Project Report

Panimalar Institute of Technology

TEAM ID : PNT2022TMID25838

PROJECT NAME : AI-Powered Nutrition Analyzer For Fitness Enthusiasts

Team Members:

K MAGESH
R.AATHARASH
S.BALAJI
P.G. DELWYN ANAND

Table of contents

1. INTRODUCTION

1.1 Project Overview

Food is the basis of human life and is the subject of many health conventions. Today, new nutrition assessment and analysis tools are opening up more opportunities to help people understand their daily diet, explore nutritional patterns, and maintain a healthy diet. Nutritional analysis is the process of determining the nutritional composition of foods. It is an important part of analytical chemistry that provides information on the chemical composition, processing, quality control and contamination of foods.

1.2 Purpose

The main objective of the project is to create a model that is used for classifying fruits depending on various features like color, shape, texture, etc. Here, the user can capture images of different fruits and then the image will be sent to the trained Model. The model analyzes the image and determines the nutrition based on the fruit such as (sugar, fiber, protein, calories, etc.).

2. LITERATURE SURVEY

2.1 Existing problem

Neutrino provides nutrition-based data services and analytics to its users and aims to become the leading source of nutrition-related platform. The platform uses NLP and mathematical models from optimization theory as well as predictive analytics to enable individualized data compilation. The app relies on artificial intelligence to generate its own data related to the AI-powered smart calorie counter. Their artificial intelligence learns individual tastes, preferences and body type. It's all wrapped up in comprehensive nutrition and activity tracking.

2.2 References

https://www.nutrinohealth.com/

2.3 Problem Statement Definition

The main objective of the project is to create a model that is used for classifying fruits depending on various features like color, shape, texture, etc. Here, the user can capture images of different fruits and then the image will be sent to the trained Model. The model analyzes the image and determines the nutrition based on the fruit such as (sugar, fiber, protein, calories, etc.).

3. IDEATION & PROPOSED SOLUTION

3.1 Empathy Map Canvas

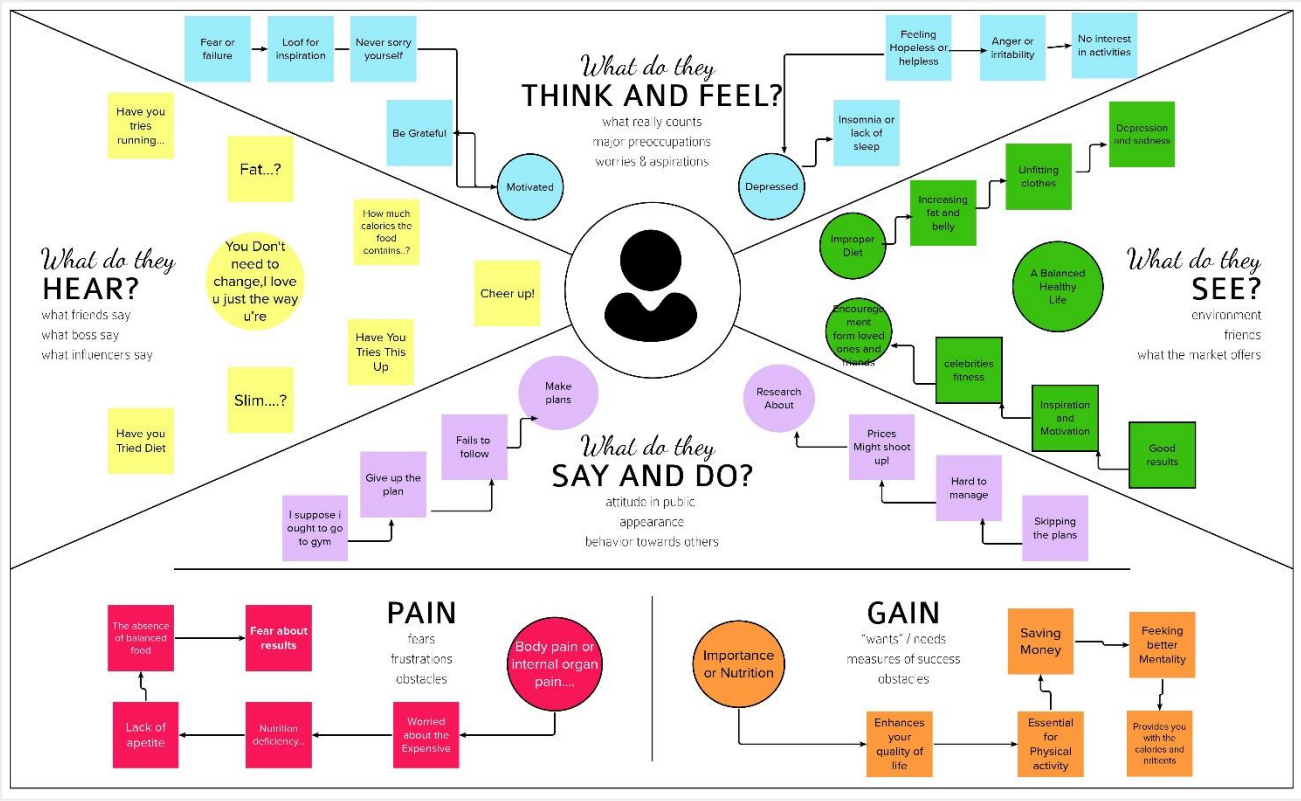
Edit this template
Right-click to unlock

Empathy Map Canvas

Gain insight and understanding on solving customer problems.

