

AI-powered Nutrition Analyzer for Fitness Enthusiasts

Team ID:PNT2022TMID25302

Project Report Format

1. INTRODUCTION

1.1 Project Overview

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

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

2. LITERATURE SURVEY

2.1 Existing problem

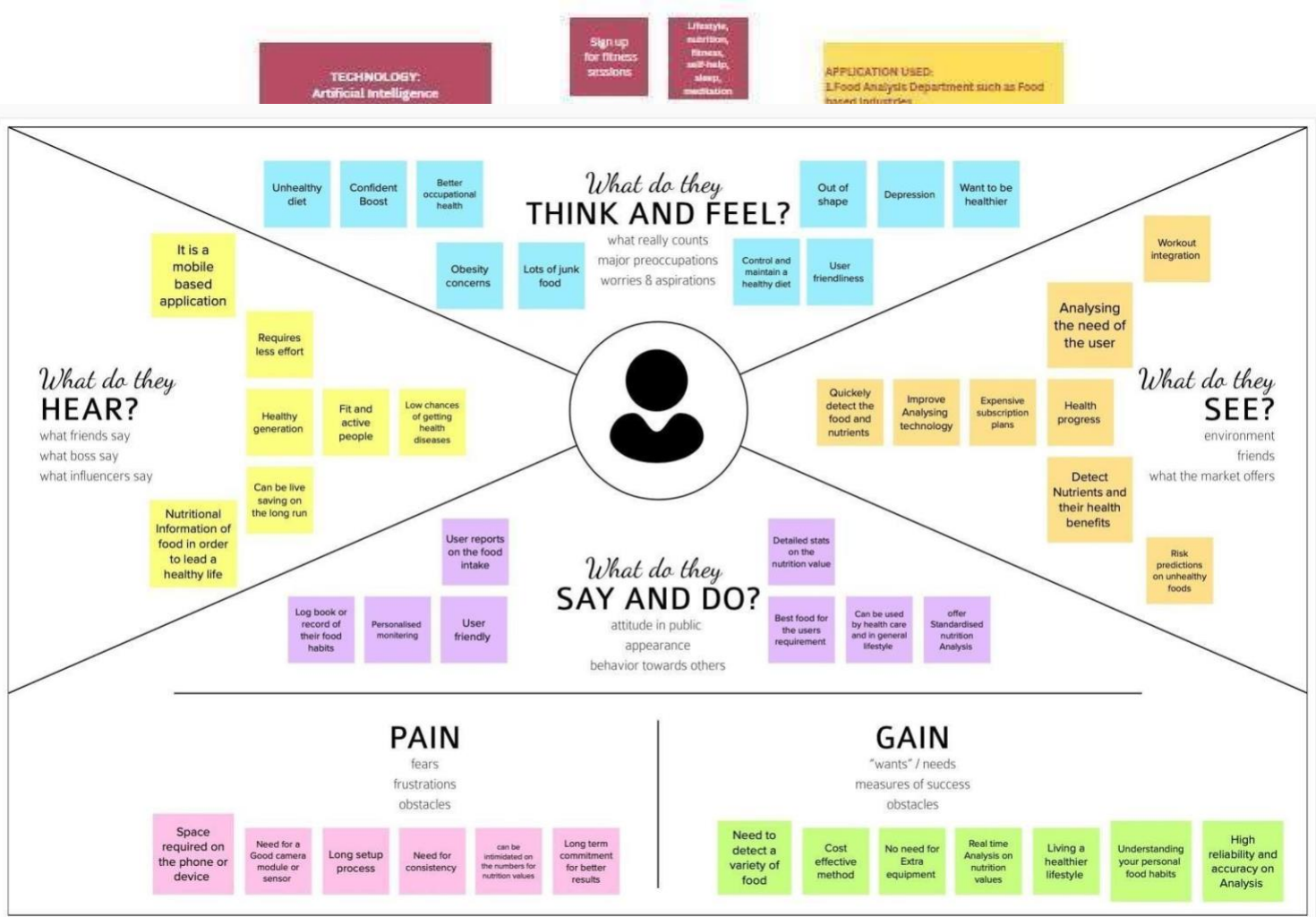
Neutrino delivers nutrition-based data services and analytics to its users and wants to turn into a leading source of the nutrition-related platform. The platform employs NLP and mathematical models from the optimization theory as well as predictive analysis to enable individualized data compilation.

