
    font-weight: 300;
```

```
    margin-bottom: 10px;
    text-transform: uppercase;
    letter-spacing: 0.2rem;
    text-align: center;
```

```
}
```

```
.brand p span {  
    color: crimson;  
}
```

```
#logo {  
float: right;  
}
```

```
.face{  
    position: relative;  
    width: 250px;  
    height: 250px;  
    border-radius: 50%;  
    background: #ffcd00;  
    display: flex;  
    justify-content: center;  
    justify-items: center;  
    align-items: center;  
}
```

```
.face::before  
{  
    content: ";  
    position: absolute;  
    top: 150px;  
    width: 150px;  
    height: 70px;  
    background: #b57700;  
    border-bottom-left-radius: 70px;  
    border-bottom-right-radius: 70px;  
    transition: 0.5s;  
}
```

```
.face::hover::before  
{  
    top: 210px;  
    width: 150px;  
    height: 20px;  
    background: #b57700;  
    border-bottom-left-radius: 0px;  
    border-bottom-right-radius: 0px;  
}
```

```
.eyes
```

```

{
    position: relative;
    top: -40px;
    display: flex;
}
.eyes .eye
{
    position: relative;
    width: 80px;
    height: 80px;
    display: block;
    background: #fff;
    margin: 0 15px;
    border-radius: 50%;
}
.eyes .eye::before
{
    content: "";
    position: absolute;
    top: 50%;
    left: 25px;
    transform: translate(-50%,-50%);
    width: 40px;
    height: 40px;
    background: #333;
    border-radius: 50%;
}

#header {
    position: fixed;
    z-index: 1000;
    left: 0;
    top: 0;
    width: 100vw;
    height: auto;
}
#header .header {
    min-height: 8vh;
    background-color: rgba(31, 30, 30, 0.24);
    transition: 0.3s ease background-color;
}
#header .nav-bar {
    display: flex;

```

```

        align-items: center;
        justify-content: space-between;
        width: 100%;
        height: 100%;
        max-width: 1300px;
        padding: 0 10px;
    }
    #header .nav-list ul {
        list-style: none;
        position: absolute;
        background-color: rgb(31, 30, 30);
        width: 100vw;
        height: 100vh;
        left: 100%;
        top: 0;
        display: flex;
        flex-direction: column;
        justify-content: center;
        align-items: center;
        z-index: 1;
        overflow-x: hidden;
        transition: 0.5s ease left;
    }
    #header .nav-list ul.active {
        left: 0%;
    }
    #header .nav-list ul a {
        font-size: 2.5rem;
        font-weight: 500;
        letter-spacing: 0.2rem;
        text-decoration: none;
        color: white;
        text-transform: uppercase;
        padding: 20px;
        display: block;
    }
    #header .nav-list ul a::after {
        content: attr(data-after);
        position: absolute;
        top: 50%;
        left: 50%;
        transform: translate(-50%, -50%) scale(0);
        color: rgba(240, 248, 255, 0.021);
        font-size: 13rem;
    }

```

```

        letter-spacing: 50px;
        z-index: -1;
        transition: 0.3s ease letter-spacing;
    }
    #header .nav-list ul li:hover a::after {
        transform: translate(-50%, -50%) scale(1);
        letter-spacing: initial;
    }
    #header .nav-list ul li:hover a {
        color: crimson;
    }
    #header .hamburger {
        height: 60px;
        width: 60px;
        display: inline-block;
        border: 3px solid white;
        border-radius: 50%;
        position: relative;
        display: flex;
        align-items: center;
        justify-content: center;
        z-index: 100;
        cursor: pointer;
        transform: scale(0.8);
        margin-right: 20px;
    }
    #header .hamburger:after {
        position: absolute;
        content: "";
        height: 100%;
        width: 100%;
        border-radius: 50%;
        border: 3px solid white;
        animation: hamburger_puls 1s ease infinite;
    }
    #header .hamburger .bar {
        height: 2px;
        width: 30px;
        position: relative;
        background-color: white;
        z-index: -1;
    }
    #header .hamburger .bar::after,
    #header .hamburger .bar::before {

```

