AI-POWERED NUTRITION ANALYZER FOR FITNESS ENTHUSIASTS

IBM NALAIYA THIRAN

PROJECT REPORT

Submitted By

SRINITHI A	**(711719205053)**
AGALYA R	**(711719205003)**
SAKTHI SUDHARSINI S	**(711719205043)**
KANAGA PRIYA S	**(711719205023)**

in partial fulfillment for the award of the degree of

BACHELOR OF TECHNOLOGY

IN

INFORMATION TECHNOLOGY

KGISL INSTITUTE OF TECHNOLOGY,

COIMBATORE-641027

ANNA UNIVERSITY::CHENNAI 600 025

NOVEMBER 2022

TABLE OF CONTENTS

CHAPTER 1

INTRODUCTION

Nutrition 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. 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.

A fruit dataset describes variety of fruit images apple, banana, orange, pineapple, watermelon. The fruit nutrition is done by analysis. The analysis can be done using convolution neural network. It has four layers as convolution layer, pooling layer, flattening layer and fully connected layer.

1.1 Project Overview

The overview of the project is to give the output by doing the image processing and make the list of nutrients to show off. The user interacts with the UI (User Interface) and give the image as input. Then the input image is then pass to our flask application, and finally with the help of the model which we build we will classify the result and showcase it on the UI.By using the Pycharm the Input is been predicted and output is showcased as nutrition content.

The process involved in this is Data Collection,Image Preprocessing,Model Building,Application Building.Dataset folder contains the training and testing images for training our model.

We are building a Flask Application that under project name Flask , HTML pages are stored in the templates folder and css codings are stored in static folder and python file app.py, Sample_images, uploads, nutrition.h5 are stored in under Flask folder, templates folder contains home.html, image.html, imageprediction.html, aboutus.html pages.

Static folder had the css 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.

1.2 Purpose

The purpose of this project is to make it beneficial for the fitness enthusiasts. Sometimes,the people tempt to eat more and make their calories increase in their diet and make the diet plan collapse.So at this point this NutriFact website make them to create awareness and make the sense of giving the correct nutrition chart.So that they will be eating only up to the level.This website gives the nutrient-content list like Potassium, Carbohydrates, Fat, Satuarted Fat, Mono-unsaturated Fat, Poly- unsaturated Fat, Energy, Protein, Sugar and Sodium.

CHAPTER 2

LITERATURE SURVEY

2.1 Existing Problem

Unfortunately, some nutritional software packages are of poor quality, and the technical support provided to users is sometimes inadequate. In addition, although many excellent software packages and databases are available, they are open to misuse by users who do not understand or appreciate the limitations of such systems. This review examines some of the sources of error associated with the use of nutritional analysis software. In many build sources the information of nutrient content is very less.So it can be improvised .

2.2 References

1) Tushar Vashisht Mathew Cherian and Sachin Shenoy (2012) . Healthify Me for calorie tracking and advice on nutrition.

2) Sean Moriarty Developer Website: Leaf Group Limited . MY PLATE CALORIE TRACKER for tracking daily calorie intake

3) Ketan Mavinkurve (7-10-2022) .ALPHA COACH EVOLVE for Losing weight with simplified dieting.

4) Charles Boes (2-10-2018) .FOODVISOR-NUTRITION AND DIET .nutrition guide

5) Donna J. Scott (October 15, 2017). "How Much Does MyFitnessPal Premium Cost and is it Worth?". ModernFit. Retrieved June 22, 2019.

6) Swearingen, Jake (2018-10-10). Calorie counter by fat secret ". Intelligencer. Retrieved 2019-09-15.

7) Babish, Sian (December 19, 2020). "Noom review: Can this program help you achieve sustainable weight loss. Chicago Tribune. Retrieved June 22, 2021

2.3 Problem Statement Definition:

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

CHAPTER 3

IDEATION & PROPOSED SOLUTION

3.1 Empathy Map Canvas

In this following empathy map we gave a clear idea about our AI-NutritionAnalyzer Website. The outlook of this is what the people think and feel, see, say

For example:

Pain: Slow process for some people, negative feedbacks and accuracy may vary.Gain:

Extra things like consultants, quick and easy and many food journals.

Think and Feel: Using a dedicated smart phone app, easy to work with it, it can reduceman's power.

figure 3.1.1

3.2 Ideation & Brainstorming

We have discussed different ideas during ideation phase and organised under person 1 and so on .We have discussed about various algorithms for implementation.This ideasgive a clear view about the process.

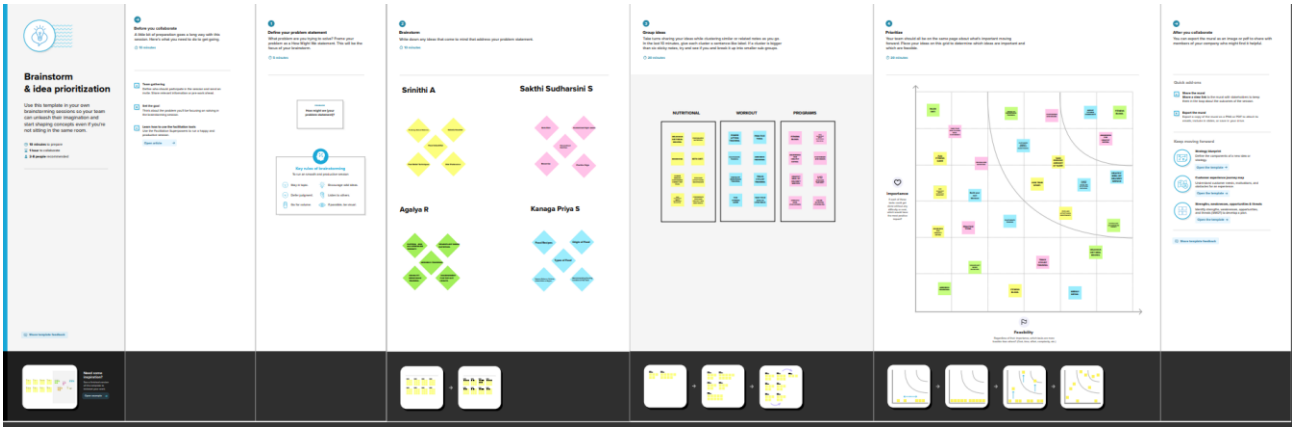
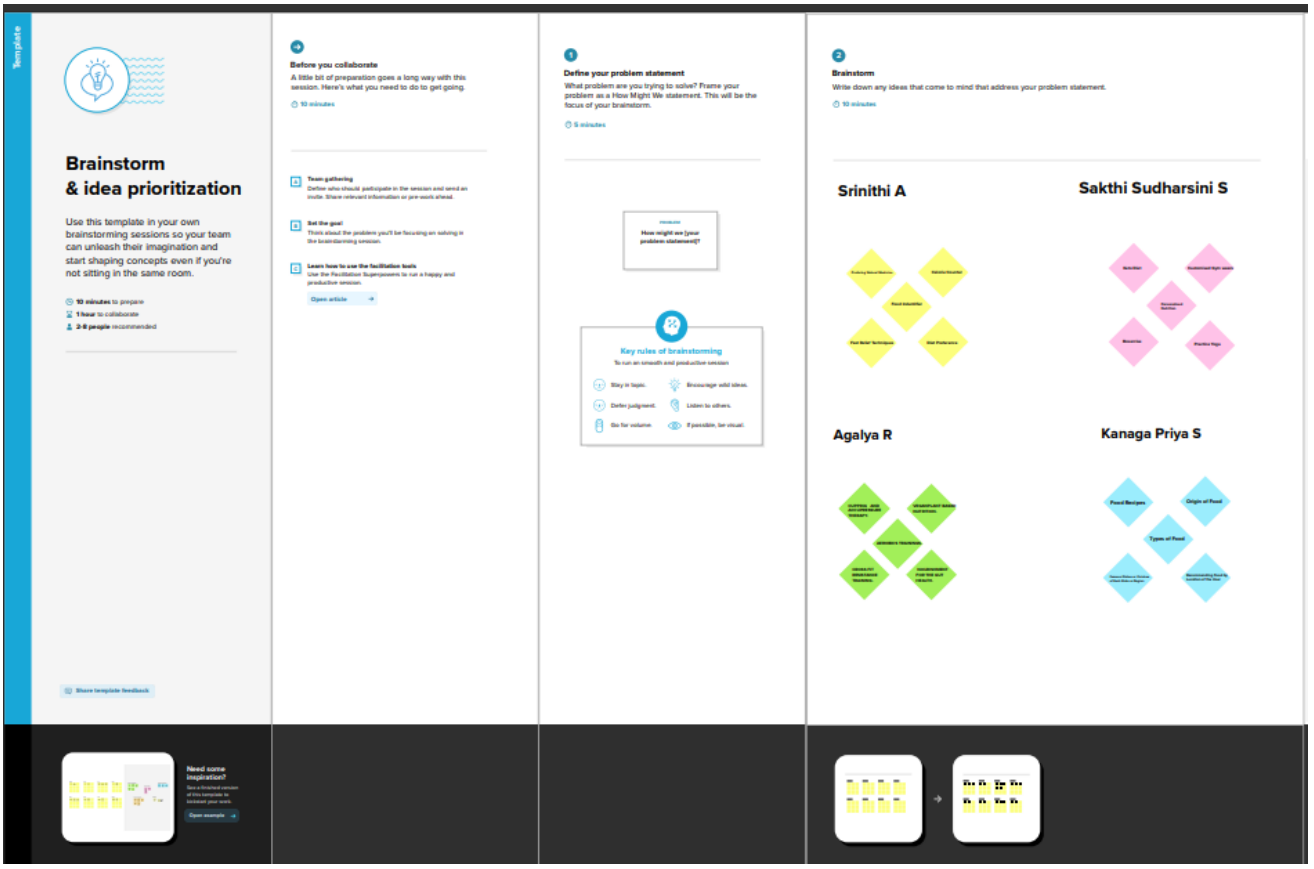