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.

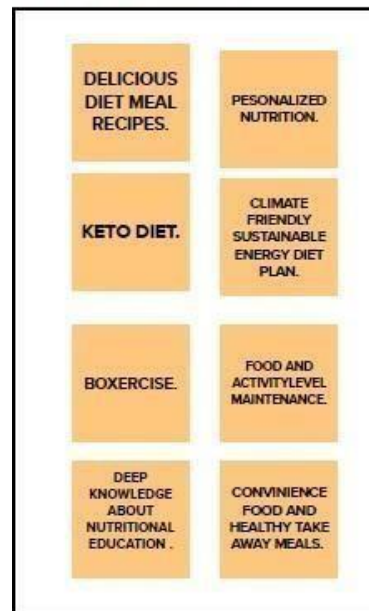
2.3 Problem Statement Definition

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.).

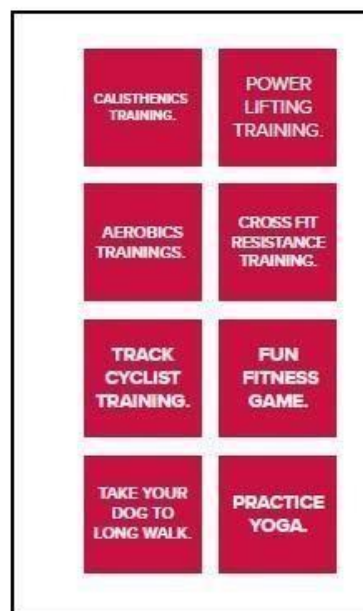
3. IDEATION & PROPOSED SOLUTION



NUTRITIONAL



WORKOUT



PROGRAMS



3.2 Proposed Solution

S.NO	PARAMETERS	SOLUTIONS
1.	Problem Statement	<ul style="list-style-type: none"> Main objective is to detect the nutrition in a fruit from camera captured image. The identification of nutrition and calories from a image is quite an interesting field. Since nutrition monitoring plays an important role in leading healthy lifestyle, this product has the potential to become an essential in our day to day life.
2.	Idea / Solution description	<ul style="list-style-type: none"> The solution is to develop AI-powered nutrition analyzer application. By giving the image of the fruit as the input to the application, it will display the nutrition content in it. By training the model with various inputs, image processing can be improved as well as the accuracy of the result.
3.	Novelty / Uniqueness	<ul style="list-style-type: none"> Personalized nutrition for individuals. Providing science based guidance for healthy living. Balanced food diet and measured intake. 24/7 support. Serving size.

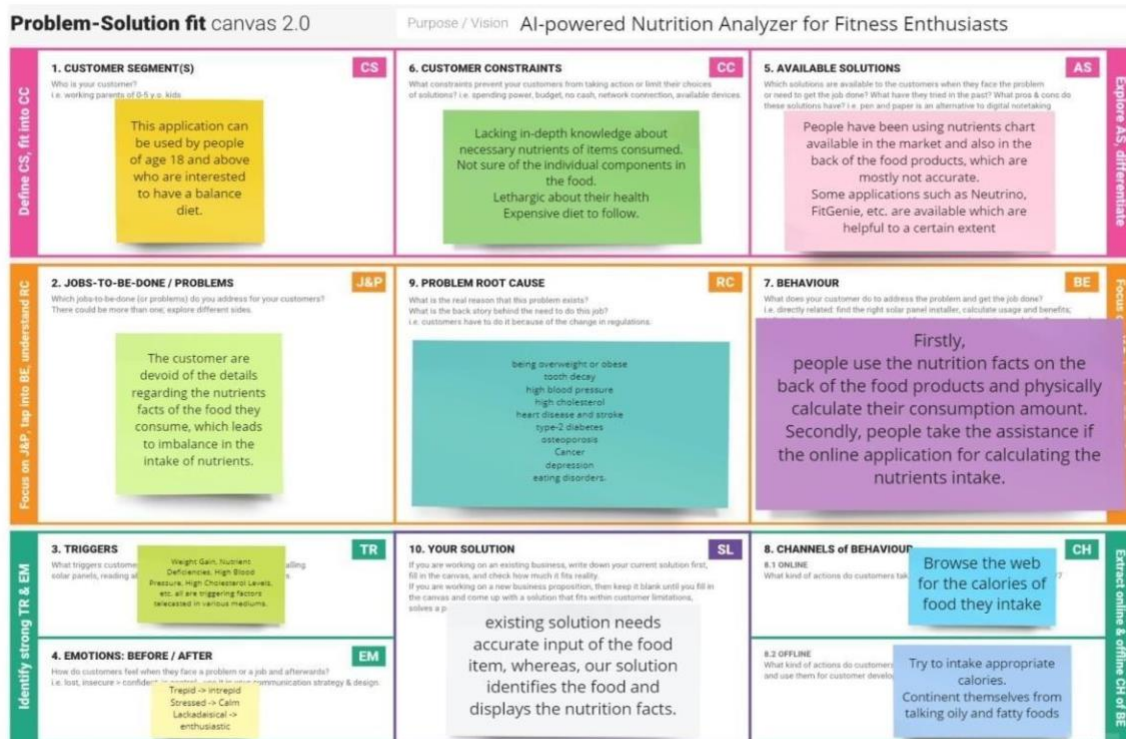
4.	Social Impact / Customer Satisfaction	<ul style="list-style-type: none"> • Economically stable product. • Change one's view towards health and fitness. • Quality of service. • High fiber food. • Accurate amount of nutrition.
5.	Business Model (Revenue Model)	<ul style="list-style-type: none"> • User friendly interface which improves the constant use of the product. • Hence, Economical growth improves. • Product will be delivered in pocket size which results in consuming low memory. • Nutrition and fitness related ads to earn profit
6.	Scalability of the Solution	<ul style="list-style-type: none"> • Offers ingredients substance detail in food • Suggest best health solution and meal plans for different criteria proposed by different individuals. • Virtualization of your long term plan to provide motivation to the customer.

