

**19L039 - PROFESSIONAL READINESS FOR  
INNOVATION,EMPLOYABILITY AND  
ENTREPRENEURSHIP**

**AI-POWERED NUTRITION ANALYSER FOR FITNESS  
ENTHUSIASTS**

**CHANDRAN V V (19L108)**

**SANJAY A (19L130)**

**SOUNDARYAA R (19L136)**

**VASANTH R (20L421)**

Dissertation submitted in partial fulfillment of the requirements for the degree of

**BACHELOR OF ENGINEERING**

**Branch: ELECTRONICS AND COMMUNICATION ENGINEERING**

of Anna University



**NOVEMBER 2022**

**PSG COLLEGE OF TECHNOLOGY**

(Autonomous Institution)

COIMBATORE – 641 004

## **Project Report**

DATE	18 November 2022
TEAM ID	PNT2022TMID12745
PROJECT NAME	AI-powered Nutrition Analyzer for Fitness Enthusiasts

## **1. INTRODUCTION**

### **1.1 Project Overview**

Nutrition analyzer aims to use personal information about individuals or groups of individuals to deliver nutritional advice that, theoretically, would be more suitable than generic advice. Deep learning, a sub branch of Artificial Intelligence, has promise to aid in the development of predictive models that are suitable for analyzing Nutrition. Using the prediction made by CNN to provide nutrition values of the food or fruits, that help fitness enthusiasts to track their daily nutrition intake to maintain a healthy life.

### **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, Fiber, Protein, Calories, etc.).

## **2. LITERATURE SURVEY**

### **2.1 Existing problem**

Sharing and collaboration are commonplace in today's social media-driven world. But the lack of regulation means it's all too easy for people to share information on fitness, health – and pretty much anything – without any factual backing. Combine this with standard challenges

like keeping clients motivated, and there are problems in the fitness industry that can affect your business.

## **2.2 References**

1. Pure-CNN: A Framework for Fruit Images Classification by Asia Kausar, Mohsin Sharif, JinHyuck Park and Dong Ryeol Shin on 2018 International Conference on Computational Science and Computational Intelligence (CSCI).

Link. [https://www.researchgate.net/publication/338360652\\_Pure-CNN\\_A\\_Framework\\_for\\_Fruit\\_Images\\_Classification](https://www.researchgate.net/publication/338360652_Pure-CNN_A_Framework_for_Fruit_Images_Classification)

2. Fruit Classification using Convolutional Neural Network via Adjust Parameter and Data Enhancement by Liuchen Wu, Hui Zhang, Rubio Chen, Ruibo Chen, Junfei Yi on 12th International Conference on Advanced Computational Intelligence (ICACI).

Link: <https://ieeexplore.ieee.org/document/9177518>

3. Improving the Prediction of Rotten Fruit Using Convolutional Neural Network by Sumitra Nuanmeesri, Lap Poomhiran, Kunalai Ploydanai on International Journal of Engineering Trends and Technology.

## **2.3 Problem Statement Definition**

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.

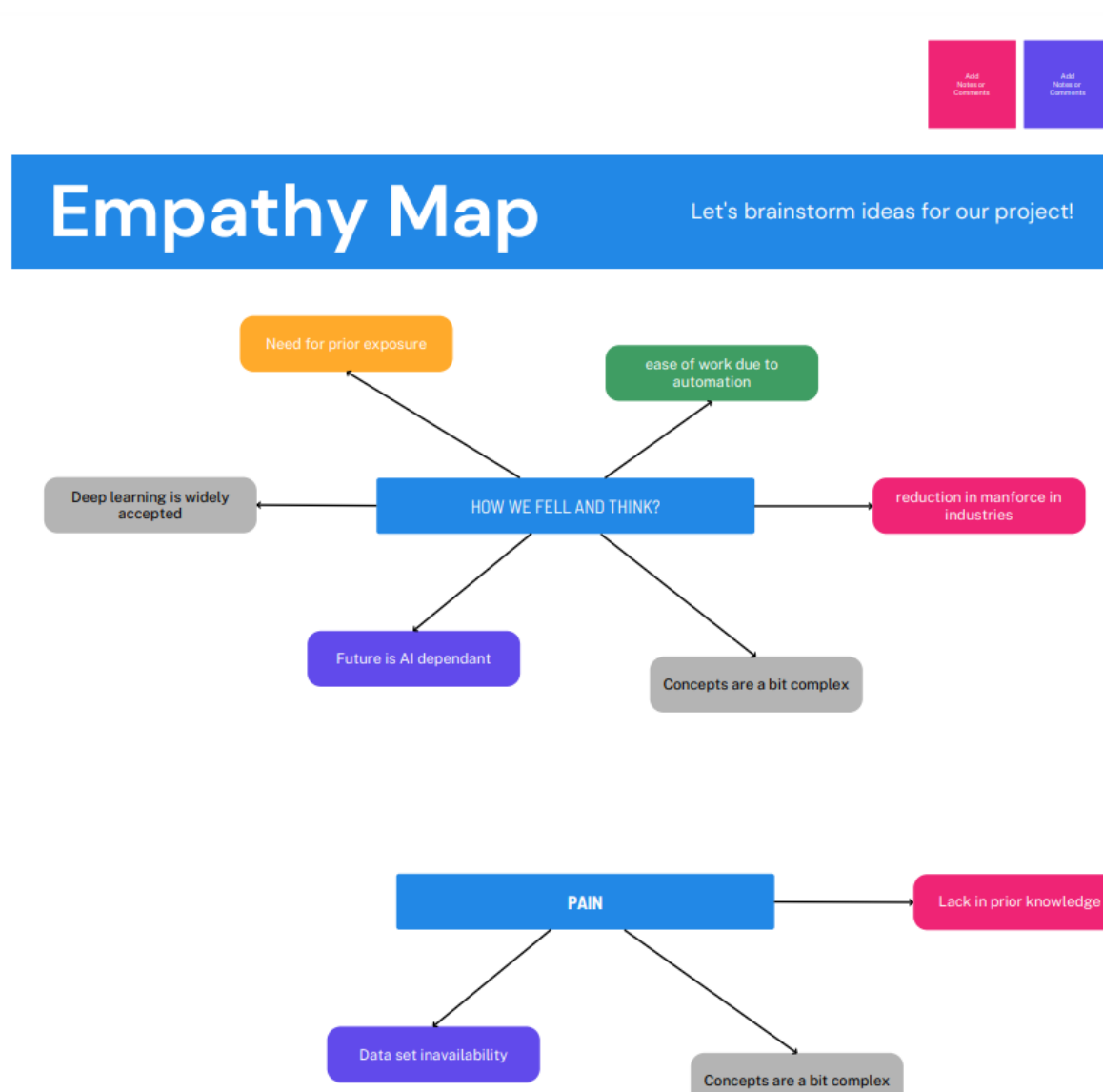
### 3. IDEATION & PROPOSED SOLUTION

#### 3.1 Empathy Map Canvas

In this Empathy Map we describe about

What do they,

Think And Feel, See, Say And Do, Hear, Pain And Gain



### 3.2 Ideation & Brainstorming

In this phase we discussed about

- Where to Start
- Time Management
- Friend and Family Support
- Bad Health Habits



### 3.3 Proposed Solution

S.No.	Parameter	Description
1.	Problem Statement (Problem solved)	A nutrition analyzer using AI-powered fruit classifier based on characteristics to give fitness enthusiasts with nutritional values like fibre, vitamins, minerals, protein etc.
2.	Idea / Solution description	Developing web-based software to track and monitor people's health conditions and assist in their improvement
3.	Novelty / Uniqueness	Artificial intelligence presents nutrition with unmatched prospects for advancement and applications. To maximise the potential of this developing field, there are still holes to fill.