figure 3.2.1



3.3 Proposed solution

We have used convolution neural network to analyse the image.It has fourlayers.They are convolution layer, pooling layer, flattening layer, fully connected layer.

S.No.	Parameter	Description
1.	Problem Statement (Problem to be solved)	A regular person must use cutting-edge AI- based analysing software to identify fruits and vegetables based on colour, texture, form, and other characteristics. At the time of identification, the user must also be aware of thenutritional content of that specific edible.
2.	Idea / Solution description	**Main Solution**: ● Clear and proper identification of the given input data. ● Provide nutritional facts based on the obtained data. ● Fitness analysis and maintenance as per the user's body conditions **Additional benefits**: ● Analysis of daily dietary requirements ● Daily tracking of dietary consumptionthoroughly.
3.	Novelty / Uniqueness	● The availability of fitness plans withaddon bonuses ● Suggestion of home remedies and simple solutions for basic problems. ● An individualized food plan based onhealth condition and deficiency. ● Allowing for diet flexibility helps promote a healthy and effective eating pattern
4.	Social Impact / Customer Satisfaction	● Healthy lifestyle development ● Constant calorie management monitoring results in a fitness mindset.

5.	Business Model (Revenue Model)	• Consultation with nearest trainers and nutritionist for personalized plans.
		• Adopt a specialized diet plan under the direction of an expert. • Advertise and offer nutritional supplements and fitness gear. • Promotion for fitness centers and hospitals.
6.	Scalability of the Solution	• Improving accuracy by expanding thedata collection using user input data • Storage requirements of a specific food. • User friendly UI for everyone to use andget benefit from it.

Convolution layer:

It has a filter passes over the image,scanning a few pixels at a time and creatinga feature map that predicts the class to which each feature belongs.

Pooling Layer:

It reduces the amount of information in each feature obtained in the convolutionlayer while maintaining the most important information.

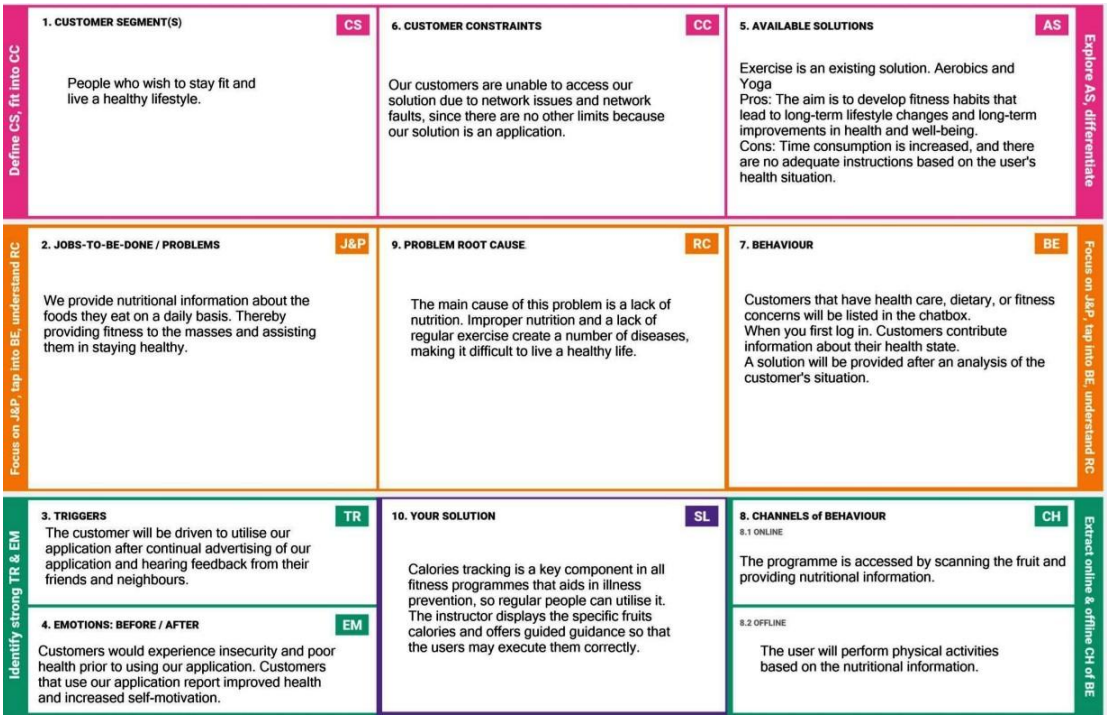
Flatten Layer:

It takes the output of the previous layers,flattens them and turns them into asingle vector that can be input for the next stage.

Fully Connected Layer:

It applies weights to predict the correct label and gives final probabilities.