3.4 Problem Solutionfit

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 and building trust by solving frequent annoyances, or urgent or costly problems.



4. REQUIREMENT ANALYSIS

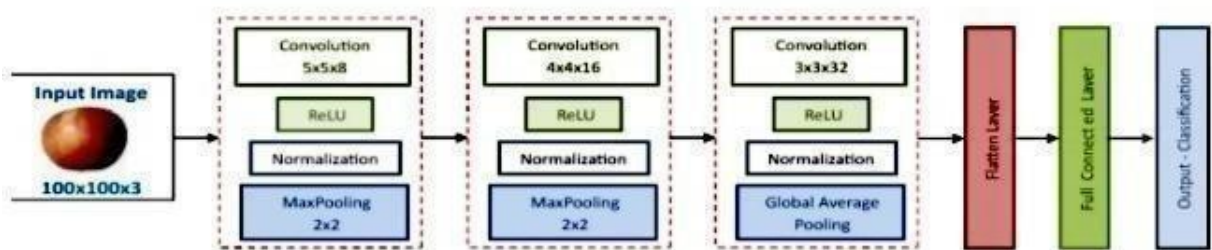
4.1 Functional requirement

- 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
- Computer-Assisted Nutritional Recognize Food Images - In order to solve this issue, a brand-new Convolutional Neural Network (CNN)- based food picture identification system was

created, as described in this study. We utilized our suggested strategy on two sets of actual food picture data.

- 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.)
- The Ultimate Workout at Home Solution This fitness AI software is designed with personalized training regimens for each individual. 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.

