AI-POWERED NUTRITION ANALYZER FOR FITNESS ENTHUSIASTS

IBM NALAIYATHIRAN

PROJECT REPORT

Submitted

by

TEAM ID : PNT2022TMID42999

TEAM LEADER: M. SABIDHA

TEAM MEMBER: S. AMBAYEERAM

TEAM MEMBER: S. SUGANTHI

TEAM MEMBER: T. RANJITHKUMAR

in partial fulfillment for the award of the degree

of

BACHELOR OF ENGINEERING IN COMPUTER SCIENCE AND ENGINEERING

RVS TECHNICAL CAMPUS COIMBATORE - COIMBATORE

ANNA UNIVERSITY: CHENNAI 600 025

NOVEMBER 2022

CONTENTS

1. INTRODUCTION

1.1 Project Overview

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

1.2 Purpose

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

2. LITERATURE SURVEY

2.1 Existing problem

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

2.2 References

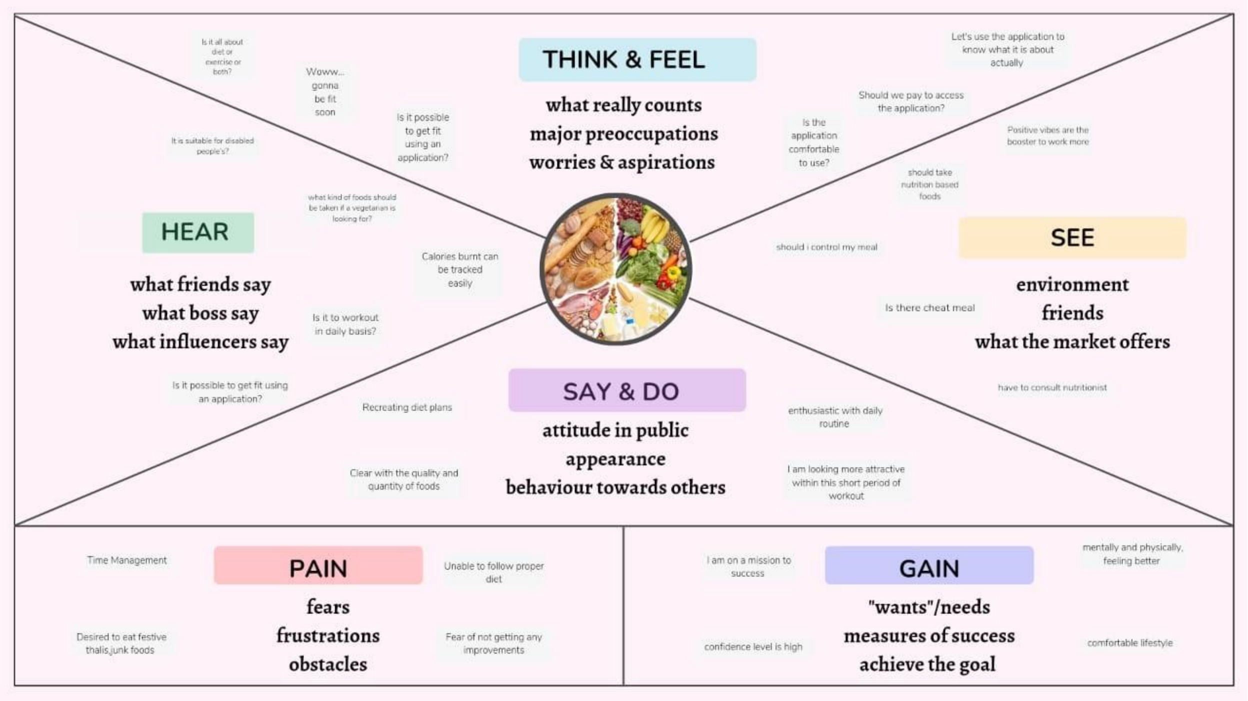
The application relies on Artificial Intelligence to produce custom data related to smart calorie counter powered by AI. Their artificial intelligence learns an individual’s tastes, preferences, and body type. All of this is packaged in a comprehensive nutrition and activity tracker.

2.3 Problem Statement Definition

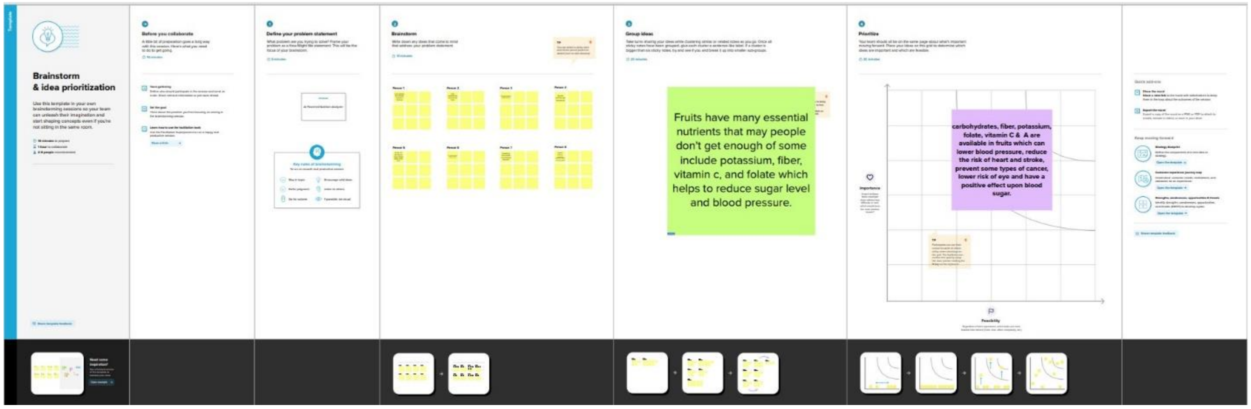
The main aim of the project is to building a model which is used for classifying the fruit depends on the different characteristicslike colour,shape, texture etc. Here the user can capture the images of different fruits and then the image will be sent the trained model. The model analyses the image and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

3. IDEATION & PROPOSED SOLUTION

3.1 Empathy Map Canvas


THINK & FEEL
what really counts
major preoccupations
worries & aspirations
HEAR
what friends say
what boss say
what influencers say
SEE
environment
friends
what the market offers
SAY & DO
attitude in public
appearance
behaviour towards others
PAIN
fears
frustrations
obstacles
GAIN
"wants"/needs
measures of success
achieve the goal
Is it all about diet or exercise or both?
Woww... gonna be fit soon
Is it possible to get fit using an application?
It is suitable for disabled people's?
what kind of foods should be taken if a vegetarian is looking for?
Calories burnt can be tracked easily
Is it to workout in daily basis?
Is it possible to get fit using an application?
Let's use the application to know what it is about actually
Should we pay to access the application?
Is the application comfortable to use?
Positive vibes are the booster to work more
should take nutrition based foods
should i control my meal
Is there cheat meal
have to consult nutritionist
Recreating diet plans
Clear with the quality and quantity of foods
enthusiastic with daily routine
I am looking more attractive within this short period of workout
Time Management
Unable to follow proper diet
Desired to eat festive thalis,junk foods
Fear of not getting any improvements
I am on a mission to success
mentally and physically, feeling better
confidence level is high
comfortable lifestyle

3.2 Ideation & Brainstorming

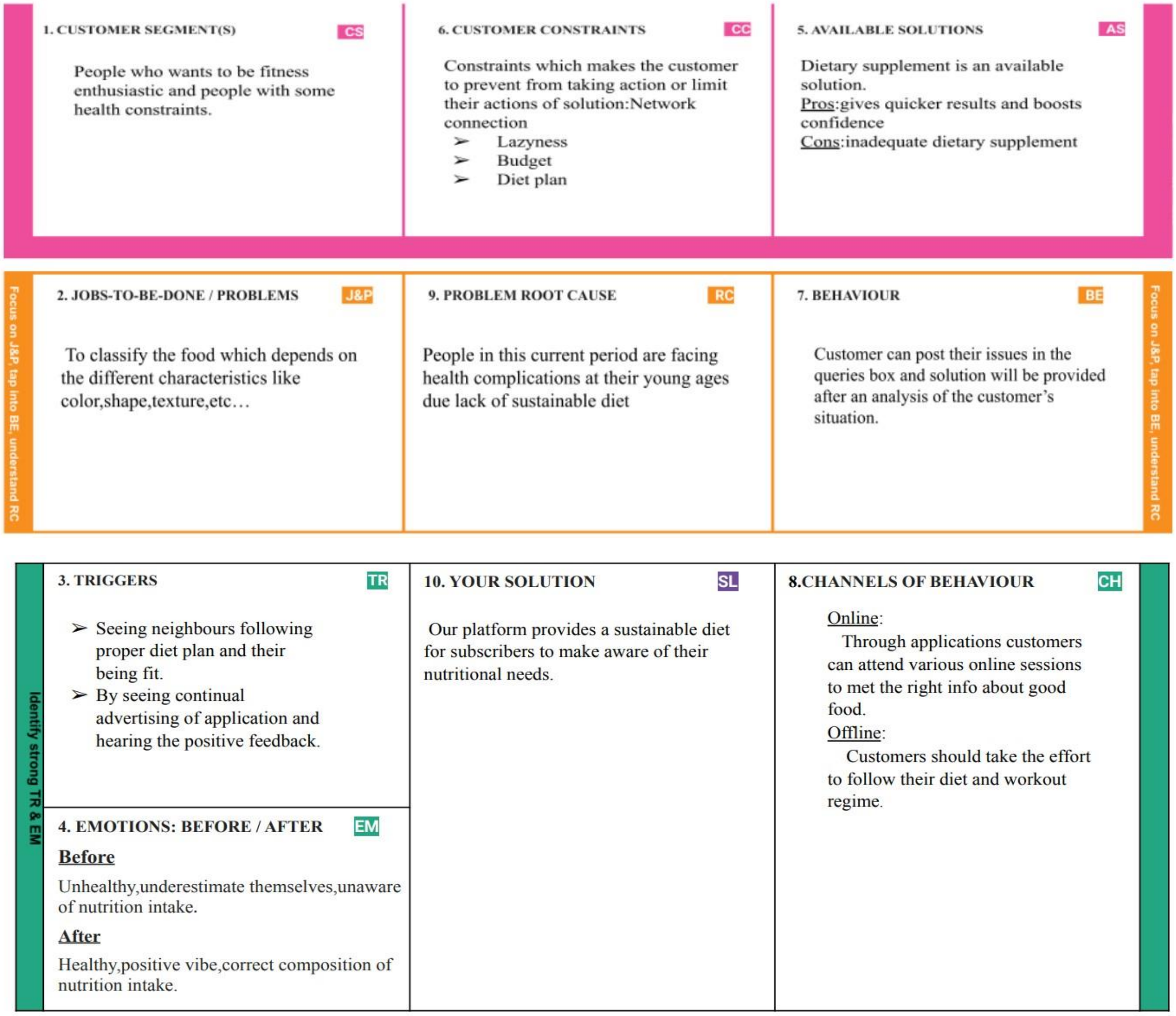

Brainstorm
& idea prioritization
Before you collaborate
Define your problem statement
Brainstorm
Group ideas
Prioritize
Fruits have many essential nutrients that may people don't get enough of some include potassium, fiber, vitamin c, and folate which helps to reduce sugar level and blood pressure.
carbohydrates, fiber, potassium, folate, vitamin C & A are available in fruits which can lower blood pressure, reduce the risk of heart and stroke, prevent some types of cancer, lower risk of eye and have a positive effect upon blood sugar.
Importance
Feasibility

3.3 Proposed Solution

S.No.	Parameter	Description
1.	Problem Statement (Problem to be solved)	An enthusiast must use an AI based analyzing software to identify the nutritional content,color,texture,form and other characteristics of food to be taken
2.	Idea / Solution description	The main aim of the project is to provide proper and clear identification(fitness level),nutritional content(proportion of vitamins,minerals,iron and fibre etc…) of the given input data(food)
3.	Novelty / Uniqueness	The AI based model using DL algorithm detects object in an image or text and provides an enthusiastic individual fitness/food plan based on nutritional requirements
4.	Social Impact / Customer Satisfaction	➢ Feature of providing a calorie measurement makes an individual to recreate or follow the schedule of fitness and diet. ➢ Makes the customer to track their improvement activity
5.	Business Model (Revenue Model)	➢ It is a free platform for the users to make use of the application to know about the calorie and fitness level. ➢ Payment is required for special features and guidance.

