1.INTRODUCTION

1.1 Project Overview:

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

1.2 Purpose:

Food is essential for human life and has been the concern of many health conventions. Nowadays new dietary assessment and nutrition analysis tools enables 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 chemical composition, processing, quality control and contamination of food. So there is a need for model to find the nutrition content present in food.

2 LITERATURE SURVEY

2.1 Existing problem:

Literature has indicated that accurate dietary assessment is very important for assessing the effectiveness of weight loss interventions. However, most of the existing dietary assessment methods rely on memory. With the help of pervasive mobile devices and rich cloud services, it is now possible to develop new computer-aided food recognition system for accurate dietary assessment. However, enabling this future Internet of Things based dietary assessment imposes several fundamental challenges on algorithm development and system design. In this paper, we set to address these issues from the following two aspects: (1) to develop novel deep learning-based visual food recognition algorithms to achieve the best-in-class recognition accuracy; (2) to design a food recognition system employing edge computing based service computing.

2.2 Reference:

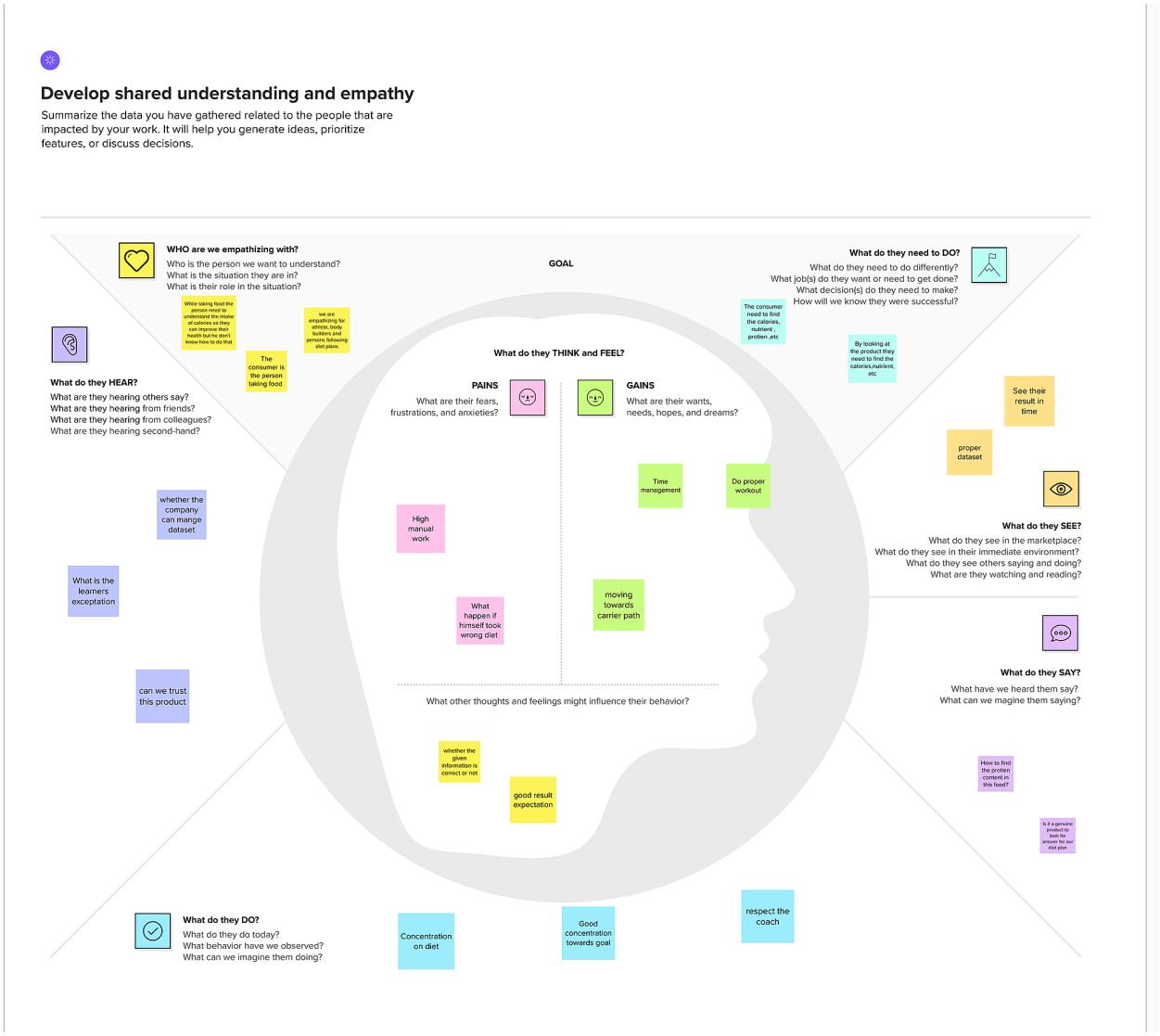
https://ieeexplore.ieee.org/ielaam/4629386/8332642/7837725-aam.pdf