3.3 Problem Solution fit



CHAPTER 4

REQUIREMENT ANALYSIS

4.1. Functional Requirement:

FR No.	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)
FR-1	HTML Page creation	To give input image
FR-2	Flask Module	To run the application
FR-3	Database	For storing the information
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	Result	The nutrient content in the particular fruit is displayed

Table 4.1.1

In this functional requirement, we first collect the dataset and do image preprocessing and build model using CNN layers and predict the required output.

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 tobe 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 shouldbe available.
NFR-6	**Scalability**	Ability of the application to handle an increase in workload without performance degradation, orits ability to quickly enlarge.

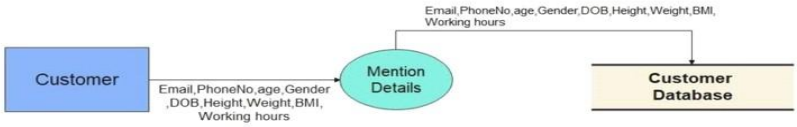
Table 4.2.1

CHAPTER 5

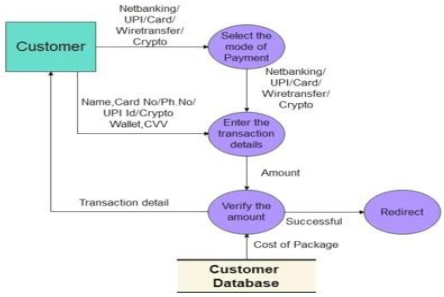
PROJECT DESIGN

5.1 Data Flow Diagrams

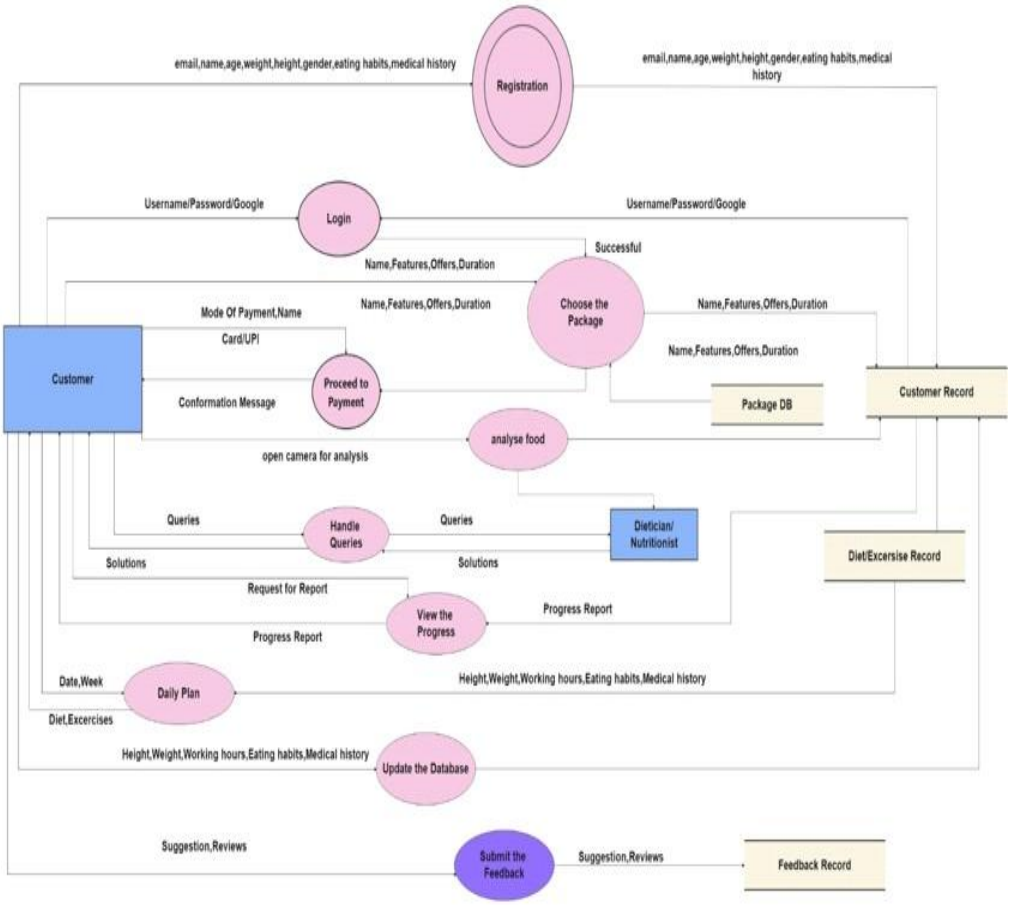
DFD-1(Registration):



DFD-1(Payment):

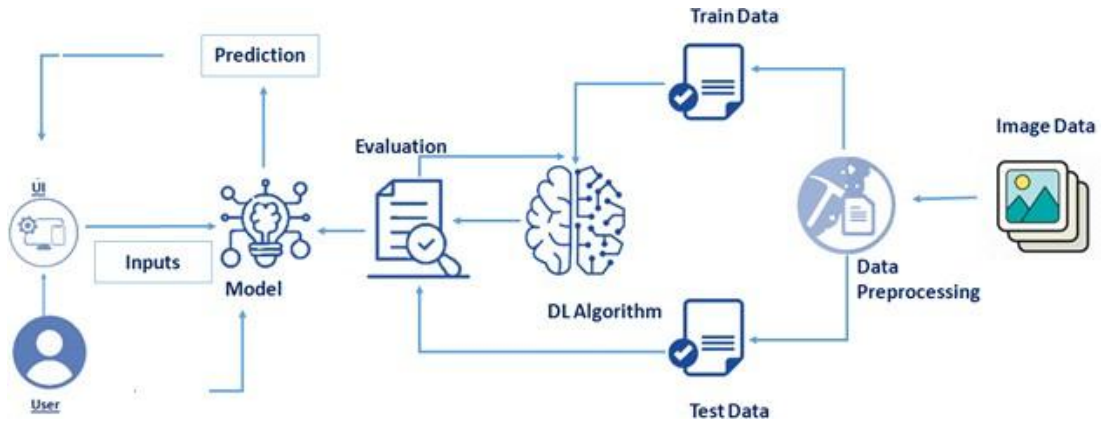


DFD-0:



5.2 Solution & Technical Architecture

figure 5.2.1



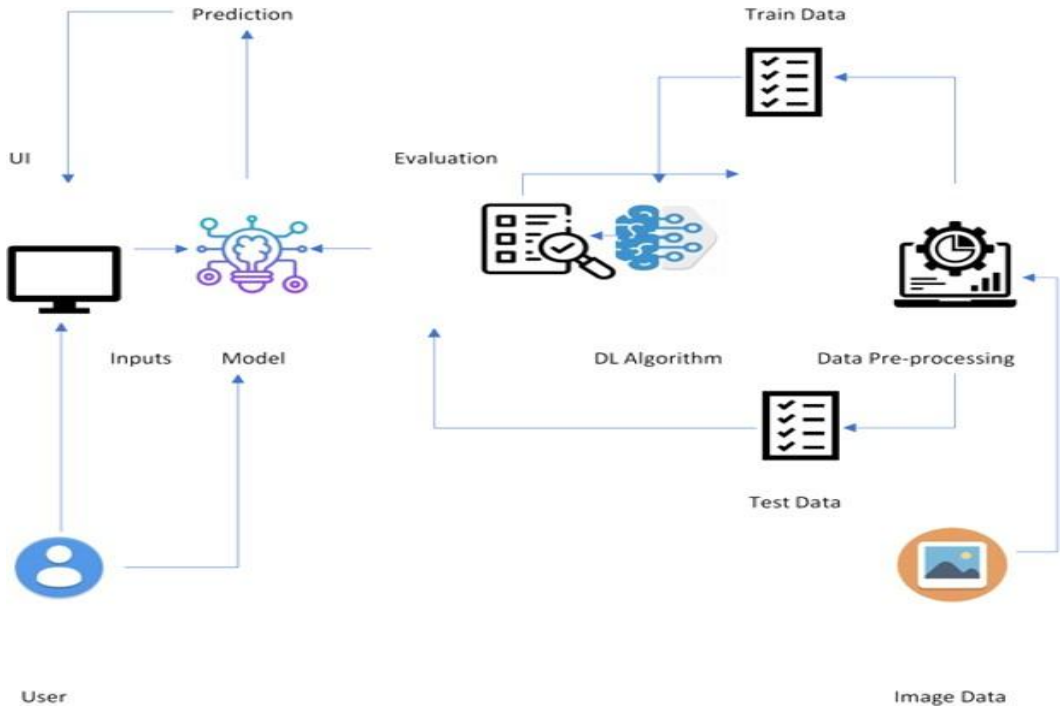


Figure 5.2.2

5.3 User Stories

User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority	Release
Customer	Registration	USN-1	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-2	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-3	As a user, I can register for the application through Google	I can register & access the dashboard with Facebook Login	Low	Sprint-2
		USN-4	As a user, I can register for the application through Microsoft	I can access the Dashboard with Microsoft.	Medium	Sprint-1
	Login	USN-5	As a user, I can log into the application by entering email & password	I can login the Application by entering password	High	Sprint-1
	Main Interface	USN-6	As a user I can view my calorie intake by clicking photo of the food I eat	Access the proper information about the nutrition and the calorie intake	High	Sprint-2
	Package DB, Dashboard	USN-7	As a user I can choose variety of packages based on my requirement	Selecting an appropriate package	Medium	Sprint-2
Customer Care Executive	Feedbacks DB , Tollfree number, chat bot	USN-8	As a customer care executive, I collect feedbacks from customers	Maintaining proper environment for the customers	High	Sprint-2
Dietitian	Customer Record	USN-9	As a dietitian I provide daily plans for the betterment of the user	Positive results from user	High	Sprint-2
Administrator	Dashboard	USN-10	As an administrator I take care of all the operations which takes place in the app	Zero issues from the user	High	Sprint-2

figure 5.3.1

CHAPTER 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	Download Food Nutrition Dataset	10	High	Srinithi A
Sprint-1	Data Preprocessing	USN-2	Importing The Dataset into Workspace	6	Medium	Sakthi Sudharsinis Agalya R
Sprint-1		USN-3	Handling Missing Data	5	Medium	Kanaga Priya S
Sprint-1		USN-4	Feature Scaling	3	Low	Kanaga Priya S
Sprint-1		USN-5	Data Visualization	4	High	Srinithi a
Sprint-1		USN-6	Spitting the Data into the Train and Test	4	Medium	Sakthi Sudharsinis Agalya R
Sprint-1		USN-7	Creating A Dataset with Sliding Windows	4	Medium	Sakthi Sudharsinis
						Agalya R
Sprint-2	Model Building	USN-8	Importing The Model Building Libraries	1	Medium	Venuka. A
Sprint-2		USN-9	Initializing The Model	3	High	Srinithi A
Sprint-2		USN-10	Adding LSTM Layers	2	Medium	Sakthi Sudharsinis Agalya R
Sprint-2		USN-11	Adding Output Layers	3	High	Srinithi A
Sprint-2		USN-12	Configure The Learning Process	2	Low	Kanaga Priya S
Sprint-2		USN-13	Train The Model	2	Medium	Srinithi A
Sprint-2		USN-14	Model Evaluation	1	Medium	Sakthi Sudharsinis Agalya R
Sprint-2		USN-15	Save The Model	2	Medium	Sakthi Sudharsinis Agalya R
Sprint-2		USN-16	Test The Model	3	High	Srinithi A
Sprint-3	Application Building	USN-17	Create An HTML File	4	Medium	Srinithi A
Sprint-3		USN-18	Build Python Code	4	High	Sakthi Sudharsinis Agalya R
Sprint-3		USN-19	Creating our Flask application and loading our model by using load_model method	4	Medium	Srinithi A
Sprint-3		USN-20	Routing to HTML page	4	High	Srinithi
Sprint-3		USN-21	Run the application	2	Medium	Sakthi Sudharsinis Agalya R
Sprint-4	Train The Model On IBM	USN-21	Register For IBM Cloud	4	Medium	Kanaga Priya S
Sprint-4		USN-22	Train The ML Model On IBM	8	High	Srinithi A
Sprint-4		USN-23	Integrate Flask with Scoring End Point	8	High	Sakthi Sudharsinis

figure 6.1.1

6.2 Sprint Delivery Schedule

Sprint	Total Story Points	Duration	Sprint Start Date	Sprint End Date (Planned)
Sprint-1	12	6 Days	24 Oct 2022	29 Oct 2022
Sprint-2	8	4 Days	30 Oct 2022	02 Nov 2022
Sprint-3	8	3 Days	03 Nov 2022	05 Nov 2022
Sprint-4	40	7 Days	06 Nov 2022	12 Nov 2022

figure 6.2.

