

PROJECT REPORT

Chennai Institute of Technology

TEAM ID:PNT2022TMID24705

PROJECT NAME: AI-Powered Nutrition Analyzer For
Fitness Enthusiasts

Team Members:

Seelam Sathwik
Shaik Afrid
S Hemanth Kumar
Shaik Avez

Table of contents

Chapter	Title	Page No.
1	1. INTRODUCTION	4
	1.1 Project Overview	4
	1.2 Purpose	4
2	2. LITERATURE SURVEY	4
	2.1 Existing problem	4
	2.2 References	4
	2.3 Problem Statement Definition	5
3	3. IDEATION & PROPOSED SOLUTION	5
	3.1 Empathy Map	5
	3.2 Ideation & Brainstorming	6
	3.3 Proposed Solution	7
	3.4 Problem Solution fit	9
4	4. REQUIREMENT ANALYSIS	10
	4.1 Functional requirement	12
	4.2 Non-Functional requirements	13
5	5. PROJECT DESIGN	16
	5.1 Data Flow Diagrams	16
	5.2 Solution & Technical Architecture	17
	5.3 User Stories	20
6	6. PROJECT PLANNING & SCHEDULING	20
	6.1 Sprint Planning & Estimation	20
	6.2 Sprint Delivery Schedule	21
	6.3 Reports from JIRA	22
7	7. CODING & SOLUTIONING (Explain the features added in the project along with code)	23
	7.1 Feature 1	23
8	8. TESTING	28
	8.1 Test Cases	28
	8.2 User Acceptance Testing	28

9	9. RESULTS	29
	9.1 Results	30
10.	10. ADVANTAGES & DISADVANTAGES	33
11.	CONCLUSION	34
12.	FUTURE SCOPE	34
13.	APPENDIX	34
	GitHub & Project Demo Link	

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

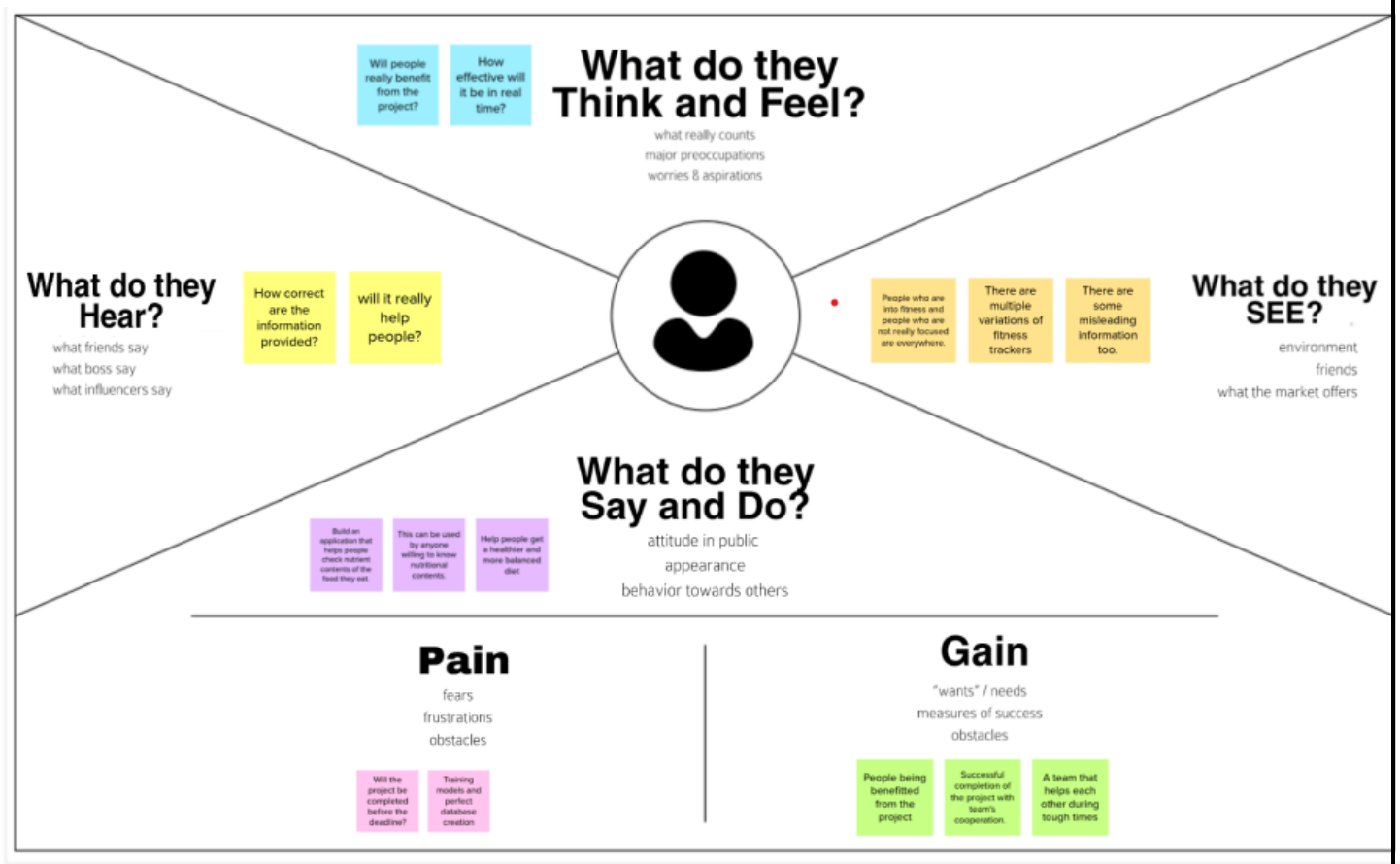
<https://www.nutrinohealth.com/>

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



3.2 Ideation & Brainstorming



Brainstorm & idea prioritization

Use this template to generate ideas and prioritize them. It includes a section for brainstorming ideas and a section for prioritizing them based on their feasibility and desirability.