2.3 Problem Statement Definition:

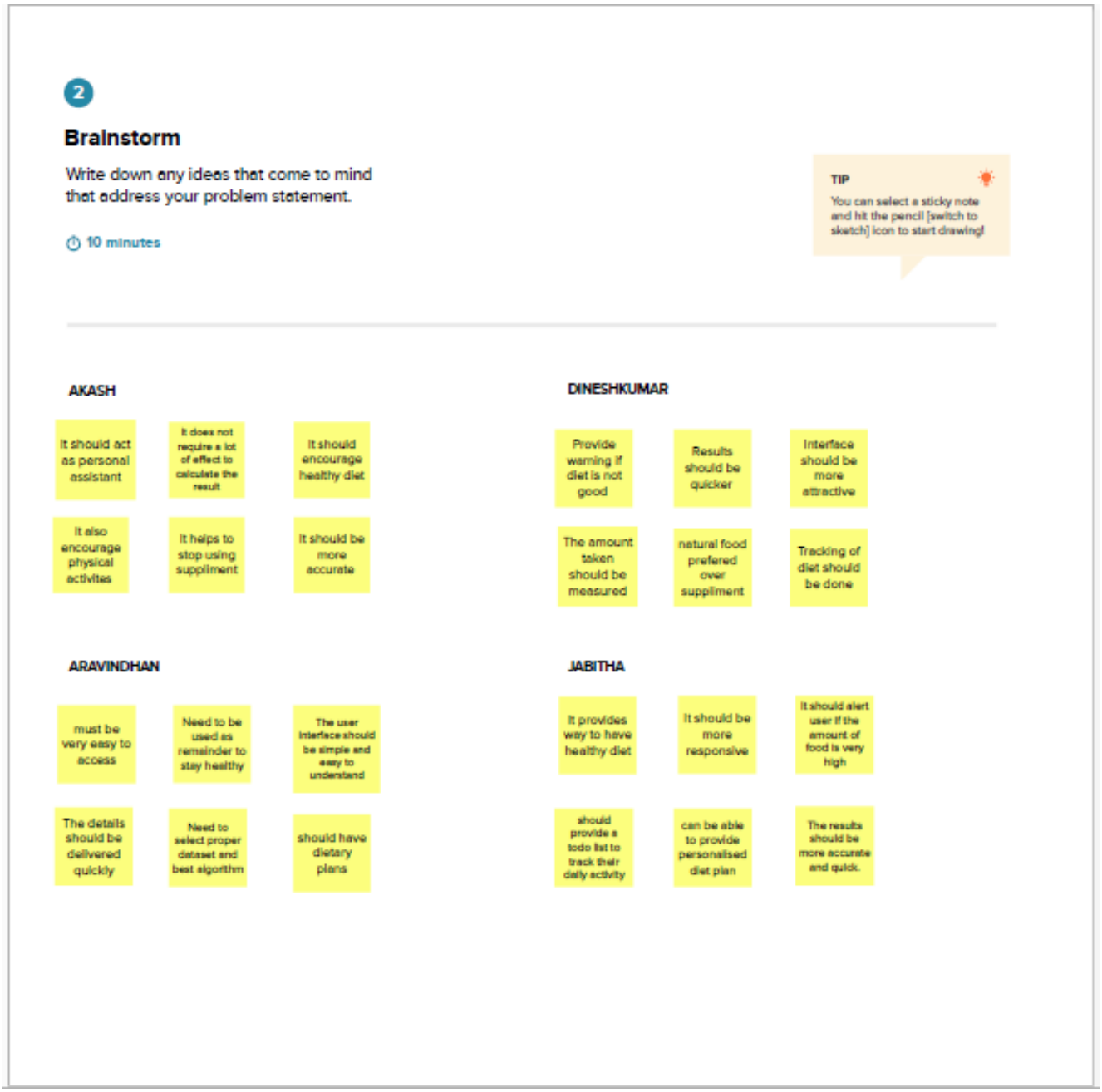
In this paper, we aimed to develop a practical deep learning based food recognition system for dietary assessment within the edge computing service infrastructure. The key technique innovation in this paper includes: the new deep learning-based food image recognition algorithms and the proposed real-time food recognition system employing edge computing service paradigm. Our experimental results on two challenging data sets using our proposed approach have demonstrated that our system has achieved the three major objectives: (1) it outperforms the results from all existing approaches in terms of recognition accuracy; (2) it develops a real-time system whose response time is close to the minimal of existing techniques; and (3) it saves the energy by keep the energy consumption equivalent to the minimum of the existing approaches.

3 IDEATION & PROPOSED SOLUTION

3.1 Empathy Map Canvas:



3.2 Ideation & Brainstorming:



3

Group Ideas

Take turns sharing your ideas while clustering similar or related notes as you go. Once all sticky notes have been grouped, give each cluster a sentence-like label. If a cluster is bigger than six sticky notes, try and see if you and break it up into smaller sub-groups.

20 minutes

TIP

Add customizable tags to sticky notes to make it easier to find, browse, organize, and categorize important ideas as themes within your mural.