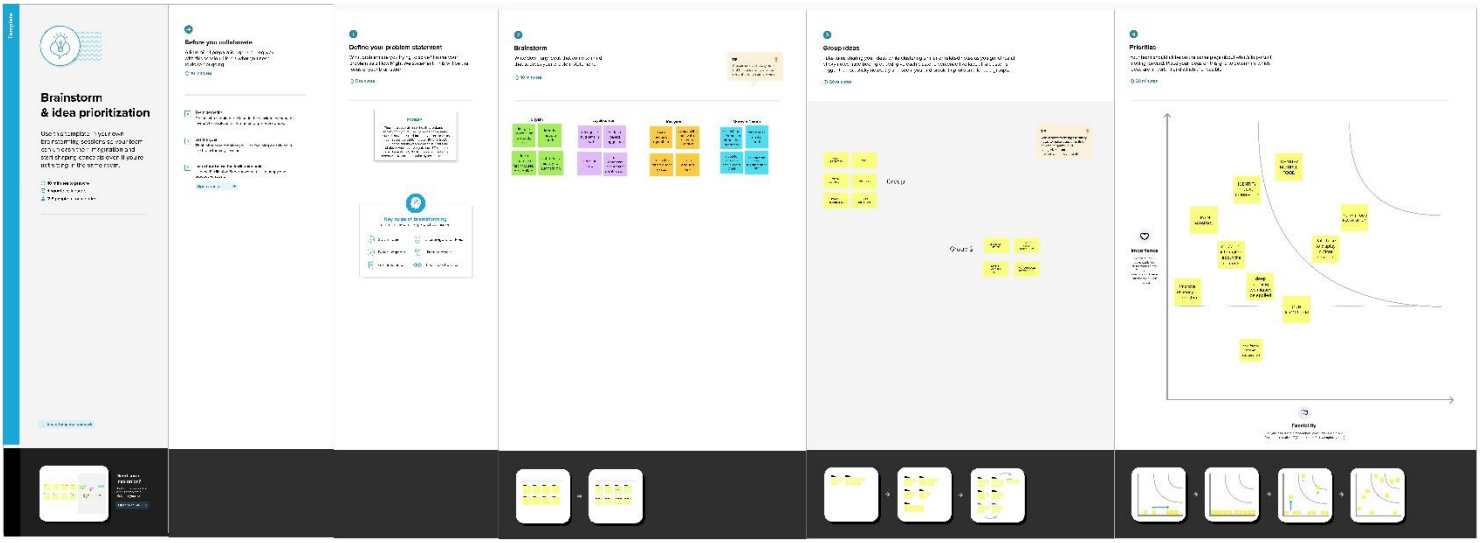
1

Build empathy and keep your focus on the user by putting yourself in their shoes.



Share your feedback

3.2 Ideation & Brainstorming



3.3 Proposed Solution

PROPOSED SOLUTION:

The main aim of the project is to build a model which is used for identifying the fruit depends on the different characteristics like colour, shape, texture etc using image processing. Here the user can capture the images of different fruits and then the image will be analysed with the trained model. The model analyses the image and lists out the nutrients present in the fruit like sugar, vitamins, minerals, protein etc.

NOVELTY:

The application has several unique features. The main feature is that the user need not have to visit or consult a Nutritionist (or) a Dietician to follow a fit and healthy diet. This application has the feature of analysing the entire nutritional content of fruits and vegetables by simply scanning them. It provides for a personalized dietary requirement for individuals who have limited preferences while choosing food.

FEASIBILITY OF IDEA:

The idea of this application is that the user can capture the images of different fruits and vegetables, and then the image will be sent to the trained model. The model analyses the image and detects the nutrition based on the fruits like (Sugar, Fibre, Protein, Calorie intake, etc.). The above idea is achieved by using the Convolution Neural Network (CNN) . It is used to pick the raw pixels present in the image . Fruit Recognition using Colour and Texture Features .

BUSINESS MODEL:

Social media is the best way to spread the word about our application and with the help of influencers we can attract normal people. Clustering and targeting the fitness people with the help of local gyms. Allowing third-party vendors(Nutritional Products) to sell their products through our app via advertisements is way to generate money. If the products sold through advertisements, then it is even better.

SOCIAL IMPACT:

This will acquire knowledge and provide information about nutrition. Now a days, no one follows the diet plan. Providing this information, they come to know about the nutrition present in each food item. It is used to schedule a diet plan by taking the image of a food item and if we send it, we can get information about each food nutrition like carbohydrates, fat, proteins, vitamins, minerals and sugar. This will help others to improve their
health and fitness.

SCALABILITY:

Artificial intelligence (AI) can be used to predict investment outcomes quickly and effectively, as well as to devise strategies or establish long-term goals. Scalable AI pertains to how data models, infrastructures, and algorithms can increase or decrease their complexity, speed, or size at scale in order to best handle the requirements of the situation at hand. As improvements continue with data storage capacities as well as computing resources, AI models can be created with billions of parameters. Scaling up nutrition is a global push for action and investment to improve maternal, child nutrition and various health problems.

3.4 Problem Solution fit

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

Purpose:

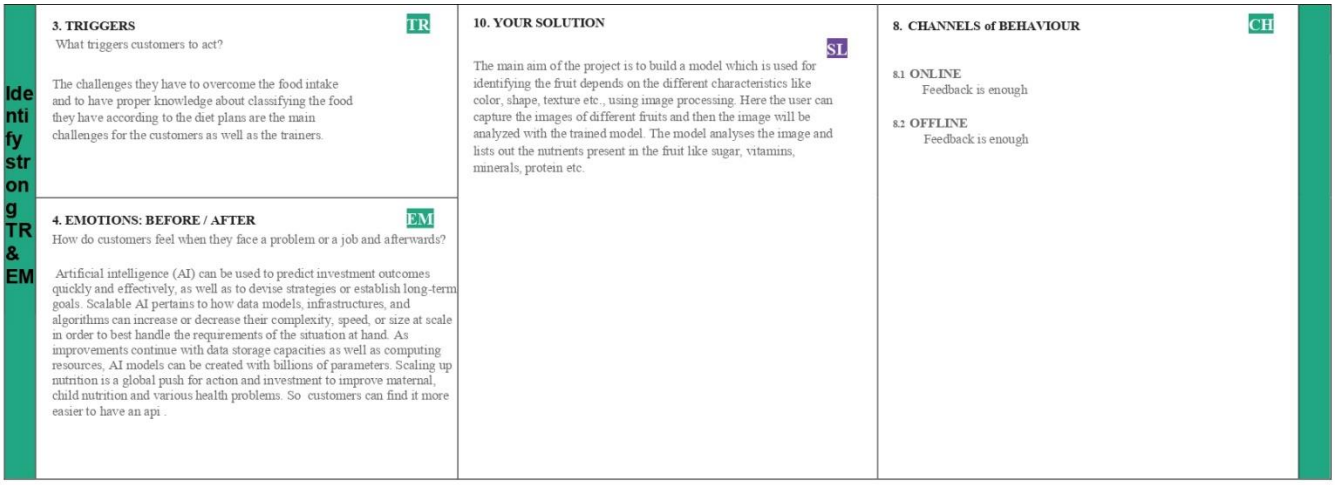
• Solve complex problems in a way that fits your customers' situation.

• Achieve faster success and increase adoption of your solution by leveraging existing media and behavioral channels.

• Enhance your communication and marketing strategy with the right triggers and messages.

•

• Increase touch points with your company by finding the right fit for problem behaviors and building trust by addressing frequent annoyances or urgent or costly issues.



4. REQUIREMENT ANALYSIS

4.1 Functional requirement

• It will generate a diet plan and also track the user's health to classify the disease category and create a diet plan. It will also reduce the cost of consulting a nutritionist.

• The task of food detection/classification is not as easy as it seems. All possible options related to a given image.

• Image classification, object detection, segmentation, face recognition.

• Crystal structure classification using a convolutional neural network

• Nutrition is vital for the growth of the human body. Nutritional analysis ensures that the food meets the appropriate vitamin and mineral requirements, and the nutrition examination of the food helps to understand the proportion of fat, dilution of carbohydrates, protein, fiber, sugar, etc. Another thing to keep in mind is not to exceed our daily requirements for calorie

• Computer Aided Nutrition for Food Image Recognition – To solve this problem, a completely new Convolutional Neural Network (CNN) based food identification system was created. Created as described in this study. We applied our proposed strategy on two real food datasets.