Brainstorming ideas

- Generate ideas
- Filter ideas
- Group ideas

Prioritizing ideas

- Feasibility
- Desirability

Brainstorming ideas

Generate ideas

Filter ideas

Group ideas

Prioritizing ideas

Feasibility

Desirability

Brainstorming ideas

Generate ideas

Filter ideas

Group ideas

Prioritizing ideas

Feasibility

Desirability

Brainstorming ideas

Generate ideas

Filter ideas

Group ideas

Prioritizing ideas

Feasibility

Desirability

Define your problem statement

Define your problem statement

Define your problem statement

Brainstorm

Brainstorm

Brainstorm

Group ideas

Group ideas

Group ideas

Prioritize

Prioritize

Prioritize

Define your problem statement

Define your problem statement

Define your problem statement

Brainstorm

Brainstorm

Brainstorm

Group ideas

Group ideas

Group ideas

Prioritize

Prioritize

Prioritize

Define your problem statement

Define your problem statement

Define your problem statement

Brainstorm

Brainstorm

Brainstorm

Group ideas

Group ideas

Group ideas

Prioritize

Prioritize

Prioritize

Define your problem statement

Define your problem statement

Define your problem statement

Brainstorm

Brainstorm

Brainstorm

Group ideas

Group ideas

Group ideas

Prioritize

Prioritize

Prioritize

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:

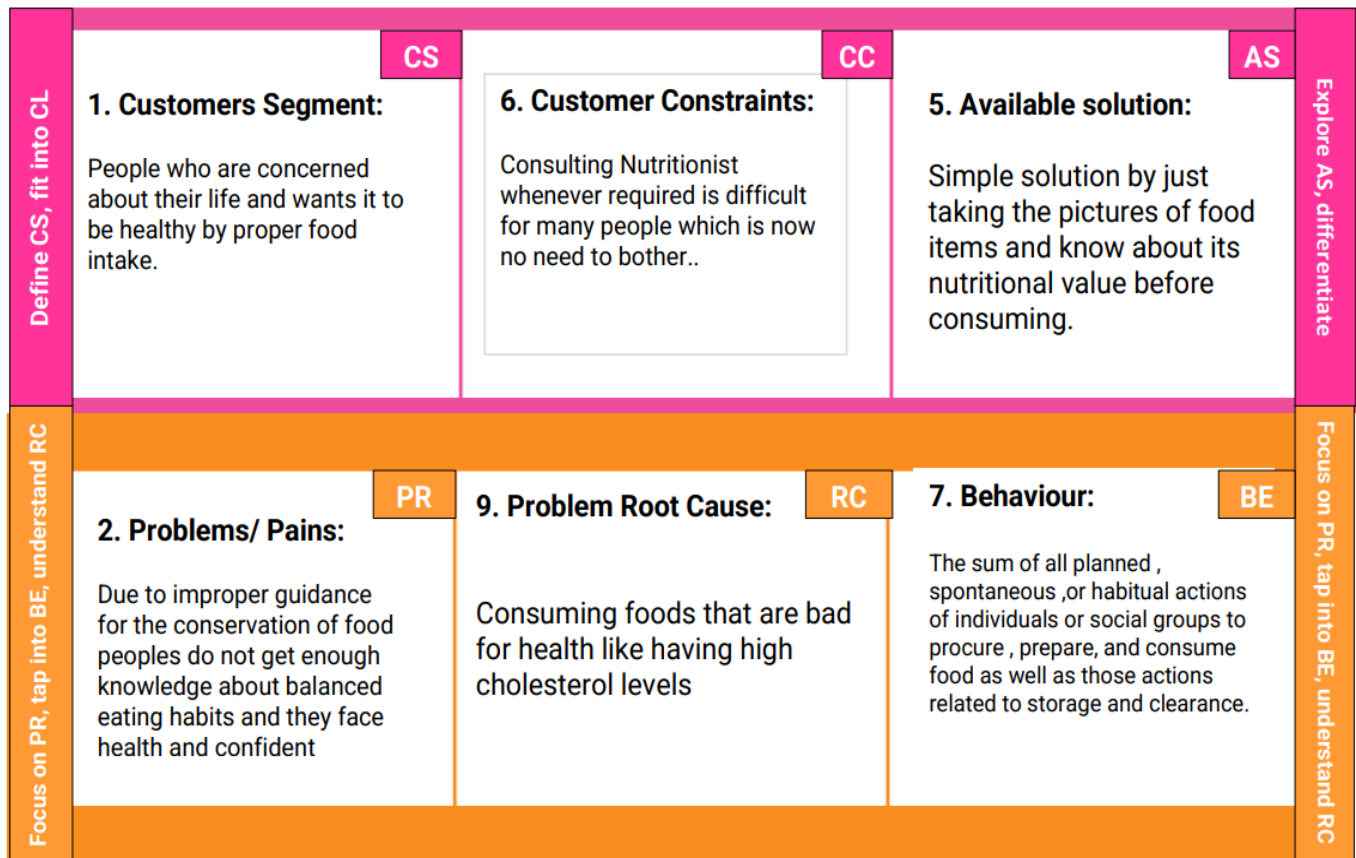
Artificial intelligence (AI) can be used to predict investment outcomes quickly and effectively, as well as to devise strategies or establish long-term goals. Scalable AI pertains to how data models, infrastructures, and algorithms can increase or decrease their complexity, speed, or size at scale in 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.