Based on diet

natural food prefered over suppliment

It helps to stop using suppliment

Provide warning if diet is not good

Based on interface

must be very easy to access

It should be more responsive

Based on product technology

It should be more accurate

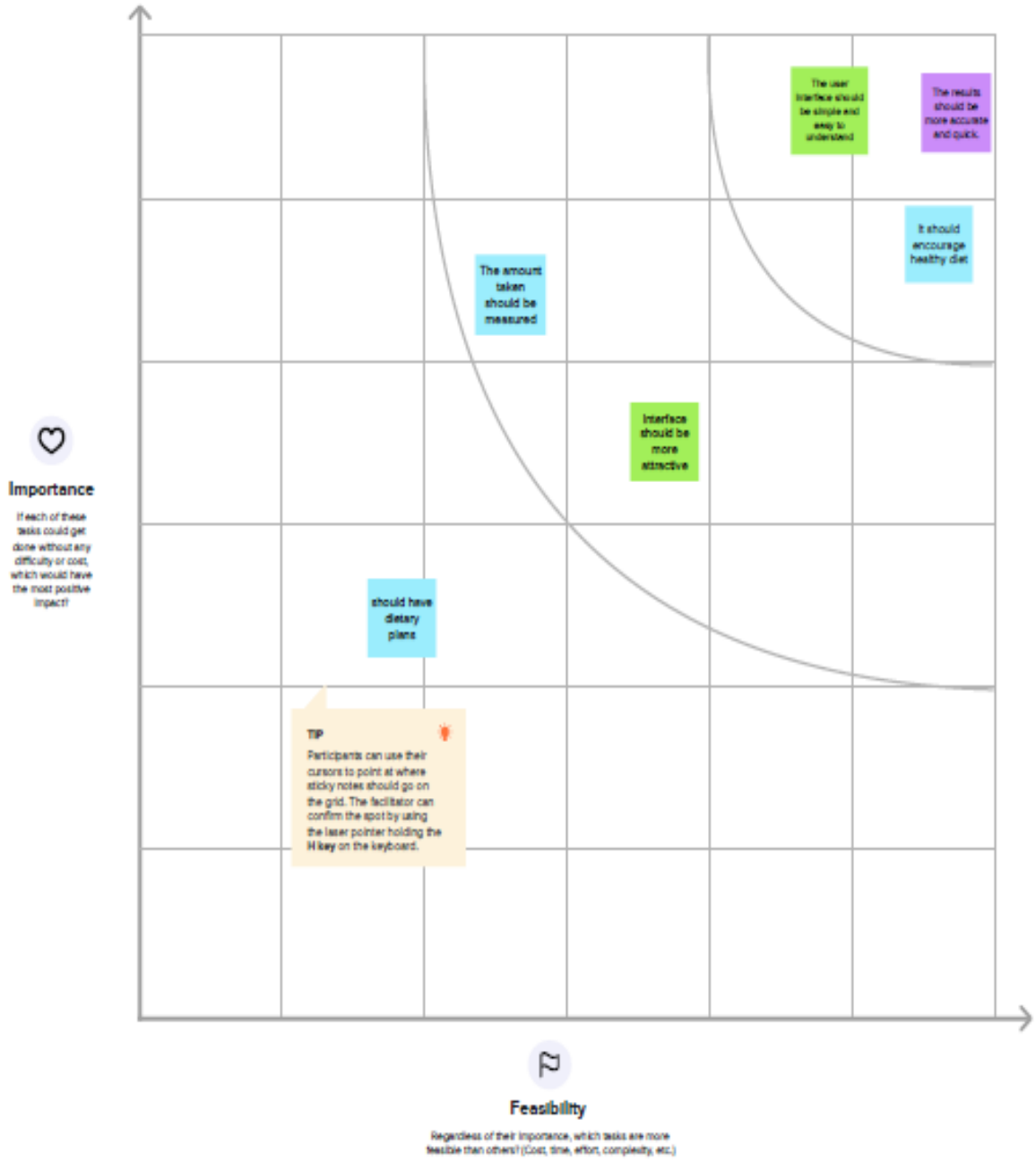
The amount taken should be measured

4

Prioritize

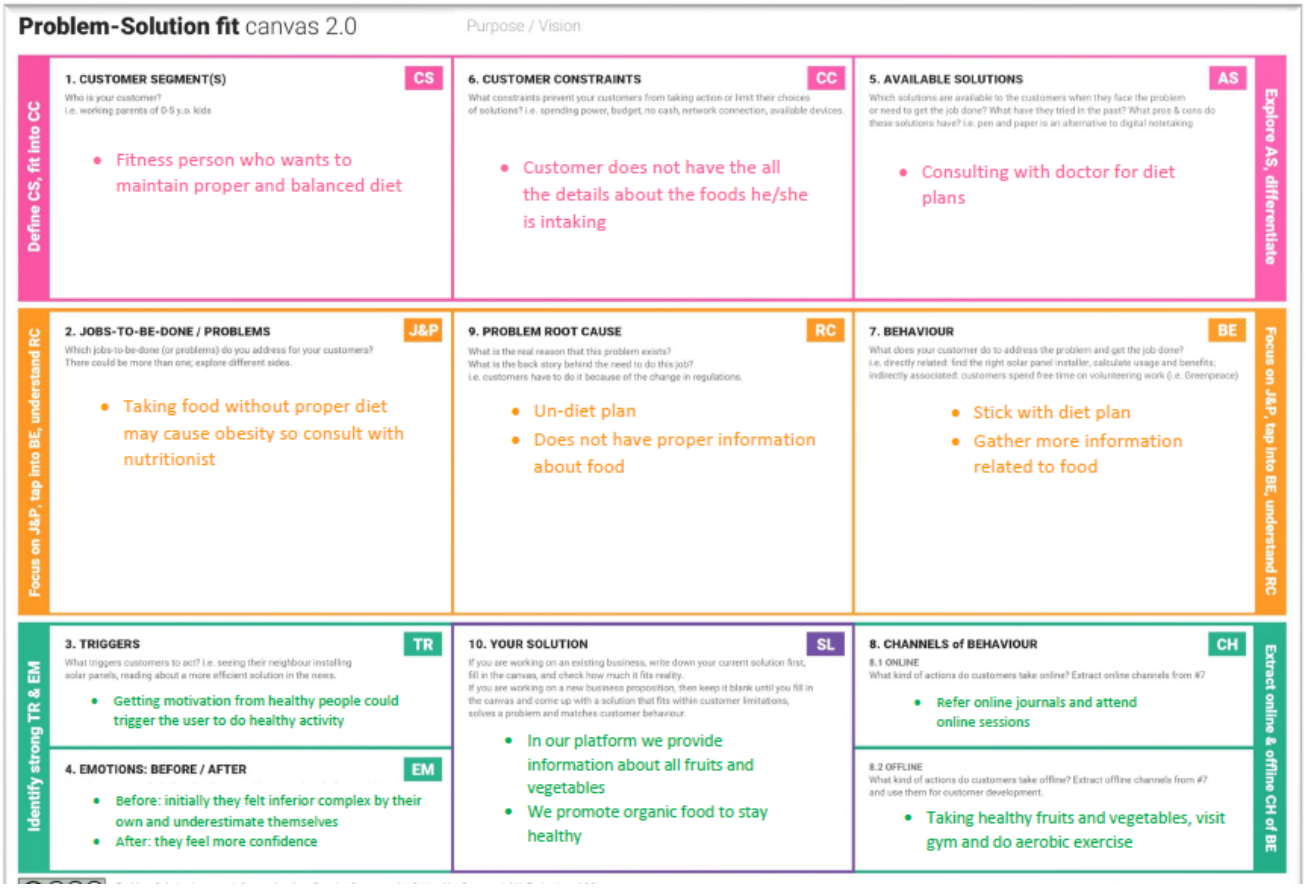
Your team should all be on the same page about what's important moving forward. Place your ideas on this grid to determine which ideas are important and which are feasible.