6.3 Reports from JIRA

Burndown Chart

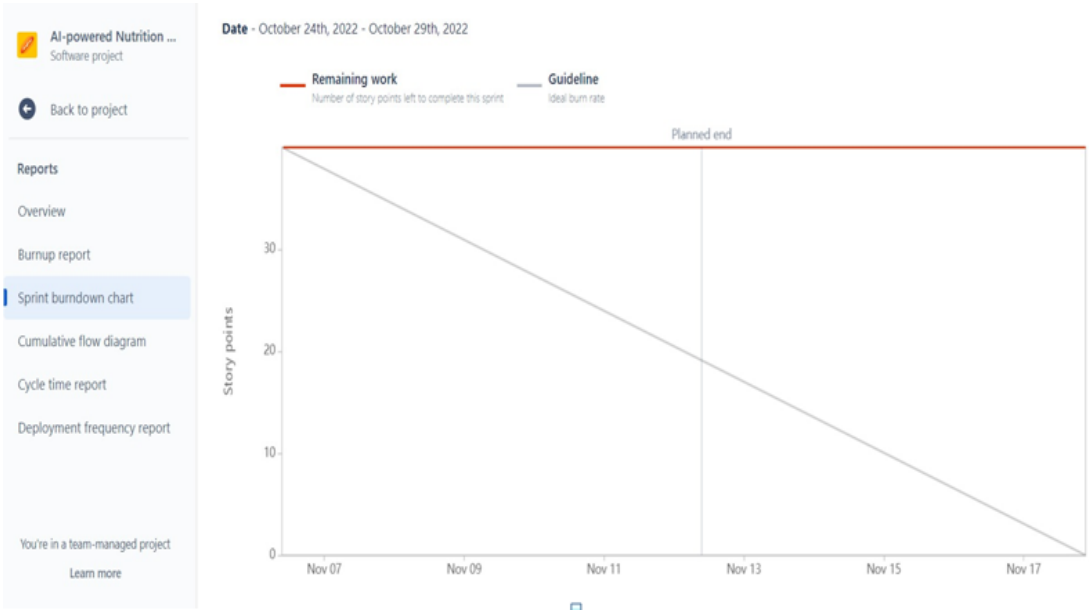


Figure 6.3.1

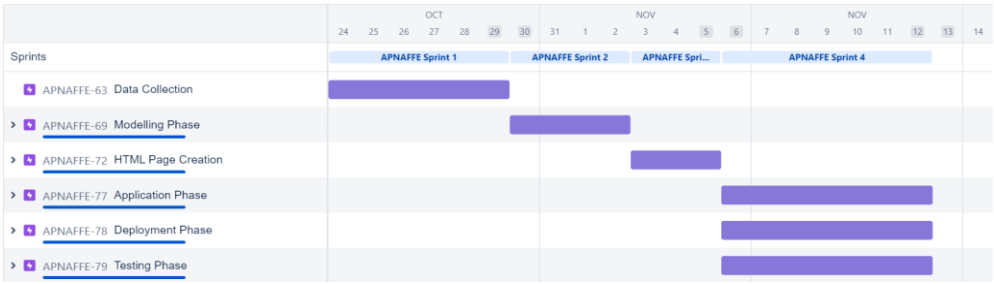


Figure 6.3.2

CHAPTER 7

CODING & SOLUTIONING

7.1 Feature 1

Python Flask Code

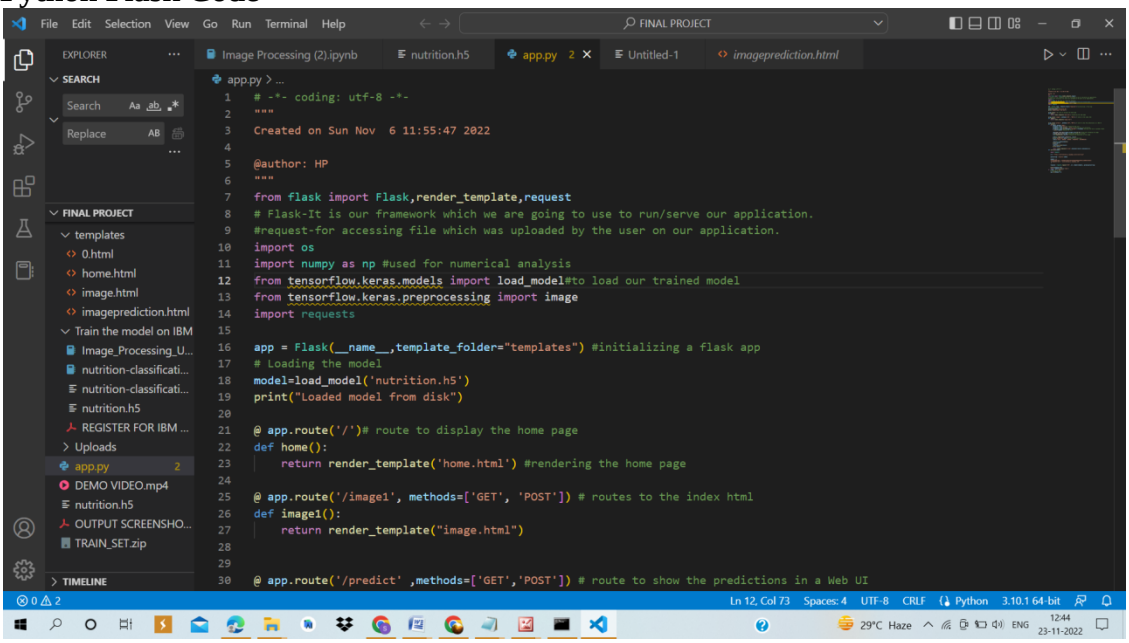


figure 7.1.1

Source code:

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

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

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

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

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


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

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

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

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

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

    import requests

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

    querystring = {"query":index}

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

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

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

7.2 Feature 2

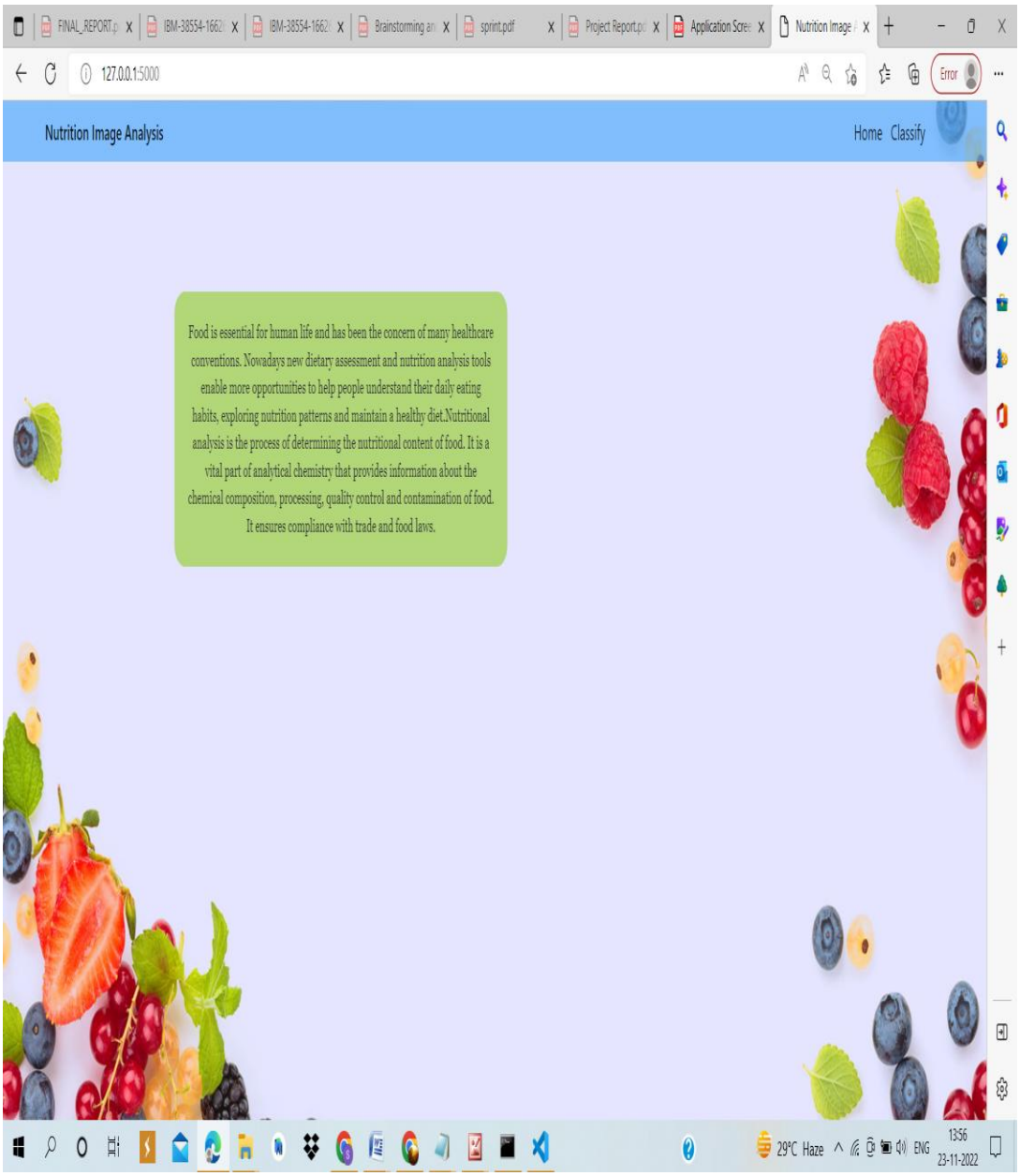
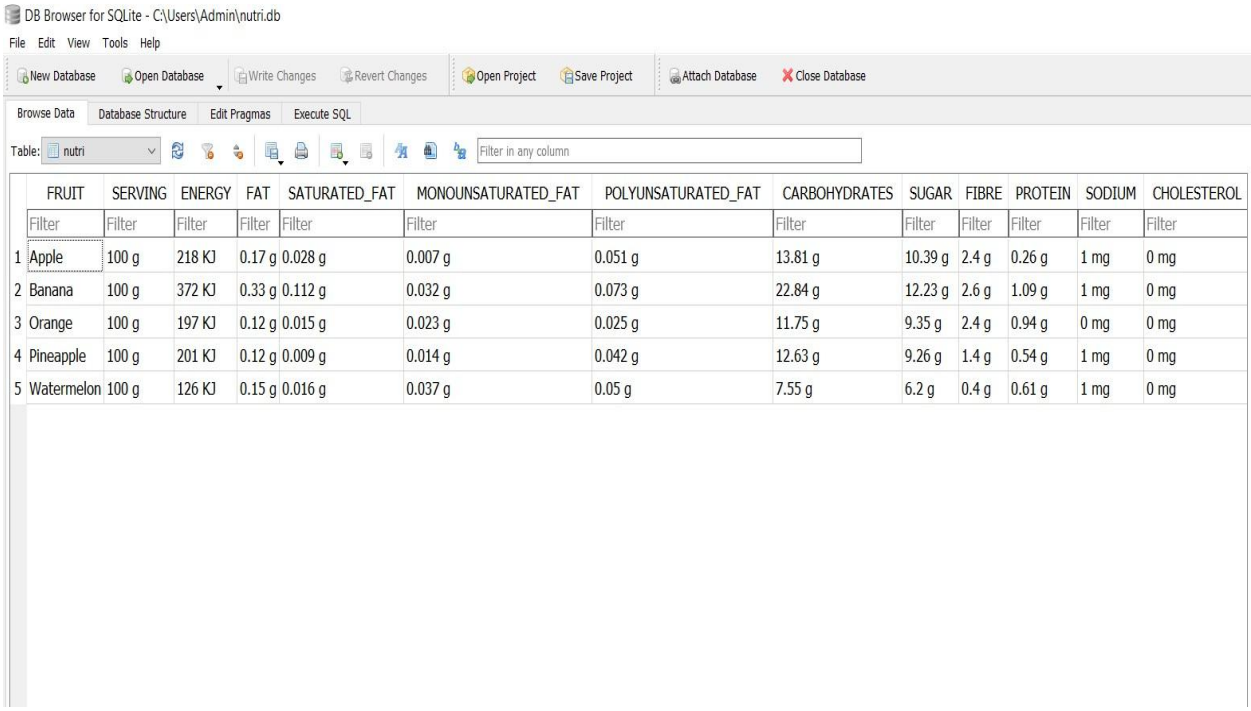