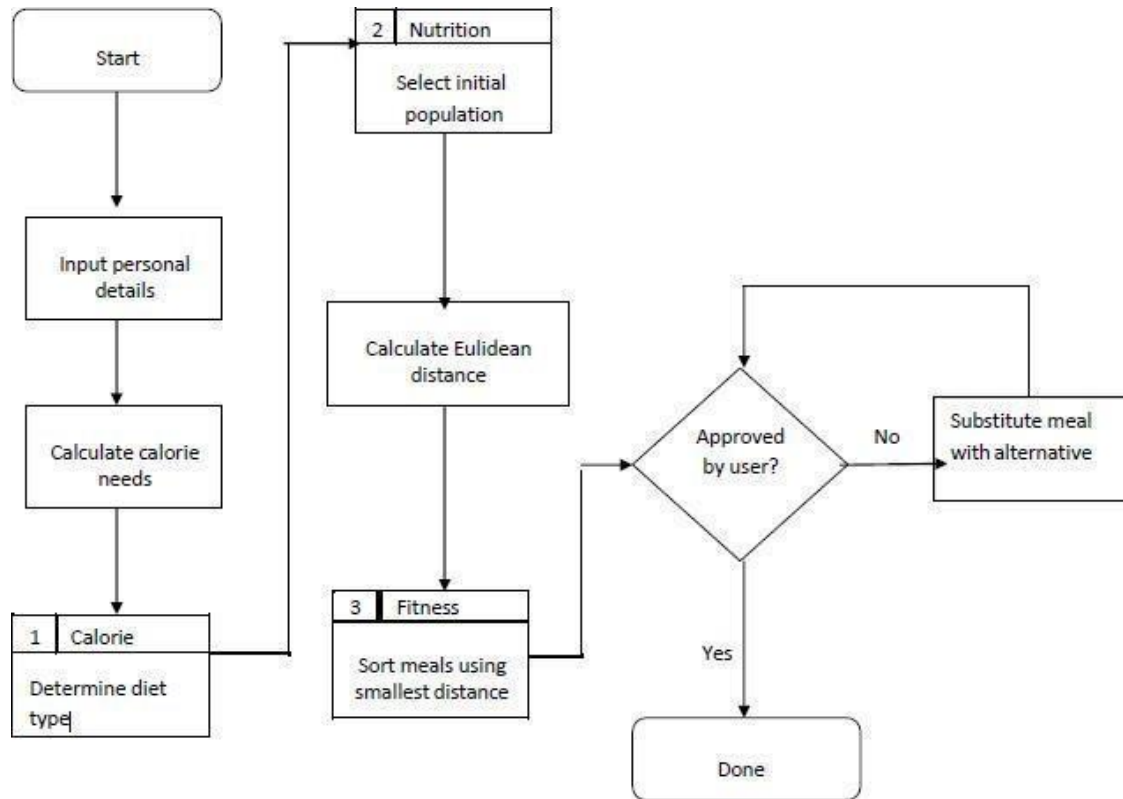
For Example:



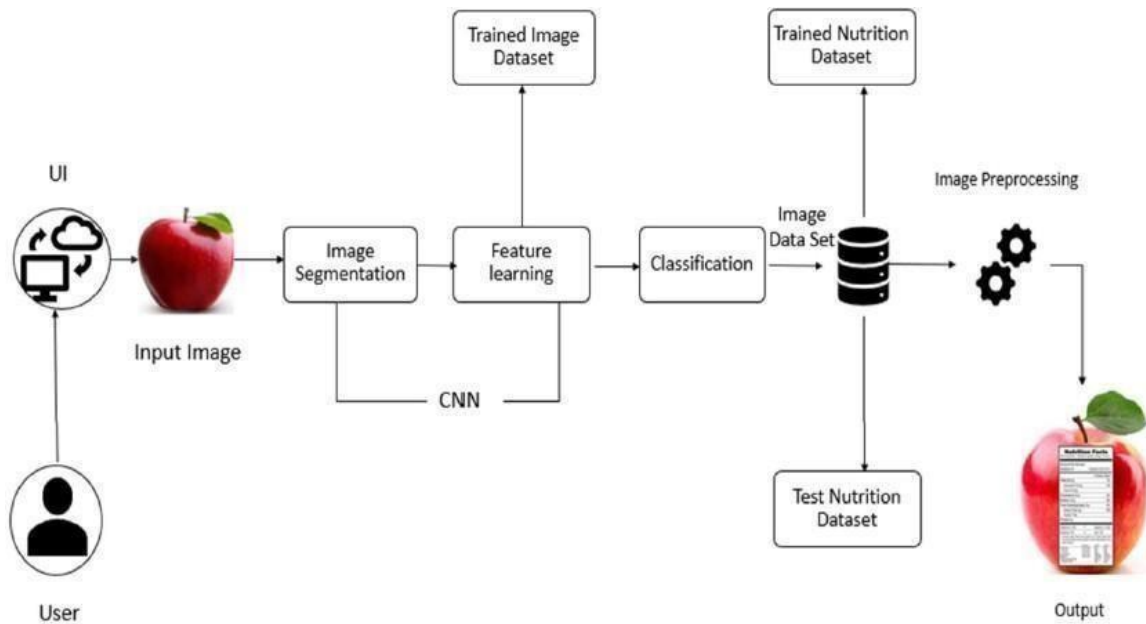
- The comparison of the proposed model with the conventional models shows that the results of this model are exceptionally good and promising to use in real-world applications.
- This sort of higher accuracy and precision will work to boost the machine’s general efficiency in fruit recognition more appropriately.
- A generic 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 of utilization.