6.	Scalability of the Solution	➢ It is an user friendly application. ➢ It provides guidelines on meal planning. ➢ Establishment of healthy lifestyle.

3.4 Problem Solution fit



1. CUSTOMER SEGMENT(S) CS

People who wants to be fitness enthusiastic and people with some health constraints.

6. CUSTOMER CONSTRAINTS CC

Constraints which makes the customer to prevent from taking action or limit their actions of solution:Network connection

➢ Lazyness
➢ Budget
➢ Diet plan

5. AVAILABLE SOLUTIONS AS

Dietary supplement is an available solution.
Pros:gives quicker results and boosts confidence
Cons:inadequate dietary supplement

2. JOBS-TO-BE-DONE / PROBLEMS J&P

To classify the food which depends on the different characteristics like color,shape,texture,etc…

9. PROBLEM ROOT CAUSE RC

People in this current period are facing health complications at their young ages due lack of sustainable diet

7. BEHAVIOUR BE

Customer can post their issues in the queries box and solution will be provided after an analysis of the customer's situation.

Focus on J&P, tap into BE, understand RC

3. TRIGGERS TR

➢ Seeing neighbours following proper diet plan and their being fit.
➢ By seeing continual advertising of application and hearing the positive feedback.

4. EMOTIONS: BEFORE / AFTER EM

Before

Unhealthy,underestimate themselves,unaware of nutrition intake.

After

Healthy,positive vibe,correct composition of nutrition intake.

10. YOUR SOLUTION SL

Our platform provides a sustainable diet for subscribers to make aware of their nutritional needs.

8.CHANNELS OF BEHAVIOUR CH

Online:
Through applications customers can attend various online sessions to met the right info about good food.
Offline:
Customers should take the effort to follow their diet and workout regime.

Identify strong TR & EM

4. REQUIREMENT ANALYSIS

4.1 Functional requirement

FR No.	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)

FR-1	User Registration	Registration through Form Registration through Gmail Registration through LinkedIN
FR-2	User Confirmation	Confirmation via Email Confirmation via OTP
FR-3	User Login	Give the login ID Give the password that is created during registration
FR-4	Dataset	Uploading the dataset consists of variety of fruit images
FR-5	Image Input	Analysing the image input provided by the user
FR-6	Process	Testing the image by various convolution layers
FR-7	Process	The nutrient content in the particular fruit is displayed

4.2 Non-Functional requirements

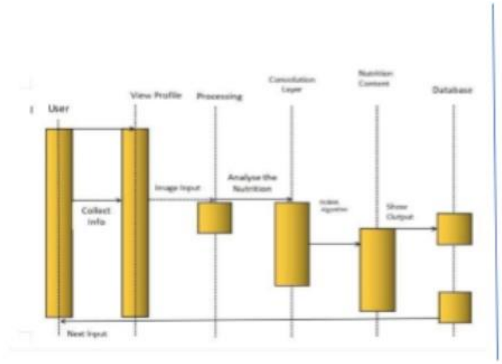
FR No.	Non-Functional Requirement	**Description**
NFR-1	Usability	The ways in which the system will accessible to users with particular or non standard accessibility requirements
NFR-2	Security	Extend to which data needs to be protected and kept confidential.
NFR-3	Capacity	Ability to continue to function well as it changes in according to input that we give.
NFR-4	Performance	Fast response is achieved

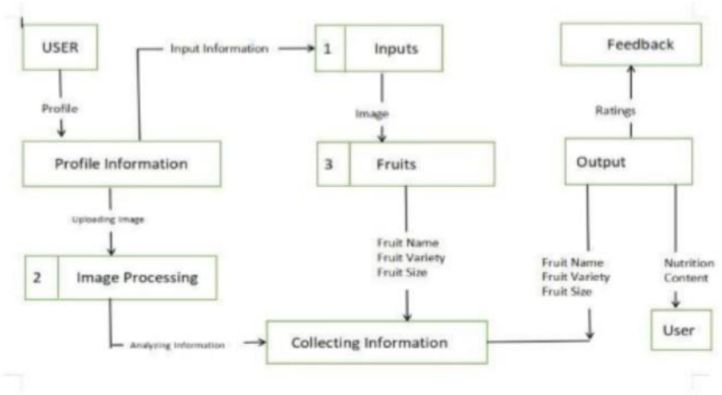
NFR-5	Availability	The minimum proportion of time given in online service hours that the service should be available.
NFR-6	Scalability	Ability of the application to handle an increase in workload without performance degradation, or its ability to quickly enlarge.

5.PROJECT DESIGN

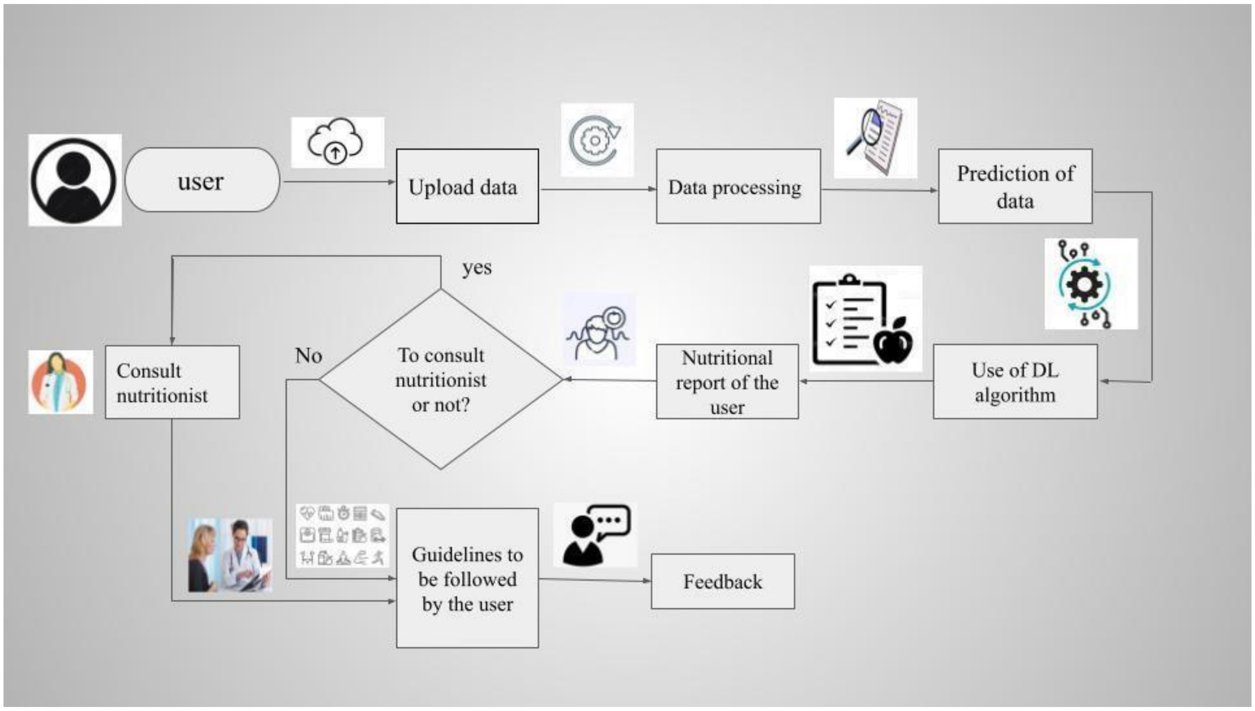
5.1 Data Flow Diagrams

Simplified Diagram





5.2 Solution & Technical Architecture



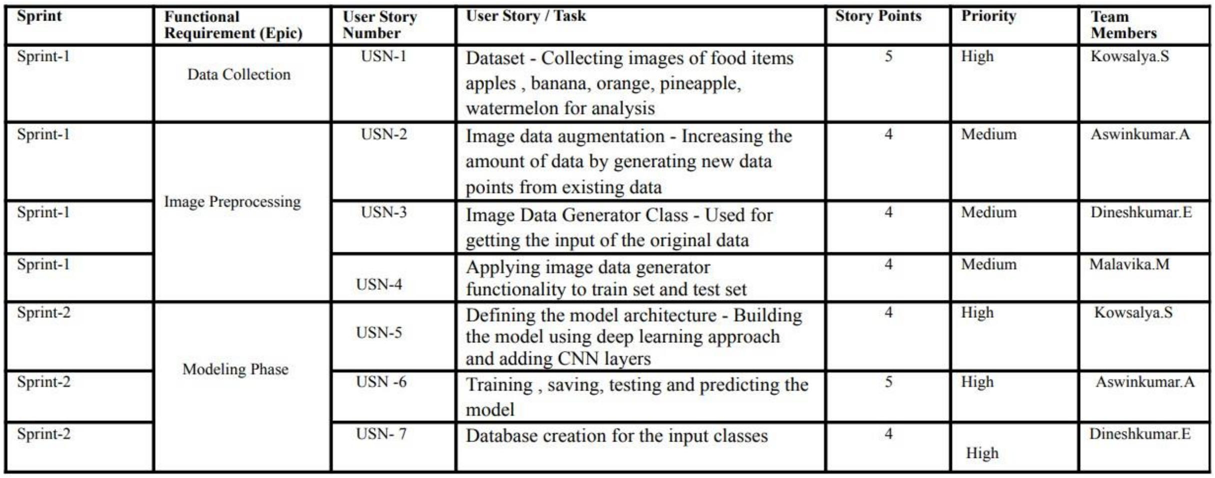
5.3 User Stories

User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority	Release
Customer (Mobile user)	Registration	USN-1	As a user, I can do registration by my mobile and get confirmation email	I can access my account / dashboard	High	Sprint-1
Customer (Web user)	Registration	USN-2	As a user, I can register for the application by entering my email, password, and confirming my password	I can access my account / dashboard	High	Sprint-1
		USN-3	As a user, I will receive confirmation email once I have registered for the application	I can receive confirmation email & click confirm	High	Sprint-1
		USN-4	As a user, I can register for the application through other social media	I can register & access the dashboard with Login	Low	Sprint-2
	Login	USN-5	As a user, I can log into the application by entering email & password	I can successfully logged in to the home page	High	Sprint-1
	Dashboard	USN-6	As a user, I can select the list of options provided in the dashboard	I can access the options according to my need	Medium	Sprint-1
	Search	USN-7	As a user, I can search for different variety of fruits	I can get the nutrition content of different fruits	High	Sprint-2
	View	USN-8	As a user, I can view the list of fruits	I will get the information such as calories,vitamins etc	High	Sprint-2