Figure 7.2.1

7.3 Database Schema

Database of fruits was created using jupyter notebook and implemented and an nutri.db file is generated after execution.The table for storing database is created and it can be viewed by using SQLlite Software.



	FRUIT	SERVING	ENERGY	FAT	SATURATED_FAT	MONOUNSATURATED_FAT	POLYUNSATURATED_FAT	CARBOHYDRATES	SUGAR	FIBRE	PROTEIN	SODIUM	CHOLESTEROL
	Filter	Filter	Filter	Filter	Filter	Filter	Filter	Filter	Filter	Filter	Filter	Filter	Filter
1	Apple	100 g	218 KJ	0.17 g	0.028 g	0.007 g	0.051 g	13.81 g	10.39 g	2.4 g	0.26 g	1 mg	0 mg
2	Banana	100 g	372 KJ	0.33 g	0.112 g	0.032 g	0.073 g	22.84 g	12.23 g	2.6 g	1.09 g	1 mg	0 mg
3	Orange	100 g	197 KJ	0.12 g	0.015 g	0.023 g	0.025 g	11.75 g	9.35 g	2.4 g	0.94 g	0 mg	0 mg
4	Pineapple	100 g	201 KJ	0.12 g	0.009 g	0.014 g	0.042 g	12.63 g	9.26 g	1.4 g	0.54 g	1 mg	0 mg
5	Watermelon	100 g	126 KJ	0.15 g	0.016 g	0.037 g	0.05 g	7.55 g	6.2 g	0.4 g	0.61 g	1 mg	0 mg

figure 7.3.1

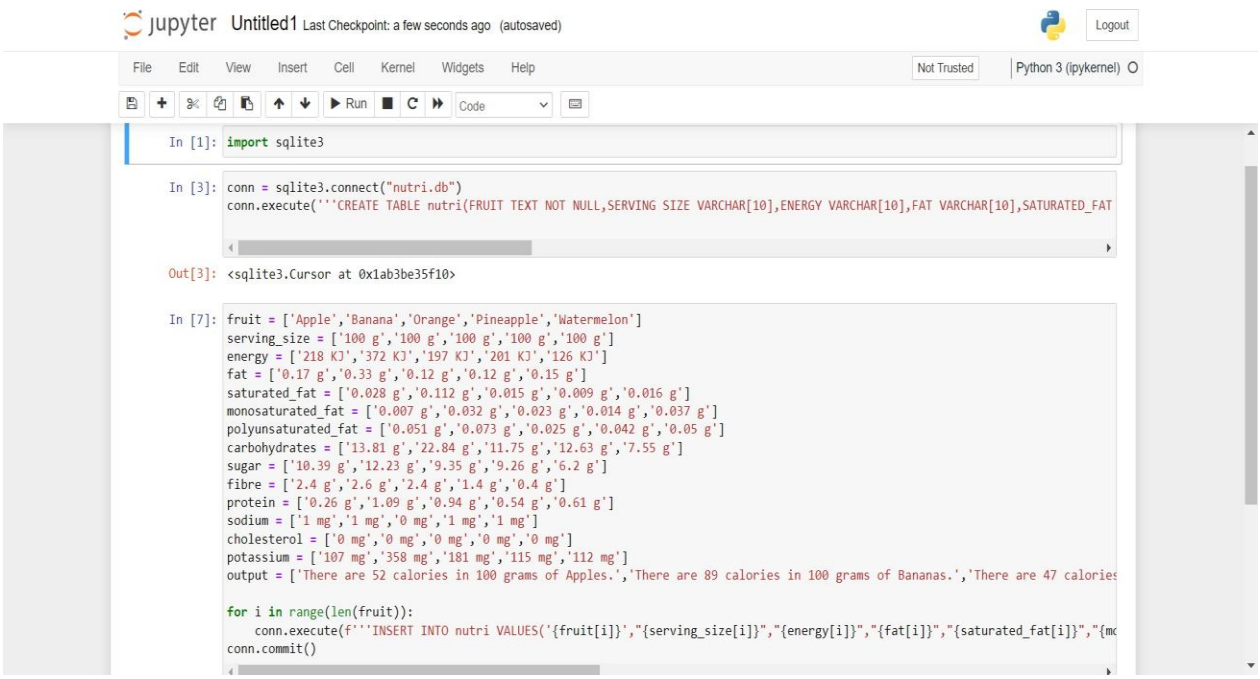


figure 7.3.2

CHAPTER 8

TESTING

8.1 Test Cases

A test case might be created as an automated script to verify the functionalityas per the original acceptance criteria.