5. PROJECT DESIGN

5.1 Data Flow Diagrams



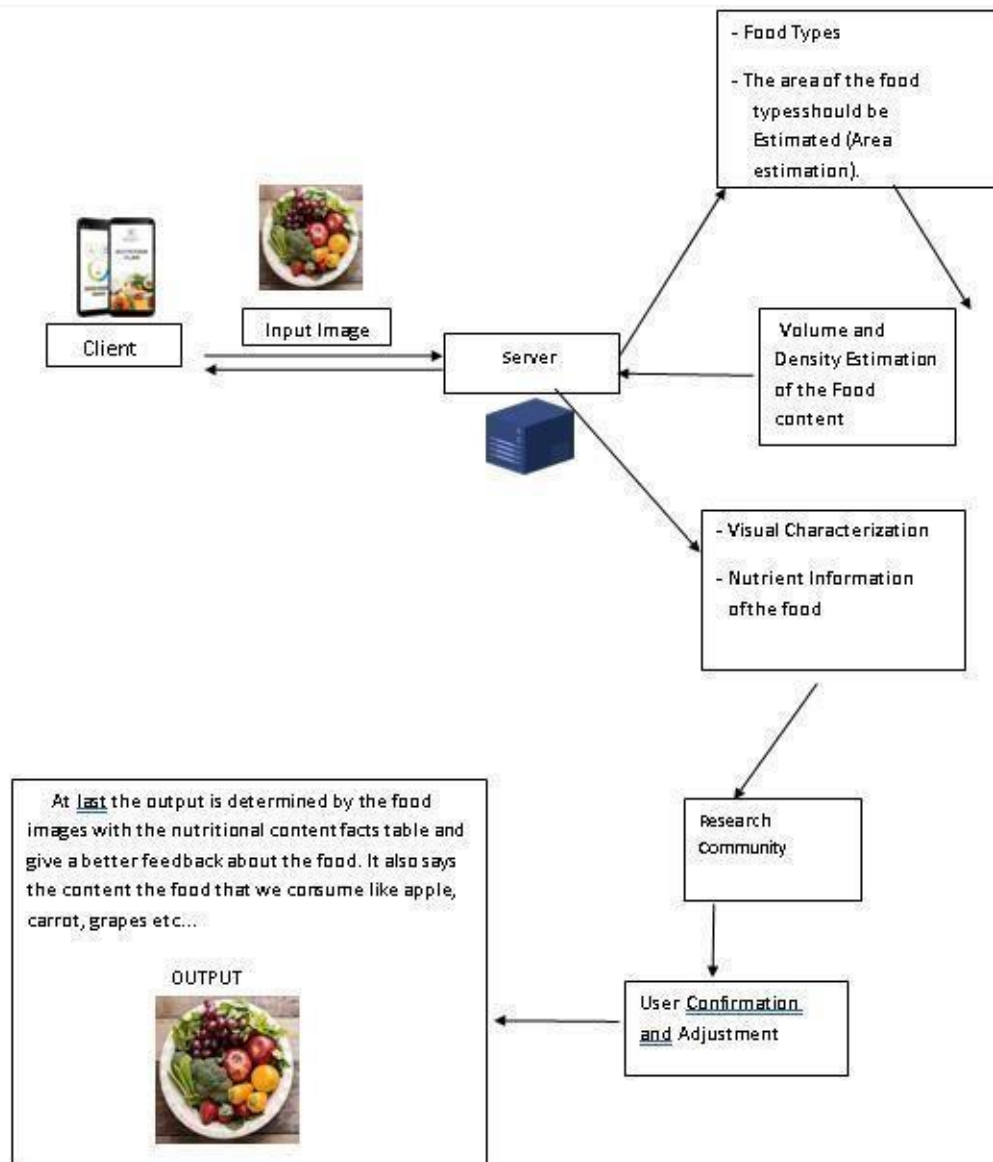
5.2 Solution & Technical Architecture



S.No	Component	Description	Technology
1.	App	User interacts with application	Python, Java, HTML, SQLite, Android studio
2.	Database	for the prediction of Nutrition Data Type, Configurations and	MySQL, JS
3.	Cloud Database	data will be stored Database Service on Cloud	IBM DB2, IBM Cloudant etc.
4.	File Storage	File storage requirements	Cloud -- > drive
5.	Machine Learning Model	Purpose of Machine Learning Model	ANN, CNN, RNN
6.	Notification	Notification will be sent from the server	SendGrid