• Here the user can capture images of different fruits and then the image will be sent to the trained model. The model analyzes the image and detects nutrition based on the fruit as (sugar, fiber, protein, calories, etc.)

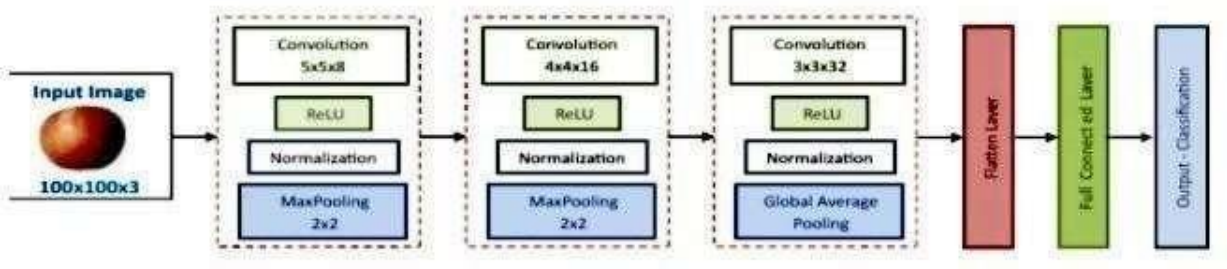
•

• The best solution for working out at home This AI fitness software is designed with individual training regimens for each individual. It started out as "gym-only software" but has now refined its system to meet "home fitness" expectations.

• You take a picture, dial in information such as whether you're having breakfast or lunch, add a quick text label, and the app estimates the calorie content.

• This software worked with IBM's natural language capability to provide 24-hour assistance and dietary recommendations.

For Example:



• Comparison of the proposed model with conventional models shows that the results of this model are extremely good and promising for use in real applications.

• This kind of higher accuracy and precision will enhance the general effectiveness of the machine in recognizing fruits more adequately.

• The general model for the need for protein in the diet (as with any nutrient) defines the requirement in terms of the organism's needs,

• i.e. metabolic demands and the dietary amount that satisfies these needs, i.e. utilization efficiency, i.e.: dietary requirement = metabolic need/utilization efficiency.

FUNCTIONAL REQUIREMENTS:

Following are the functional requirements for the proposed solution

FR NO.	FUNCTIONAL REQUIREMENTS(EPIC)	SUB REQUIREMENT(STORY/SUBTASK)
FR-1	USER REGISTRATION	Interacting the user through web interface and automated voice to answer the user queries and to guide them in a proper way to maintain their fitness. In the web interface, ● There will be separate and special features for the registered user to get personalized and well defined advice and good practice lectures to maintain their fitness. ● All the registered users will be verified with either email or mobile number based on their interest in giving their information, but the verification is a must one. ● For non-registered users, the user can visit the website free of cost and can check the nutrient value in the fruits and vegetables, and also can view the common practices for fitness.
FR-2	USER MANAGEMENT	Creating a group of people, who are willing to be fit in their health and making them organized in a sampe place, through which they can collaborate and also can achieve their goals with others, by encouraging each other. The application gives the ability to ask questions about a problem in the fitness groups, through which they can work effectively.
FR-3	USER SATISFYING	The satisfaction of each user is a must, so UI/UX should be more than enough to engage the user in the platform and the performance of the application should be optimized in order to keep every user for a long time. On an periodic interval (like once in month), we need to interact one to one with each and every user to solve the queries
FR-4	USER ENGAGEMENT	The user should be engaged in the application at least Once a day to get notified about the latest and good practice on fitness which is recommended by the backend model.

NON-FUNCTIONAL REQUIREMENTS:

Following are the functional requirements for the proposed solution.

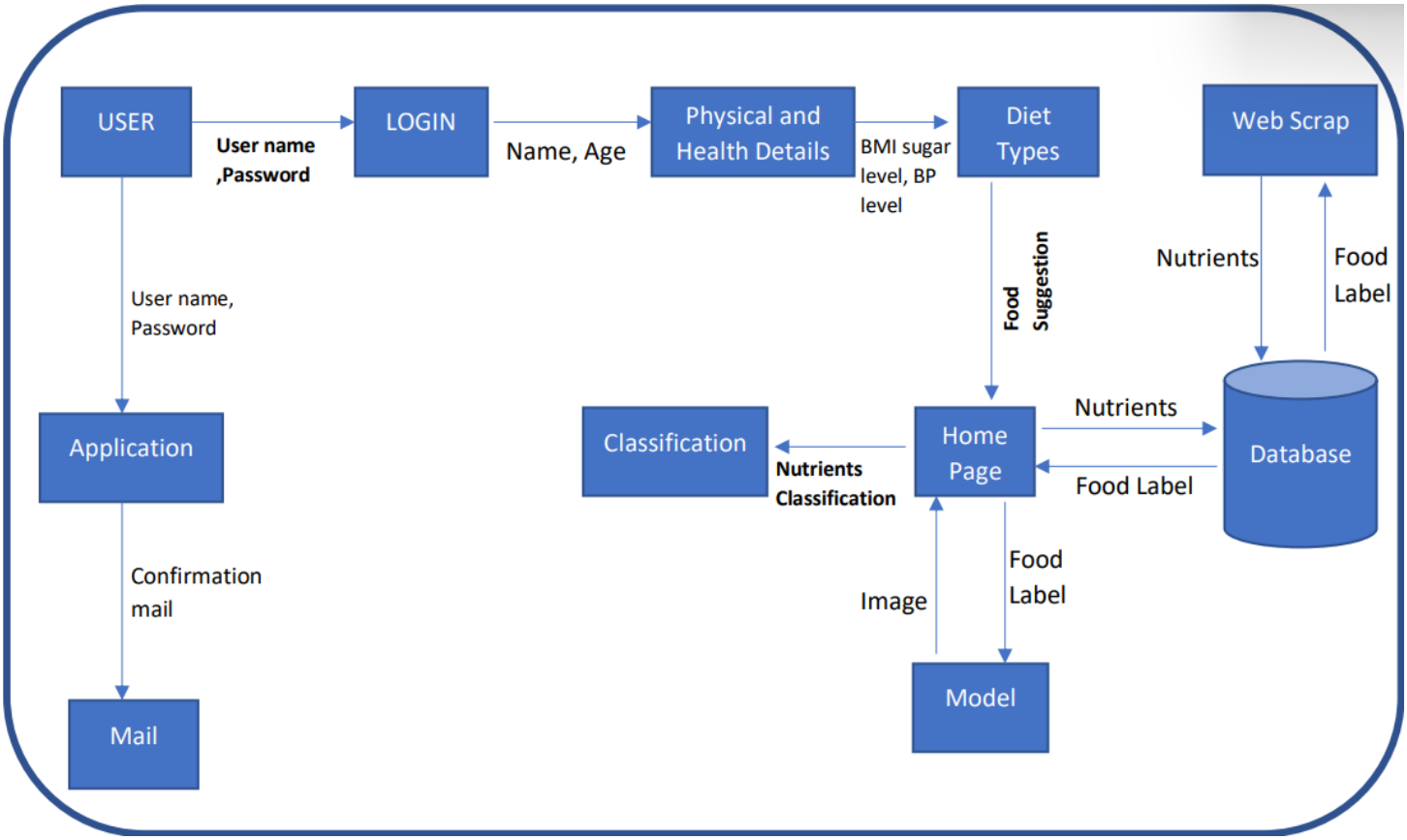
FR.NO	NON-FUNCTIONAL REQUIREMENTS	DESCRIPTION
NFR-1	USABILITY	• No training is required to access the Nutrition Analyzer. • The results should be loaded within30 seconds. • It should be user friendly and comfortable. • It should be simple and easy to use. • The results should be self explanatory so that it can be understood by common people.
NFR-2	SECURITY	• AI powered nutrition analyzer for fitness should contain more security in which our data which entered or maintained should be more security. • With the help of the username and password it provides more security in which it can access more securable and the data are private. • It should be social-economic which should access to sufficient and safe touse. • It is Important that the AI powered nutrition analyzer for fitness provides should Must reliable.