```

        content: "";
        position: absolute;
        height: 100%;
        width: 100%;
        left: 0;
        background-color: white;
        transition: 0.3s ease;
        transition-property: top, bottom;
    }
    #header .hamburger .bar::after {
        top: 8px;
    }
    #header .hamburger .bar::before {
        bottom: 8px;
    }
    #header .hamburger.active .bar::before {
        bottom: 0;
    }
    #header .hamburger.active .bar::after {
        top: 0;
    }
    #hero {
        background-image: url('logo.png');
        background-repeat: no-repeat;
        background-attachment: fixed;
        background-position: 90% 40%;
        position: relative;
        z-index: 1;
    }
    #hero::after {
        content: "";
        position: absolute;
        left: 0;
        top: 0;
        height: 100%;
        width: 100%;
        background-color: black;
        opacity: 0.7;
        z-index: -1;
    }
    #hero .hero {
        max-width: 1200px;
        margin: 0 auto;
    }

```

```

        padding: 0 50px;
        justify-content: flex-start;
    }
    #hero h1 {
        display: block;
        width: fit-content;
        font-size: 4rem;
        position: relative;
        color: transparent;
        animation: text_reveal 0.5s ease forwards;
        animation-delay: 1s;
    }
    #hero h1:nth-child(1) {
        animation-delay: 1s;
    }
    #hero h1:nth-child(2) {
        animation-delay: 2s;
    }
    #hero h1:nth-child(3) {
        animation: text_reveal_name 0.5s ease forwards;
        animation-delay: 3s;
    }
    #hero h1 span {
        position: absolute;
        top: 0;
        left: 0;
        height: 100%;
        width: 0;
        background-color: crimson;
        animation: text_reveal_box 1s ease;
        animation-delay: 0.5s;
    }
    #hero h1:nth-child(1) span {
        animation-delay: 0.5s;
    }
    #hero h1:nth-child(2) span {
        animation-delay: 1.5s;
    }
    #hero h1:nth-child(3) span {
        animation-delay: 2.5s;
    }
    #hero h2 {
        display: block;
        width: fit-content;

```

```
    font-size: 4rem;
    text-align: top;
    position: relative;
    color: orange;
    background-color: Tomato;
    animation: text_reveal 0.5s ease forwards;
    animation-delay: 1s;
}

#services .services {
    flex-direction: column;
    text-align: center;
    max-width: 1500px;
    margin: 0 auto;
    padding: 100px 0;
}

#services .service-top {
    max-width: 500px;
    margin: 0 auto;
}

#services .service-bottom {
    display: flex;
    align-items: center;
    justify-content: center;
    flex-wrap: wrap;
    color: red;
    margin-top: 50px;
}

#services .service-item {
    flex-basis: 80%;
    display: flex;
    align-items: flex-start;
    justify-content: center;
    flex-direction: column;
    color: red;
    padding: 30px;
    border-radius: 10px;
    background-image: url(/img/img-1.png);
    background-size: cover;
    margin: 10px 5%;
    position: relative;
    z-index: 1;
    overflow: hidden;
}
```



```
#services .service-item::after {
    content: "";
    position: absolute;
    left: 0;
    top: 0;
    height: 100%;
    width: 100%;
    background-image: linear-gradient(60deg, #29323c 0%, #485563 100%);
    opacity: 0.9;
    z-index: -1;
}
#services .service-bottom .icon {
    height: 80px;
    width: 80px;
    margin-bottom: 20px;
}
#services .service-item h2 {
    font-size: 2rem;
    color: red;
    margin-bottom: 10px;
    text-transform: uppercase;
    text-align: left;
}
#services .service-item p {
    color: white;
    text-align: left;
}

#services .service-item a {
    color: white;
    text-align: center;
}

#footer {
    background-image: linear-gradient(60deg, #29323c 0%, #485563 100%);
}
#footer .footer {
    min-height: 200px;
    flex-direction: column;
    padding-top: 50px;
    padding-bottom: 10px;
}
#footer h2 {
```