4.	Social Impact / Customer Satisfaction	An individual's nutritional state and social, psychological, and cultural circumstances are related in a cause-and-effect manner. One may monitor how many nutrients they can balance in their diet by using this programme.
5.	Business Model (Revenue Model)	offering premium features on a monthly or annual subscription basis. Providing revenue from application data.
6.	Scalability of the Solution	The nutrition analyzer now only includes fruits, but it may be expanded to include more foods and implemented as a mobile app.

### **3.4 Problem Solution fit**

#### **HONEY-PROFILING:**

Bruker's FoodScreener Essential Honey, a reasonably priced system geared for beekeepers, honey packers, and honey associations, offers the Honey-Profiling approach (dedicated to honey analysis only). It is also compatible with Bruker's Wine-Profiling and Juice-Profiling techniques as well as the flexible NMR FoodScreener™ platform.



#### **SOLID FAT CONTENT:**

The best place to get benchtop NMR analyzers for solid fat content is Bruker (SFC). The mq series offers a field-upgradable SFC Analyzer, but the mq-one SFC Analyzer delivers the greatest possible price and footprint-to-performance ratio. To provide reliable findings that may be compared with those from other factories throughout the world, Bruker minispec systems are calibrated using standardised procedures. The only method for determining SFC that has received formal approval is the NMR technique. It is more efficient than the laborious, time-consuming dilatometry approach, which is used to calculate the Solid Fat Index (SFI) values.





## 4. REQUIREMENT ANALYSIS

### 4.1 Functional requirement

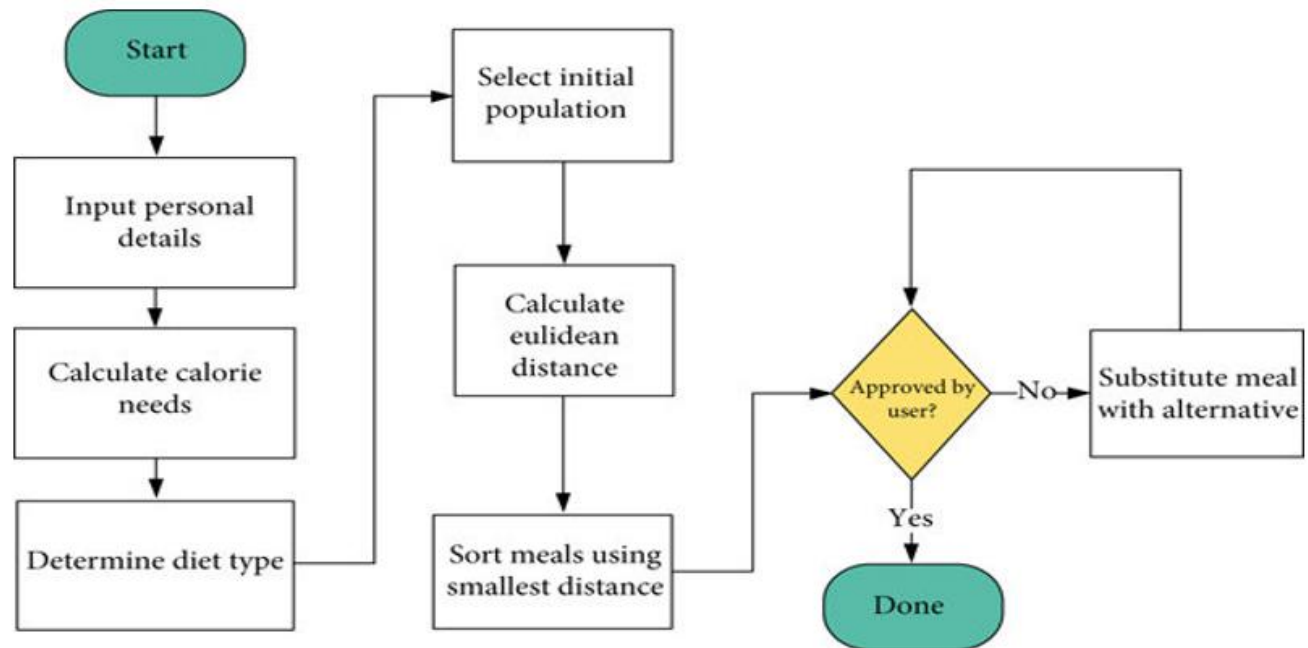
FR NO.	FUNCTIONAL REQUIREMENTS(EPIC)	SUB REQUIREMENT(STORY/SUBTASK)
FR-1	USER REGISTRATION	-Registration through Gmail  -Registration through Mobile Number -Registration through Face-book
FR-2	USER CONFIRMATION	-Confirmation via Email  -Confirmation via OTP
FR-3	USER DETAILS	PERSONAL DETAILS FOOD DETAILS Age Food Height Recipe Weight Added  ingredients Diseases if any Age Conditions is any Allergies is any
FR-4	USER REQUIREMENTS	-The user simply inputs your recipe ingredients and amounts. The software will instantly produce an accurate readout of your dish in terms of nutritional analysis in a readable format that consumers are familiar with.  -With already given details the system can alert the consumer if any content of their allergies ,it can alert the consumer