20 minutes



3.3 Proposed Solution:

3.4 Problem Solution fit:



4. REQUIREMENT ANALYSIS

4.1 Functional Requirement:

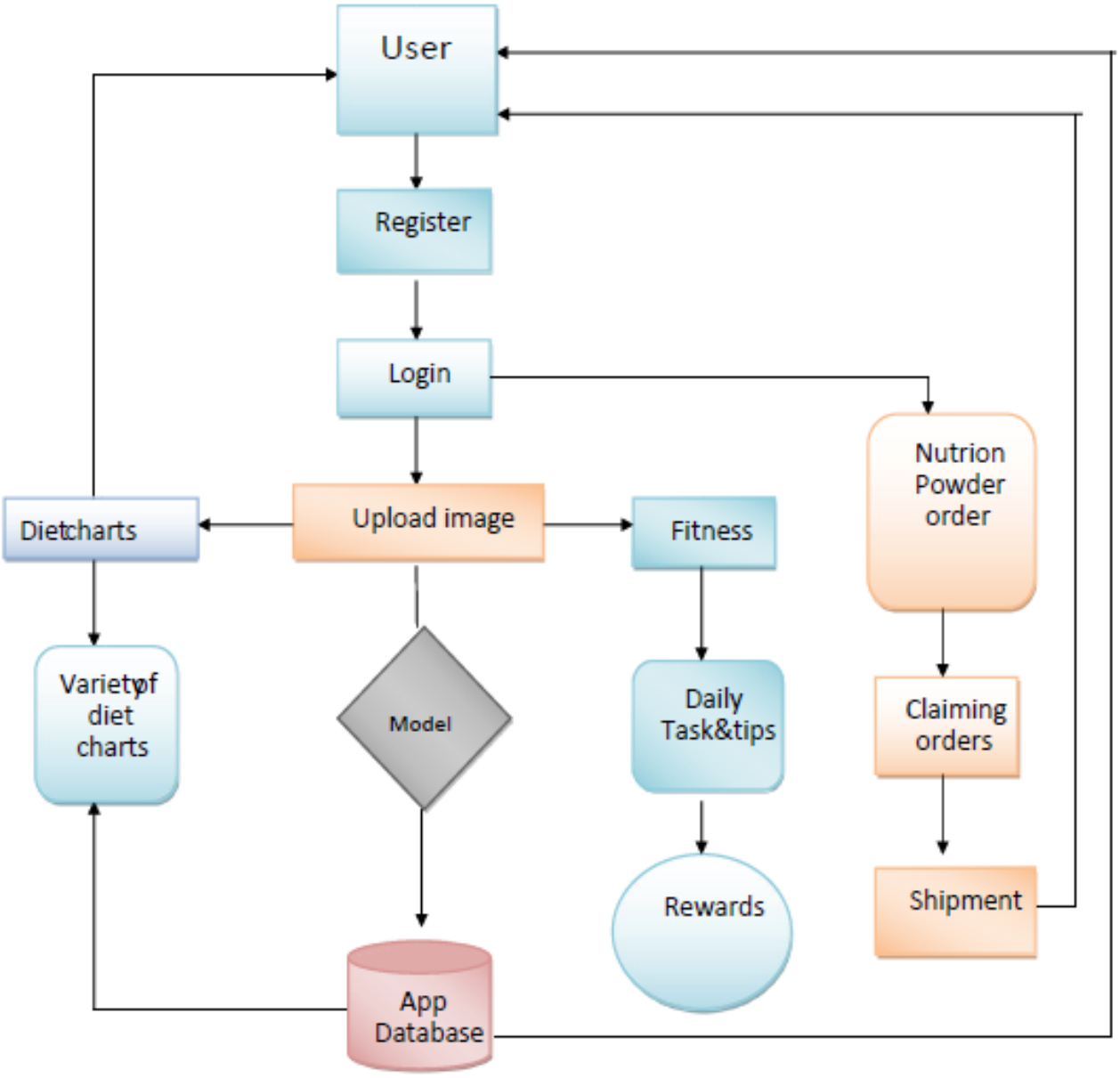
FR No.	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)
FR-1	User Registration	Install and register through Gmail or google (Two step verification) and updating the profile.
FR-2	User Confirmation	Confirmation via mail
FR-3	User Uploads	Interacts with user interface and upload image which they want to analyze.
FR-4	User Accessibility	Ordering nutrition powders, chat with nutritionist, analyzing food, access different diet charts and fitness tips, can solve their queries with nutritionist and can know the importance of food item.
FR-5	User Benefits	Have a knowledge about nutrients in food, lead a healthy life, knowing daily exercises necessary to live a peaceful life.
FR-6	User Instructions	Follow the diet charts and completing the daily tasks

4.2 Non-Functional requirements:

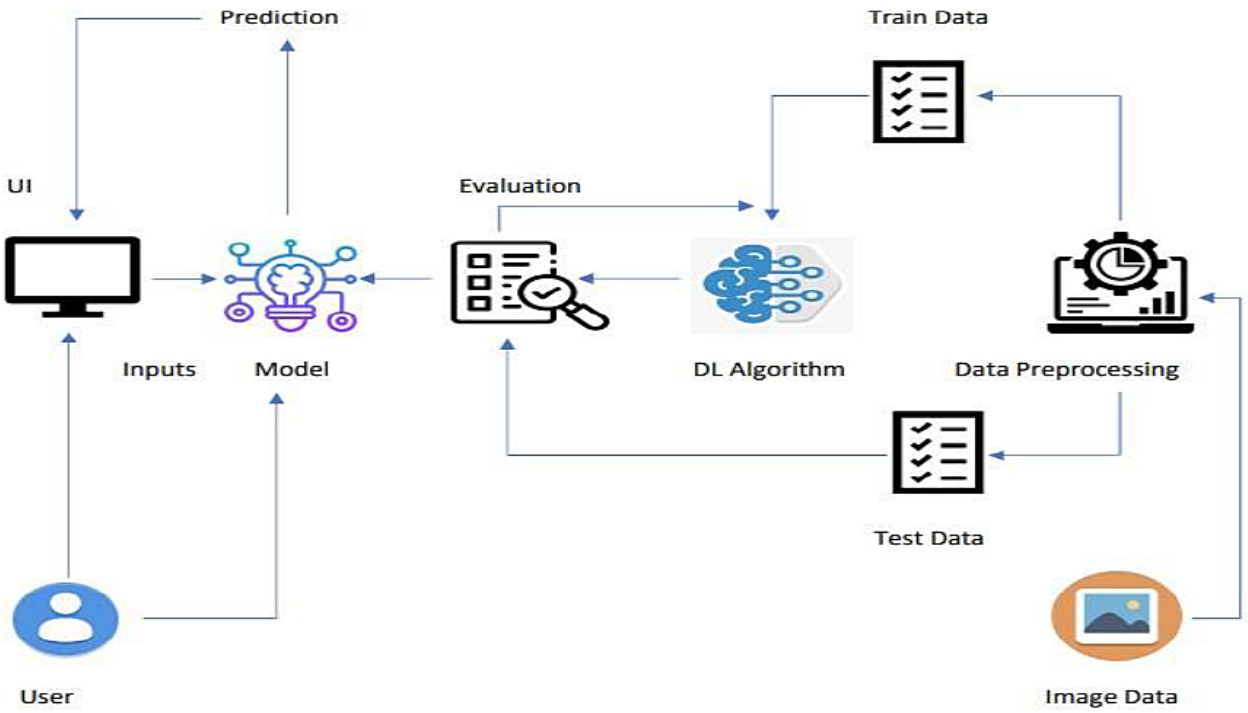
FR No.	Non-Functional Requirement	Description
NFR-1	**Usability**	An user friendly web application, dynamic and attractive user interface, maintaining the records of users and easily satisfies the user needs.
NFR-2	**Security**	Users' information in the web application are protected by advanced security system.Implementation of proper logging.Keep your application up to date.Authentication.
NFR-3	**Reliability**	Fault less application.A piece of software operating without failure. The website's load time is not more than one second for users.
NFR-4	**Performance**	Fast and quick analyzation of food images, is done as a GPU used for the model is more fast in analyzing and uploading the user uploaded the image. Occupation of less storage space.
NFR-5	**Availability**	Available in the internet, in application diet charts, excersing tips, food analyzer through camera, online deliveries are available.
NFR-6	**Scalability**	It works in high-speed authentication of people's information, quick response for queries from nutritionist, highly reliable, on time delivery and provide appropriate information.