3. Triggers:

By getting awareness to maintain their body fit . Seeing their neighbour using this to maintain balanced diet.
Through internet, by seeing the feedback of others .

TR

4. Emotions:

BEFORE: Eating lot of food leads to many health issues. So they worry about their health, they become anxiety and fear.
AFTER: Customer feels motivated while using this model to live a healthy life.

EM

10. Your Solutions:

Existing solution takes time to identify the image. But this model takes less time than the existing one. It will also display the benefit of eating that fruit.

SL

8. Channels of behaviour:

Online: It allows analysis software to detect nutritional contents without physical contact.

Offline: Traditional methods of nutritional therapy can be done via offline

CH

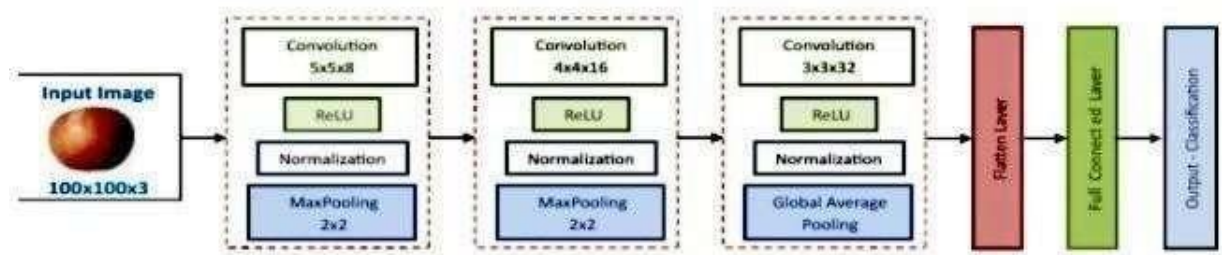
4. REQUIREMENT ANALYSIS

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	<p>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.</p> <p>In the web interface,</p> <ul style="list-style-type: none">• 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	<p>Creating a group of people, who are willing to be fit in their health and making them organized in a same place, through which they can collaborate and also can achieve their goals with others, by encouraging each other.</p> <p>The application gives the ability to ask questions about a problem in the fitness groups, through which they can work effectively.</p>
FR-3	USER SATISFYING	<p>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.</p> <p>On an periodic interval (like once in month), we need to interact one to one with each and every user to solve the queries</p>
FR-4	USER ENGAGEMENT	<p>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.</p>

NON-FUNCTIONAL REQUIREMENTS:

Following are the functional requirements for the proposed solution.

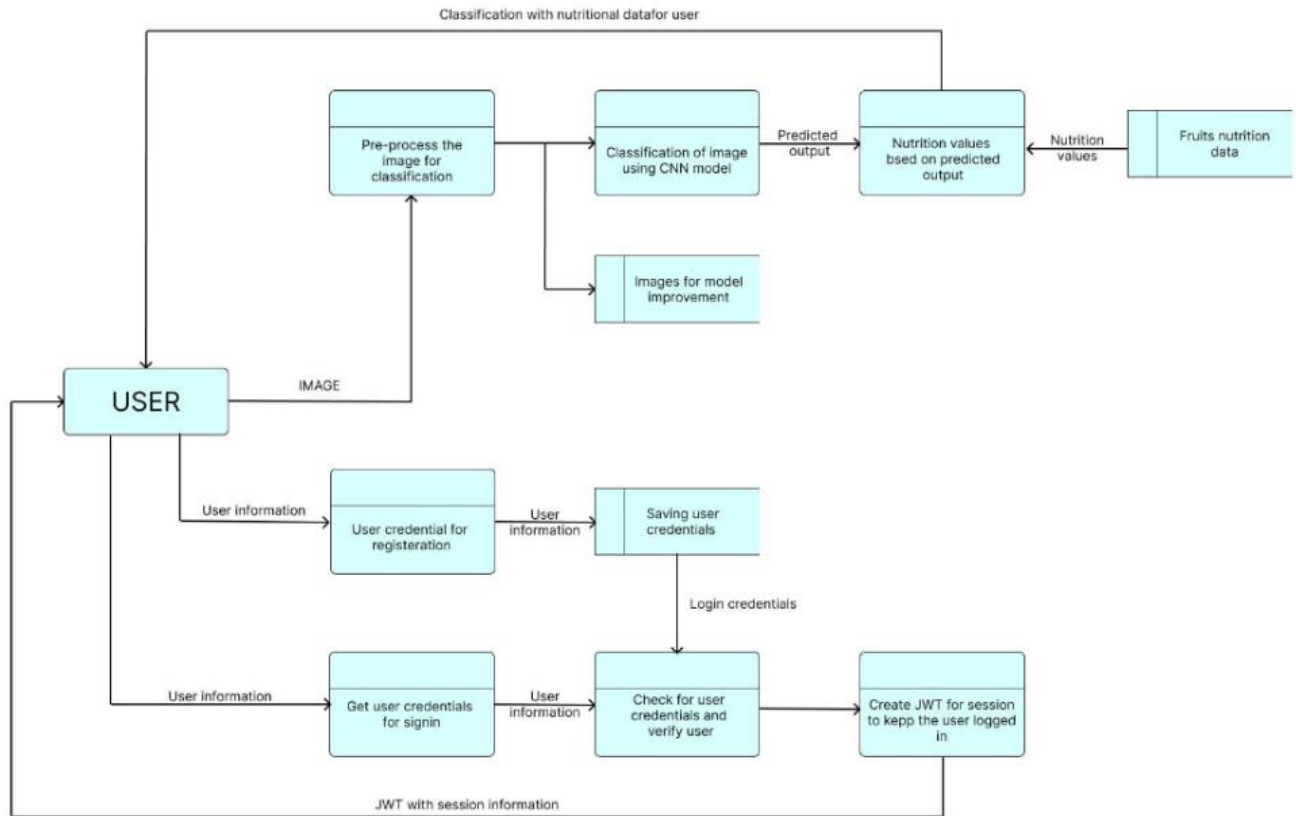
FR.NO	NON-FUNCTIONAL REQUIREMENTS	DESCRIPTION
NFR-1	USABILITY	<ul style="list-style-type: none">• No training is required to access the Nutrition Analyzer.• The results should be loaded within 30 seconds.• It should be user friendly and comfortable.• It should be simple and easy to use.• The results should be self explanatory so that it can be understood by common people.
NFR-2	SECURITY	<ul style="list-style-type: none">• 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	<ul style="list-style-type: none"> • How a person can find it is reliable? It is easy 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 takes too much time, to avoid this a reliable application should be 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 a proper fitness plan. • It should also provide the information on nutrition and health which it should prevent from health information on diseases, health risks and prevention guidelines. It should also provide 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 contain the calorie information, balanced diet plans, what type food can be consumed at what time etc..... So, by this way it can be reliable.
NFR-4	PERFORMANCE	<ul style="list-style-type: none"> • 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