## 4.2 Non Functional Requirements

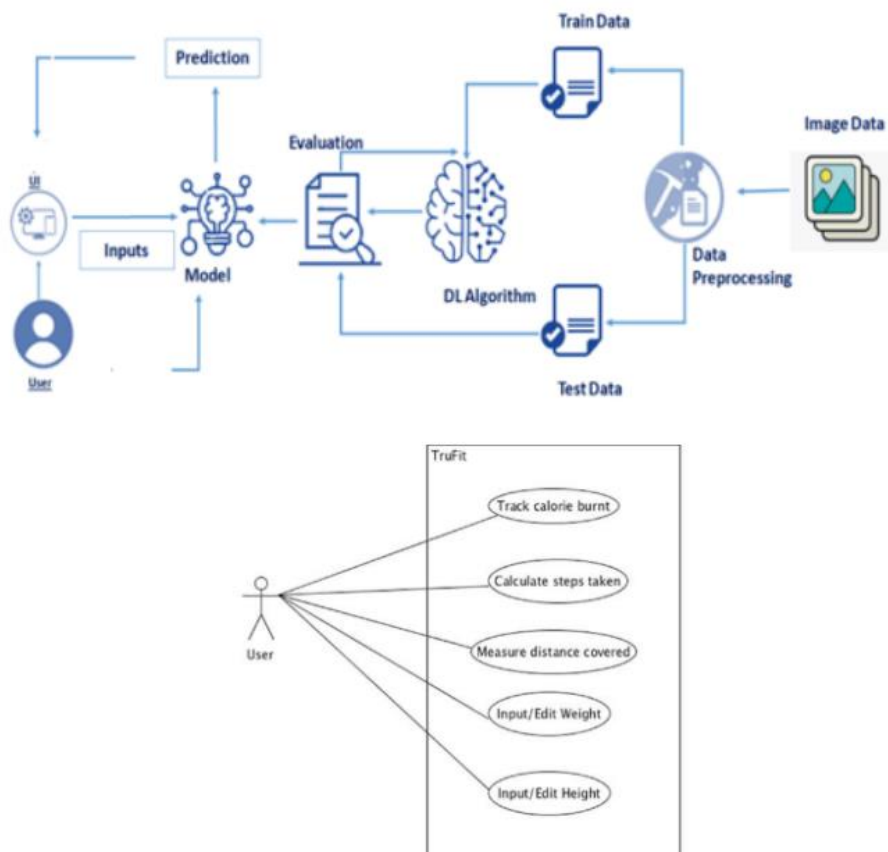
FR.NO	NON-FUNCTIONAL REQUIREMENTS	DESCRIPTION
NFR-1	USABILITY	<ul style="list-style-type: none"><li>❖ No training is required to access the Nutrition Analyzer.</li><li>❖ The results should be loaded within 30 seconds.</li><li>❖ It should be user friendly and comfortable.</li><li>❖ It should be simple and easy to use.</li><li>❖ The results should be self explanatory so that it can be understood by common people.</li></ul>
NFR-2	SECURITY	<ul style="list-style-type: none"><li>❖ AI powered nutrition analyzer for fitness should contain more security in which our data which entered or maintained should be more security.</li><li>❖ With the help of the username and password it provides more security in which it can access more securable and the data are private.</li><li>❖ It should be social-economic which should access to sufficient and safe to use.</li></ul>
		<ul style="list-style-type: none"><li>❖ It is Important that the AI powered nutrition analyzer for fitness provides should Must reliable.</li></ul>

## 5. PROJECT DESIGN

### 5.1 Data Flow Diagrams



### 5.2 Solution & Technical Architecture



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-2
	Login	USN-5	As a user, I can log into the application by entering email & password		High	Sprint-2
	Dashboard	USN-6	As a user, I can select the activity I wish to do in the application.	Desired activity is opted	High	Sprint-3
	Food recognition	USN-7	The food item under the lens is recognised	Food item is recognised	High	Sprint-4
	Nutrient Content	USN-8	Once the food item is recognised, its nutrient content is displayed to the user.	Nutrient content is displayed	High	Sprint-5
Administrator	Updation	USN-9	The food dataset is updated by the administrator periodically.		High	Sprint-6

## 6. PROJECT PLANNING & SCHEDULING

### 6.1 Sprint Planning & Estimation

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint1	Registration	USN-1	As a analyzer , I can register for the application by entering my email, password, and confirming my password	2	High	CHANDRAN V V
Sprint1		USN-2	As a analyzer, I can register for the application through Gmail	1	Medium	SOUNDARYAA R
Sprint1	User Confirmation	USN-3	As a analyzer, I will receive confirmation email once I have registered for the application	1	High	SANJAY A
Sprint1	Login	USN-4	As a analyzer, I canlog into the application by entering email & Password	2	High	VASANTH R

## 6.2 Sprint Delivery Schedule

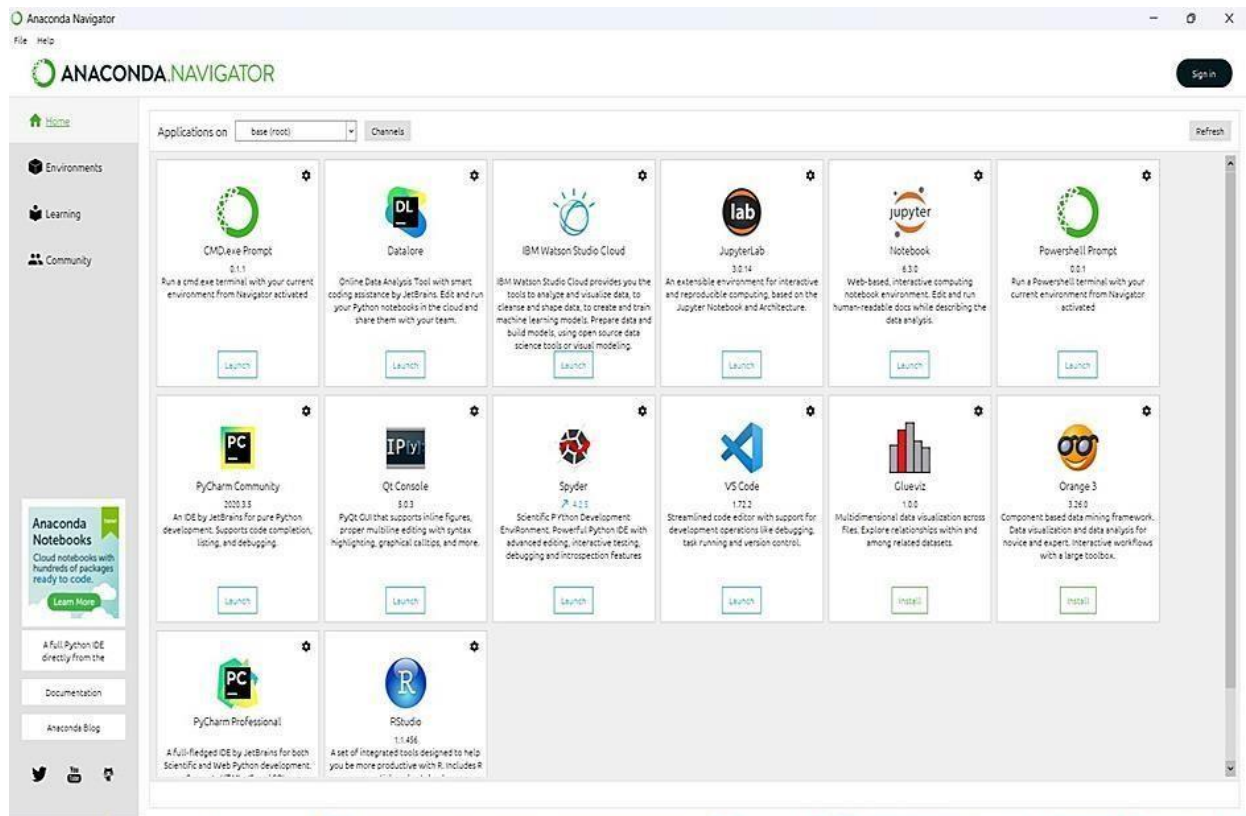
Sprint	Total Story Points	Total Story Points	Sprint Start Date	Sprint End Date (Planned)	Story Points Completed (as on Planned End Date)	Sprint Release Date(Actual)
Sprint-1	20	4 Days	24 Oct 2022	27 Oct 2022	20	29 Oct 2022
Sprint-2	20	5 Days	28 Oct 2022	01 Nov 2022	20	04 Nov 2022
Sprint-3	20	8 Days	02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