```

        color: white;
        font-weight: 500;
        font-size: 1.8rem;
        letter-spacing: 0.1rem;
        margin-top: 10px;
        margin-bottom: 10px;
    }
    #footer .social-icon {
        display: flex;
        margin-bottom: 30px;
    }
    #footer .social-item {
        height: 50px;
        width: 50px;
        margin: 0 5px;
    }
    #footer .social-item img {
        filter: grayscale(1);
        transition: 0.3s ease filter;
    }
    #footer .social-item:hover img {
        filter: grayscale(0);
    }
    #footer p {
        color: white;
        font-size: 1.3rem;
    }
}

@keyframes hamburger_puls {
    0% {
        opacity: 1;
        transform: scale(1);
    }
    100% {
        opacity: 0;
        transform: scale(1.4);
    }
}

@keyframes text_reveal_box {
    50% {
        width: 100%;
        left: 0;
    }
    100% {

```

```

        width: 0;
        left: 100%;
    }
}
@keyframes text_reveal {
    100% {
        color: white;
    }
}
@keyframes text_reveal_name {
    100% {
        color: crimson;
        font-weight: 500;
    }
}

@media only screen and (min-width: 768px) {
    .cta {
        font-size: 2.5rem;
        padding: 20px 60px;
    }
    h1.section-title {
        font-size: 6rem;
    }

    #hero h1 {
        font-size: 7rem;
    }

    #services .service-bottom .service-item {
        flex-basis: 45%;
        margin: 2.5%;
    }
}

@media only screen and (min-width: 1200px) {
    #header .hamburger {
        display: none;
    }
    #header .nav-list ul {

```

```

        position: initial;
        display: block;
        height: auto;
        width: fit-content;
        background-color: transparent;
    }
    #header .nav-list ul li {
        display: inline-block;
    }
    #header .nav-list ul li a {
        font-size: 1.8rem;
    }
    #header .nav-list ul a:after {
        display: none;
    }

    #services .service-bottom .service-item {
        flex-basis: 22%;
        margin: 1.5%;
    }
}
</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%;">NUTRITION IMAGE ANALYSIS</div>
    <div class="topnav-right"style="padding-top:0.5%;">

        <a class="active" href="{{ url_for('home')}}" "><b>HOME</b></a>
        <a href="{{ url_for('image1')}}" "><b>CLASSIFY</b></a>
    </div>
</div>
</div>
<br>
<br>

<section id="about">

```

```

<div class="container mt-4 pt-4">
    <br><br><br>
    <h1 class="text-center"><center><b>&emsp;OBJECTIVE OF THE
PROJECT</center></b></h1>
    <div class="row mt-4">
        <div class="col-lg-4">
            
        </div>

        <div class="col-lg-8">
            <br>

            <ul>
<li>Food is essential for human life and has been the concern of many healthcare conventions.
</li>
<li>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. </li>
<li>Nutritional analysis is the process of determining the nutritional content of food. </li>
<li>It is a vital part of analytical chemistry that provides information about the chemical
composition, processing, quality control and contamination of food.</li>
</ul>

            </div>
        </div>
    </section>
    <br>
    <br>
    <br><br>
    <section id="about">
        <div class="container mt-4 pt-4">
            <br><br><br>
            <h1 class="text-center"><b>AIM OF THE PROJECT</b></h1>
            <div class="row mt-4">
                <div class="col-lg-4">
                    