Application Characteristics:

S.No	Characteristics	Description	Technology
1.	Open-Source Frameworks	Open-source frameworks used	SendGrid, Python, JQuery
2.	Security Implementations	Request authentication using encryption	Encryptions, SSL certs
3.	Scalable Architecture	The scalability of architecture consists of 3 tiers	Web Server - HTML, CSS ,Javascript Application Server -Python Flask Database Server - IBM Cloud IBM Cloud hosting
4.	Availability	Availability is increased by loads balancers in cloud VPS	
5.	Performance	The application is expected to handle up to 4000 predications per second	IBM Load Balance



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

6.1 Feature 1

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')
```

Model Building

1. Importing The Model Building Libraries

```
[ ] import numpy as np
import tensorflow as tf
from tensorflow.keras.models import Sequential
from tensorflow.keras import layers
from tensorflow.keras.layers import Dense, Flatten
from tensorflow.keras.layers import Conv2D, MaxPooling2D, Dropout
```

2. Initializing The Model

```
[ ] classifier = Sequential()
```

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

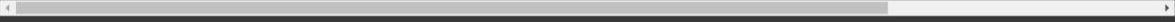
```
[ ] classifier.compile(optimizer='adam', loss='sparse_categorical_crossentropy', metrics=['accuracy'])
```

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: 0:52 - loss: 0.7194 - accuracy: 0.7174



7. Saving The Model

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

8. Testing The Model

```
[ ] from tensorflow.keras.models import load_model  
from keras.preprocessing import image  
model = load_model("nutrition.h5")
```

```
from tensorflow.keras.models import load_model  
from tensorflow.keras.preprocessing import image  
model = load_model("nutrition.h5")  
img = image.load_img(r'/content/drive/MyDrive/Colab Notebooks/Sample_Images/Test_Image1.jpg',grayscale=False,target_size= (64,64))  
x = img_to_array(img)  
x = np.expand_dims(x,axis = 0)  
predict_x=model.predict(x)  
classes_x=np.argmax(predict_x,axis=-1)  
classes_x
```

1/1 [=====] - 0s 62ms/step
array([0])

```
[ ] index=['APPLES', 'BANANA', 'ORANGE','PINEAPPLE','WATERMELON']  
result=str(index[classes_x[0]])  
result
```

6.2 Feature 2

FOLDERS

templates

O.html

home.html

image.html

imageprediction

home.html

X

O.html

<!-- templates -->

```
1 <!DOCTYPE html>
2 <html lang="en" dir="ltr">
3 <head>
4 <style>
5
6 </style>
7 <meta charset="utf-8">
8 <title>Nutrition Image Analysis</title>
9 <link rel="shortcut icon" href="{{ url_for('static', filename='diabetes-favicon.ico') }}">
10 <link rel="stylesheet" type="text/css" href="{{ url_for('static', filename='style.css') }}">
11 <script src="https://kit.fontawesome.com/5f3f547870.js" crossorigin="anonymous"></script>
12 <link href="https://fonts.googleapis.com/css2?family=Pacifico&display=swap" rel="stylesheet">
13 </head>
14
15 <!-- Result -->
16 <div class="results">
17     <!-- <p style="padding-top: 150px; color:blue;"><h4 style="color:blue;">Food Classified is: <b><br>
18         style="color:red;"><u>{{showcase}}<b><br><h4 style="color:red;"><u>{{showcase}}<b><br>
19         <p style="padding-top: 150px;width: 122px;color:#F2E7D5;text-align: left;"> <span style="text-align: center;">Food Classified:</span>
20         <span style="text-align: center;"><br>{{showcase}}</span><br>{{showcase}}<br></p>
21     </div>
22 <br>
23 <br>
24 </div>
25 </body>
26 </html>
```

Line 1, Column 1

Tab Size: 4

HTML

The screenshot shows the VS Code editor interface with the following details:

- Editor Title Bar:** home.html
- FOLDERS Panel:** templates, 0.html, home.html, image.html, imageprediction.
- Main Editor:** Displays the content of index.html. The code includes a DOCTYPE declaration, meta tags for charset and viewport, Bootstrap CSS and JS links, and a series of CSS rules for styling a hero section, a navigation bar, and two columns of links.
- Status Bar:** Line 1, Column 1: Detect Indentation, Setting indentation to 2 spaces