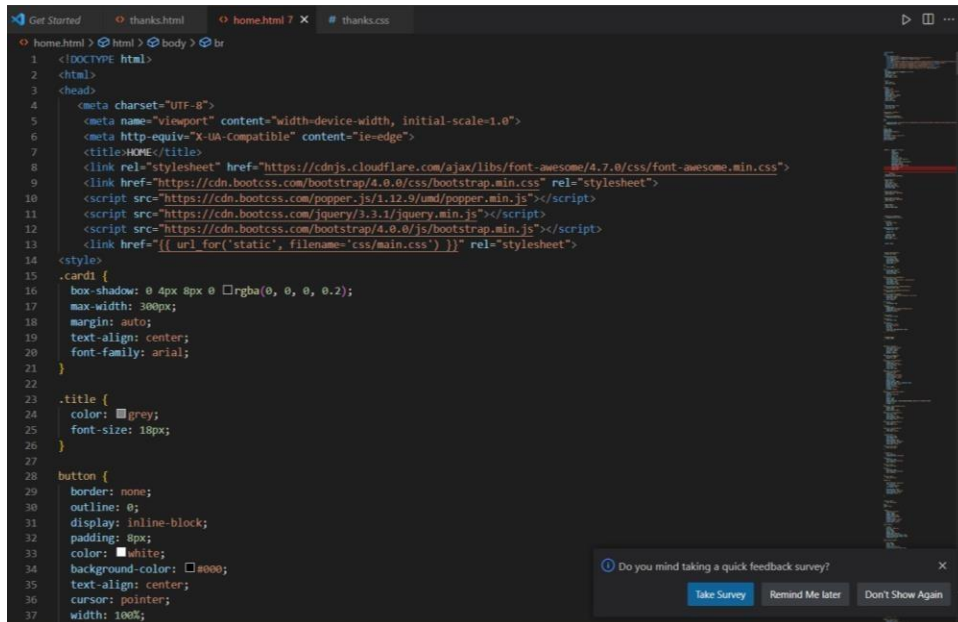
## 7. CODING & SOLUTIONING (Explain the features added in the project along with code)

### 7.1 Feature 1

If you are using anaconda navigator, follow the below steps to download the required packages:

1. Open anaconda prompt as administrator.
2. Type "pip install tensorflow==1.14.0" and click enter.
3. Type "pip install keras=2.2.4" and click enter.
4. Type "pip install opencv-python" and click enter. • Type "pip install imutils" and click enter





```
1 <!DOCTYPE html>
2 <html>
3 <head>
4   <meta charset="UTF-8">
5   <meta name="viewport" content="width=device-width, initial-scale=1.0">
6   <meta http-equiv="X-UA-compatible" content="ie=edge">
7   <title>HOME</title>
8   <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/4.7.0/css/font-awesome.min.css">
9   <link href="https://cdn.bootcss.com/bootstrap/4.0.0/css/bootstrap.min.css" rel="stylesheet">
10  <script src="https://cdn.bootcss.com/popper.js/1.12.9/umd/popper.min.js"></script>
11  <script src="https://cdn.bootcss.com/jquery/3.3.1/jquery.min.js"></script>
12  <script src="https://cdn.bootcss.com/bootstrap/4.0.0/js/bootstrap.min.js"></script>
13  <link href="{[ url_for('static', filename='css/main.css') ]}" rel="stylesheet">
14 </head>
15 <body>
16   <div class="card">
17     <div class="card-body">
18       <div class="text-align: center">
19         <h1>HOME</h1>
20         <button class="btn btn-primary">Click Me</button>
21       </div>
22     </div>
23   </div>
24 </body>
25 </html>
```

```
1 .card {
2   box-shadow: 0 4px 8px 0 rgba(0, 0, 0, 0.2);
3   max-width: 300px;
4   margin: auto;
5   text-align: center;
6   font-family: arial;
7 }
8
9 .title {
10  color: grey;
11  font-size: 18px;
12 }
13
14 button {
15  border: none;
16  outline: 0;
17  display: inline-block;
18  padding: 8px;
19  color: white;
20  background-color: #000000;
21  text-align: center;
22  cursor: pointer;
23  width: 100%;
24 }
```

## COLAB :

### Data Collection

Download the dataset [here](#)

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

Mounted at /content/drive

[ ] cd /content/drive/MyDrive/Colab Notebooks

/content/drive/MyDrive/Colab Notebooks

[ ] # Unzipping the dataset
    !unzip 'Dataset.zip'
```



## Image Preprocessing

```
[ ] from keras.preprocessing.image import ImageDataGenerator
```

## Image Data Augmentation

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

## Applying Image DataGenerator Functionality To Trainset And Testset

```
▶ x_train = train_datagen.flow_from_directory(
    r'/content/drive/MyDrive/Colab Notebooks/Dataset/TRAIN_SET',
    target_size=(64, 64), batch_size=5, color_mode='rgb', class_mode='sparse')
x_test = test_datagen.flow_from_directory(
    r'/content/drive/MyDrive/Colab Notebooks/Dataset/TEST_SET',
    target_size=(64, 64), batch_size=5, color_mode='rgb', class_mode='sparse')
```

### 3. Adding CNN Layers