NFR-3	RELIABILITY	• How a person can find it is reliable? It iseasy to find that is he/she can compare the nutrition based food with other nutrition related application so, it can easily rectify whether it is reliable or not. • But it take too much time, to avoid this a reliable application should made in which it itself produces whether we can get correct solution or not. So, it is necessary that the AI powered nutrition analyzer for fitness should have proper data and information in which we can get a correct information about it and also get a proper guidance about it. • With the proper guide and proper information in which we can get a nutrition properly and we can have get aproper fitness plan. • It should also provides the information on nutrition and health which it should prevent from health information on diseases, health risks and prevention guidelines. It should also provides an extension a research based online learning network with several resource areas, so it provides more reliability in that area. For more reliable it can also contains the calorie information, balanced diet plans, what type food can consumed at what time etc..... So, by this way it can reliable.
NFR-4	PERFORMANCE	• It should provide more number of users to consume at any time and at any place. • It should provide Reliability, Scalability, Security and Usability. • It should contain minimum data while over-paging the websites or application and it is necessary that it

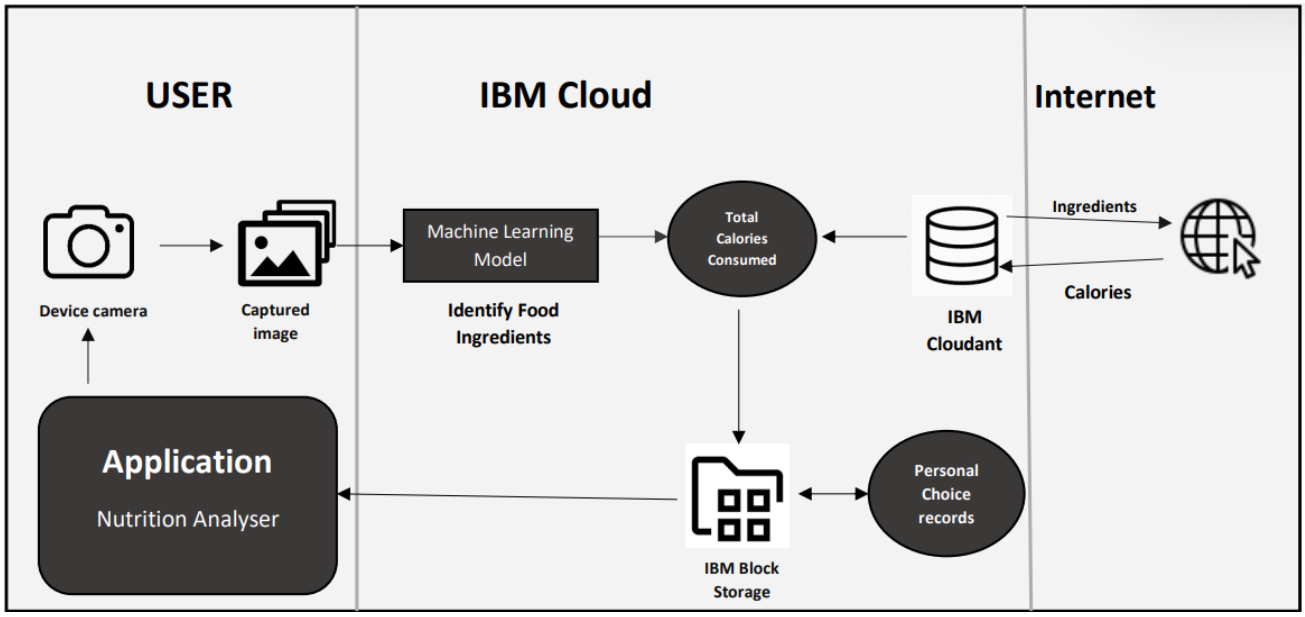
		should not exceed more than 20mb. • While consuming the page it should provide the response as much as possible without any delay or time traffic. • The connection should e properly maintained so that it can use while travelling or in remote places. • The nutritious food to meet their dietary needs and the food preferences for an active and healthy life. • It should be consistently access, availability and affordability of foods and beverages that promote well- being and prevent from diseases. • It should suitable in all situations that exists to all people, at all times.
NFR-5	AVAILABILITY	• Easy to access Data. • Avoids Data redundancy and inconsistency. • Fast and Efficient. • User Friendly.
NFR-6	SCALABILITY	• The architecture for AI powered Nutrition Analyzer for fitness provides the clear procedure daily consumption of food and helps the user to maintain ahealthy diet. • According to their tracking system implemented in architecture provide theproper mechanism to the every individual of their nutrients intake which can be increased or decreased. • The premium amount for analyzer is verymuch optimum.