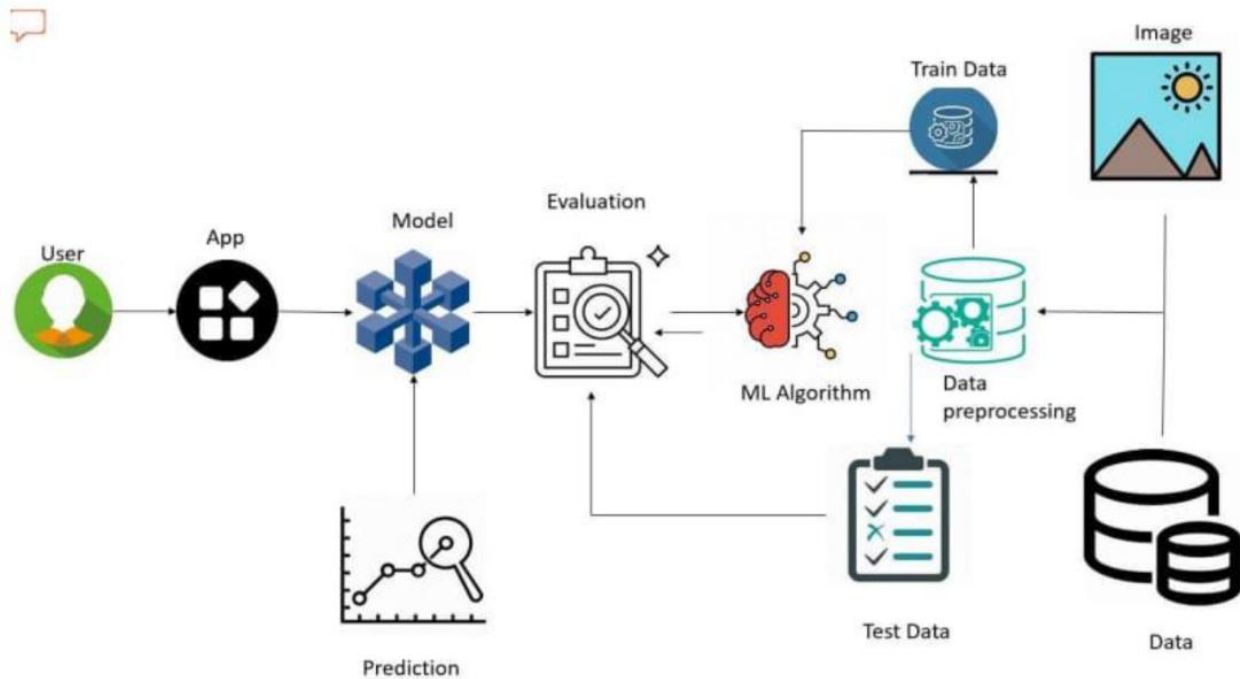
		<p>should not exceed more than 20mb.</p> <ul style="list-style-type: none"> • While consuming the page it should provide the response as much as possible without any delay or time traffic. • The connection should be properly maintained so that it can be used while travelling or in remote places. • The nutritious food to meet their dietary needs and the food preferences for an active and healthy life. • It should be consistently accessible, availability and affordability of foods and beverages that promote well-being and prevent from diseases. • It should be suitable in all situations that exist for all people, at all times.
NFR-5	AVAILABILITY	<ul style="list-style-type: none"> • Easy to access Data. • Avoids Data redundancy and inconsistency. • Fast and Efficient. • User Friendly.
NFR-6	SCALABILITY	<ul style="list-style-type: none"> • The architecture for AI powered Nutrition Analyzer for fitness provides the clear procedure for daily consumption of food and helps the user to maintain a healthy diet. • According to their tracking system implemented in architecture, it provides the proper mechanism to every individual of their nutrients intake which can be increased or decreased. • The premium amount for analyzer is very much 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 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. PROJECT PLANNING & SCHEDULING