```
[ ] 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())
```

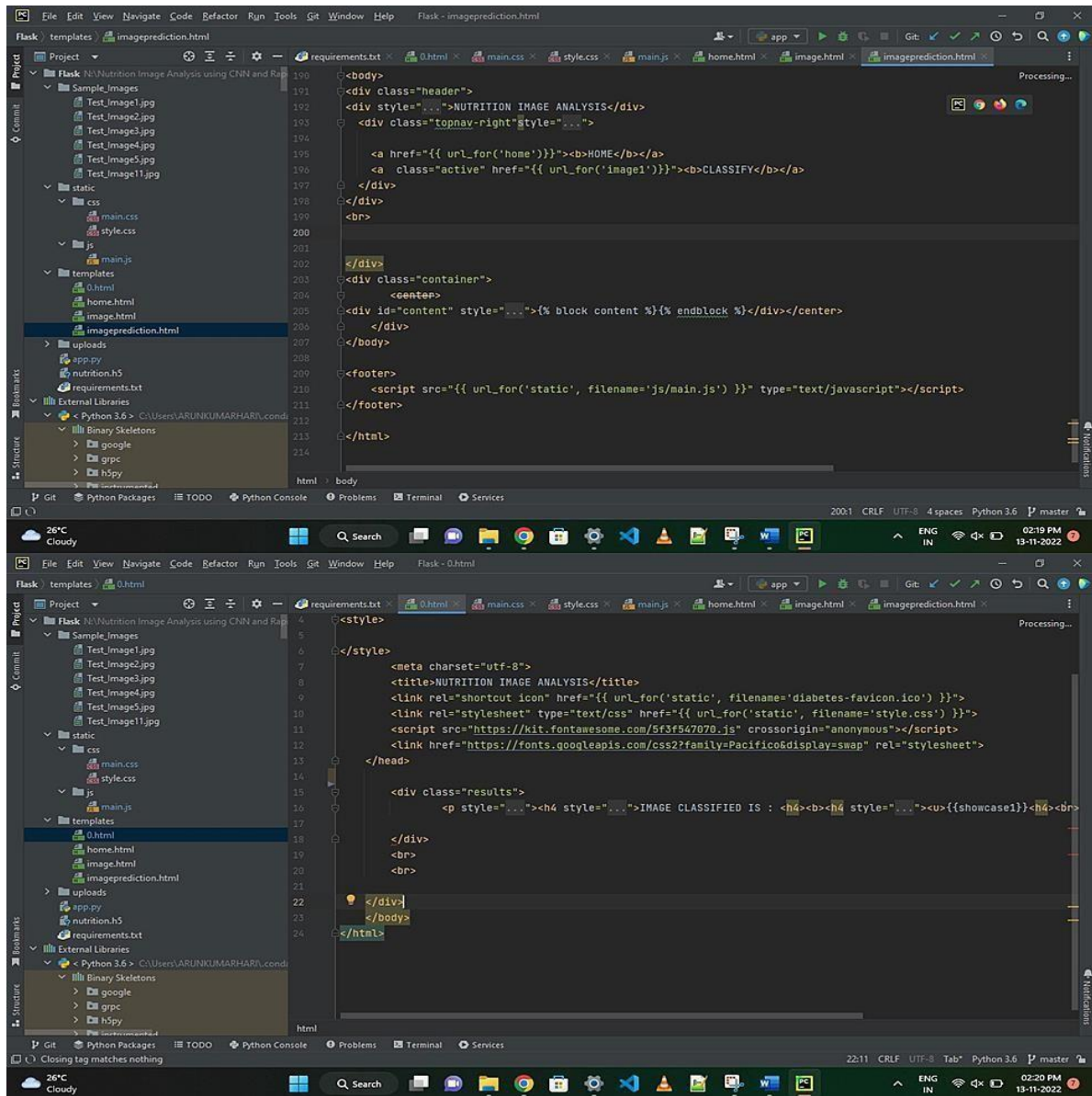
### 4. Adding Dense Layers

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

```
▶ classifier.summary()
```

Model: "sequential\_1"

Layer (type)	Output Shape	Param #
conv2d (Conv2D)	(None, 62, 62, 32)	896



#### 5. Configure The Learning Process

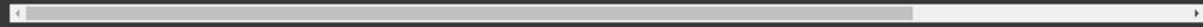
```
[ ] classifier.compile(optimizer='adam', loss='sparse_categorical_crossentropy', metrics=['accuracy'])
```

#### 6. Train The Model

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

/usr/local/lib/python3.7/dist-packages/ipykernel\_launcher.py:2: UserWarning: 'Model.fit\_generator' is deprecated and will be removed in a future version. P]

Epoch 1/20  
494/824 [=====>.....] - ETA: 6:52 - loss: 0.7194 - accuracy: 0.7174

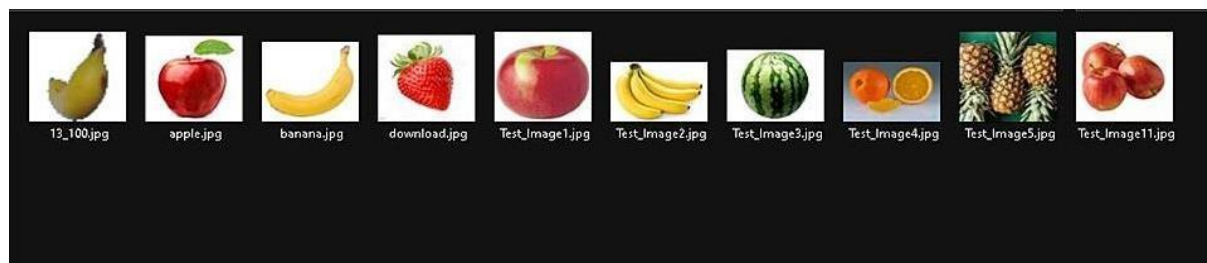
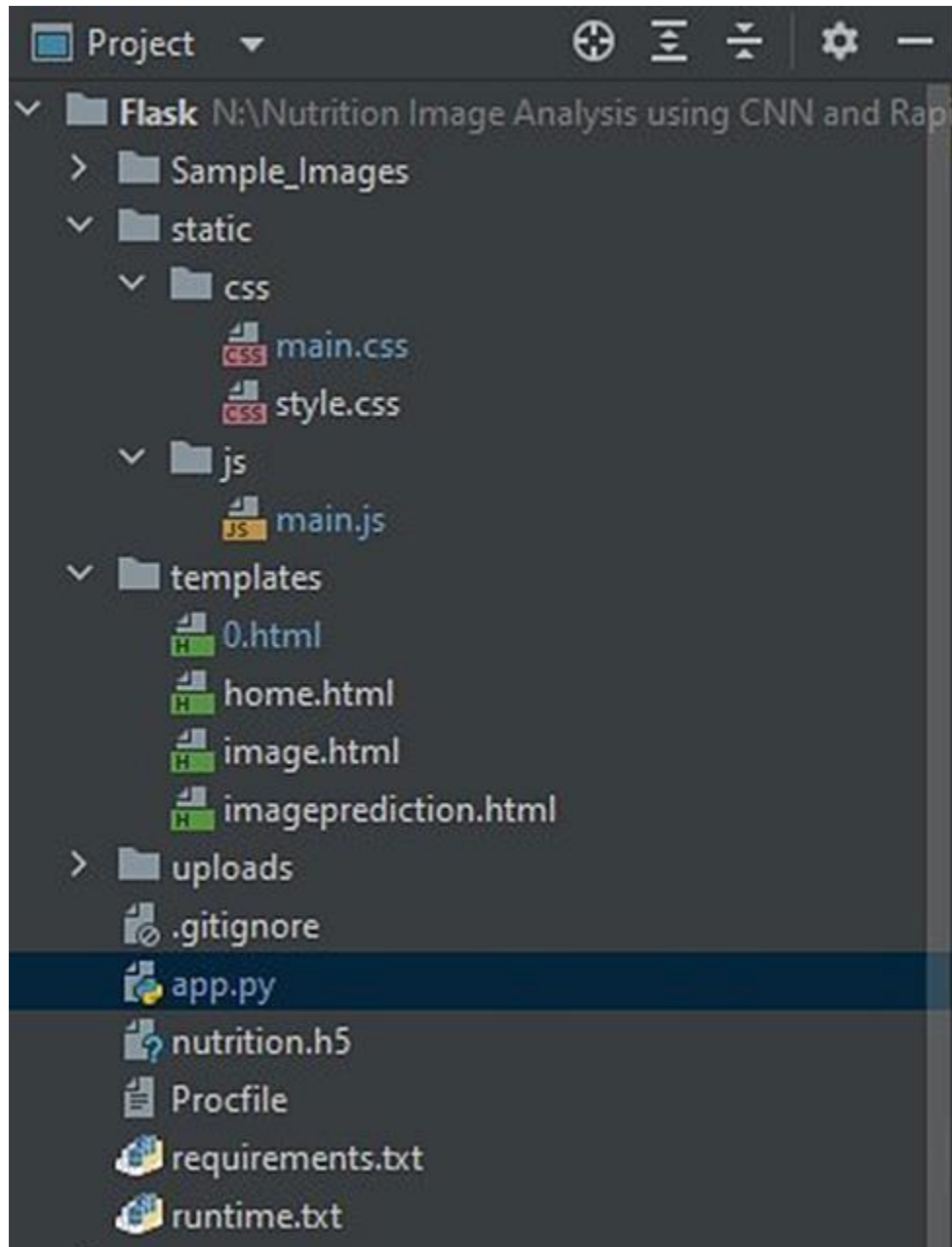


#### 7. Saving The Model