                </div>

                <div class="col-lg-8">

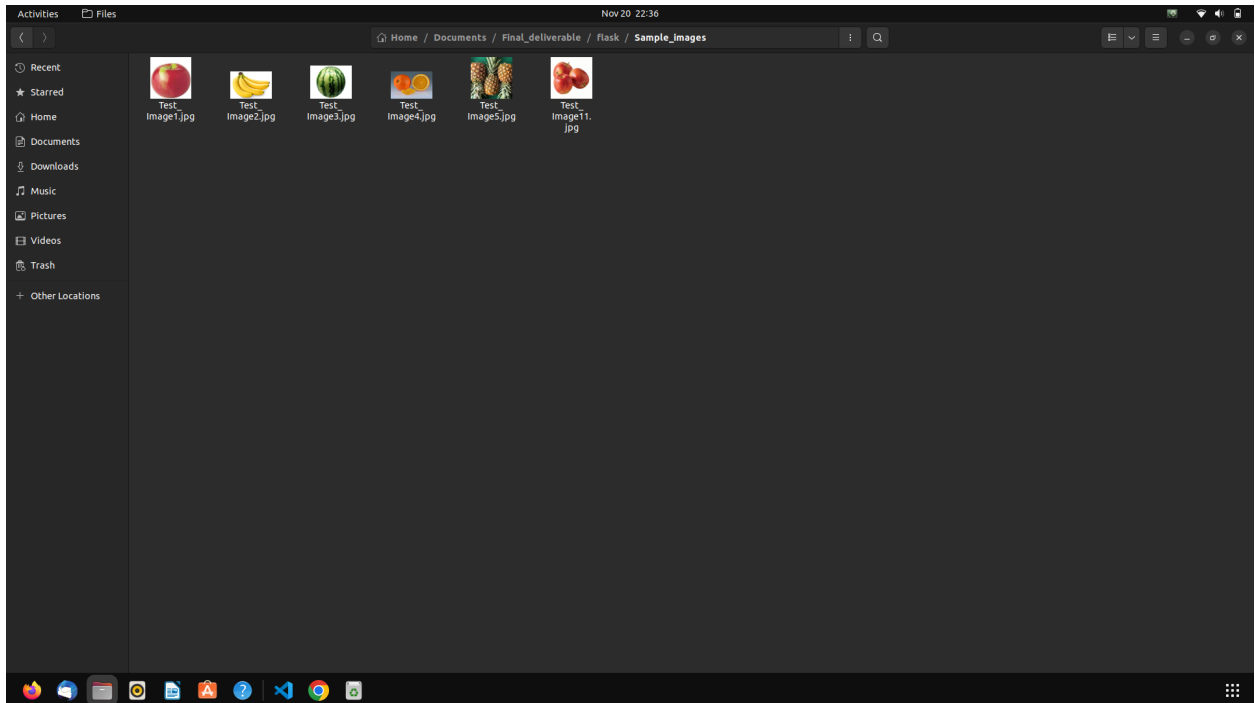
```

```
<br>
<ul>
<li>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. </li>
<li>Here the user can capture the images of different fruits and then the image will be sent the
trained model.</li>
<li>The model analyses the image and detect the nutrition based on the fruits like (Sugar, Fibre,
Protein, Calories, etc.).</li>
</ul>
</div>
</div>
</section>
<br>
<br>
<br>
<br>
<br>

</body>
</html>
```

8. TESTING

8.1 Test Cases



9. ADVANTAGES & DISADVANTAGES

Advantages:

- 1. Interface is extremely friendly and easy to use. – Helps count calories while tracking weight loss.**
- 2. Analyzes a person's age and weight with how they want to improve their health.**
- 3. You can scan in popular foods by their barcodes without having to type in the nutritional values.**
- 4. Provides a daily report of caloric intake.**
- 5. Has a feature where users can track their food portion sizes by taking a picture of their dish.**

Disadvantages:

- 1. You have to manually input your daily food intake once a weight loss plan has been created**
- 2. While the app itself is free, all the premium features are locked away unless you pay \$9.99 for it.**
- 3. It doesn't track your intake of minerals and vitamins.**
- 4. Food database is not that large. It is missing a lot of the popular food brands, which means users have to manually type the nutritional values in."**

10. CONCLUSION

In the journey of developing this nutrify application, I got an opportunity to learn the detail process of developing Android applications with the help of material design concepts to built beautiful yet elegant user interface. In this process I also able to learn how to develop and use a RESTful API from scratch with the help of Python. Moreover, I even get a chance to learn how to use different Android libraries such as MPAndroidChart, Zxing, FitChart and so on. I also learn about the various food ontology APIs on how to use them and even by developing the client-server application helped me to understand the communication calls between them.

FUTURE SCOPE

In the future scope, the application can enhance its functionality by adding image recognition which can be used to analyze the food image and produce the result with the nutrition values contained in that particular food item. A Google map can be added to track the distance covered by the user using the Activity Tracker to provide a more visual representation of the activity to the user. The activity tracker can also be updated using the Google fit API for the more accurate result.