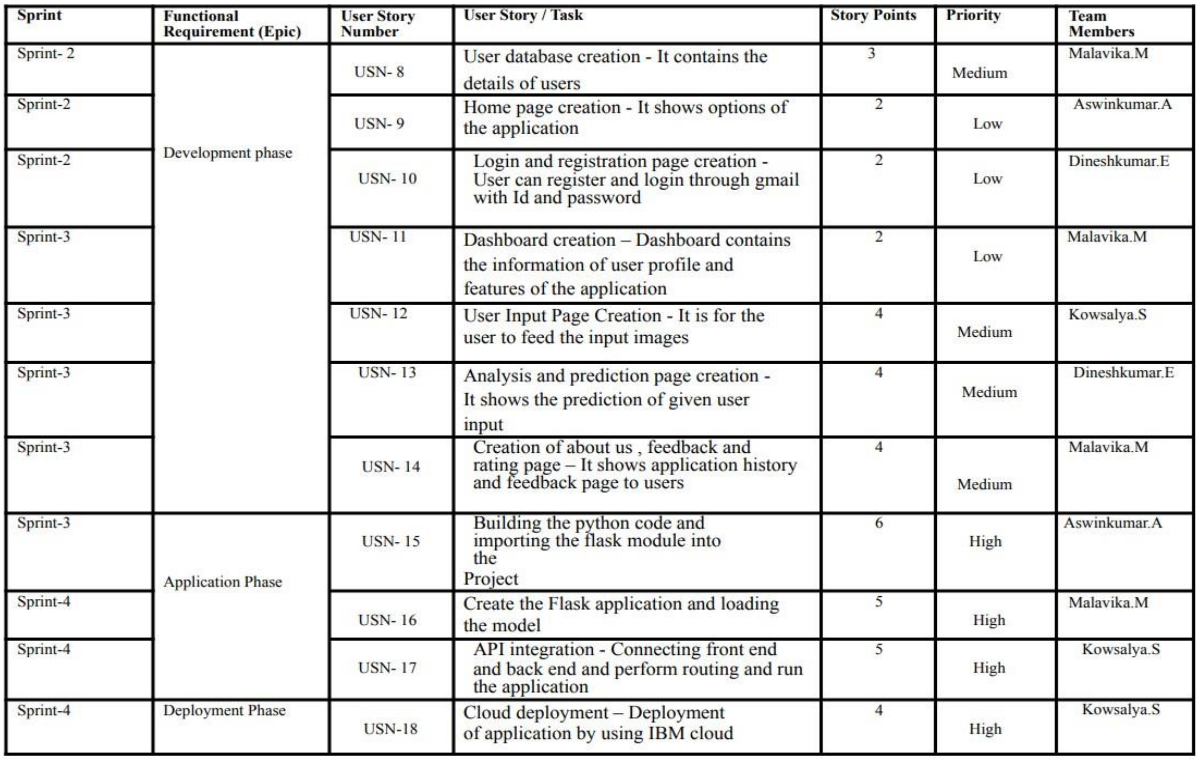
	Notifications	USN-9	As a user, I will receive notification about variety and textures of different fruits	I will get the frequent updates of different fruits	Low	Sprint-2
Customer Care Executive	Mediator	USN-10	As a customer care executive ,they could take care of customer feedbacks and solve user requirements	Users can get help and support from customer care executives	Medium	Sprint-2
Administrator	Database	USN-11	As a admin,I will store the user database confidentially	I can store and access data if it is needed in future	High	Sprint-1
	Data Information	USN-12	As a admin, I will include the dataset for performing various processes	I can store dataset and analyse it	High	Sprint-2
	Processing	USN-13	As a admin, I will use various convolution layers for image analysis	I can process using various convolution layers	High	Sprint-2
	Nutrition Analyzer	USN-14	As a admin, I will predict the fruit that has send as input	I can get the nutrition content of particular food after processing and display it	High	Sprint-2

6.PROJECT PLANNING & SCHEDULING

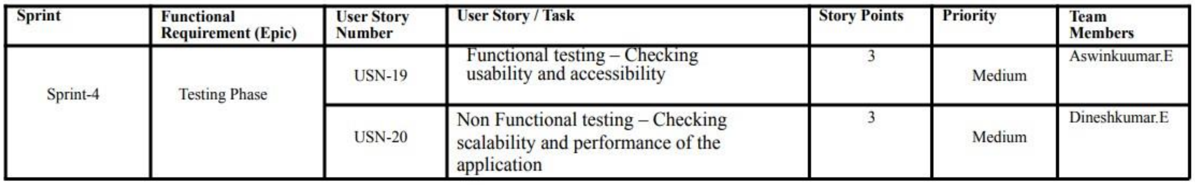
6.1 Sprint Planning & Estimation



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 food items apples , banana, orange, pineapple, watermelon for analysis	5	High	Kowsalya.S
Sprint-1	Image Preprocessing	USN-2	Image data augmentation - Increasing the amount of data by generating new data points from existing data	4	Medium	Aswinkumar.A
Sprint-1		USN-3	Image Data Generator Class - Used for getting the input of the original data	4	Medium	Dineshkumar.E
Sprint-1		USN-4	Applying image data generator functionality to train set and test set	4	Medium	Malavika.M
Sprint-2	Modeling Phase	USN-5	Defining the model architecture - Building the model using deep learning approach and adding CNN layers	4	High	Kowsalya.S
Sprint-2		USN -6	Training , saving, testing and predicting the model	5	High	Aswinkumar.A
Sprint-2		USN- 7	Database creation for the input classes	4	High	Dineshkumar.E

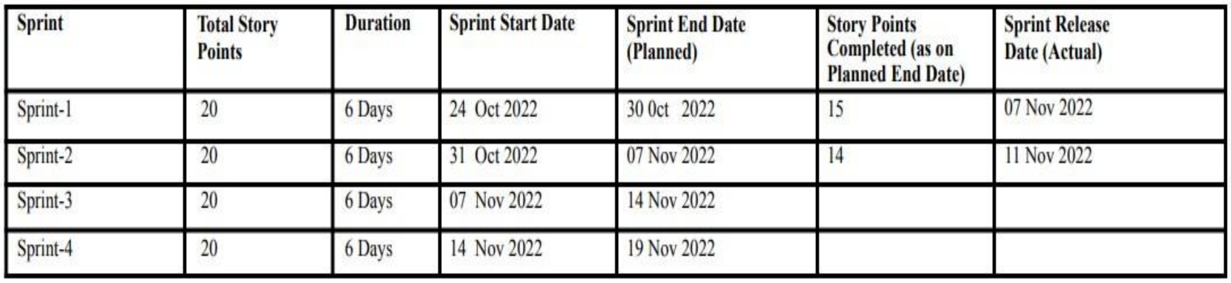


Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint- 2	Development phase	USN- 8	User database creation - It contains the details of users	3	Medium	Malavika.M
Sprint-2		USN- 9	Home page creation - It shows options of the application	2	Low	Aswinkumar.A
Sprint-2		USN- 10	Login and registration page creation - User can register and login through gmail with Id and password	2	Low	Dineshkumar.E
Sprint-3		USN- 11	Dashboard creation – Dashboard contains the information of user profile and features of the application	2	Low	Malavika.M
Sprint-3		USN- 12	User Input Page Creation - It is for the user to feed the input images	4	Medium	Kowsalya.S
Sprint-3		USN- 13	Analysis and prediction page creation - It shows the prediction of given user input	4	Medium	Dineshkumar.E
Sprint-3		USN- 14	Creation of about us , feedback and rating page – It shows application history and feedback page to users	4	Medium	Malavika.M
Sprint-3	Application Phase	USN- 15	Building the python code and importing the flask module into the Project	6	High	Aswinkumar.A
Sprint-4		USN- 16	Create the Flask application and loading the model	5	High	Malavika.M
Sprint-4		USN- 17	API integration - Connecting front end and back end and perform routing and run the application	5	High	Kowsalya.S
Sprint-4	Deployment Phase	USN-18	Cloud deployment – Deployment of application by using IBM cloud	4	High	Kowsalya.S



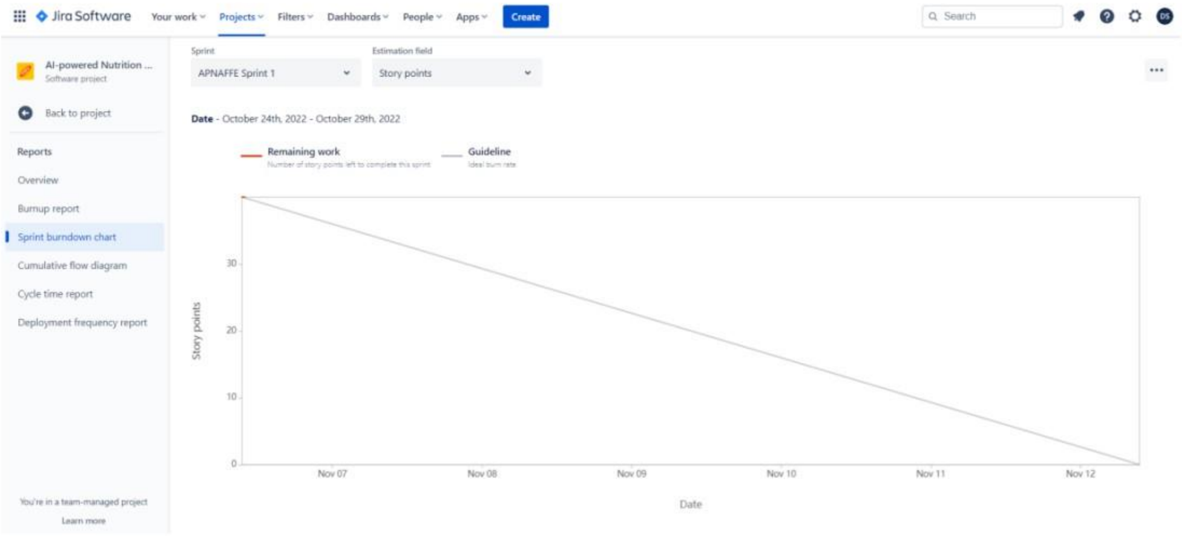
Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-4	Testing Phase	USN-19	Functional testing – Checking usability and accessibility	3	Medium	Aswinkuumar.E
		USN-20	Non Functional testing – Checking scalability and performance of the application	3	Medium	Dineshkumar.E

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	30 Oct 2022	15	07 Nov 2022
Sprint-2	20	6 Days	31 Oct 2022	07 Nov 2022	14	11 Nov 2022
Sprint-3	20	6 Days	07 Nov 2022	14 Nov 2022		
Sprint-4	20	6 Days	14 Nov 2022	19 Nov 2022		