6.1 Sprint Planning & Estimation

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-1	Data Collection	USN-1	Dataset - Collecting images of fruits like apples , cherries, lemons, dragonfruits, oranges for analysis	5	High	Afrid
Sprint-1	Image Preprocessing	USN-2	Image data augmentation - Increasing the amount of data by generating new data points from existing data	4	Medium	Sathwik
Sprint-1		USN-3	Image Data Generator Class - Used for getting the input of the original data	4	Medium	Hemanth
Sprint-1		USN-4	Applying image data generator functionality to train set and test set	4	Medium	Avez
Sprint-2	Modeling Phase	USN-5	Defining the model architecture - Building the model using deep learning approach and adding CNN layers	4	High	Afrid
Sprint-2		USN -6	Training , saving, testing and predicting the model	5	High	Avez
Sprint-2		USN- 7	Database creation for the input classes	4	High	Sathwik

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint- 2	Development phase	USN- 8	User database creation - It contains the details of users	3	Medium	Hemanth
Sprint-2		USN- 9	Home page creation - It shows options of the application	2	Low	Hemanth
Sprint-2		USN- 10	Login and registration page creation User can register and login through gmail with Id and password	2	Low	Avez
Sprint-3		USN- 11	Dashboard creation – Dashboard contains the information of user profile and features of the application	2	Low	Sathwik
Sprint-3		USN- 12	User Input Page Creation - It is for the user to feed the input images	4	Medium	Afrid
Sprint-3		USN- 13	Analysis and prediction page creation It shows the prediction of given user input	4	Medium	Hemanth
Sprint-3		USN- 14	Creation of about us , feedback and rating page – It shows application history and feedback page to users	4	Medium	Avez
Sprint-3	Application Phase	USN- 15	Building the python code and importing the flask module into the Project	6	High	Afrid
Sprint-4		USN- 16	Create the Flask application and loading the model	5	High	Sathwik

Sprint-4		USN- 17	API integration - Connecting front end and back end and perform routing and run the application	5	High	Afrid
Sprint-4	Deployment Phase	USN-18	Cloud deployment – Deployment of application by using IBM cloud	4	High	Hemanth
Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-4	Testing Phase	USN-19	Functional testing – Checking usability and accessibility	3	Medium	Avez
		USN-20	Non functional testing – Checking scalability and performance of the application	3	Medium	Sathwik

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-PMT2022TMD01549/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-PMT2022TMD01549/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 (MaxPooling2D)	(None, 14, 14, 32)	0
flatten (Flatten)	(None, 6272)	0
dense (Dense)	(None, 128)	802944
dense_1 (Dense)	(None, 5)	645
Total params: 813,733		
Trainable params: 813,733		
Non-trainable params: 0		

Configure The Learning Process

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

Train The Model

[illegible]

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/100Dataset-FIT2022/FIT00104R/Dataset/TEST_SET/ORANGE/30_100.jpg', target_size=(64,64))
img
```



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

[47] X

```
array([[255., 255., 252.],
       [249., 239., 231.],
       [235., 255., 255.],
       ...,
       [255., 255., 255.],
       [239., 239., 239.],
       [235., 235., 235.],
       ...,
       [255., 255., 251.],
       [254., 253., 231.],
       [255., 253., 254.],
       ...,
       [255., 235., 255.],
       [235., 255., 235.],
       [239., 239., 239.]])
```

```
[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'
```

```
def home():
    return render_template("home.html")

@app.route("/image", methods=['GET', 'POST'])
def image():
    return render_template("image.html")

@app.route("/predict", methods=['GET', 'POST'])
def predict():
    if request.method == 'POST':
        f = request.files["file"]
        basepath = os.path.dirname(__file__)
        filepath = os.path.join(basepath, "uploads", f.filename)
        f.save(filepath)

        img = image.load_img(filepath, target_size=(64, 64))
        x = image.img_to_array(img)
        x = np.expand_dims(x, axis=0)

        pred = np.argmax(model.predict(x), axis=-1)
        print("prediction:", pred)
        index = ['APPLES', 'BANANA', 'ORANGE', 'PINEAPPLE', 'WATERMELON']

        result = str(index[pred[0]])