5. PROJECT DESIGN

5.1 Data Flow Diagrams



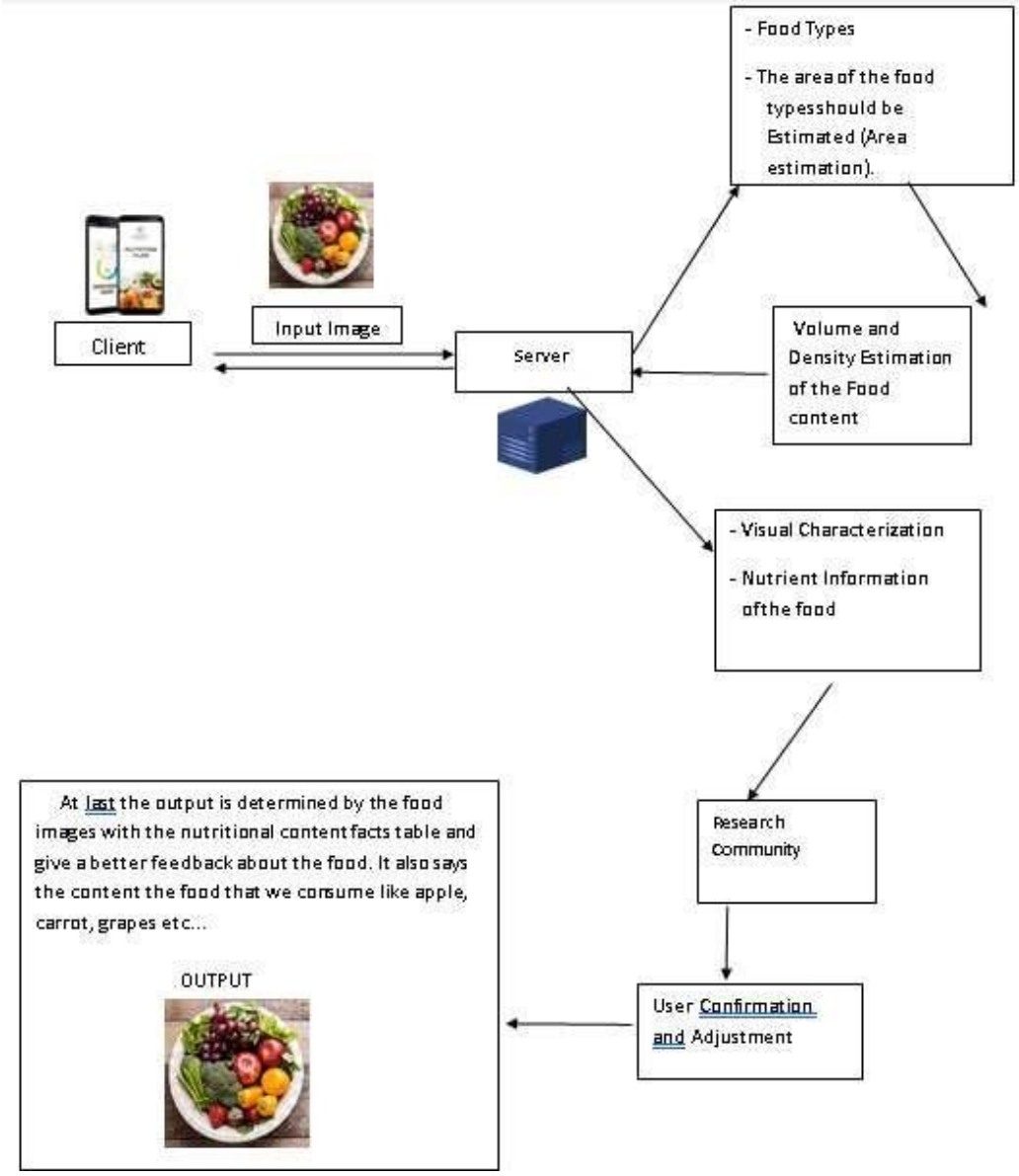
5.2 Solution & Technical Architecture



S.no	component	Description	Technologies
1.	User Interface	The user interacts with application Web UI for the nutrition content for the given food.	HTML, CSS, JavaScript
2.	Database	Data Type, Configurations will be stored	MySQL
3.	Cloud Database	Database Service on IBM Cloud	IBM DB2, IBM Cloudant etc.
4.	File Storage	File storage requirements	Storage will be based on Cloud
5.	Machine Learning Model	To classify the image of food and provide the nutrient content of the same.	OPEN CV,MATPLOTLIB, ANN ,CNN, RNN
6.	Infrastructure (Server / Cloud)	Application Deployment on Cloud	IBM CLOUD

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 architectureconsists of 3 tiers	Web Server – HTML,CSS ,Javascript Application Server –Python Flask Database Server – IBM Cloud
4.	Availability	Availability is increased by loads balancers in cloud VPS	IBM Cloud hosting
5.	Performance	The application is expected to handle up to 4000 predications per second	IBM Load Balance


- Food Types
- The area of the food typesshould be Estimated (Area estimation).
Client
Input Image
Server
Volume and Density Estimation of the Food content
- Visual Characterization
- Nutrient Information ofthe food
Research Community
User Confirmation and Adjustment
At last the output is determined by the food images with the nutritional content facts table and give a better feedback about the food. It also says the content the food that we consume like apple, carrot, grapes etc...
OUTPUT

5.3 User Stories

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-1	Data selection and Image Preprocessing	USN-1	we will be improving the image data that suppresses unwilling distortions or enhances some image features important for further processing, although performing some geometric transformations of images like rotation, scaling, translation, etc. The ImageDataGenerator accepts the originaldata, randomly transforms it, and returns only the new, transformed data.	2	High	Shyam Kaviyaraj Jeyakumar Sherwin Dennis
Sprint-2	Model Building	USN-1	**Steps to Build a Deep Learning Model** 1. Defining the model architecture 2. Configure the learning process 3. Train The Model 4. Save the Model 5. Predictions	1	Medium	Shyam Kaviyaraj Jeyakumar Sherwin Dennis
Sprint-3	Application Building	USN-1	Now that we have trained our model, let us build our flask applicationwhich will be running in our local browser with a user interface. In the flask application, the input parameters are taken from the HTML page. These factors are then given to the model to predict the type of food and to know the nutrition content in it. In order to know the nutrition content we will be using an API in this project.	2	High	Shyam Kaviyaraj Jeyakumar Sherwin Dennis
Sprint-4	Train the model on IBM	USN-1	In this milestone, we will register in the IBM cloud and Train the Model in thecloud. Finally we will build a deep learning model.	2	High	Shyam Kaviyaraj Jeyakumar Sherwin Dennis

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 selection and Image Preprocessing	USN-1	we will be improving the image data that suppresses unwilling distortions or enhances some image features important for further processing, although performing some geometric transformations of images like rotation, scaling, translation, etc. The ImageDataGenerator accepts the originaldata, randomly transforms it, and returns only the new, transformed data.	2	High	Shyam Kaviyaraj Jeyakumar Sherwin Dennis
Sprint-2	Model Building	USN-1	**Steps to Build a Deep Learning Model** 1. Defining the model architecture 2. Configure the learning process 3. Train The Model 4. Save the Model 5. Predictions	1	Medium	Shyam Kaviyaraj Jeyakumar Sherwin Dennis

Sprint-3	Application Building	USN-1	Now that we have trained our model, let us build our flask applicationwhich will be running in our local browser with a user interface. In the flask application, the input parameters are taken from the HTML page. These factors are then given to the model to predict the type of food and to know the nutrition content in it. In order to know the nutrition content we will be using an API in this project.	2	High	Shyam Kaviyaraj Jeyakumar Sherwin Dennis
Sprint-4	Train the model on IBM	USN-1	In this milestone, we will register in the IBM cloud and Train the Model in thecloud. Finally we will build a deep learning model.	2	High	Shyam Kaviyaraj Jeyakumar Sherwin Dennis

6.2 Sprint Delivery Schedule

Milestone	Activity
Data Collection	Collecting images of food items apples,bananas, oranges, pineapples, watermelons for analysis.
Image Preprocessing	Increasing the amount of data by generating new data points from existing data. Applying image data generator functionality to train and test dataset.
Modeling Phase	Building the model using a deep learning approach and adding CNN layers. Training,saving,testing and predicting the model. Database creation for the input classes.
Development phase	Dashboard creation. Analysis and prediction page creation. Creating feedback and rating page.
Application Phase	Building the python code and importing the flask module into the project. Create the Flask application and load the model. Connecting front end and back end and performing routing and running the application.
Testing Phase	Checking usability and accessibility. Checking scalability and performance of the application.