5. PROJECT DESIGN

5.1 Data Flow Diagrams:



5.2 Solution & Technical Architecture:



5.3 User Stories:

User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority	Release
Customer (Web user)	Dashboard	USN-3	As a user, I can see the home page of the website of the nutrition analyzer	I can register & access the dashboard	Low	Sprint-3
		USN-4	As a user, I can input image of food in the upload field	I can access the upload field without any problem.	Low	Sprint-3
Administrator	Prediction	USN-1	Here the model will be created and it predict the image using deep learning Algorithms such as CNN.	In this I can have correct prediction on the particular algorithms.	High	Sprint-1
		USN-2	The model is trained on the IBM cloud	In this the model is trained on the IBM cloud	Medium	Sprint-2
	Classifier	USN-5	Here I will send all the model outputs to classifie to produce final results.	In this I will find the correct results.	Low	Sprint-4

6. PROJECT PLANNING & SCHEDULING

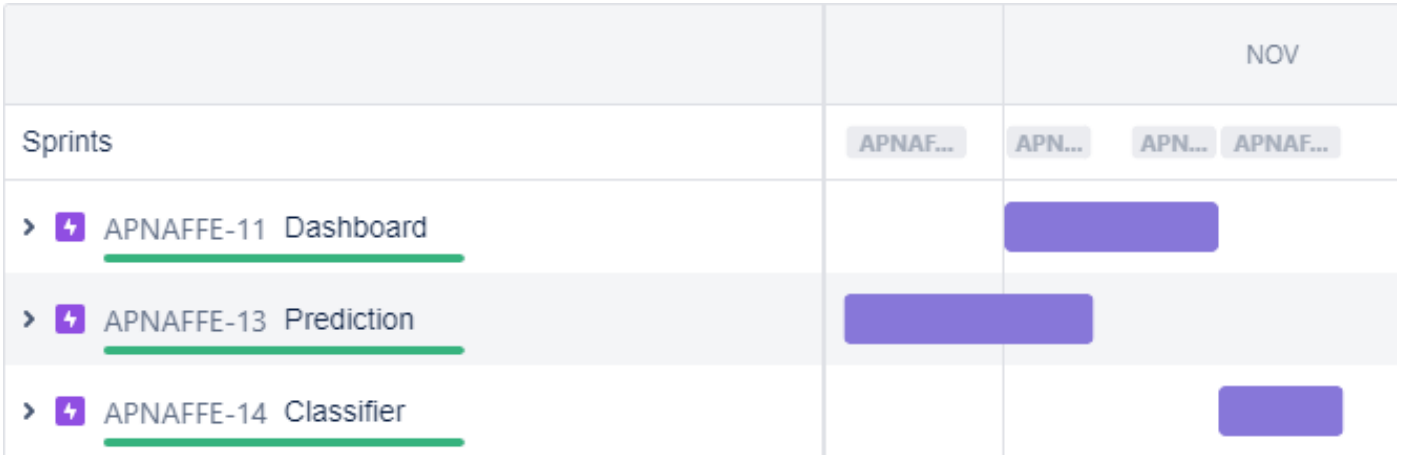
6.1 Sprint Planning & Estimation:

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-1	Prediction	USN-1	Here the model will be created and it predict the image using deep learning Algorithms such as CNN.	20	High	Akash, Aravindhan, Dineshkumar, Jabitha
Sprint-2	Prediction	USN-2	The model is trained on the IBM cloud	10	Medium	Akash, Aravindhan, Dineshkumar, Jabitha
Sprint-2	Dashboard	USN-3	As a user, I can see the home page of the website of the nutrition analyser	10	Low	Akash, Aravindhan, Dineshkumar Jabitha
Sprint-3	Dashboard	USN-4	As a user, I can input image of food in the upload field	20	Low	Akash, Aravindh, Dineshkumar, Jabitha
Sprint-4	Classifier	USN-5	Here I will send all the model outputs to classifier in order to produce final results.	20	Low	Akash, Aravindh, Jabitha, Dineshkumar,

6.2 Sprint Delivery Schedule:

Sprint	Total Story Points	Duration	Sprint Start Date	Sprint End Date (Planned)	Story Points Completed (as on Planned End Date)	Sprint Release Date (Actual)
Sprint-1	20	6 Days	24 Oct 2022	29 Oct 2022	20	29 Oct 2022
Sprint-2	20	6 Days	31 Oct 2022	05 Nov 2022	20	05 Nov 2022
Sprint-3	20	6 Days	07 Nov 2022	12 Nov 2022	20	12 Nov 2022
Sprint-4	20	6 Days	14 Nov 2022	19 Nov 2022	20	19 Nov 2022

6.3 Reports from JIRA:



7. CODING & SOLUTIONING

7.1 Feature 1:(Added home page)

home.html:

```
<!DOCTYPE html>
<html lang="en" id="home">
<head >
    <meta charset="UTF-8">
    <title>Home_page</title>
<link rel="stylesheet" href="{{
url_for('static',filename='css/home.css')}}">
</head>
<body >
    <img src="{{
url_for('static',filename='image/home.jpg')}}">
    <div class="topnav">
        <a class="active" href="/" >Home</a>
        <a href="/image">Classify</a>

        <h1>AI-powered Nutrition Analyzer for Fitness
Enthusiasts</h1>

    </div>
    <div id="div_cont">
        <div class="content">
            <p>Food is essential for human life and
<br>has been the concern of many healthcare
<br>conventions. Nowadays new dietary assessment<br> 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.</p>
        </div>
        <a href="/image"><button>Click To
Start</button></a>
    </div>

</body>
</html>
```

home.css:

```
body{
    margin:0;
    font-family:Arial,Helvetica,sans-serif;
    overflow-x: hidden;
}
img{
    pointer-events:none;
    position:absolute;
    width:100vw;
    height:100vh;
    z-index: -1;
}
.topnav{
    overflow:hidden;
    background-color:#333;
    position:relative;
```

```
}

.topnav a {
    float:left;
    color: #f2f2f2;
    text-align:center;
    padding: 16.5px 16px;
    text-decoration:none;
    font-size: 17px;
}

.topnav a:hover{
    background-color: #ddd;
    color:black;
}

.topnav a.active {
    background-color: #04AA6D;
    color:white;
}
.topnav h1 {
    font-size:1.5em;
    margin-left:30%;
    color:#0fca0f;
    margin-top:10px;
}
.content{
    width:800px;
    padding: 10px;
    font-size:1.6em;
    color:#00ff40;
}
```

```
#div_cont{
    margin-top:3%;
    margin-left:2%;
    display: inline-block;
    overflow:hidden;
    position:absolute;
    width:100%
}


button{
    font-size:4em;
    background:#0fca0f;
    color: #fff;
    padding: 10px 30px;
    border: none;
    border-radius: 3em;
    cursor: pointer;
}
```

app.py

```
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="template")

# initializing a flask app
# Loading the model
model = load_model('nutrition.h5')
@app.route('/')  # route to display the home page
def home():
    return render_template('home.html')#rendering the
home page

if__name__=='__main__':
    app.run()
```

output:

7.2 Feature 2(Added image uploading feature):

image.html

```
<!DOCTYPE html>
<html lang="en" >
<head>
    <meta charset="UTF-8">
    <title>nutrition_analyzer</title>
    <link rel="stylesheet" href="{{
url_for('static',filename='css/image.css')}}">
<script src="{{url_for('static',
filename='js/image.js')}}"></script>
</head>
<body>
    <img id="image_bc" src="{{
url_for('static',filename='image/image.jpg')}}">
    <div class="topnav">
        <a class="active" href="/" >Home</a>
        <a href="/image">Classify</a>

       <h1>AI-powered Nutrition Analyzer for Fitness
Enthusiasts</h1>

        </div>
    <h1>Upload image to classify</h1>
    <div id="cont">

        <form action="/predict" id="upload-file"
method="post" enctype="multipart/form-data">
```

```
            <div class="label_div">
                <label for="imageUpload" class="upload-
label" style="cursor: pointer">choose...
                </label></div>
            <input type="file" name="image"
id="imageUpload" accept="image/*" style="display:none"
onchange="loadFile(event)">
            <img id="output" width="300">
            <button type="submit" id="submit"
disabled>classify</button>
        </form>
    </div>

</body>
</html>
```

image.css

```
body{
    margin:0;
    font-family:Arial,Helvetica,sans-serif;
    overflow-x: hidden;
}

.topnav{
    overflow:hidden;
    background-color:#333;
}

.topnav a {
    float:left;
    color: #f2f2f2;
```

```
    text-align:center;
    padding: 16.5px 16px;
    text-decoration:none;
    font-size: 17px;
}

.topnav a:hover{
    background-color: #ddd;
    color:black;
}

.topnav a.active {
    background-color: #ff4d4d;
    color:white;
}
.topnav h1 {
    font-size:1.5em;
    margin-left:30%;
    color:#ff4d4d;
    margin-top:10px;
}
input, label ,button{
    display:block;
}


.label_div{
    background:#ff1a1a ;
    padding: 10px 30px;
    border-radius: 3em;
    width:200px;
    margin-left:-30%;
}
```

```
.upload-label{
    font-size:3em;
}

#output{
    margin-top:20px;
}

button{
    font-size:2em;
    cursor:pointer;
}


#image_bc{
    pointer-events:none;
    position:absolute;
    width:100vw;
    height:100%;
    z-index: -1;
}

#cont{
    margin-top:5%;
    margin-left:55%;
}

h1{
    margin-left:40%;
}
button{
    border-radius: 3em;
    font-size:2em;
```

```
    padding: 10px 30px;
}
```

image.js

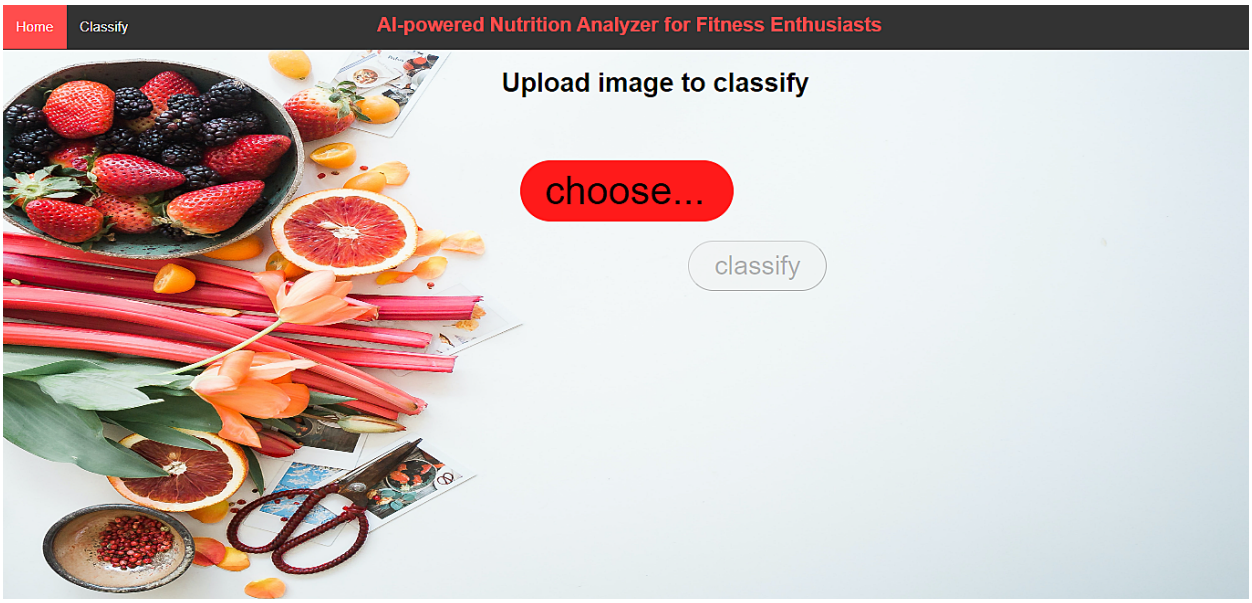
```
var loadFile = function(event) {
    document.getElementById('submit').disabled = false;

document.getElementById('submit').style.backgroundColor='
#ff0000';
    var image = document.getElementById('output');
    image.src =
URL.createObjectURL(event.target.files[0]);
};
```

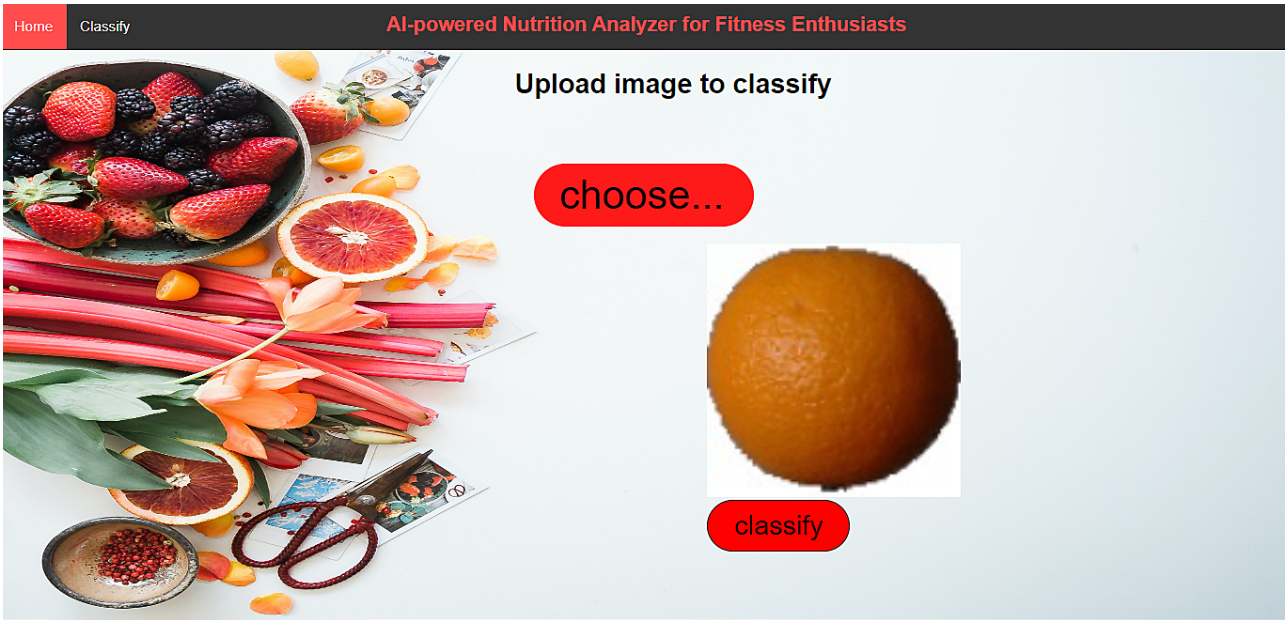
app.py

```
from flask import Flask, render_template, request
import os
import numpy as np  # used for numerical analysis
from tensorflow.keras.models import load_model
from tensorflow.keras.preprocessing import image
import requests

app = Flask(__name__, template_folder="template")
model = load_model('nutrition.h5')

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

@app.route('/image')
def image1():
```

```
    return render_template("image.html")



if __name__=='__main__':
    app.run()
```