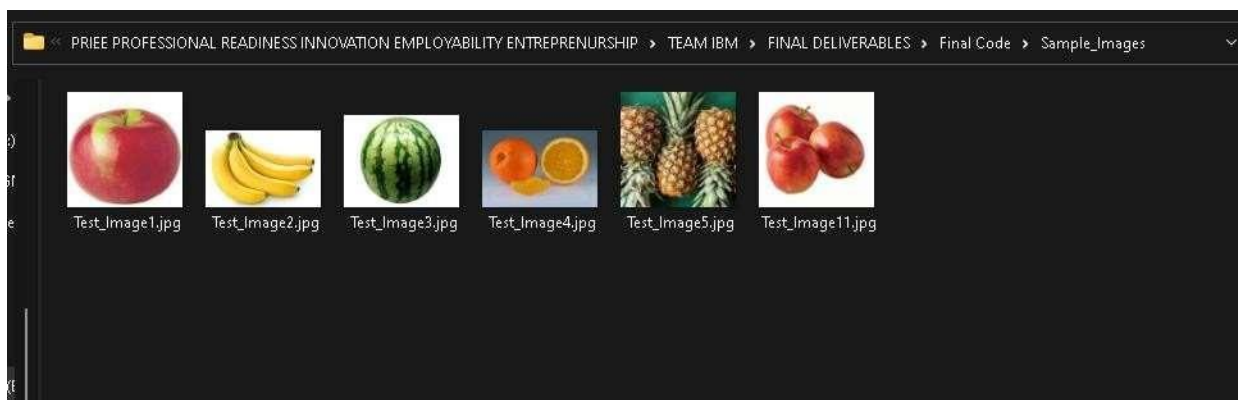
        s = result
        print(s)
        result = nutrition(result)
```

8. TESTING

8.1 Test Cases

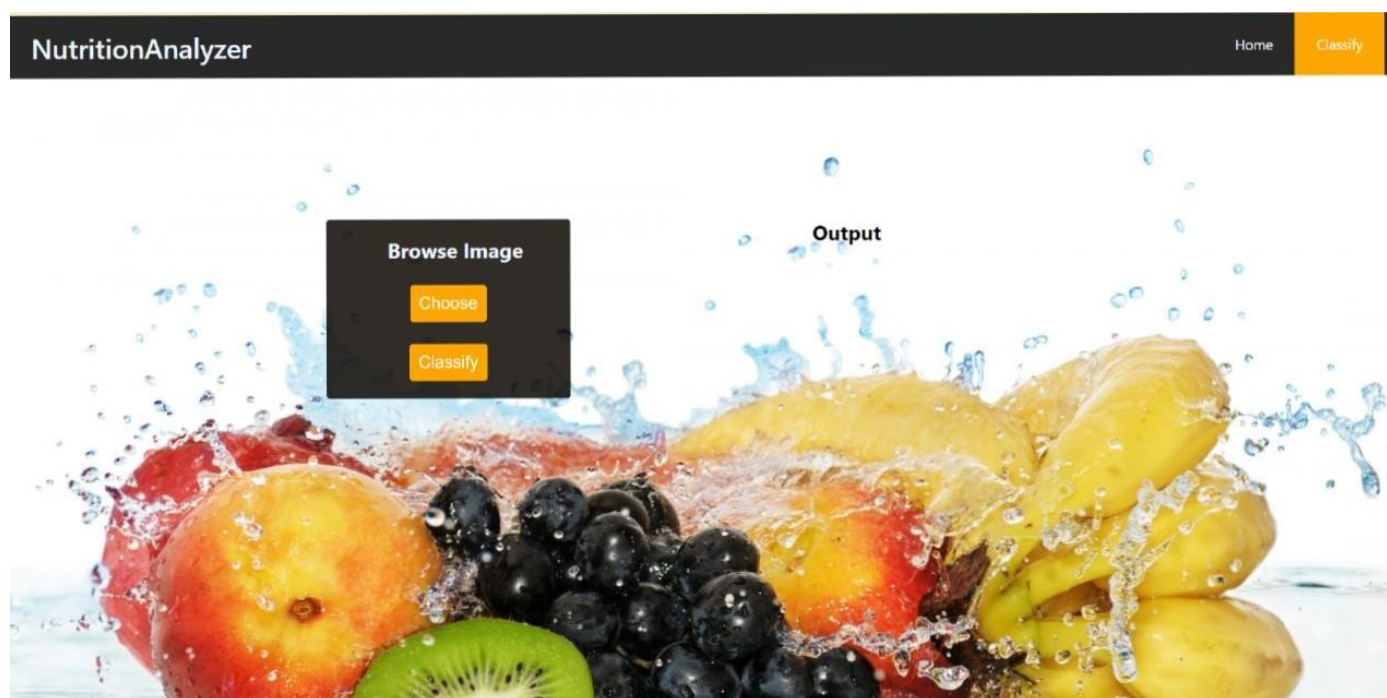
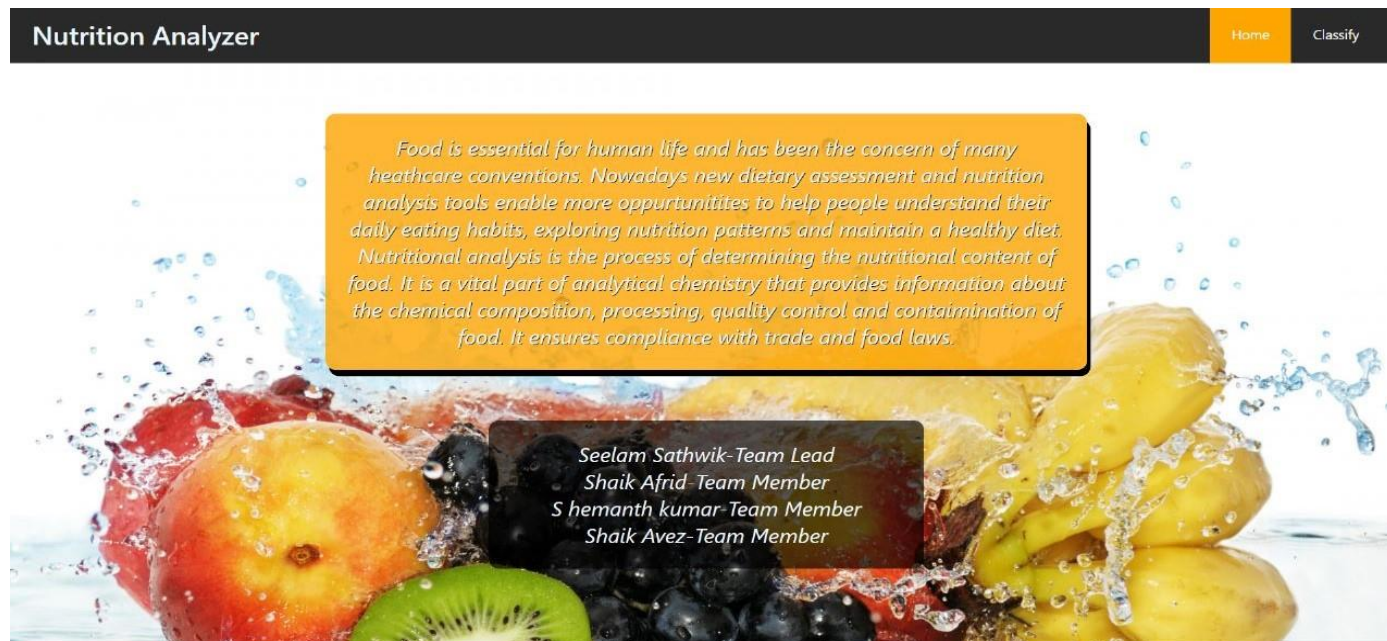