6.3 Reports from JIRA



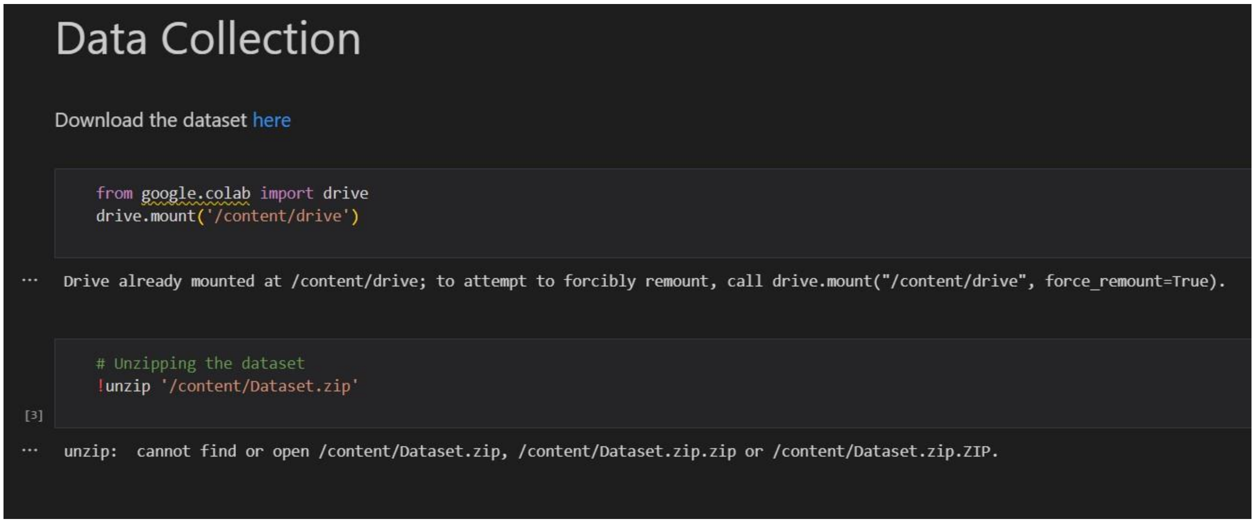
Sprint: APNAFFE Sprint 1

Estimation field: Story points

Date - October 24th, 2022 - October 29th, 2022

7. CODING & SOLUTIONING

7.1 Feature 1



Data Collection

Download the dataset here

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

```
Drive already mounted at /content/drive; to attempt to forcibly remount, call drive.mount("/content/drive", force_remount=True).
```

```
# Unzipping the dataset
!unzip '/content/Dataset.zip'
```

```
unzip:  cannot find or open /content/Dataset.zip, /content/Dataset.zip.zip or /content/Dataset.zip.ZIP.
```

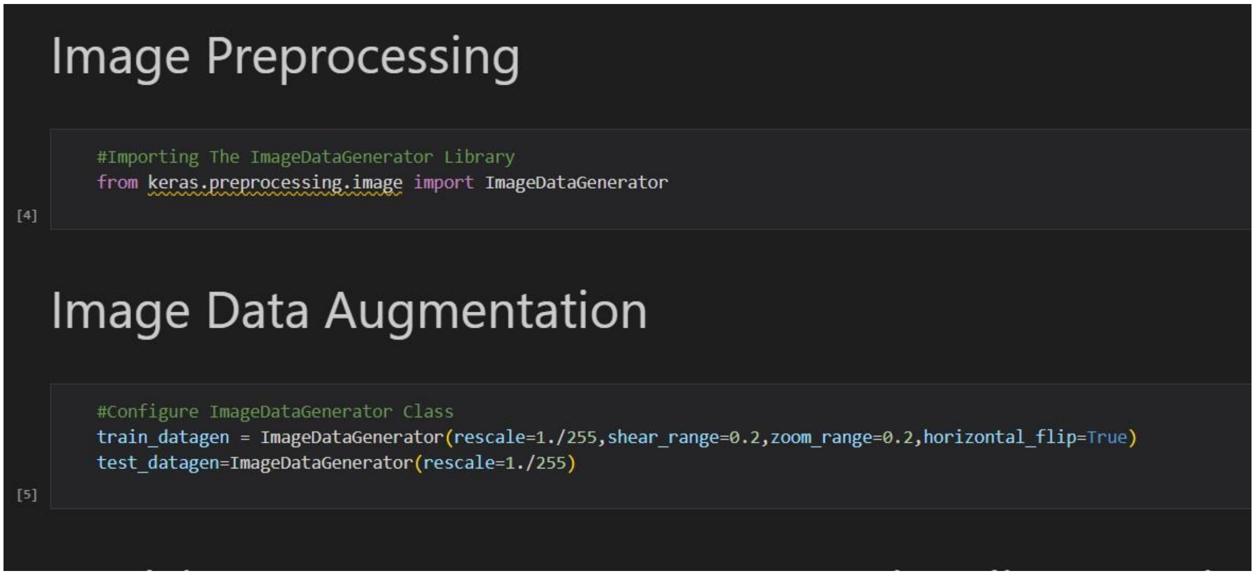
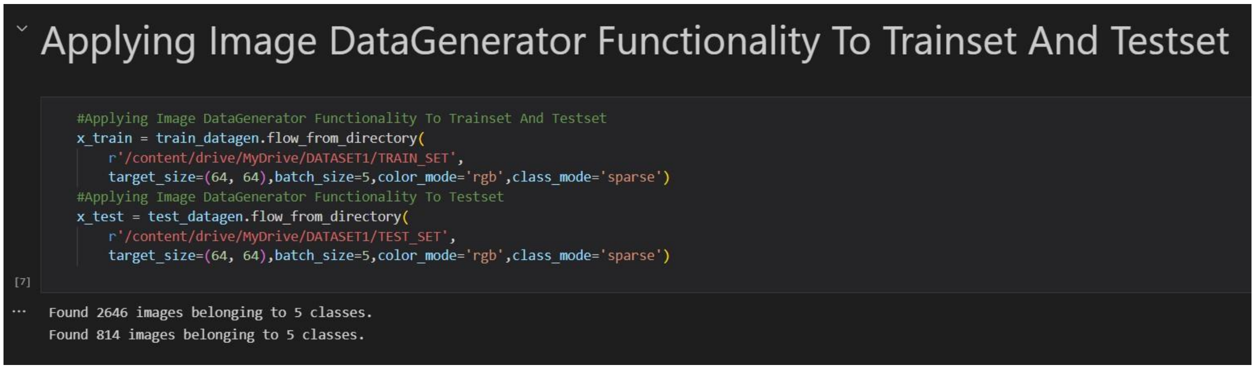


Image Preprocessing

```
#Importing The ImageDataGenerator Library
from keras.preprocessing.image import ImageDataGenerator
```

Image Data Augmentation

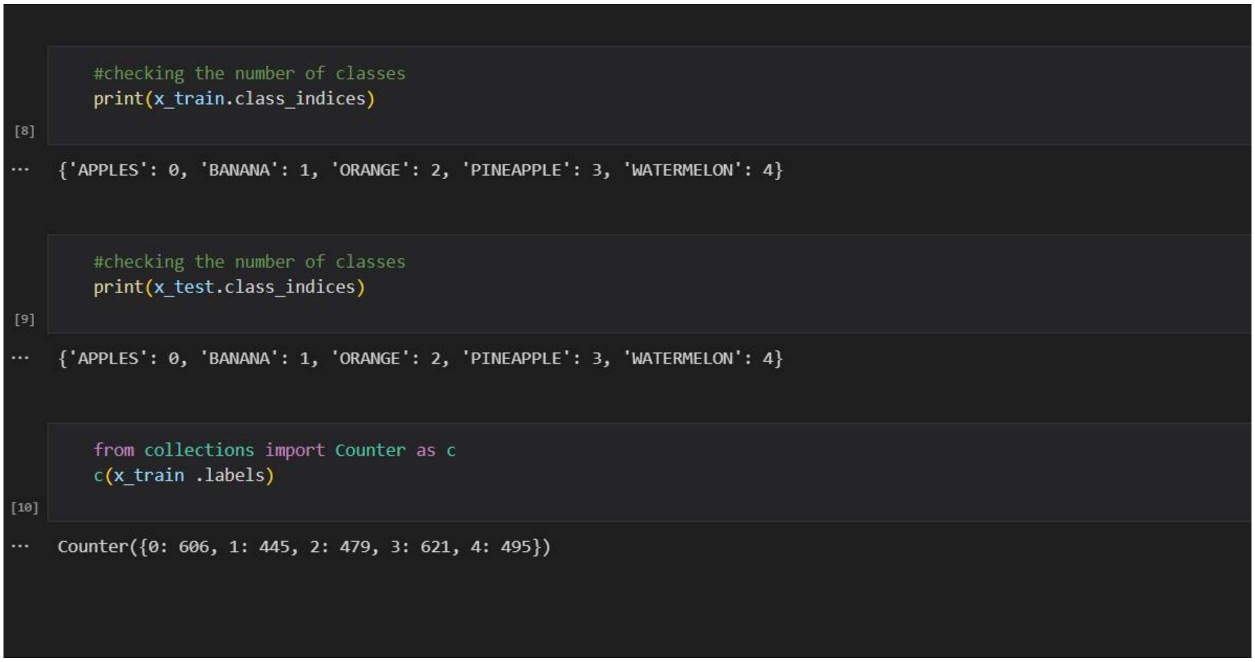
```
#Configure ImageDataGenerator Class
train_datagen = ImageDataGenerator(rescale=1./255,shear_range=0.2,zoom_range=0.2,horizontal_flip=True)
test_datagen=ImageDataGenerator(rescale=1./255)
```



Applying Image DataGenerator Functionality To Trainset And Testset

```
#Applying Image DataGenerator Functionality To Trainset And Testset
x_train = train_datagen.flow_from_directory(
    r'/content/drive/MyDrive/DATASET1/TRAIN_SET',
    target_size=(64, 64),batch_size=5,color_mode='rgb',class_mode='sparse')
#Applying Image DataGenerator Functionality To Testset
x_test = test_datagen.flow_from_directory(
    r'/content/drive/MyDrive/DATASET1/TEST_SET',
    target_size=(64, 64),batch_size=5,color_mode='rgb',class_mode='sparse')
```

```
Found 2646 images belonging to 5 classes.
Found 814 images belonging to 5 classes.
```



```
#checking the number of classes
print(x_train.class_indices)
```

[8]

```
{'APPLES': 0, 'BANANA': 1, 'ORANGE': 2, 'PINEAPPLE': 3, 'WATERMELON': 4}
```

```
#checking the number of classes
print(x_test.class_indices)
```

[9]

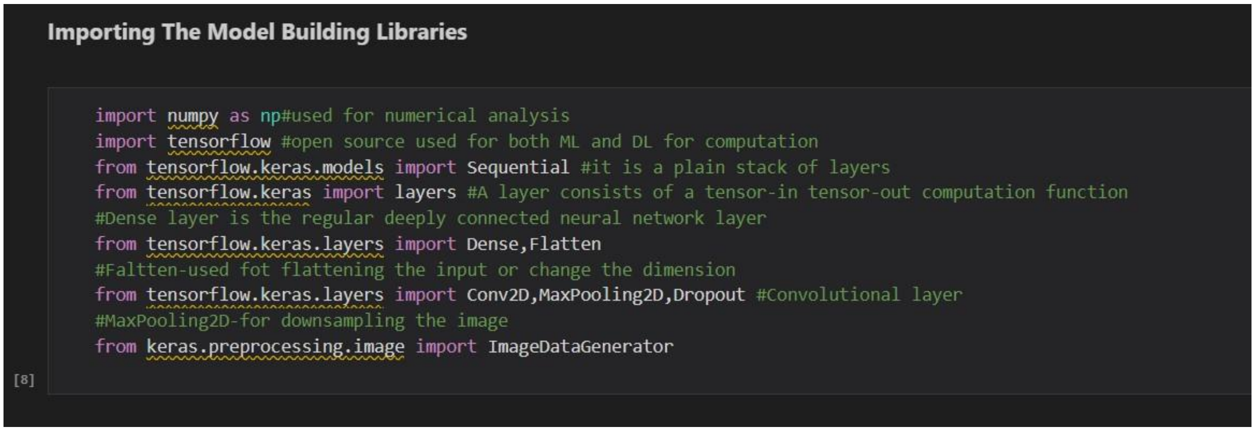
```
{'APPLES': 0, 'BANANA': 1, 'ORANGE': 2, 'PINEAPPLE': 3, 'WATERMELON': 4}
```

```
from collections import Counter as c
c(x_train .labels)
```