output:





7.3 Feature 3(image classifying feature added)

imageprediction.html

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <title>prediction</title>
    <link rel="stylesheet" href="{{
url_for('static',filename='css/imagepredition.css')}}">

</head>
<body>
    <img id="image_bc" src="{{
url_for('static',filename='image/predict.jpg')}}">
    <div class="topnav">
        <a class="active" href="/" >Home</a>
        <a href="/image">Classify</a>
```

```
        <h1>AI-powered Nutrition Analyzer for Fitness
Enthusiasts</h1>

    </div>
    <h1>Food classified as <span>{{x}}</span></h1>
    <div class="content">
        {{y}}
    </div>
</body>
</html>
```

imageprediction.css

```
body{
    margin:0;
    font-family:Arial,Helvetica,sans-serif;
}

.topnav{
    overflow:hidden;
    background-color:#333;
}

.topnav a {
    float:left;
    color: #f2f2f2;
    text-align:center;
    padding: 16.5px 16px;
    text-decoration:none;
    font-size: 17px;
```

```
}

.topnav a:hover{
    background-color: #ddd;
    color:black;
}

.topnav a.active {
    background-color: #a633cc;
    color:white;
}


.topnav h1 {
    font-size:1.5em;
    margin-left:30%;
    color:#a633cc;
    margin-top:10px;
}


h1{
    margin-top:40px;
    margin-left:10%;
    font-size:2em;
}
span{
    color:#a633cc;
    font-size:2em;
}
.content{
    margin-top:10%;
    margin-left:1%;
```

```
    width:800px;
    border: 3px solid #a633cc;
    padding: 10px;
    font-size:2em;
}
img{
    pointer-events:none;
    position:absolute;
    width:100vw;
    height:100vh;
    z-index: -1;
```

app.py

```
from flask import Flask, render_template, request
import os
import numpy as np  # used for numerical analysis
from tensorflow.keras.models import load_model
from tensorflow.keras.preprocessing import image
import requests

app = Flask(__name__, template_folder="template")  #
model = load_model('nutrition.h5')


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

```
@app.route('/image')
def image1():
    return render_template("image.html")


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

        img = image.load_img(filepath, grayscale=False,
                             target_size=(64, 64))
        x = image.img_to_array(img)  # image to array
        x = np.expand_dims(x, axis=0)  # changing the
shape
        pred = np.argmax(model.predict(x))
        print(pred, model.predict(x))
        op = ['APPLES', 'BANANA', 'ORANGE', 'PINEAPPLE',
'WATERMELON']  # Creating list of output categories
        result = op[pred]
        print(result)
        x = result
        result = nutrition(result)
        print(result)
        return render_template("imageprediction.html",
y=(result), x=(x))

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,
headers=headers, params=querystring)
    print(response.text)
    return response.json()['items']
if __name__=='__main__':
    app.run()
```

output:

8. TESTING

8.1 Test Cases:

	Test Scenarios	
1	Verify user is able to see the home page and click on click to start button	
2	Verify the UI elements in home page	
3	Verify user is able to upload and redirect to image_prediction page	
4	Verify the UI elements in image page	
5	Veriify user is able to see image_prediction page and see the result	
	Upload	
1	Verify user is able to upload image for getting nutrient content	
2	Verify user is able to see the nutrient content of the uploaded image	

Date	14-Nov-22
Team ID	PNT2022TMID33113
Project Name	AI - powered Nutrition Analyzer for Fitness Enthusiasts
Maximum Marks	4 marks

Test case ID	Feature Type	Component	Test Scenario	Pre-Requisite	Steps To Execute	Test Data	Expected Result	Actual Result	Status	Commnets	BUG ID	Executed By
HomePage_TC_001	Functional	Home Page	Verify user is able to see the home page and click on click to start button	internet,browser	1.Enter URL and click go 2.Click on Click To Start	http://127.0.0.1:5000/	home page should be displayed and click to start button should be visible	Working as expected	Pass	Steps are clear to follow	nil	AKASH
HomePage_TC_002	UI	Home Page	Verify the UI elements in home page	internet,browser	1.Enter URL and click go 2.Click on Click To Start butoon 3.Verify page redirect to image.html page	http://127.0.0.1:5000/	Application should show below UI elements: a.Click To Start b.Home c.Classify	Working as expected	Pass	Steps are clear to follow	nil	ARAVINDHAN
ImagePage_TC_003	Functional	Image page	Verify user is able to upload and redirect to image_prediction page	internet,browser,image	1.Click on Choose button for uploading image 2.Click on classify button to classify image	http://127.0.0.1:5000/image	image uploading page should be displayed and uploading of image should be possible.	Working as expected	Pass	Steps are clear to follow	nil	DINESHKUMAR
ImagePage_TC_004	UI	Image page	Verify the UI elements in image page	internet,browser	1.Click on Choose button for uploading image 2.Click on classify button to classify image	http://127.0.0.1:5000/image	Application should show below UI elements: a.Choose... b.Home c.Classify	Working as expected	Pass	Steps are clear to follow	nil	JABITHA
ImagePrediction Page_TC_005	Functional	Image Prediction page	Verify user is able to see image_prediction page and see the result	internet,browser	Output should be displayed	http://127.0.0.1:5000/predict	Application should show classified image nutrient content and it should be	Working as expected	Pass	Steps are clear to follow	nil	AKASH

Shopenzer Testcases | Testsceamios

8.2 User Acceptance Testing:

Purpose of Document

The purpose of this document is to briefly explain the test coverage and open issuesof the [ProductName] project at the time of the release to User Acceptance Testing (UAT).

