

# **NUTRITION ASSISTANT APPLICATION**

USING CLOUD

*A Project report submitted in partial fulfilment of 7<sup>th</sup> semester in degree  
of*

**BACHELOR OF ENGINEERING  
IN**

## **COMPUTER SCIENCE AND ENGINEERING**

*Submitted by*

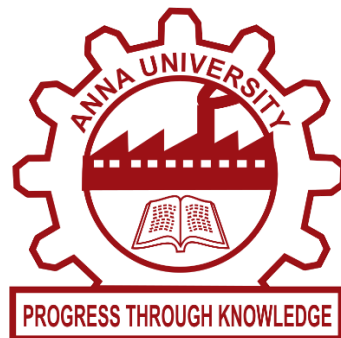
Team ID: PNT2022TMID38229

SRITHAR                      412319104029

HARI PRASAD              412319104008

DHILIP KUMAR            412319104005

KAMALESH                412319104010



**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING  
SRI RAMANUJAR ENGINEERING COLLEGE, CHENNAI-127**

ANNA UNIVERSITY: CHENNAI 600025

**NOV-2022**

**SRI RAMANUJAR ENGINEERING COLLEGE,CHENNAI-127**  
**(Affiliated to Anna University, Chennai)**



**BONAFIDE CERTIFICATE**

Certified that this project report "NUTRITION ASSISTANT APPLICATION" is the bonafide record work done by **SRITHAR (412319104029)**, **HARIPRASAD (412319104008)**, **DHILIPKUMAR (412319104005)** and **KAMALESH (412319104010)** for **IBM- NALAIYATHIRAN** in **VII** semester of **B.E.**, degree course in **Computer Science and Engineering** branch during the academic year of 2022 - 2023.

**Staff-In charge**

**Evaluator**

**Sai Priya**

**Head of the Department**  
**Dr.GOWRI SANKARAN**

---

---

# NUTRITION ASSISTANT APPLICATION

**DONE BY**

**TEAM ID: PNT2022TMID38229**

## 1. INTRODUCTION

A primary goal of the project is to provide you with information backed by nutritional science, and a variety of resources that use scientific evidence to optimize health and prevent disease. This text was designed to support, enrich, and expand the materials provided. The objective of this study is to identify dietary selfmonitoring implementation strategies on a mobile application. Nutritional knowledge is essential for promoting good eating habits since it ensures that necessary nutrient requirements are met to avoid malnutrition.

### 1.1 PROJECT OVERVIEW

This project aims at building a web App that automatically estimates food Attributes Such as ingredients and nutritional value by classifying the input image of food. Our method employs **Clarifai's AIDriven Food Detection Model** for accurate food identification and Food API's to give the nutritional value of the identified food.

### 1.2 PURPOSE

One of the most basic functions is to guide its users towards a healthy diet and assist them to achieve their health goals. So, once your user specifies the goal like desired weight goal, body type, food habits, and preferred food items, your app must suggest them with a proper diet accordingly. You can automatically calculate the nutritional information for any recipe, analyze recipe costs, visualize ingredient lists, find recipes for what's in your fridge, find recipes based on special diets, nutritional requirements, or favorite ingredients, classify recipes into types and cuisines, convert ingredient amounts, or even compute an entire meal plan.

## **ABSTRACT**

Malnutrition refers to deficiencies, excesses or imbalances in a person's intake of energy and/or nutrients. The term malnutrition covers 2 broad groups of conditions. One is 'undernutrition'—which includes stunting (low height for age), wasting (low weight for height), underweight (low weight for age) and micronutrient deficiencies or insufficiencies (a lack of important vitamins and minerals). The other is overweight, obesity and diet-related noncommunicable diseases (such as heart disease, stroke, diabetes, and cancer). Due to the ignorance of healthy food habits, obesity rates are increasing at an alarming speed, and this is reflective of the risks to people's health. People need to control their daily calorie intake by eating healthier foods, which is the most basic method to avoid obesity. However, although food packaging comes with nutrition (and calorie) labels, it's still not very convenient for people to refer to App-based nutrient dashboard systems which can analyze real-time images of a meal and analyze it for nutritional content which can be very handy and improves the dietary habits, and therefore, helps in maintaining a healthy lifestyle.