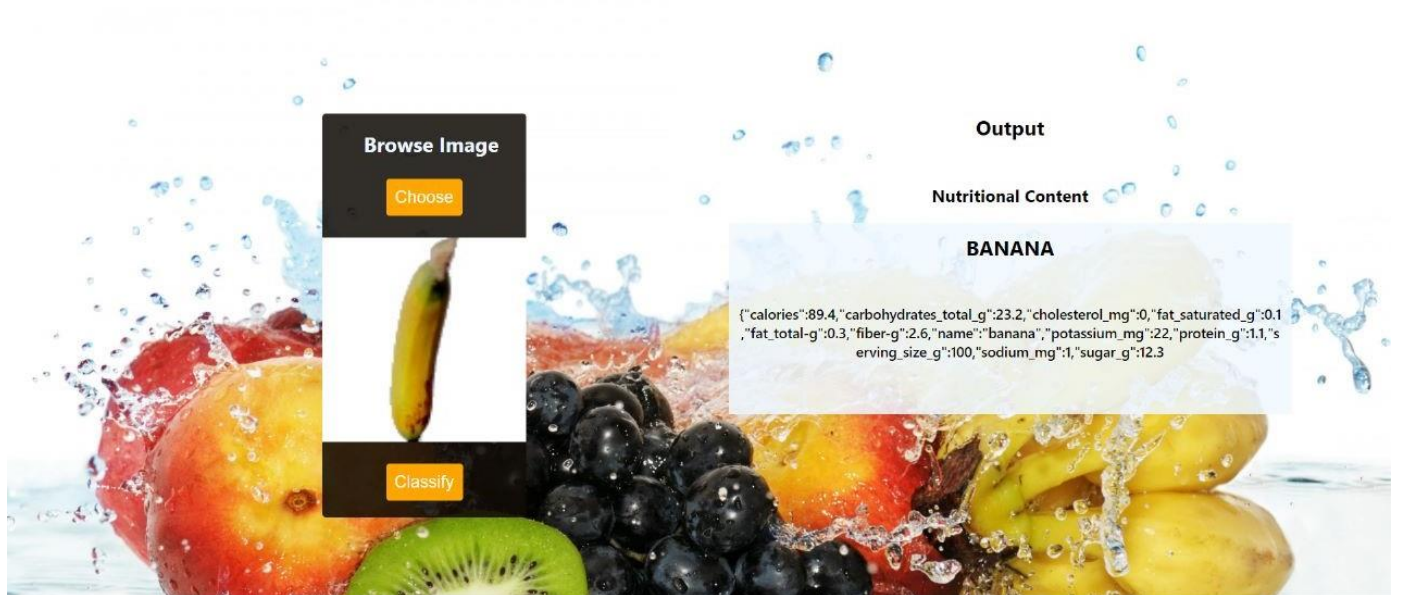
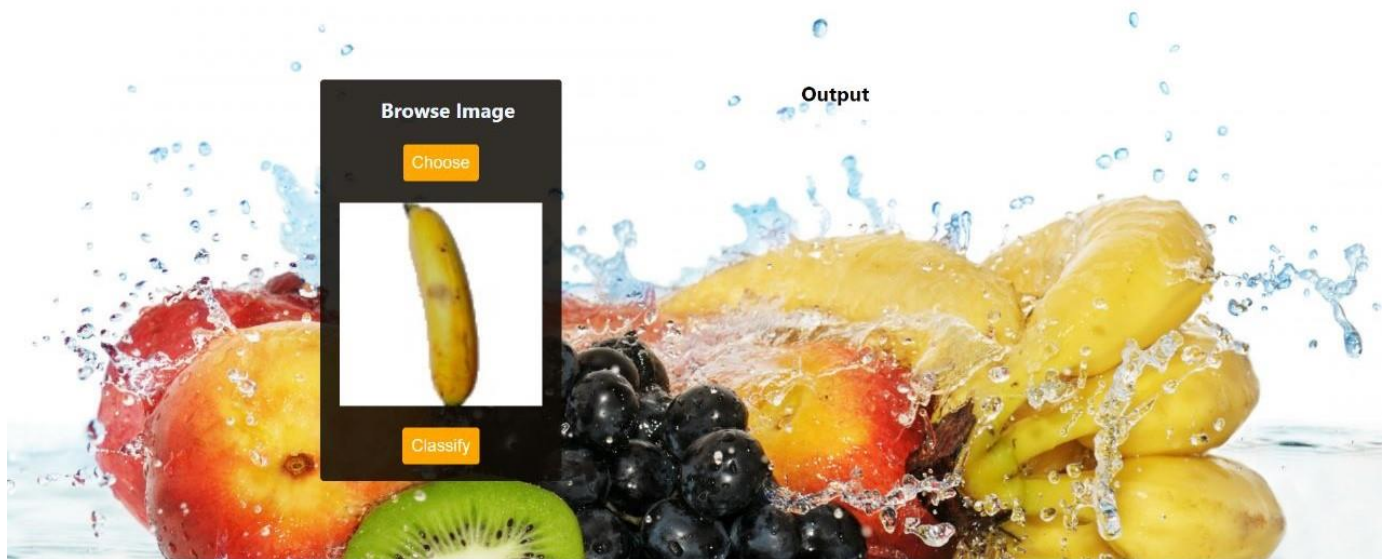
8.2 User Acceptance Testing



9.RESULTS

9.1 Output





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