6.3 Reports from JIRA

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

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

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

```
x_train = train_datagen.flow_from_directory(
    r'/content/drive/MyDrive/IBM/Dataset-PNT2022TMID01549/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/IBM/Dataset-PNT2022TMID01549/Dataset/TEST_SET',
    target_size=(64, 64),batch_size=5,color_mode='rgb',class_mode='sparse')
```

```
Found 2656 images belonging to 5 classes.
Found 1055 images belonging to 5 classes.
```

F

Initializing the Model

```
[7] model = Sequential()
```

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())
```

Adding Dense Layers

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

```
[10] classifier.summary()
```

```
Model: "sequential_1"
_________________________________________________________________
 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 (MaxPooling  (None, 14, 14, 32)       0
 2D)

 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
_________________________________________________________________
```

Configure The Learning Process

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

Train The Model

```
[13] classifier.fit_generator(
         generator= train,steps_per_epoch = len(x_train),
         epochs=10, validation_data=x_test,validation_steps = len(x_test))
```

```
/usr/local/lib/python3.7/dist-packages/ipykernel_launcher.py:1: UserWarning: `Model.fit_generator` is deprecated and will be removed in a future version. Please use `Model.fit`,
  """Entry point for launching an IPython kernel.
Epoch 1/10
532/532 [==============================] - 488s 1s/step - loss: 0.1738 - accuracy: 0.9497 - val_loss: 0.1084 - val_accuracy: 0.9703
Epoch 2/10
532/532 [==============================] - 283s 529ms/step - loss: 0.0468 - accuracy: 0.9872 - val_loss: 0.0744 - val_accuracy: 0.9739
Epoch 3/10
532/532 [==============================] - 281s 528ms/step - loss: 1.8249e-04 - accuracy: 1.0000 - val_loss: 0.0586 - val_accuracy: 0.9824
Epoch 4/10
532/532 [==============================] - 281s 528ms/step - loss: 0.0115 - accuracy: 0.9960 - val_loss: 0.1008 - val_accuracy: 0.9725
Epoch 5/10
532/532 [==============================] - 281s 528ms/step - loss: 1.1175e-04 - accuracy: 1.0000 - val_loss: 0.0595 - val_accuracy: 0.9858
Epoch 6/10
532/532 [==============================] - 281s 528ms/step - loss: 5.3459e-05 - accuracy: 1.0000 - val_loss: 0.0589 - val_accuracy: 0.9881
Epoch 7/10
532/532 [==============================] - 281s 528ms/step - loss: 0.0241 - accuracy: 0.9986 - val_loss: 0.0221 - val_accuracy: 0.9867
Epoch 8/10
532/532 [==============================] - 281s 528ms/step - loss: 1.1e-04e-04 - accuracy: 1.0000 - val_loss: 0.0716 - val_accuracy: 0.9744
Epoch 9/10
532/532 [==============================] - 281s 528ms/step - loss: 2.8955e-05 - accuracy: 1.0000 - val_loss: 0.0474 - val_accuracy: 0.9782
Epoch 10/10
532/532 [==============================] - 279s 524ms/step - loss: 1.2144e-05 - accuracy: 1.0000 - val_loss: 0.0445 - val_accuracy: 0.9781
<keras.callbacks.History at 0x7fcb1f2b5d0>
```

Save the Model

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

Test the Model

```
[14] from tensorflow.keras.models import load_model
     from tensorflow.keras.preprocessing import image
     import numpy as np
```

```
img = image.load_img("/content/drive/MyDrive/IBM/Dataset-PNT2022TMID01548/Dataset/TEST_SET/ORANGE/30_100.jpg",target_size= (64,64))
img
```

```
[16] x=image.img_to_array(img)
```

```
[17] x
```

```
array([[[255., 255., 242.],
        [255., 255., 251.],
        [255., 255., 255.],
        ...,
        [255., 255., 255.],
        [255., 255., 255.],
        [255., 255., 255.]],

       [[255., 255., 251.],
        [254., 253., 251.],
        [255., 253., 254.],
        ...,
        [255., 255., 255.],
        [255., 255., 255.],
        [255., 255., 255.]],
```

```
[18] x.ndim
```

```
3
```

```
[19]
x=np.expand_dims(x,axis=0)
```

```
[20] x.ndim
```

```
4
```

```
[21]
pred = classifier.predict(x)
```

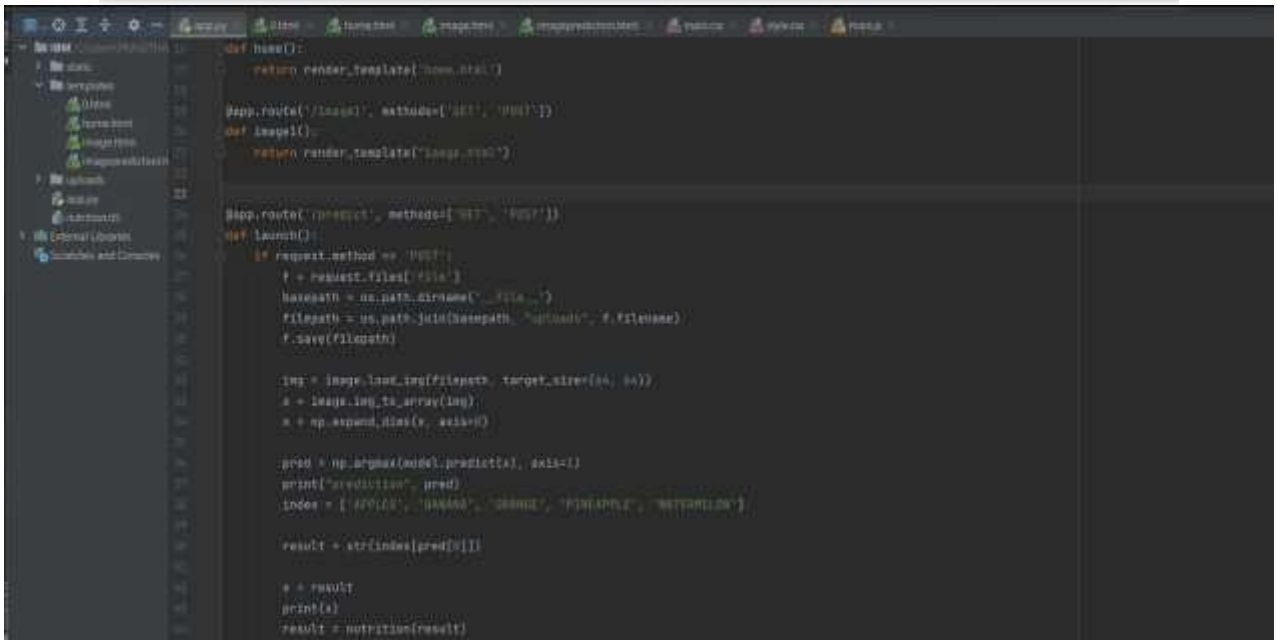
```
1/1 [==============================] - 0s 112ms/step
```

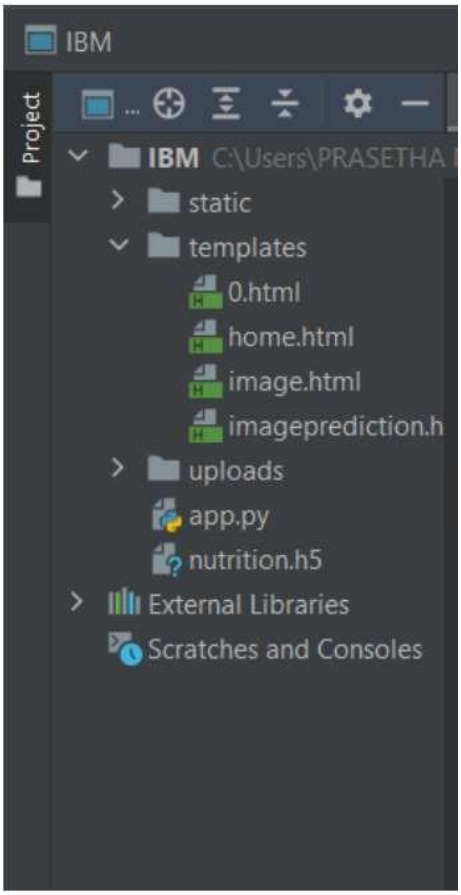
```
[22] pred
```