Defect Analysis

This reportshows the numberof resolved or closed bugs at each severity level, and how they were resolved

Resolution	Severity 1	Severity 2	Severity 3	Severity 4	Subtotal
By Design	0	0	0	2	2
Duplicate	0	0	0	0	0
External	0	0	3	0	3
Fixed	0	0	0	0	0
Not Reproduced	0	0	1	0	1
Skipped	0	0	0	2	2
Won't Fix	0	0	0	1	1
Totals	0	0	4	5	9

Test Case Analysis

This report shows the numberof test cases that have passed, failed,and untested

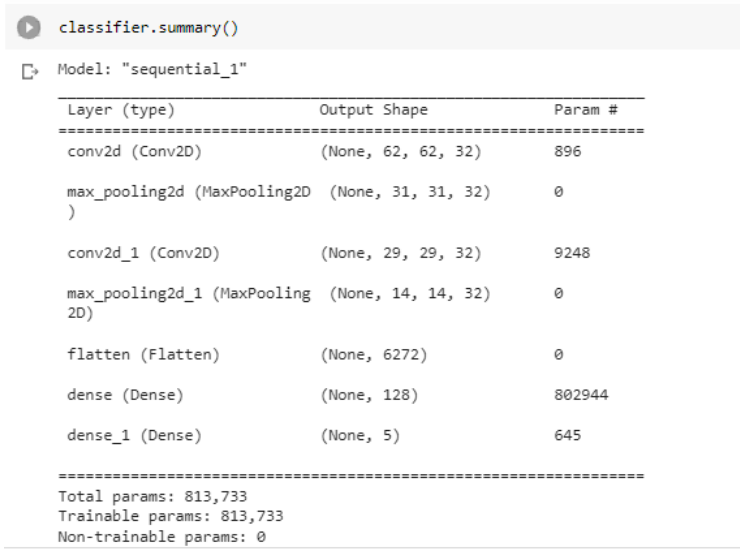
Section	Total Cases	Not Tested	Fail	Pass
Home page	2	0	0	2
Image page	2	0	0	2
Image Prediction page	1	0	0	1

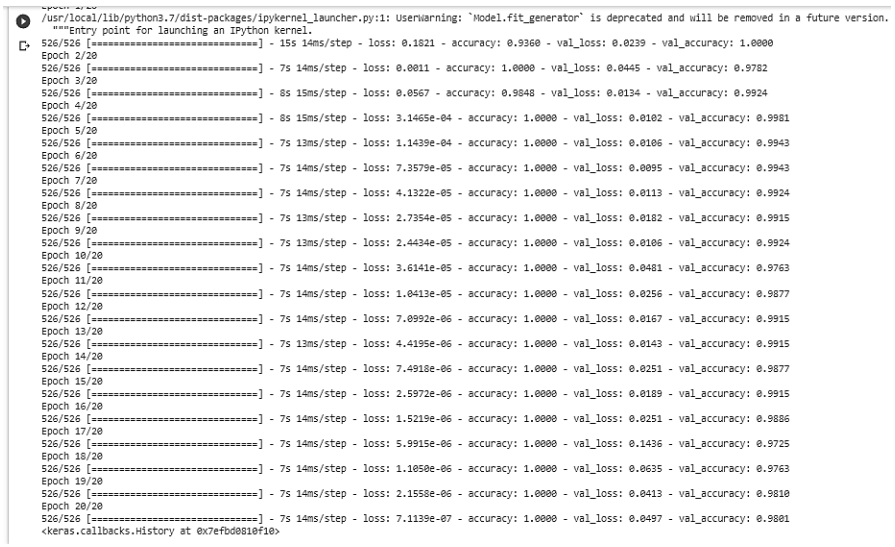
9. RESULTS

9.1 Performance Metrics:

Model Performance Testing:

S.No.	Parameter	Values	Screenshot

1.	Model Summary	`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())`
		`classifier.add(Dense(units=128, activation='relu'))`
		`classifier.add(Dense(units=5, activation='softmax'))`
			

1	Accuracy	Training Accuracy – 1.0000 Validation Accuracy - 0.9801	

10. ADVANTAGES & DISADVANTAGES

ADVANTAGES:

- Easy way to find nutrient content about the food
- Provides user friendly interface
- No login required so user no need to worry about data collection
- It saves time

DISADVANTAGES:

- No diet plans provided
- Sometimes the model may predict wrong output

11. CONCLUSION

The main aim of the project is to building a model which is used for classifying the fruit depends on the different characteristics like color, shape,texture etc. Here the user can capture the images of different fruits and the image will be sent to the trained model.The model analyses the image and detect nutrition based on the fruits like(Sugar, Fiber, Protein, Calories, etc...). We had successfully created a nutrition analyzer website which can predict the nutrient content of food, However currently it can predict the nutrient content of food to some extent.

12. FUTURE SCOPE:

- New features like providing a diet plan could be added.
- Shopping feature for buying dietary supplements could be added

13. APPENDIX

[1] C. L. Ogden, M. D. Carroll, B. K. Kit, and K. M. Flegal, "Prevalence of childhood and adult obesity in the United States, 2011-2012," Jama, vol. 311, pp. 806-814, 2014.
[2] (2016). World Health Organization: fact sheets on obesity and overweight, Available at http://www.who.int/mediacentre/factsheets/fs311/ en/.
[3] G. H. Beaton, J. Milner, P. Corey, V. McGuire, M. Cousins, E. Stewart, et al., "Sources of variance in 24-hour dietary recall data: implications for nutrition study design and interpretation," American Journal of Clinical Nutrition, vol. 32, 1979.
[4] J. Cade, R. Thompson, V. Burley, and D. Warm, "Development, validation and utilisation of food-frequency questionnaires–a review," Public health nutrition, vol. 5, pp. 567-587, 2002.
[5] R. Steele, "An overview of the state of the art of automated capture of dietary intake information," Critical Reviews in Food Science and Nutrition, 2013.
[6] Y. Matsuda and K. Yanai, "Multiple-food recognition considering co-occurrence employing manifold ranking," in Pattern Recognition (ICPR)
[7] S. Yang, M. Chen, D. Pomerleau, and R. Sukthankar, "Food recognition using statistics of pairwise local features," in Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on, 2010, pp. 2249-2256.
[8] F. Zhu, M. Bosch, I. Woo, S. Kim, C. J. Boushey, D. S. Ebert, et al., "The use of mobile devices in aiding dietary assessment and evaluation," Selected Topics in Signal Processing, IEEE Journal of, vol. 4, pp. 756-766, 2010.

GitHub & Project Demo Link:

GitHub - https://github.com/IBM-EPBL/IBM-Project-1197-1658378181
Project Demo Link -
https://drive.google.com/drive/folders/1u3unbet38PFAWrnlQuzVXLtIdrovcPVU?usp=share_link