```
[ ] classifier.save('nutrition.h5')
```

## 8. TESTING

### 8.1 Test Cases



## 8.2 User Acceptance Testing



We are performing White Box Testing for select the package module.

### **Pseudocode for select the package module is-**

1. select\_the\_package () procedure begins
2. READ the package name, features, offers and duration from the package database
3. DISPLAY the package name, features, offers and duration
4. DO
5. GET the package name, features, offers and duration
6. STORE the package selected to the customer's database
7. PROCEED to payment screen //another module

8. WHILE select package is NULL

9. //End DO...WHILE

10.procedure ends

## **FLOWGRAPH**

### **CYCLOMATIC COMPLEXITY OF RESULTANT GRAPH**

$V(G) = \text{Number of regions}$

$= 2$

$V(G) = \text{Edges} - \text{Nodes} + 2$

$= 8 - 8 + 2$

$= 2$

$V(G) = \text{Predicate nodes} + 1$

$= 1 + 1$

$= 2$

### **LINEARLY INDEPENDENT PATHS FOR FLOW GRAPHS**

Path 1: 1-2-3-4-5-6-7-8-9-10

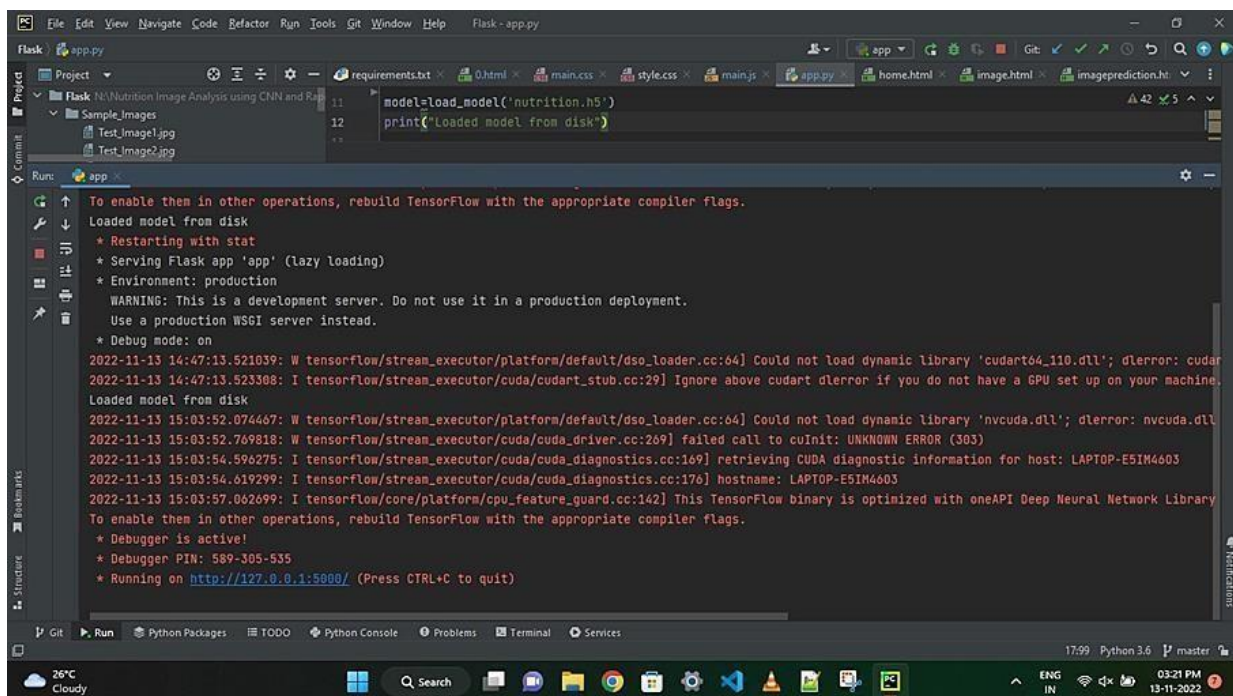
***Path 2: 1-2-3-4-5-6-7-8-4-5-6-7-8-9-10***

TEST ID	INPUT VALUES	ACTUAL OUTPUT	EXPECTED OUTPUT
1	Package is selected	To be observed after execution	Display the selected package
2	Package is not selected	To be observed after execution	Show the packages to select until one is selected

**TEST CASE TABLE**

## 9. RESULTS

### 9.1 Performance Metrics



```

Flask - app.py
Project: Flask - Nutrition Image Analysis using CNN and RNN
Sample_Images
Test_Image1.jpg
Test_Image2.jpg
requirements.txt
0.html
main.css
style.css
main.js
app.py
home.html
image.html
imageprediction.ht

11 model=load_model('nutrition.h5')
12 print("Loaded model from disk")