```

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 href="https://cdn.bootcss.com/bootstrap/4.0.0/css/bootstrap.min.css" rel="stylesheet">
9   <script src="https://cdn.bootcss.com/popper.js/1.12.0/umd/popper.min.js"></script>
10  <script src="https://cdn.bootcss.com/jquery/3.3.1/jquery.min.js"></script>
11  <script src="https://cdn.bootcss.com/bootstrap/4.0.0/js/bootstrap.min.js"></script>
12  <link href="{{ url_for('static', filename='css/main.css') }}" rel="stylesheet">
13 </head>
14 <body>
15 {
16   background-image: url("https://www.livingproofnyc.com/wp-content/themes/livingproof/assets/img/hero-background.jpg");
17   background-size: cover;
18 }
19 .bar
20 {
21   margin: 0px;
22   padding: 20px;
23   background-color:white;
24   opacity:0.6;
25   color:black;
26   font-family: Roboto ,sans-serif;
27   font-style: italic;
28   border-radius:20px;
29   font-size:25px;
30 }
31 h3
32 {
33   margin: 0px;
34   padding:20px;
35   background-color:#A8E4CB;
36   width: 800px;
37   opacity:0.6;
38   color:ffffff;
39   font-family: Roboto ,sans-serif;
40   font-style: italic;
41   border-radius:20px;
42   font-size:25px;
43 }
44 a
45 {
46   color:grey;
47   float:right;
48   text-decoration:none;
49   font-style:normal;
50   padding-right:20px;
51 }
52 a:hover{
53   background-color:black;
54   color:white;

```

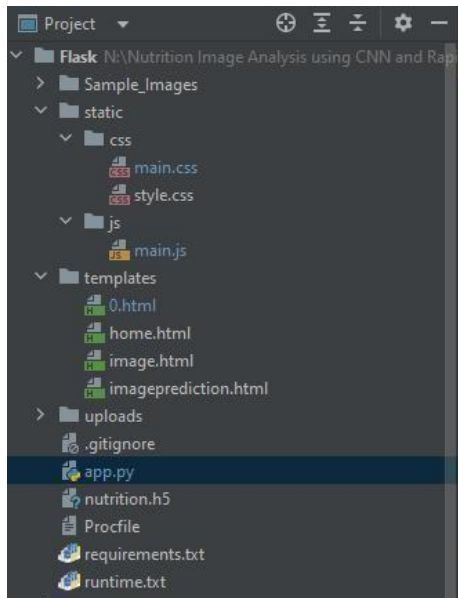


```
home.html image.html — templates UNREGISTERED
FOLDERS
  templates
    0.html
    home.html
    image.html
    imageprediction.html
  imageprediction.html
  imageprediction.html