[10]

```
Counter({0: 606, 1: 445, 2: 479, 3: 621, 4: 495})
```

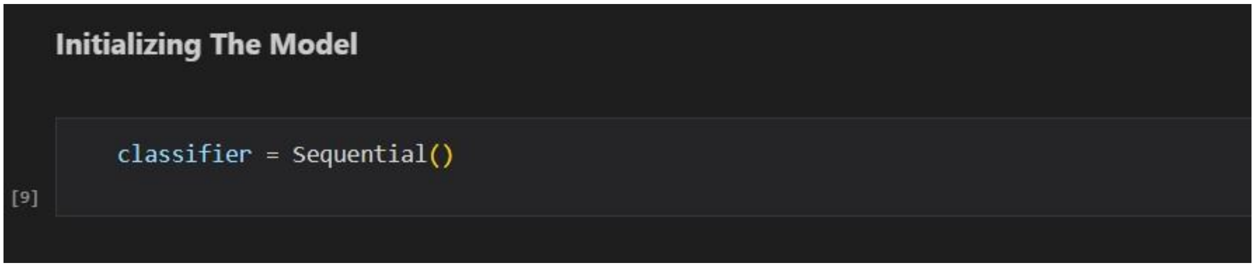
7.2 Feature 2



Importing The Model Building Libraries

```
import numpy as np#used for numerical analysis
import tensorflow #open source used for both ML and DL for computation
from tensorflow.keras.models import Sequential #it is a plain stack of layers
from tensorflow.keras import layers #A layer consists of a tensor-in tensor-out computation function
#Dense layer is the regular deeply connected neural network layer
from tensorflow.keras.layers import Dense,Flatten
#Faltten-used fot flattening the input or change the dimension
from tensorflow.keras.layers import Conv2D,MaxPooling2D,Dropout #Convolutional layer
#MaxPooling2D-for downsampling the image
from keras.preprocessing.image import ImageDataGenerator
```

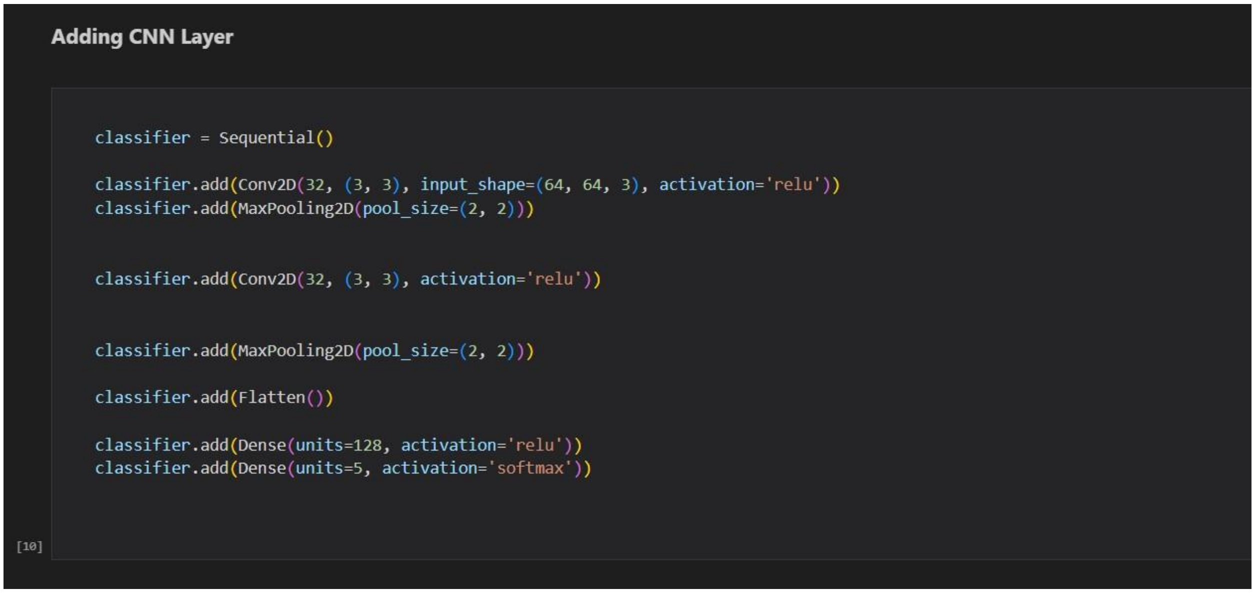
[8]



Initializing The Model

```
classifier = Sequential()
```

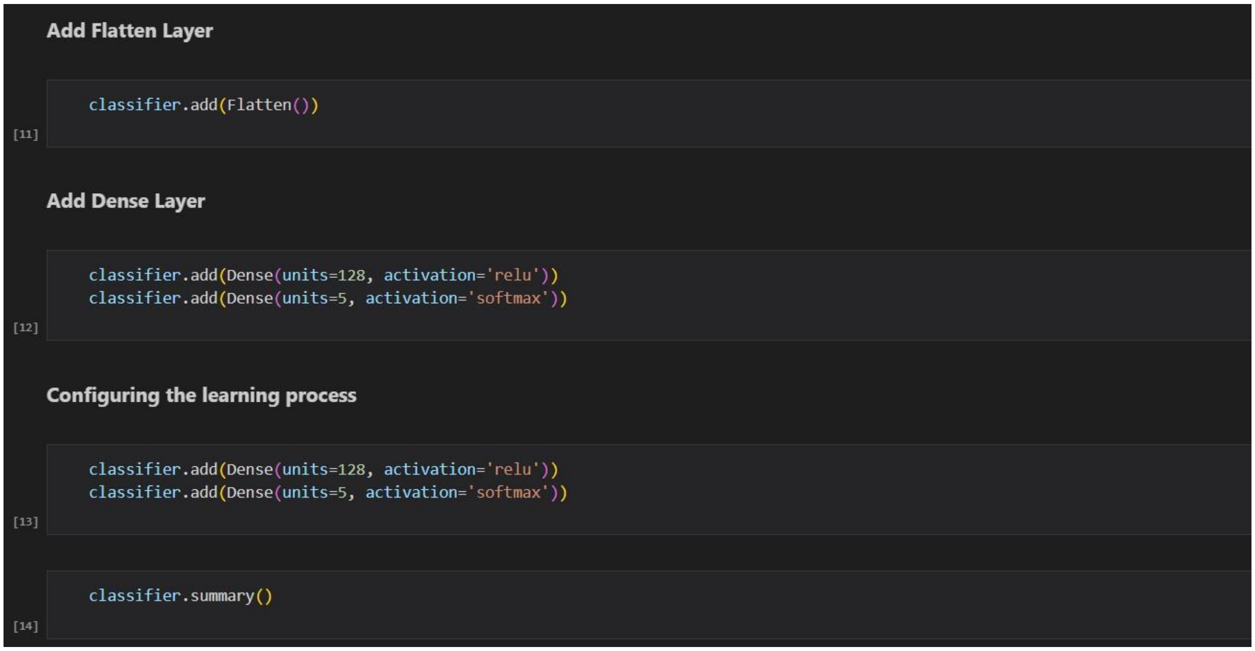
[9]



Adding CNN Layer

```
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'))
```

[10]



Add Flatten Layer

```
classifier.add(Flatten())
```

[11]

Add Dense Layer

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

[12]

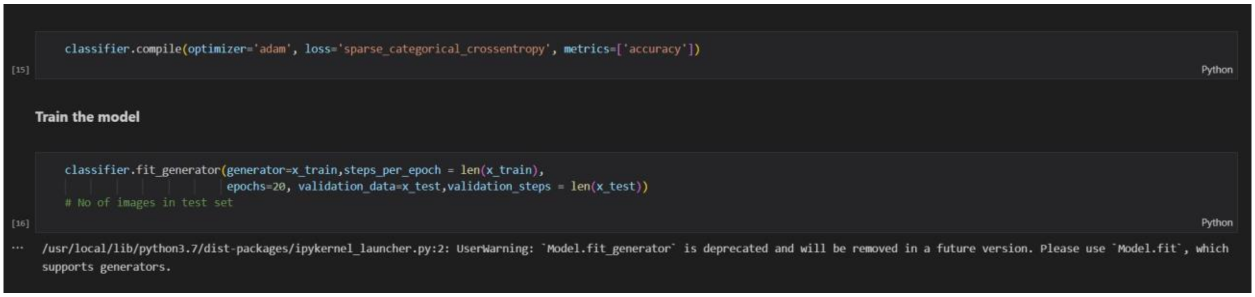
Configuring the learning process

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

[13]

```
classifier.summary()
```

[14]



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

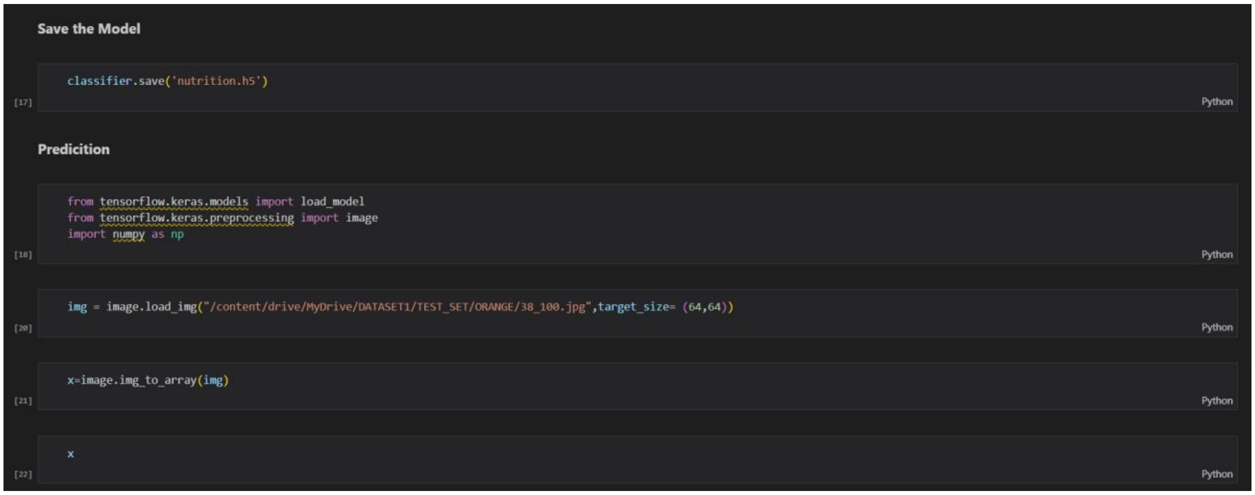
[15]

Train the model

```
classifier.fit_generator(generator=x_train,steps_per_epoch = len(x_train),
                         epochs=20, validation_data=x_test,validation_steps = len(x_test))
# No of images in test set
```

[16]

```
/usr/local/lib/python3.7/dist-packages/ipykernel_launcher.py:2: UserWarning: `Model.fit_generator` is deprecated and will be removed in a future version. Please use `Model.fit`, which
supports generators.
```



Save the Model

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

[17]

Predicition

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

[18]

```
img = image.load_img("/content/drive/MyDrive/DATASET1/TEST_SET/ORANGE/38_100.jpg",target_size= (64,64))
```