```
# -*- coding: utf-8 -*-  
"""app
```

Automatically generated by Colaboratory.

Original file is located at

<https://colab.research.google.com/drive/17Yx8hioHPopCN4EFPvJYw6TuixjXcrwY>

Data Collection & Image Processing

Collecting images of different food items organized into subdirectories based on their respective names
"""

ls

cd /content/drive/MyDrive/dataset

```
pwd
```

```
!unzip TRAIN_SET.zip
```

```
!unzip TEST_SET-20221103T125409Z-001.zip
```

```
"""Importing and configuring the Image data generator library from Keras"""
```

```
from tensorflow.keras.preprocessing.image import ImageDataGenerator#scaling, zooming
```

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

```
test_datagen=ImageDataGenerator(rescale=1./255)
```

```
"""Applying Image data generator functionality to training set and testing set"""
```

```
x_train=train_datagen.flow_from_directory(r"/content/drive/MyDrive/dataset/TRAIN_SET", target_size=(64,64), color_mode='rgb', class_mode="categorical", batch_size=24)
```

```
x_test=test_datagen.flow_from_directory(r"/content/drive/MyDrive/dataset/TEST_SET", target_size=(64,64), color_mode='rgb', class_mode="categorical", batch_size=24)
```

```
x_train.class_indices
```

```
""" Model Building & Testing
```

```
"""
```

```
from tensorflow import keras
```

```
from tensorflow.keras.models import Sequential
```

```
from tensorflow.keras import layers
```

```
from tensorflow.keras.layers import Dense, Flatten
```

```
from tensorflow.keras.layers import Conv2D
```

```
from tensorflow.keras.layers import MaxPooling2D
```

```
from tensorflow.keras.layers import Dropout
```

```
"""Initializing the model"""
```

```
model=Sequential()
```

```
"""Creating the model"""
```

```
model.add(Conv2D(32,(3,3),activation="relu",strides=(1,1),input_shape=(64,64,3)))
```

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

```
model.add(Flatten())
```

```
model.add(Dense(300,activation="relu"))
```



```
model.add(Dense(300,activation="relu"))

model.add(Dense(5,activation="softmax"))

model.summary()

model.add(Dense(300,activation='relu'))
model.add(Dense(300,activation='relu'))

model.add(Dense(4,activation='softmax'))

model.compile(loss="categorical_crossentropy",optimizer="adam",metrics=['accuracy'])

len(x_train)

model.fit(x_train,epochs=10,validation_data=x_test,steps_per_epoch=len(x_train),validation_steps=len(x_test))

"""Saving the Model"""

model.save('train.h5')

model.save('dataset.h5')

model.save('fruits.h5')

model.save('nutrition.h5')

"""Testing the Model"""

import numpy as np
from tensorflow.keras.models import load_model
from tensorflow.keras.preprocessing import image

model=load_model('train.h5')

model=load_model('dataset.h5')

model=load_model('fruits.h5')

model=load_model('nutrition.h5')

img=image.load_img(r"/content/drive/MyDrive/dataset/TEST_SET/APPLES/32_100.jpg")

img

img=image.load_img(r"/content/drive/MyDrive/dataset/TEST_SET/APPLES/32_100.jpg",target_size=(64,64))

img

x=image.img_to_array(img)
```

```
x
```

```
x=np.expand_dims(x,axis=0)
```

```
x
```

```
pred = model.predict  
pred
```

```
predict_x=model.predict(x_test)  
classes_x=np.argmax(predict_x,axis=0)
```

```
predict_x
```

```
classes_x
```

```
x_test.class_indices
```

```
index=['APPLES','BANANA','ORANGE','PINEAPPLE','WATERMELON']
```

```
index[np.argmax(pred)]
```

```
"""Build Python 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
```

```
"""Creating our flask application and loading our model by using the load_model method"""
```

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

```
"""Routing to *Html* page"""
```

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

```
@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 launch():
    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=['APPLES','BANANA','ORANGE','PINEAPPLE','WATERMELON']
        result=str(index[pred[0]])
        x=result
        print(x)
        result=nutrition(result)
        print(result)
        return render_template("0.html",showcase=(result))

x=result
print(x)
result=nutrition(result)
print(result)

def nutrition(index):

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

    querystring = {"query":index}

    headers = {
        'x-rapidapi-key': "5d797ab107mshe668f26bd044e64p1ffd34jsnf47bfa9a8ee4",
        'x-rapidapi-host': "calorieninjas.p.rapidapi.com"
    }
    response = requests.request("GET", url="https://calorieninjas.p.rapidapi.com/v1/nutrition", headers =
        {
            'x-rapidapi-key': "5d797ab107mshe668f26bd044e64p1ffd34jsnf47bfa9a8ee4",
            'x-rapidapi-host': "calorieninjas.p.rapidapi.com"
        }, params= {"query":index} )
    print(response.text)
    return response.json()['items']

if __name__ == "__main__":
    # running the app
    app.run(debug=False)

```