Test case ID	Test case ID	Test case ID	Test case ID
Model Building_TC_OO1	Training and Testing	Python	Verify whether the image prediction is proper or not
Backend_TC_OO2	App Configuration	Python	It will get data from front end and processit
Frontend_TC_OO3	UI	Home page(user),user input Page,image prediction page page,about us page	user can give inputas jpg,jpeg,png format and displayoutput
Datebase_TC_OO4	Prediction	Python	Verify that it display the information as correct

Table 8.1.1

Steps To Execute	Test Data	Expected Result	Status	Executed By
1. Importing dataset and unzip it 2. Image preprocessing 3. Add convolution layers and predict fruit	http://127.0.0.1:5000	Predict the fruit	Pass	KANAGA PRIYA S
1. APP configuration 2. APP Route	/http://127.0.0.1:5000	Users data should processIn Backend it should get data from frontend and display output	Pass	SRINITHI A
1. Enter the input image 2. Pick the image format as jpg,png,jpeg 3. Click submit	http://127.0.0.1:5000 _____ http://127.0.0.1:5000 _____ http://127.0.0.1:5000 _____ http://127.0.0.1:5000 _____	User should navigate to home page andrequired pages they want to go.	Pass	AGALYA R
1. Declaring the table using database code 2. Data should store with the various featuresof fruit	http://127.0.0.1:5000 http://127.0.0.1:5000\	Display datafrom given database created	Pass	SAKTHI SUDHARSINI S

Table 8.1.2

8.2 User Acceptance Testing

Defect Analysis:

Resolution	Severity 1	Severity 2	Severity 3	Severity 4	Subtotal
By Design	8	3	2	1	14
Duplicate	0	1	0	2	3
External	3	2	0	1	6
Fixed	6	1	3	12	22
Not Reproduced	0	0	0	1	1
Skipped	0	0	1	0	1
Won't Fix	0	1	0	0	1
Totals	15	08	6	17	48

Table 8.2.1

Testcase Analysis:

Section	Total Cases	Not Tested	Fail	Pass
Routing to pages	2	0	0	2
Input Page	4	0	0	4
Storing Image	4	0	0	4
Image Prediction	5	0	0	5
Final Output	5	0	0	5

Table 8.2.2

CHAPTER 9

RESULTS

9.1 Performance Metrics

NFT - Risk Assessment

Project Name	Scope	Functional Changes	Hardware Changes	Software Changes	Impact of Downtime	Risk Score
AI-powered Nutrition Analyzer for Fitness Enthusiats	New	Low	Moderate	Moderate	Low	GREEN

Table 9.1.1

NFT - Detailed Test Plan

S.No	Project Overview	NFT Test approach	Assumptions/Dependencies/Risks	Approvals/SignOff
1	Display nutrient content	Stress	App Crash/ Developer team/ Site Down	Approved
2	Display nutrient content	Load	Server Crash/ Developer team	Approved

Table 9.1.2

End Of Test Report

NFR - Met	Test Outcome	GO/NO-GO decision	Recommendatio ns	IdentifiedDefects (Detected/Close d/Open)
Performance	CPU -01	GO	High Performance	Closed
Database Information	Storage	NO-GO	SQLlite for access	Closed

Table 9.1.3

CHAPTER 10

ADVANTAGES AND DISADVANTAGES

Advantages:

- The platform provides nutrition-based data services, analytics, and technologies to its consumers .
- To enable individualised compilation of data, the platform uses NLP and mathematical models from the optimation theory and predictive analysis.
- The app uses AI and image classification technology to identify the food correctly and accurately and also calculated the amount the calories just from the picture.
- API with diverse data sets, the food which is automatically recognised by the platform is paired with detailed nutrition information.
- Online platform which uses deep learning and image recognition to analyse what the users eat and determine what is trending in terms of each popular dish that the user eats and consumption time.
- Time is saved and managed for the people who are using the website like this.

Disadvantages:

- Sometimes the fruits which are rotten that is also been analyzed.
- Only for the limited number of fruits the image & data has been processed andcoded to give the output.
- It supports only the jpeg, png,jpeg file not the other format.If the file which isgiven in any other format it does not support it.
- It does not say the accurate fruits if the different types of fruit variety such asgreen apple or cut fruits it will not show the accurate fruit variety in it

CHAPTER 11

11. CONCLUSION

NutriFact website help others to create awareness for the fitness enthusiasits and created successfully using Pycharm and image input is accepted and it does image preprocessing and under convolution layers and predict the output .It will display the name of fruit that we give as image input and its nutritional facts. User can know more details and get quality information from our website.It helps in maintaining health and fitness.

NutriFact is understandable to all people so anyone can use this.It is very simple and easy to use. Users will definitely enjoy this application and know more intersting facts about fruits.

CHAPTER 12

FUTURE SCOPE

1. **Based On Reviews:**

In future the feedbacks are asked from users to meet their expectations like additional features such as BMI calculation,daily calorie intake calculator various features.

2. **Nutritional Chatbot:**

In future the nutrition chatbot will be used.It can have various type of input such as speech recognition,image input,input as name and will give output in form of voice or text .Analyzing according to the bowl size.We will try to take the disadvantages as the challenge and make it possible in the upcomings.