1
2
3
4 {% extends "imageprediction.html" %} {% block content %}
5 <div style="float:left">
6 <br>
7 <br>
8 <h5><font color="#F2E7D5" size="3" font-family="sans-serif"><b>Upload image to classify</b></font></h5><br><br>
9
10 <div>
11   <form id="upload-file" method="post" enctype="multipart/form-data">
12     <label form="imageUpload" class="upload-label" style="background-color: #FF9F9F;color: black;">
13       Choose...
14     </label>
15     <input type="file" name="file" id="imageUpload" accept=".png, .jpg, .jpeg">
16   </form>
17
18   <center> <div class="image-section" style="display:none;">
19     <div class="img-preview">
20       <div id="imagePreview">
21         </div></center>
22     </div>
23     <center><div>
24       <button type="button" class="btn btn-primary btn-lg" id="btn-predict">Classify</button>
25     </center></div>
26   </div>
27
28   <div class="loader" style="display:none;margin-left: 450px;"></div>
29
30   <div id="result">
31
32     <!-- <span><p style="padding-top: 25px;"><h4>Food Classified is : <h4><b>{{(showcase)}}{{(showcase1)}}</p> </span> -->
33     <p style="padding-top: 25px;">Food Classified is : {{(showcase)}}{{(showcase1)}}</p>
34   </div>
35
36 </div>
37 </div>
38
39
40
41 {% endblock %}
```

7. TESTING

7.1 TestCases



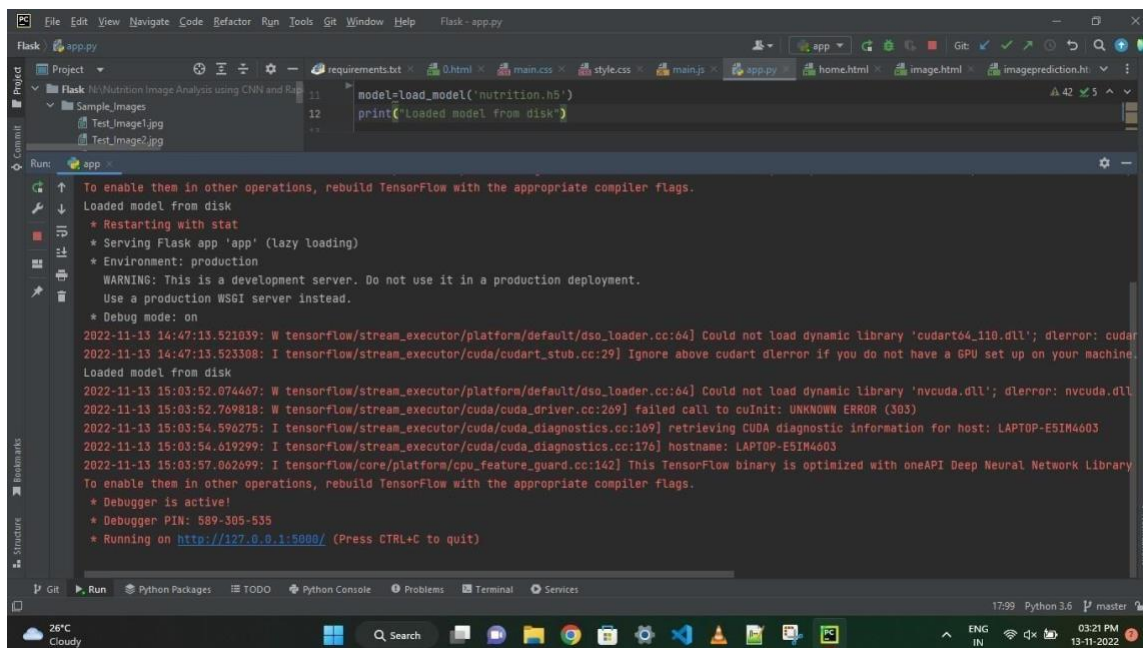


7.2 User Acceptance Testing



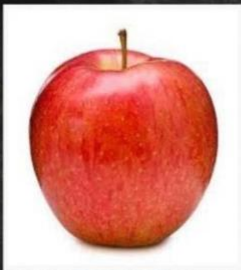
8. RESULTS

8.1 Performance Metrics



Upload image to classify

Choose...



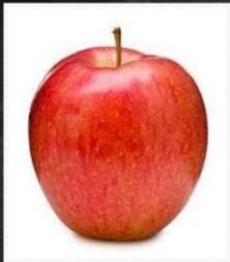
Classify

#253376670



Upload image to classify

Choose...



Food Classified:
APPLES
[{'sugar_g': 10.3,
'fiber_g': 2.4,
'serving_size_g':
100.0,
'sodium_mg': 1,
'name': 'apples',
'potassium_mg':
11,
'fat_saturated_g':
0.0, 'fat_total_g':
0.2, 'calories':
53.4,
'cholesterol_mg':
0, 'protein_g':
0.3,
'carbohydrates_total_g':
13.8}]

#253376670



9. CONCLUSION

By the end of this project we will

- know fundamental concepts and techniques of Convolutional Neural Network.
- gain a broad understanding of image data
- know how to build a web application using the Flask framework.
- know how to pre-process data and
- know how to clean the data using different data preprocessing techniques.

10. FUTURE SCOPE

- AI is revolutionizing the health industry.
- It is majorly used in improving marketing and sales decisions, AI is now also being used to reshape individual habits.
- In future we don't want to go to gym and do any diets. By using this nutrition fitness analyzer we can maintain our diet plans without any help from others and we can lead a happy and healthy life with good wealth.
- AI can easily track health behaviors and repetitive exercise patterns and use the data to guide you towards your fitness journey and diet plans .

11. APPENDIX

Github Link - <https://github.com/IBM-EPBL/IBM-Project-55506-1669097710>

Demonstration Video Link -

https://drive.google.com/file/d/19TraHtSiRDBC0mYhPJX_9BoPfyEgUYkn/view?usp=share_link