```
array([[0., 0., 1., 0., 0.]], dtype=float32)
```

```
labels=['APPLES', 'BANANA', 'ORANGE','PINEAPPLE','WATERMELON']
labels[np.argmax(pred)]
```

```
'ORANGE'
```




IBM
Project
IBM C:\Users\PRASETHA
static
templates
0.html
home.html
image.html
imageprediction.h
uploads
app.py
nutrition.h5
External Libraries
Scratches and Consoles

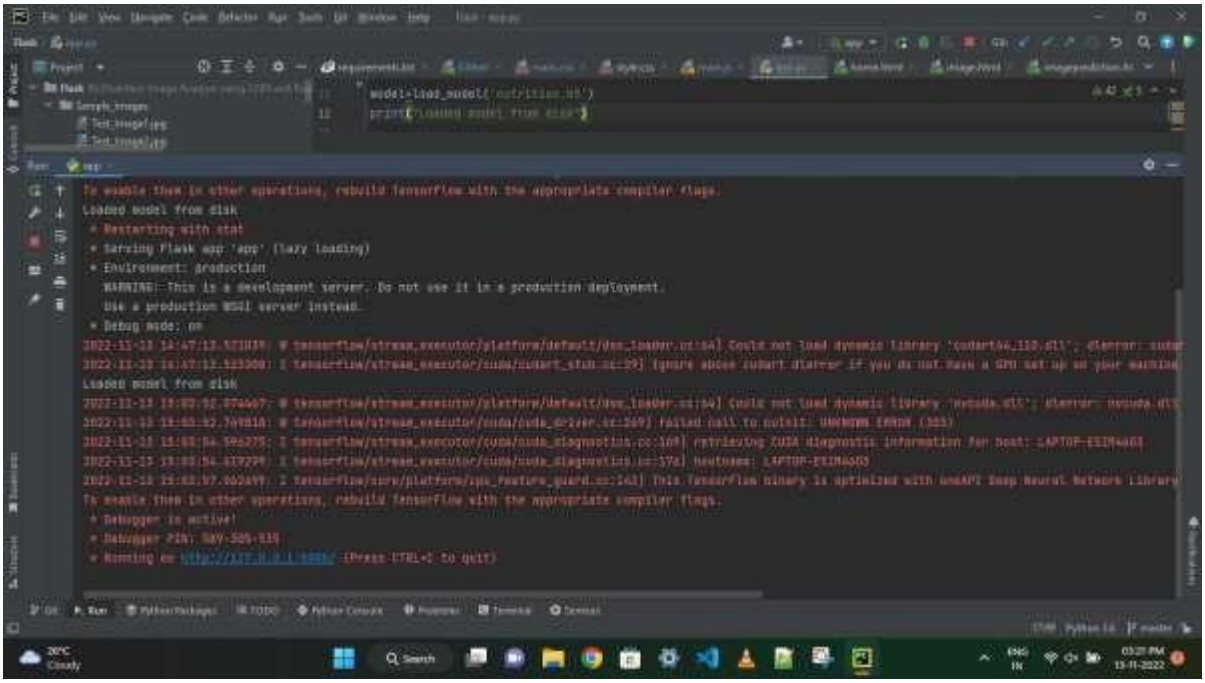
8. TESTING

8.1 Test Cases

8.2 User Acceptance Testing

9.RESULTS

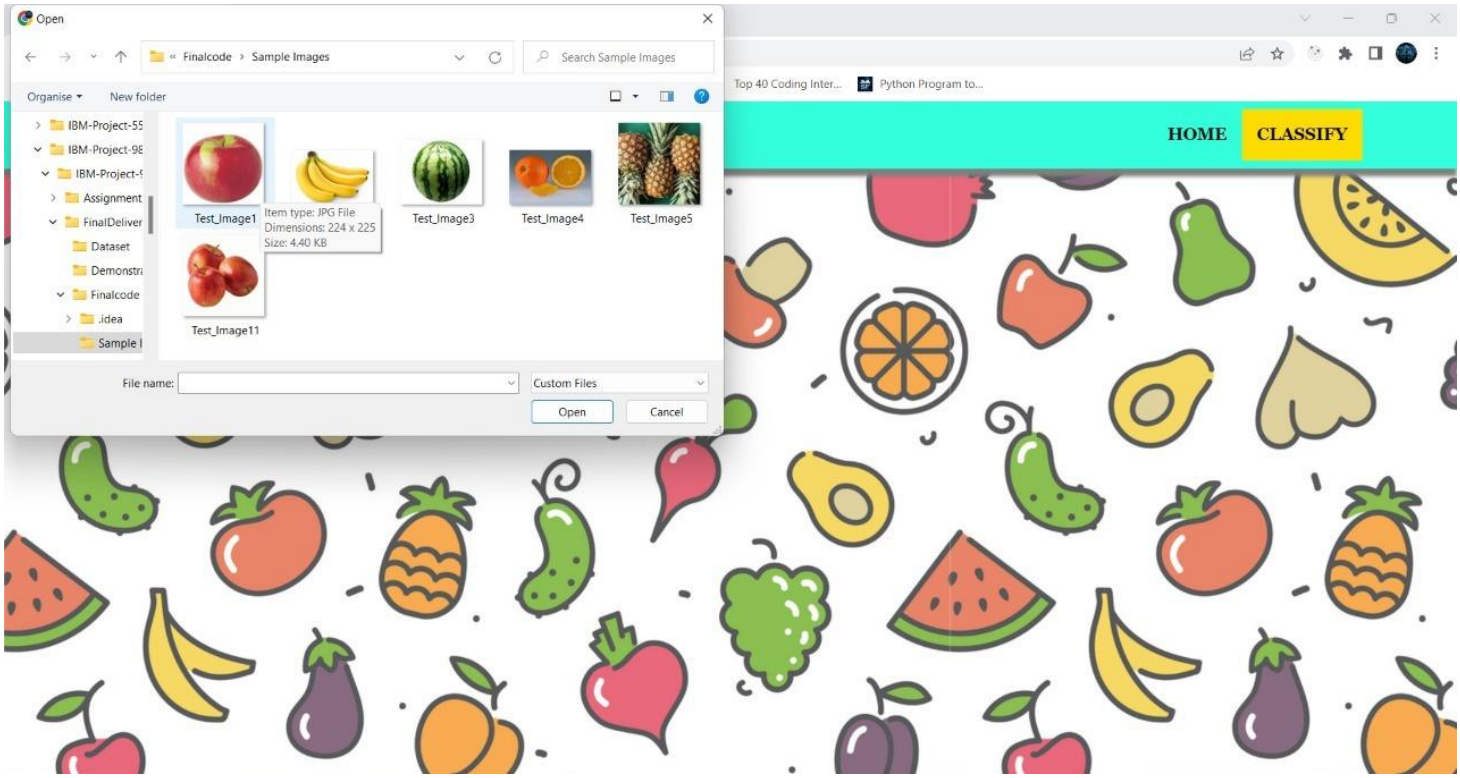
9.1 Performance Metrics



9.2 Output


Nutrition Image Analysis
Home Classify
PURPOSE
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.
GOAL
Aim of the project is to find fruits by the colour,texture,shape of the given fruits and then to find the vitamin ,protiens ,sugar level etc of the desired fruit
VITAMINS


NUTRITION IMAGE ANALYSIS
HOME
CLASSIFY
UPLOAD IMAGE
CHOOSE...
ANALYZE


Open
Finalcode › Sample Images
Search Sample Images
Organise
New folder
IBM-Project-55
IBM-Project-98
IBM-Project-
Assignment
FinalDeliver
Dataset
Demonstra
Finalcode
.idea
Sample I
Test_Image1
Item type: JPG File
Dimensions: 224 x 225
Size: 4.40 KB
Test_Image3
Test_Image4
Test_Image5
Test_Image11
File name:
Custom Files
Open
Cancel
Top 40 Coding Inter...
Python Program to...
HOME
CLASSIFY


NUTRITION IMAGE ANALYSIS
HOME
CLASSIFY
UPLOAD IMAGE
CHOOSE...
ANALYZE


NUTRITION IMAGE ANALYSIS
HOME
CLASSIFY
UPLOAD IMAGE
CHOOSE...
IMAGE CLASSIFIED IS :
APPLES
[{'sugar_g': 2.6, 'fiber_g': 1.2, 'serving_size_g': 100.0, 'sodium_mg': 4, 'name': 'tomato', 'potassium_mg': 23, 'fat_saturated_g': 0.0, 'fat_total_g': 0.2, 'calories': 18.2, 'cholesterol_mg': 0, 'protein_g': 0.9, 'carbohydrates_total_g': 3.9}]

10. ADVANTAGES & DISADVANTAGES

ADVANTAGES

- New dietary assessment and nutrition analysis tools provide more opportunities to help people understand their daily eating habits
- It helps in exploring nutritional patterns in their daily routines and this is very helpful for people to maintain a balanced healthy diet.
- Nutritional analysis is used to determine the nutritional content of foods.
- This app eliminates travel costs when visiting a nutritionist.
- Using this app greatly reduces the time required to get the best diet plan

DISADVANTAGES

- Android mobile user will not be able to insert or view details if the server is down.
- So the disadvantage is single point failure.

11. CONCLUSION

We will be by the end of this project

- know the basic concepts and techniques of a convolutional neural network.
- gain a broad understanding of image data
- know how to create a web application using the Flask framework.
- know how to preprocess data and
- know how to clean data using various data pre-processing techniques.

12. FUTURE SCOPE

- Artificial intelligence is revolutionizing healthcare.
- Mainly used to improve marketing and sales decisions, AI is now also being used to reshape individual habits.
- We don't want to go to the gym and follow any diets in the future. With this nutrition analyzer we can maintain our diet plans without the help of others and 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 on your fitness journey and diet plans.

13. APPENDIX

Source code