Run: app
To enable them in other operations, rebuild TensorFlow with the appropriate compiler flags.
Loaded model from disk
* Restarting with stat
* Serving Flask app 'app' (lazy loading)
* Environment: production
  WARNING: This is a development server. Do not use it in a production deployment.
  Use a production WSGI server instead.
* Debug mode: on
2022-11-13 14:47:13.521039: W tensorflow/stream_executor/platform/default/dso_loader.cc:64] Could not load dynamic library 'cudart64_110.dll'; dlderror: cudart
2022-11-13 14:47:13.523308: I tensorflow/stream_executor/cuda/cudart_stub.cc:29] Ignore above cudart dlerror if you do not have a GPU set up on your machine.
Loaded model from disk
2022-11-13 15:03:52.074467: W tensorflow/stream_executor/platform/default/dso_loader.cc:64] Could not load dynamic library 'nvcuda.dll'; dlderror: nvcuda.dll
2022-11-13 15:03:52.769818: W tensorflow/stream_executor/cuda/cuda_driver.cc:269] failed call to cuInit: UNKNOWN ERROR (303)
2022-11-13 15:03:54.596275: I tensorflow/stream_executor/cuda/cuda_diagnostics.cc:169] retrieving CUDA diagnostic information for host: LAPTOP-E5IM4603
2022-11-13 15:03:54.619299: I tensorflow/stream_executor/cuda/cuda_diagnostics.cc:176] hostname: LAPTOP-E5IM4603
2022-11-13 15:03:57.062699: I tensorflow/core/platform/cpu_feature_guard.cc:142] This TensorFlow binary is optimized with oneAPI Deep Neural Network Library
To enable them in other operations, rebuild TensorFlow with the appropriate compiler flags.
* Debugger is active!
* Debugger PIN: 589-305-535
* Running on http://127.0.0.1:5000/ (Press CTRL+C to quit)
  
```



## **10.ADVANTAGES & DISADVANTAGES**

### **ADVANTAGES:**

1. Monitor Your Diet Easily. Weight watchers or people who want to gain weight can mention the type and amount of foods consumed at each meal.
2. Monitor Your Progress.
3. Give Free Health and Fitness Tips.
4. Provide Personal Health Coaches.

### **DISADVANTAGES:**

1. In some cases, it can lead us in failing sick.
2. Cost of using this application could be high.

## **11.CONCLUSION**

Engaging in regular physical activity may produce improvements in an individual's physical health, cognitive performance, and psychological well-being. Physical benefits include, but are not limited to, reduced risk for diseases, and improvements in physical functioning, fitness, and overall quality of life. The purpose of a fitness app is to provide the user with instructions and examples of one or more types of exercise, physical activity, nutritional programs.

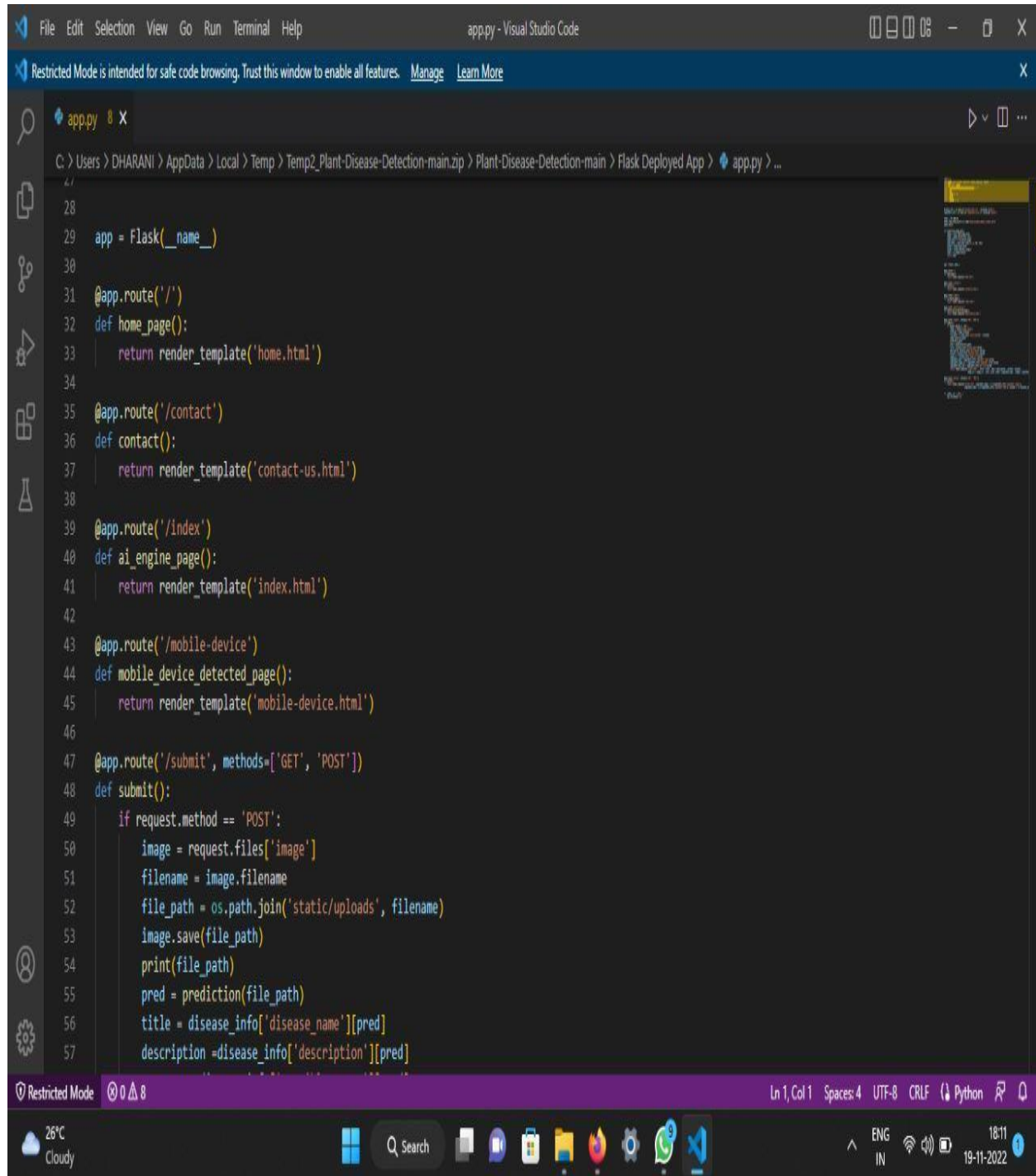
## **12.FUTURE SCOPE**

- Offers payment convenience in personal training subscriptions.
- Useful articles.
- Video instructions.
- Diet Plans.
- Individual progress tracking.
- Live video from training sessions



## 13.APPENDIX

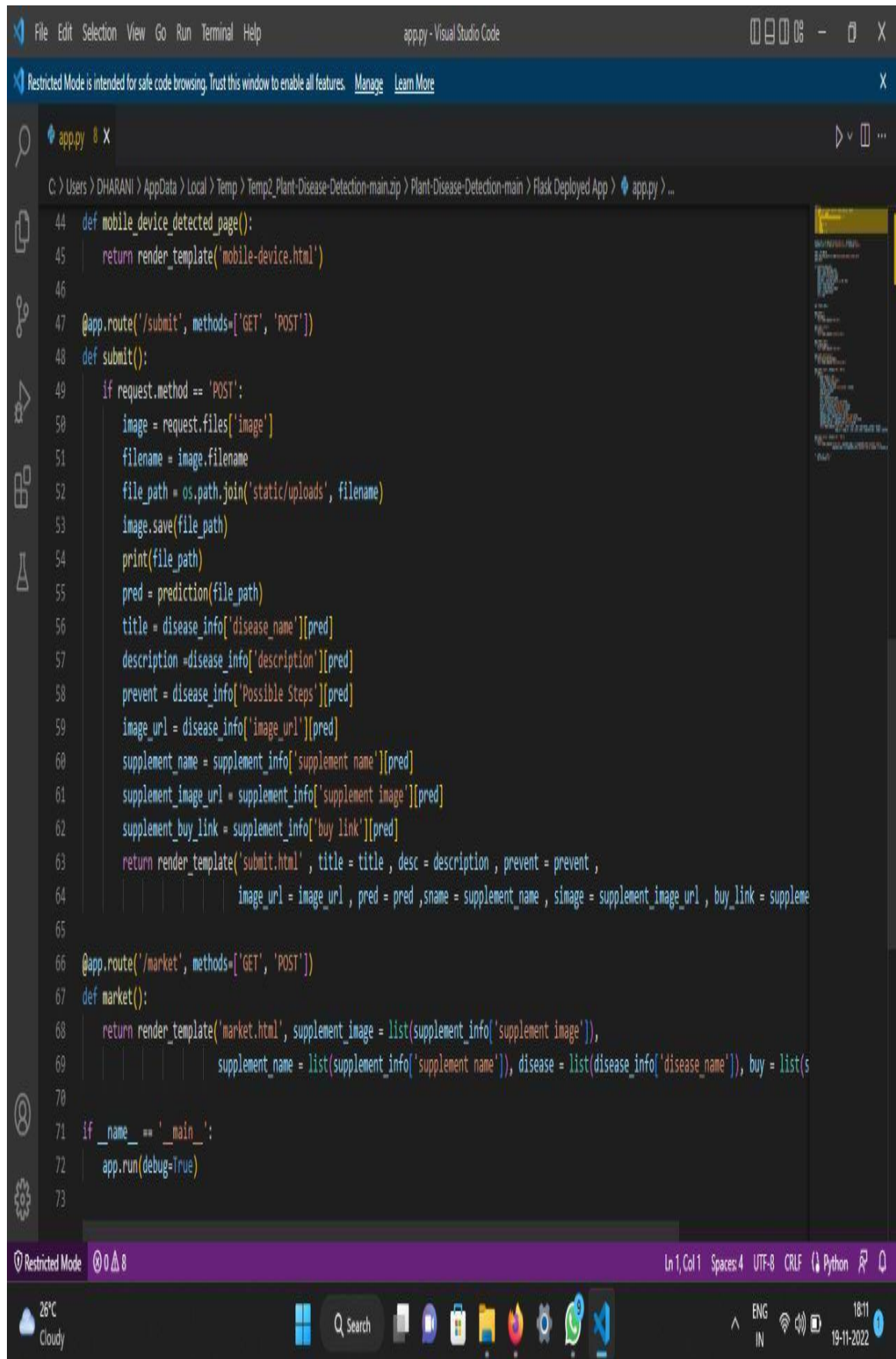
### Source Code