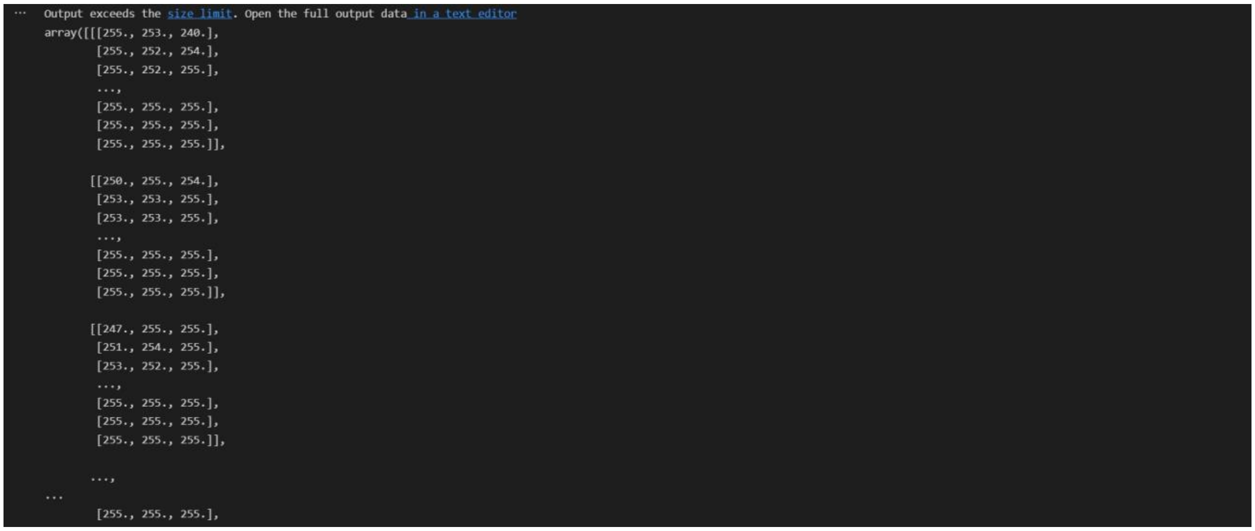
[20]

```
x=image.img_to_array(img)
```

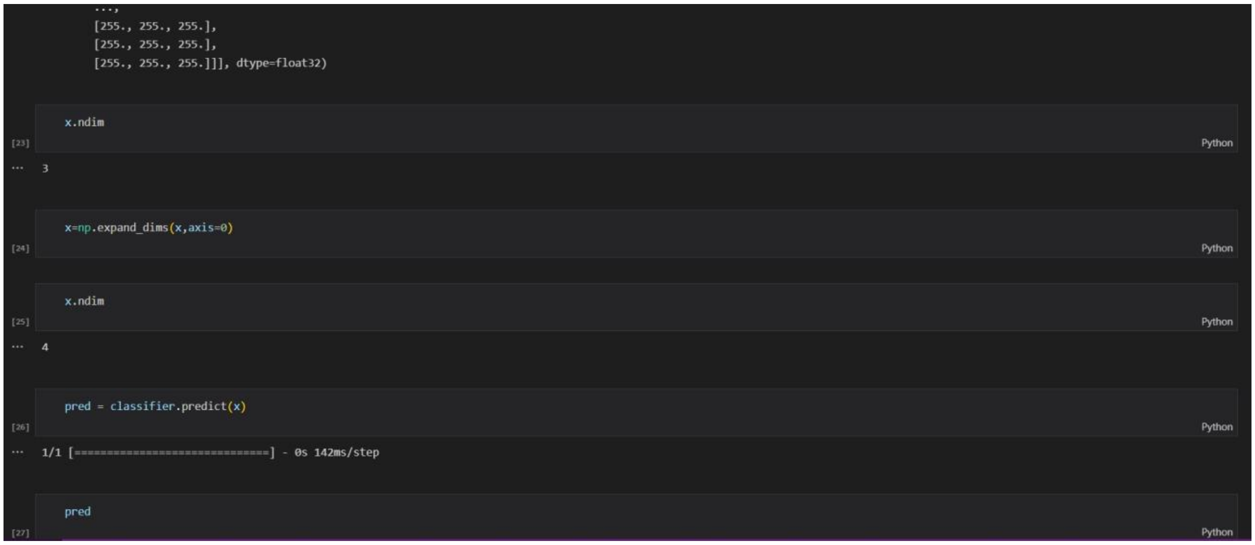
[21]

```
x
```

[22]



```
Output exceeds the size limit. Open the full output data in a text editor
array([[[255., 253., 240.],
        [255., 252., 254.],
        [255., 252., 255.],
        ...,
        [255., 255., 255.],
        [255., 255., 255.],
        [255., 255., 255.]],

       [[250., 255., 254.],
        [253., 253., 255.],
        [253., 253., 255.],
        ...,
        [255., 255., 255.],
        [255., 255., 255.],
        [255., 255., 255.]],

       [[247., 255., 255.],
        [251., 254., 255.],
        [253., 252., 255.],
        ...,
        [255., 255., 255.],
        [255., 255., 255.],
        [255., 255., 255.]],

       ...,
...
        [255., 255., 255.],
```



```
        ...,
        [255., 255., 255.],
        [255., 255., 255.],
        [255., 255., 255.]]], dtype=float32)
```

```
x.ndim
```

[23]

```
3
```

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

[24]

```
x.ndim
```

[25]

```
4
```

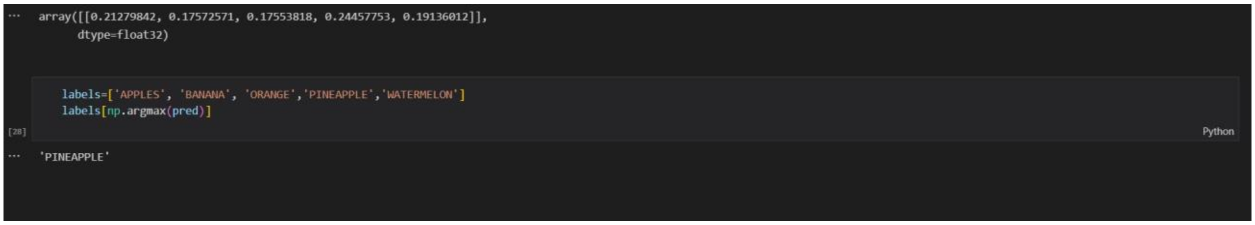
```
pred = classifier.predict(x)
```

[26]

```
1/1 [==============================] - 0s 142ms/step
```

```
pred
```

[27]



```
array([[0.21279842, 0.17572571, 0.17553818, 0.24457753, 0.19136012]],
      dtype=float32)
```

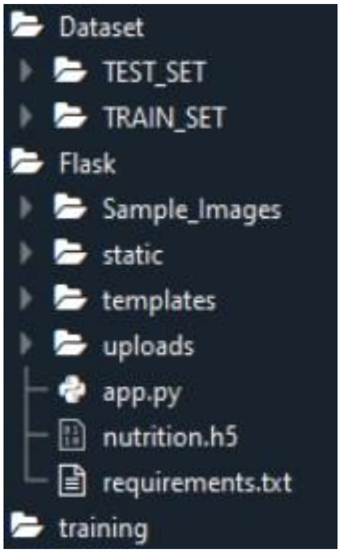
```
labels=['APPLES', 'BANANA', 'ORANGE','PINEAPPLE','WATERMELON']
labels[np.argmax(pred)]
```

[28]

```
'PINEAPPLE'
```

8.TESTING

8.1 Test Cases



```
Dataset
    TEST_SET
    TRAIN_SET
Flask
    Sample_Images
    static
    templates
    uploads
    app.py
    nutrition.h5
    requirements.txt
training
```

- Dataset folder contains the training and testing images for training our model.
- We are building a Flask Application that needs HTML pages stored in the templates folder and a python script app.py for serverside scripting
- we need the model which is saved and the saved model in this content is a nutrition.h5 ➢ templates folder contains home.html, image.html, imageprediction.html pages.
- Statis folder had the css and js files which are necessary for styling the html page and for executing the actions.
- Uploads folder will have the uploaded images(which are already tested).
- Sample_images will have the images which are used to test or upload.
- Training folder contains the trained model file.

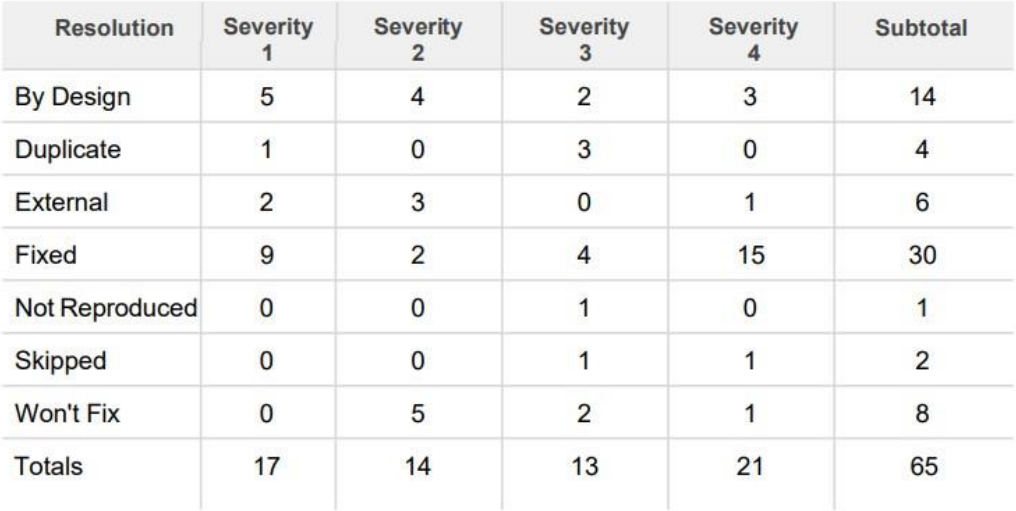
8.2 User Acceptance Testing

1.Purpose of Document

This document serves as a quick reference for the Deep Learning Fundus Image Analysis for Early Detection of project's test coverage and open issues as of the project's release for user acceptance testing

2.Defect Analysis

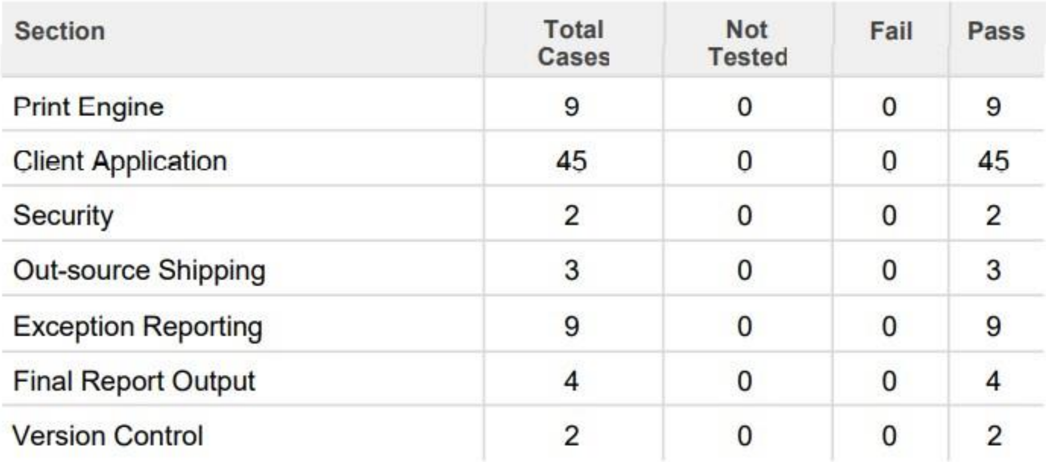
This shows how many bugs were fixed or closed at each severity level and how they were fixed.