CHAPTER 13

APPENDIX

13.1 Source Code

1. Python flask code - app.py

```
# -*- coding: utf-8 -*-
"""
Created on Sun Nov  6 11:55:47 2022

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

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

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

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


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

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

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

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

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

    import requests

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

    querystring = {"query":index}

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

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

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

HTML and CSS Pages Screenshots:

Home Page

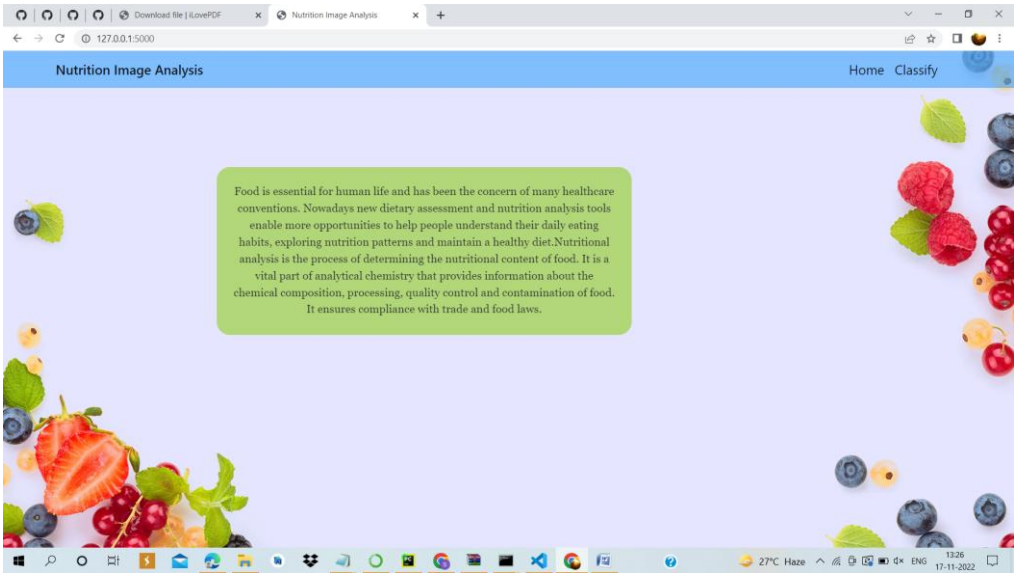


figure 13.2.1

Input Page

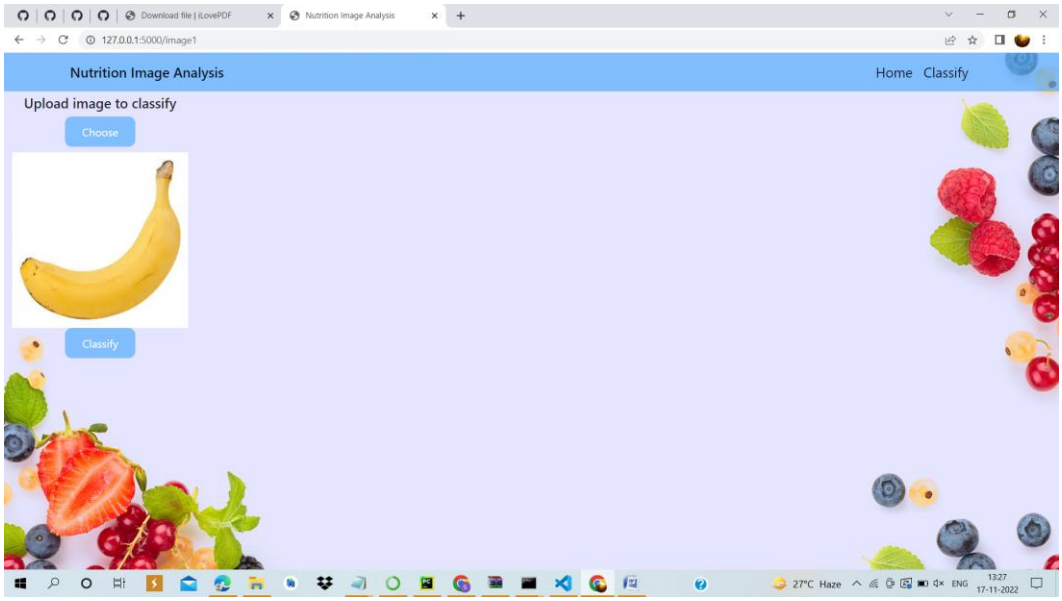


figure 13.2.2

Imageprediction Page

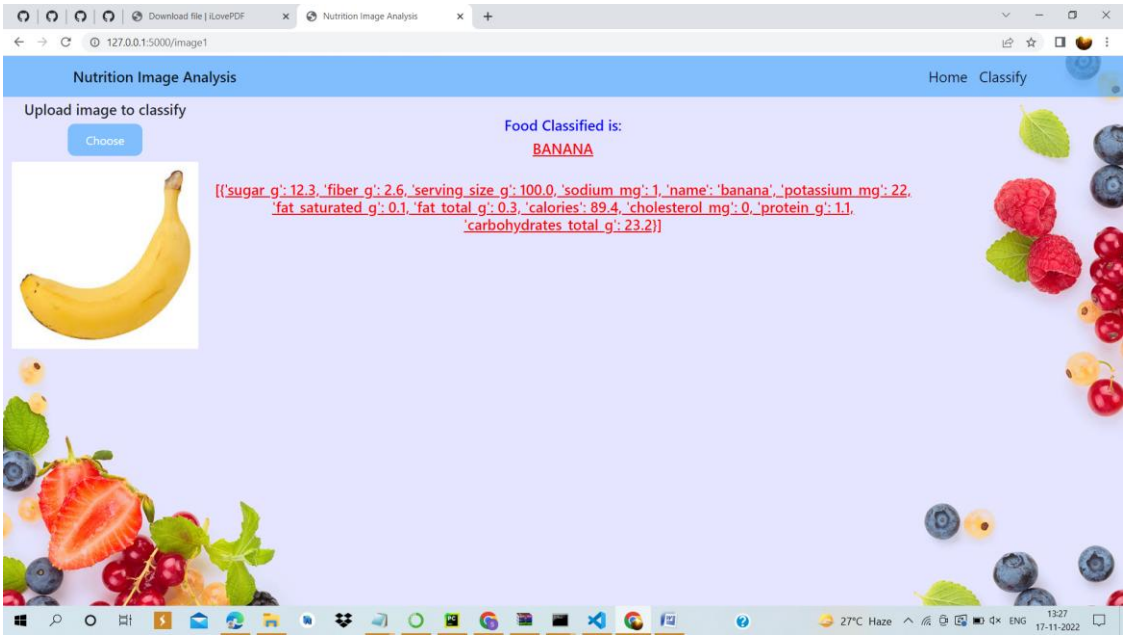


figure 13.2.3

About us Page

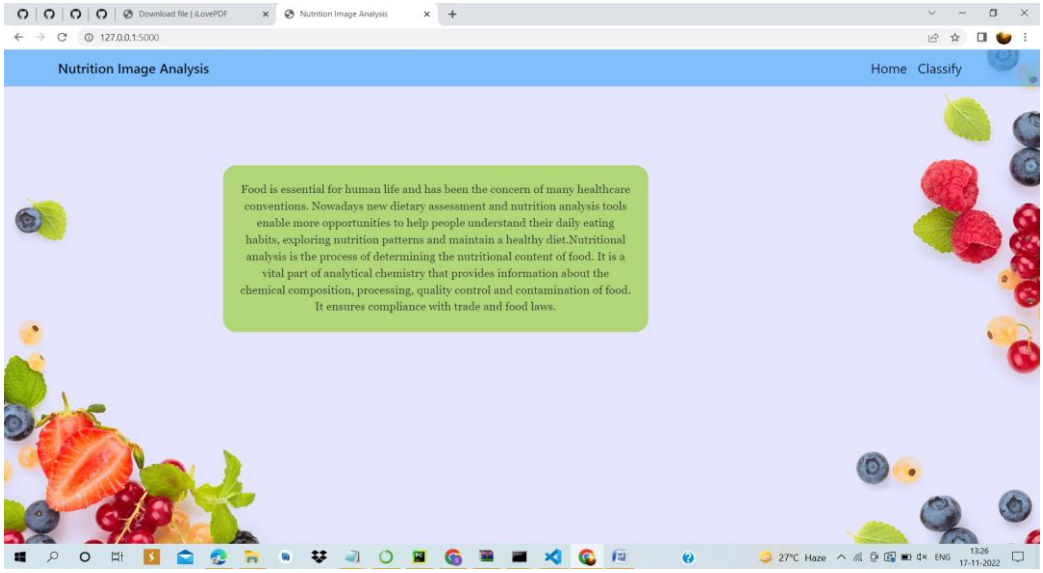


figure 13.2.4

13.2 GitHub & Project Demo Link

GitHub Link:
https://github.com/IBM-EPBL/IBM-Project-46427-1660746969

Project Demo Link

https://drive.google.com/file/d/1nvamrTJ7pEFNkrjpk6mNTE3Yyg2scGiI/view?usp=share_link