```
from flask import Flask,render_template,request
# Flask-It is our framework which we are going to use to run/serve our
application.
#request-for accessing file which was uploaded by the user on our application.
import os
import numpy as np #used for numerical analysis
from tensorflow.keras.models import load_model#to load our trained model
from tensorflow.keras.preprocessing import image
import requests

app = Flask(__name__,template_folder="templates") #initializing a flask app
# Loading the model
model=load_model('nutrition.h5')
print("Loaded model from disk")

@ app.route('/')# route to display the home page
def home():
```

```
    return render_template('home.html') #rendering the home page

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


@ app.route('/predict' ,methods=['GET','POST']) # route to show the predictions in
a Web UI
def lanuch():
    if request.method=='POST':
        f=request.files['file'] # requesting the file
        basepath=os.path.dirname('__file__') #storing the file directory
        filepath=os.path.join(basepath,"uploads",f.filename) #storing the file in
uploads folder
        f.save(filepath) #saving the file

        img=image.load_img(filepath,target_size=(64,64)) #load and reshaping the
image
        x=image.img_to_array(img) #converting image to an array
        x=np.expand_dims(x,axis=0) #changing the dimensions of the image

        pred=np.argmax(model.predict(x), axis=1)
        print("prediction",pred) #printing the prediction
        index=['APPLE','BANANA','ORANGE','PINEAPPLE','WATERMELON']

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

        return render_template("0.html",showcase=(result),showcase1=(x))
def nutrition(index):

    import requests

    url = "https://calorieninjas.p.rapidapi.com/v1/nutrition"

    querystring = {"query":index}

    headers = {
      "X-RapidAPI-Key": "226fdb7ca6mshc43f1bfd5e9705dp164933jsn6809eaf3d5e3",
      "X-RapidAPI-Host": "calorieninjas.p.rapidapi.com"
      }

    response = requests.request("GET", url, headers=headers, params=querystring)

    print(response.text)
    return response.json()['items']
if __name__ == "__main__":
    # running the app
    app.run(debug=False)
```

GitHub - https://github.com/IBM-EPBL/IBM-Project-21468-1664179525

Demo link -

https://drive.google.com/drive/u/0/folders/1gpHscmq94WVSOY8pfdh14nTAhJ6MjTLr