Resolution	Severity 1	Severity 2	Severity 3	Severity 4	Subtotal
By Design	5	4	2	3	14
Duplicate	1	0	3	0	4
External	2	3	0	1	6
Fixed	9	2	4	15	30
Not Reproduced	0	0	1	0	1
Skipped	0	0	1	1	2
Won't Fix	0	5	2	1	8
Totals	17	14	13	21	65

3.Test-Case Analysis

This report shows the number of test cases that have passed, failed, and untested.

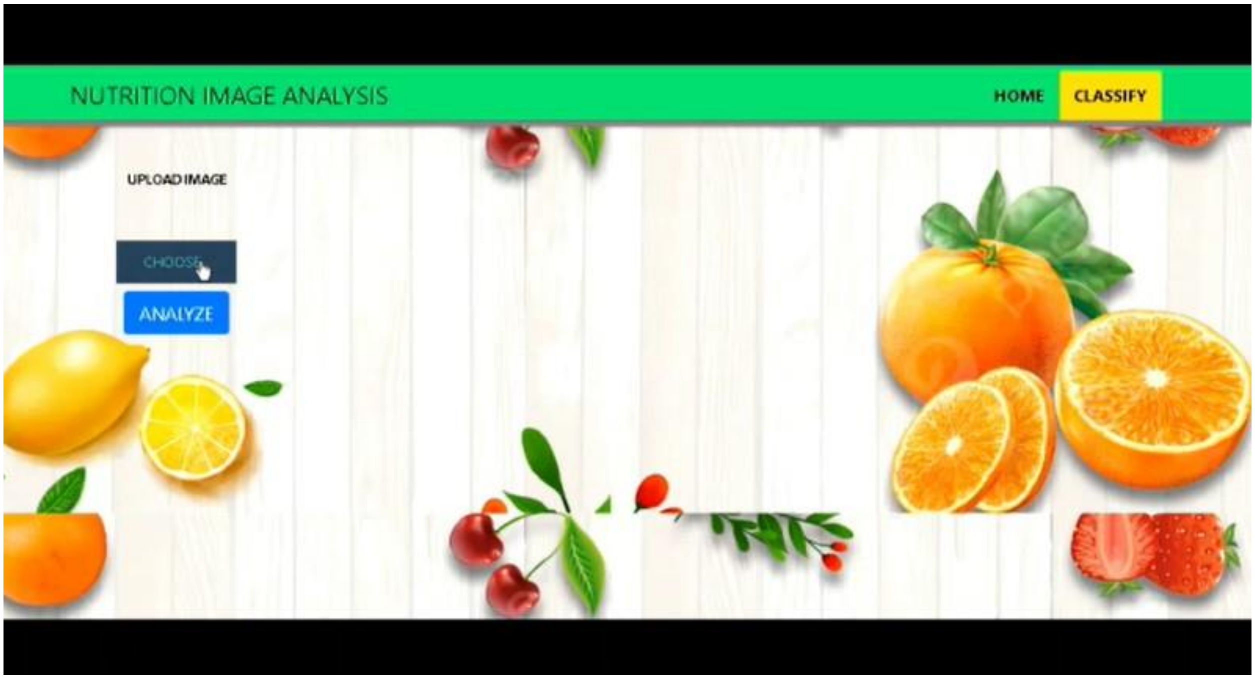


Section	Total Cases	Not Tested	Fail	Pass
Print Engine	9	0	0	9
Client Application	45	0	0	45
Security	2	0	0	2
Out-source Shipping	3	0	0	3
Exception Reporting	9	0	0	9
Final Report Output	4	0	0	4
Version Control	2	0	0	2

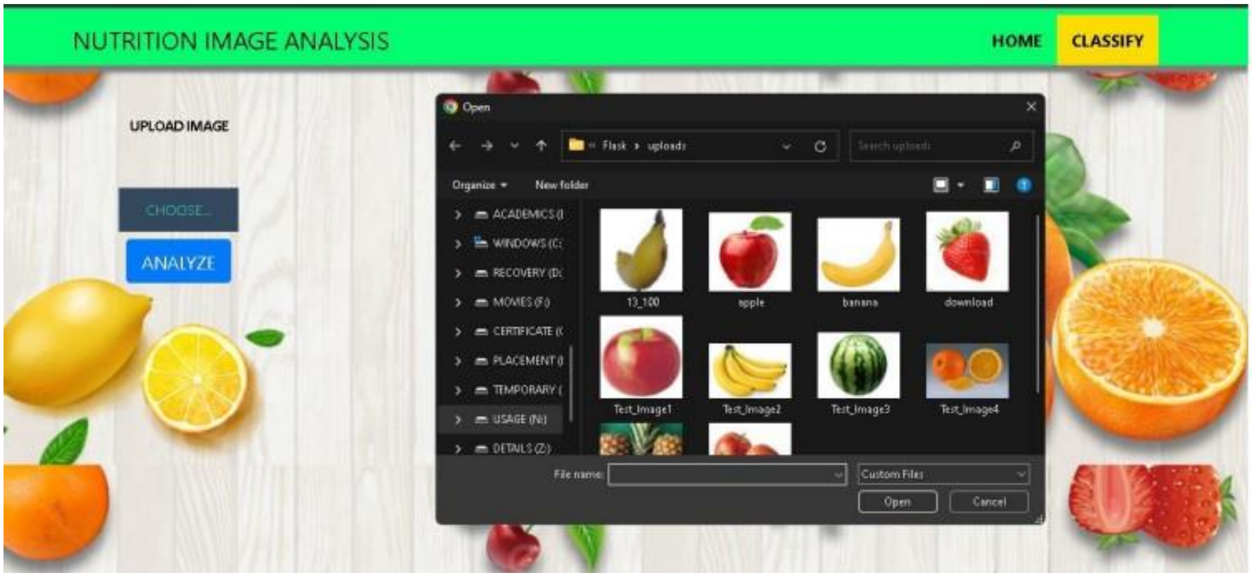
9. RESULTS

9.1Performance Metrics

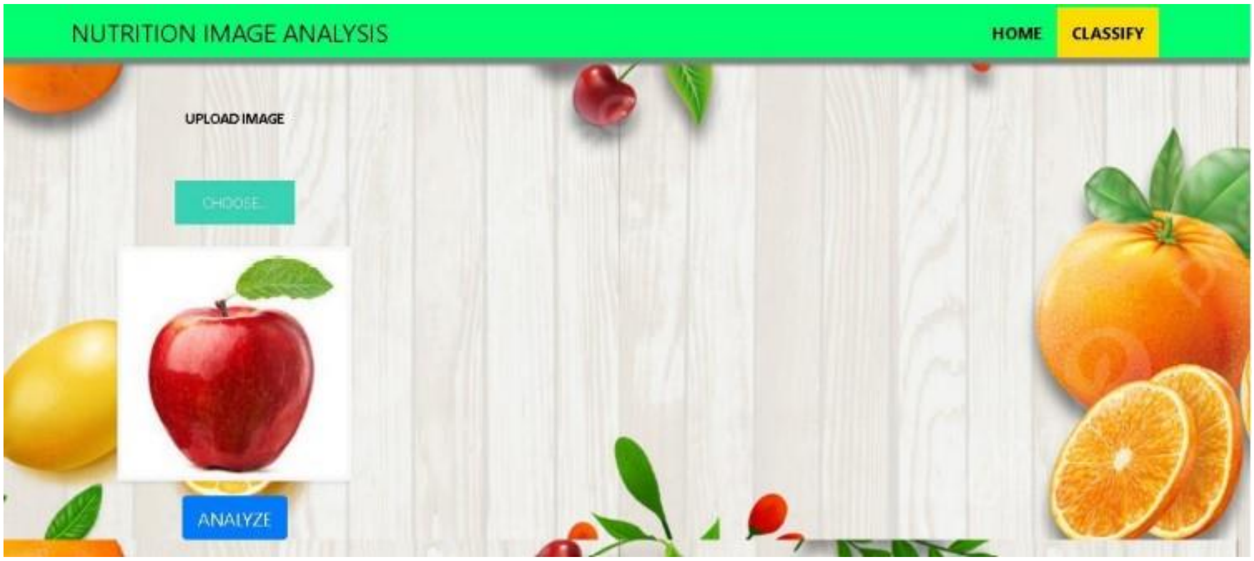
STEP 1:CHOOSE FILE



STEP 2:HOME



STEP 3:IMAGE



STEP 4:IMAGE PREDICTION

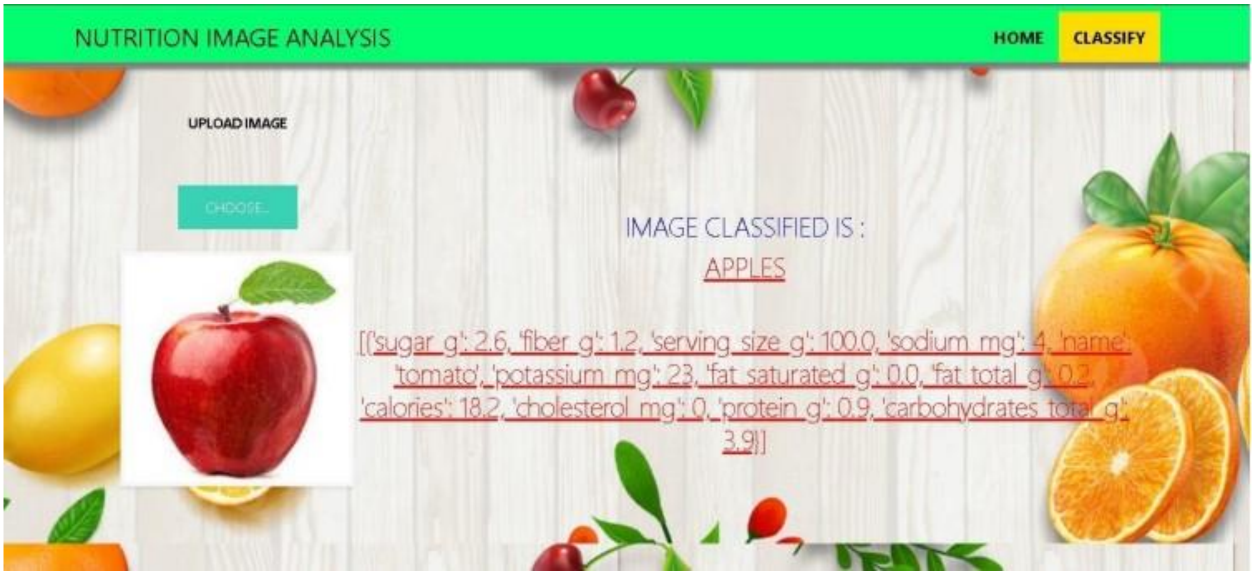


IMAGE CLASSIFIED IS :

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

10.1 ADVANTAGES

- Conducting the nutritional analysis helps to know if any product causes allergens or affects the health of the consumers.
- Conducting Nutritional Analysis allows to know the high levels of micronutrients that reduce the nutritional value of the product.

- ➢ Provides overall insights into the products.