```
File Edit Selection View Go Run Terminal Help
app.py - Visual Studio Code
Restricted Mode is intended for safe code browsing. Trust this window to enable all features. Manage Learn More
app.py 8 X
C:\Users\DHARANI> AppData\Local>Temp>Temp2_Plant-Disease-Detection-main.zip> Plant-Disease-Detection-main> Flask Deployed App> app.py > ...
28
29 app = Flask(__name__)
30
31 @app.route('/')
32 def home_page():
33     return render_template('home.html')
34
35 @app.route('/contact')
36 def contact():
37     return render_template('contact-us.html')
38
39 @app.route('/index')
40 def ai_engine_page():
41     return render_template('index.html')
42
43 @app.route('/mobile-device')
44 def mobile_device_detected_page():
45     return render_template('mobile-device.html')
46
47 @app.route('/submit', methods=['GET', 'POST'])
48 def submit():
49     if request.method == 'POST':
50         image = request.files['image']
51         filename = image.filename
52         file_path = os.path.join('static/uploads', filename)
53         image.save(file_path)
54         print(file_path)
55         pred = prediction(file_path)
56         title = disease_info['disease_name'][pred]
57         description = disease_info['description'][pred]
```

Ln 1, Col 1 Spaces: 4 UTF-8 CRLF Python

26°C Cloudy Search ENG IN 18:11 19-11-2022



```
File Edit Selection View Go Run Terminal Help
app.py - Visual Studio Code
Restricted Mode is intended for safe code browsing. Trust this window to enable all features. Manage Learn More
app.py x
C:\Users\DHARANI> AppData\Local\Temp\Temp2_Plant-Disease-Detection-main.zip\Plant-Disease-Detection-main> Flask Deployed App > app.py > ...
44 def mobile_device_detected_page():
45     return render_template('mobile-device.html')
46
47 @app.route('/submit', methods=['GET', 'POST'])
48 def submit():
49     if request.method == 'POST':
50         image = request.files['image']
51         filename = image.filename
52         file_path = os.path.join('static/uploads', filename)
53         image.save(file_path)
54         print(file_path)
55         pred = prediction(file_path)
56         title = disease_info['disease_name'][pred]
57         description = disease_info['description'][pred]
58         prevent = disease_info['Possible Steps'][pred]
59         image_url = disease_info['image_url'][pred]
60         supplement_name = supplement_info['supplement name'][pred]
61         supplement_image_url = supplement_info['supplement image'][pred]
62         supplement_buy_link = supplement_info['buy link'][pred]
63         return render_template('submit.html', title = title, desc = description, prevent = prevent,
64                                image_url = image_url, pred = pred, sname = supplement_name, simage = supplement_image_url, buy_link = supplement_buy_link)
65
66 @app.route('/market', methods=['GET', 'POST'])
67 def market():
68     return render_template('market.html', supplement_image = list(supplement_info['supplement image']),
69                            supplement_name = list(supplement_info['supplement name']), disease = list(disease_info['disease_name']), buy = list(supplement_info['buy link']))
70
71 if __name__ == '__main__':
72     app.run(debug=True)
73
Ln 1, Col 1 Spaces: 4 UTF-8 CRLF Python
26°C Cloudy
Search
19-11-2022 18:11
```

```
File Edit Selection View Go Run Terminal Help app.py - Visual Studio Code
Restricted Mode is intended for safe code browsing. Trust this window to enable all features. Manage Learn More

app.py 8 X
C:\Users\DHARANI> AppData\Local\Temp\Temp2_Plant-Disease-Detection-main.zip\Plant-Disease-Detection-main\Flask Deployed App> app.py > ...

28
29 app = Flask(__name__)
30
31 @app.route('/')
32 def home_page():
33     return render_template('home.html')
34
35 @app.route('/contact')
36 def contact():
37     return render_template('contact-us.html')
38
39 @app.route('/index')
40 def ai_engine_page():
41     return render_template('index.html')
42
43 @app.route('/mobile-device')
44 def mobile_device_detected_page():
45     return render_template('mobile-device.html')
46
47 @app.route('/submit', methods=['GET', 'POST'])
48 def submit():
49     if request.method == 'POST':
50         image = request.files['image']
51         filename = image.filename
52         file_path = os.path.join('static/uploads', filename)
53         image.save(file_path)
54         print(file_path)
55         pred = prediction(file_path)
56         title = disease_info['disease_name'][pred]
57         description = disease_info['description'][pred]
```

Restricted Mode 0 8 Ln 1, Col 1 Spaces: 4 UTF-8 CRLF Python 18:11 19-11-2022

**GitHub Link :**

**<https://github.com/IBM-EPBL/IBM-Project-36791-1660297912>**