10.2 DISADVANTAGES

- ➢ It can be difficult to identify which nutrients are positive depending on the type of model used.
- ➢ The software application seems to be paid at certain cause.
- ➢ People with lack of internet connection face difficulties in using the application.

11.CONCLUSION

By the end of this project you will:

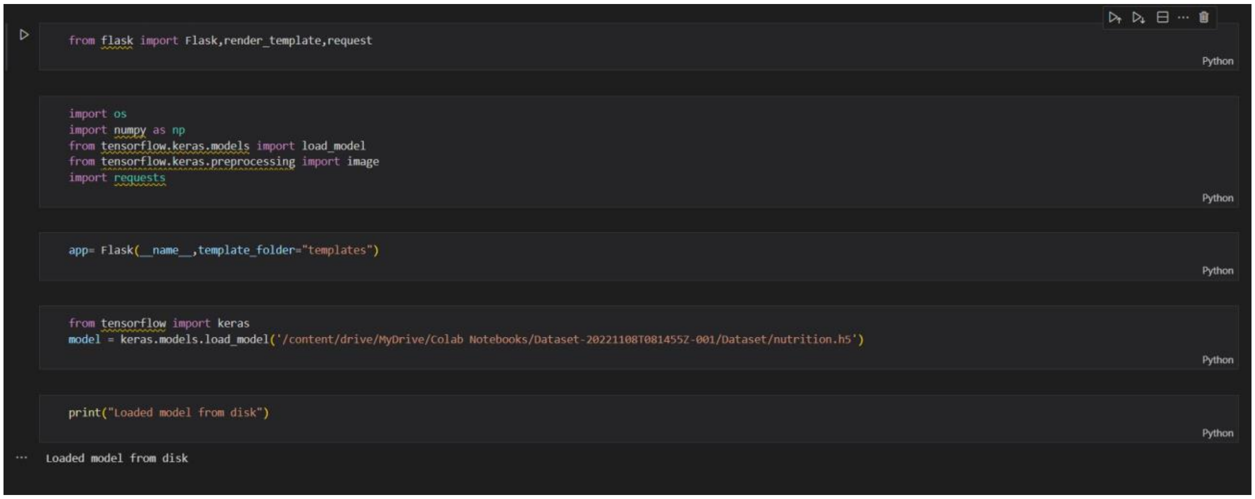
- ➢ know fundamental concepts and techniques of Convolutional Neural Network.
- ➢ gain a broad understanding of image data.
- ➢ Know how to pre-process/clean the data using different data preprocessing techniques.
- ➢ know how to build a web application using the Flask framework.

12. FUTURE SCOPE

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

13.APPENDIX

13.1 Source Code



```
from flask import Flask,render_template,request
```

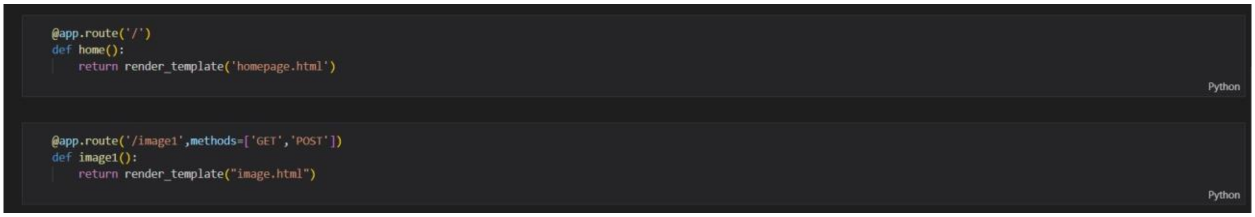
```
import os
import numpy as np
from tensorflow.keras.models import load_model
from tensorflow.keras.preprocessing import image
import requests
```

```
app= Flask(__name__,template_folder="templates")
```

```
from tensorflow import keras
model = keras.models.load_model('/content/drive/MyDrive/Colab Notebooks/Dataset-20221108T081455Z-001/Dataset/nutrition.h5')
```

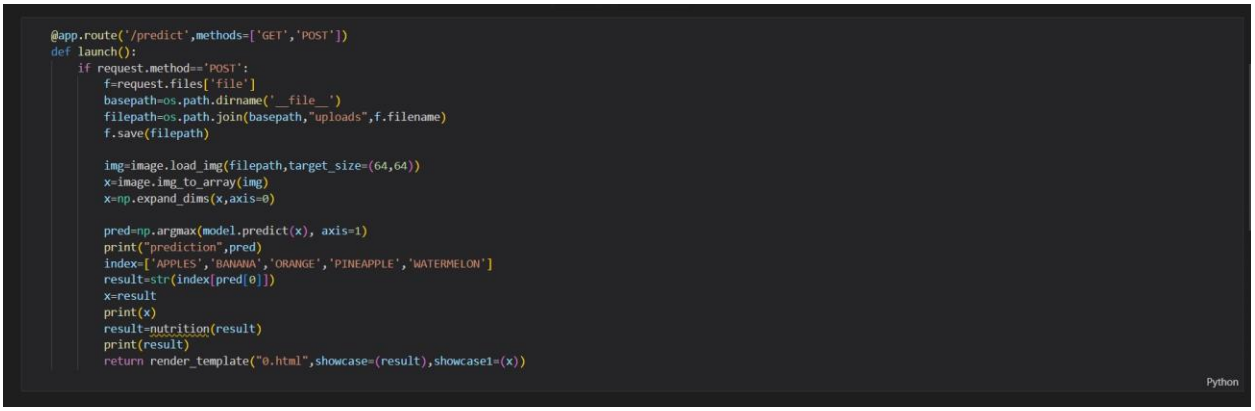
```
print("Loaded model from disk")
```

```
Loaded model from disk
```

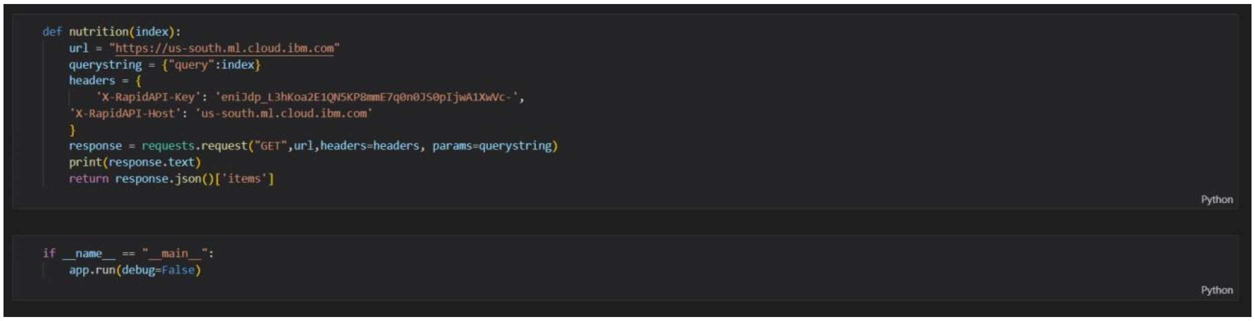


```
@app.route('/')
def home():
    return render_template('homepage.html')
```

```
@app.route('/image1',methods=['GET','POST'])
def image1():
    return render_template("image.html")
```



```
@app.route('/predict',methods=['GET','POST'])
def launch():
    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]])
        x=result
        print(x)
        result=nutrition(result)
        print(result)
        return render_template("0.html",showcase=(result),showcase1=(x))
```



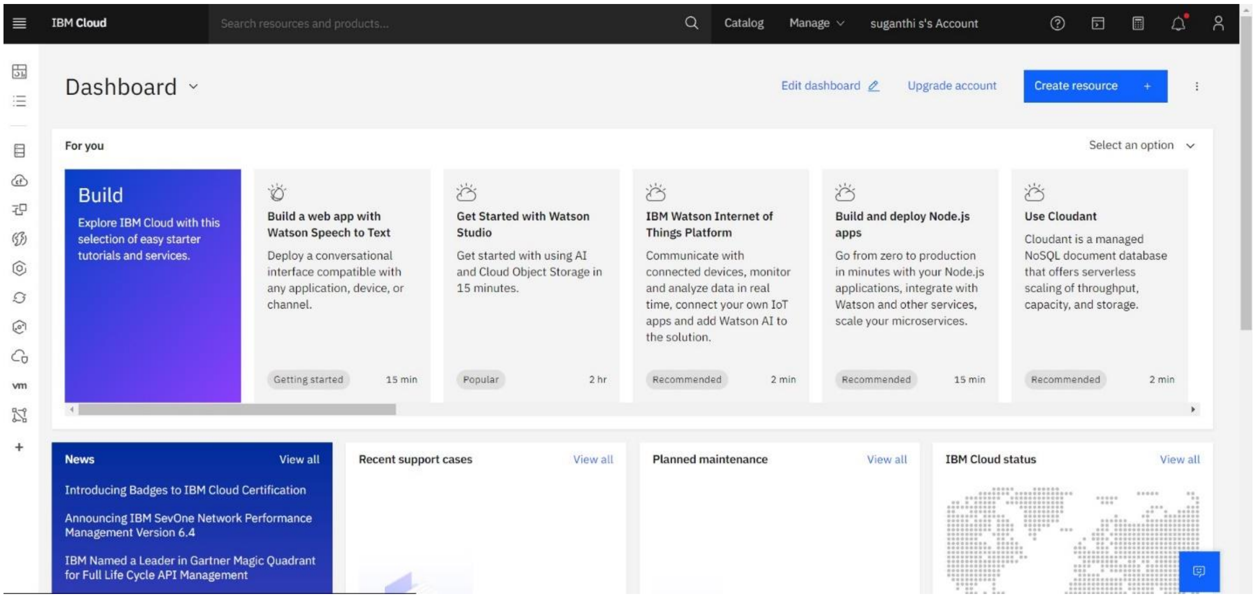
```
def nutrition(index):
    url = "https://us-south.ml.cloud.ibm.com"
    querystring = {"query":index}
    headers = {
        'X-RapidAPI-Key': 'eniJdp_L3hKoa2E1QN5KP8mmE7q0n0JS0pIjwA1XwVc-',
    'X-RapidAPI-Host': 'us-south.ml.cloud.ibm.com'
    }
    response = requests.request("GET",url,headers=headers, params=querystring)
    print(response.text)
    return response.json()['items']
```

```
if __name__ == "__main__":
    app.run(debug=False)
```

14. 0<u>IBM CLOUD REGISTRATION</u>

14.1 Registration snap

TEAM MEMBER :S. SUGANTHI



Build

Explore IBM Cloud with this selection of easy starter tutorials and services.

Build a web app with Watson Speech to Text

Deploy a conversational interface compatible with any application, device, or channel.

Getting started 15 min

Get Started with Watson Studio

Get started with using AI and Cloud Object Storage in 15 minutes.

Popular 2 hr

IBM Watson Internet of Things Platform

Communicate with connected devices, monitor and analyze data in real time, connect your own IoT apps and add Watson AI to the solution.

Recommended 2 min

Build and deploy Node.js apps

Go from zero to production in minutes with your Node.js applications, integrate with Watson and other services, scale your microservices.

Recommended 15 min

Use Cloudant

Cloudant is a managed NoSQL document database that offers serverless scaling of throughput, capacity, and storage.

Recommended 2 min

News View all

Introducing Badges to IBM Cloud Certification

Announcing IBM SevOne Network Performance Management Version 6.4

IBM Named a Leader in Gartner Magic Quadrant for Full Life Cycle API Management

Recent support cases View all

Planned maintenance View all

IBM Cloud status View all