## **2.LITERATURE SURVEY**

### **2.1 EXISTING PROBLEM**

The way food is produced, marketed and made available to people has changed tremendously in the past 50 years. Many advances have been made, leading to more efficient food supply chains which have in turn generated improvements in food security and nutrition. Yet, malnutrition in all its forms continues to be one of the greatest challenges faced by our generation, and unhealthy diets are among the leading causes of death and disability. In some parts of the world, people are not eating sufficient amounts of food to provide the calories, vitamins and minerals they need for optimal health, while in others, people are eating far more food than is necessary for their health and well-being, or simply too much of foods high in fat, sugar and/or salt. This brief describes what parliamentarians need to know about the current nutrition situation in the world and how our food systems are shaping food environments that steer people towards unhealthy diets which are one of the causal factors of malnutrition. Parliamentarians are well placed to facilitate action to transform the world's current food

systems. This brief gives concrete examples of measures through which policymakers can influence food systems so as to promote healthy diets and prevent malnutrition in all its forms, including undernourishment, stunting, wasting, micronutrient deficiencies, overweight and obesity, as well as diet-related non-communicable diseases (NCDs).

## 2.2 REFERENCES

1. Rationale and developmental methodology for the simple approach: A Systematised, Interdisciplinary

Malnutrition Pathway for Implementation and Evaluation in hospitals

AUTHOR : Jack J Bell <sup>1 2</sup>, Adrienne Young <sup>3 4</sup>, Jan Hill <sup>5</sup>,

Merrilyn Banks <sup>3 4</sup>, Tracy Comans <sup>6</sup>, Rhiannon Barnes <sup>7</sup>,

Heather H Keller <sup>8 9</sup> ABSTRACT

:

Changing population demographics, service demands, and healthcare provider expectations suggest that a shift is required regarding how malnutrition care is managed in hospitals. The present study aims to build the reason for required change, and to describe the process used to develop a model for managing malnutrition for implementation across six Queensland

hospitals. A cross-sectional survey of approaches to managing malnutrition in Queensland public hospitals, and development of a new model of care (guided by Knowledge-to-Action Framework and qualitative interviews) for testing within a broader implementation program. Twenty-three surveys were distributed with 21 completed by metropolitan (n = 11), regional (n = 8), and rural/remote (n = 2) settings. Substantial within and across site variance was observed, with care processes focused towards highly individualized, dietitian delivered care. Some early adopter sites demonstrated systematic, interdisciplinary or delegated malnutrition care processes; however, the latter was rarely or never undertaken in eight sites. A model for the Systematised,

Interdisciplinary Malnutrition Pathway for impLementation and Evaluation (SIMPLE) in hospitals was drafted based on identified contemporary models and supporting literature. A mixed-methods approach combined survey data with structured interviews conducted in six sites, purposely sampled for maximal variation to iteratively refine the model. Consensus for implementation of the final model was achieved across site clinicians, leaders, and governance structures. Systematised, delegated, and interdisciplinary nutrition care activities are realistic in at least some settings. A model is now available to provide interdisciplinary care. Next steps including testing implementation will determine if this interdisciplinary model improves malnutrition care delivered in hospitals.

#### REFERLINK

:<https://pubmed.ncbi.nlm.nih.gov/29436107/>

2. Orthogeriatric care for the elderly with hip fractures: where are we? AUTHOR :

Giulio Pioli 1, Andrea Giusti, Antonella Barone ABSTRACT

:

Hip fracture (HF) is a major health care problem in the Western world, associated with significant morbidity, mortality and loss of function. Its incidence is expected to increase as the population ages. The authors discuss the role of a coordinated multidisciplinary team in the management of patients during hospital stay, at discharge and during rehabilitation. Orthogeriatric care should not just be viewed as a multidisciplinary activity, but as a radical alternative to the traditional model of care, an alternative based on all those strategies in which evidence shows an improvement in outcomes in the fractured elderly. Therefore, key points of the care are early surgery, immediate mobilization, prevention and management of delirium, pain and malnutrition, as well as an integrated and multidisciplinary approach. Comprehensive geriatric assessment is useful in identifying frail elderly and in providing information that is essential in formulating clinical recommendations and making care plans. In each hospital, the orthogeriatric unit should represent a center of excellence for treating elderly patients with major fractures. However, when an orthogeriatric project is implemented, it is essential that detailed data about the casemix of patients, process of care and outcomes are collected, to compare the results with historical data and to be able to participate in audit processes. REFERLINK :<https://pubmed.ncbi.nlm.nih.gov/18431078/>

## **2.3 PROBLEM STATEMENT**

A variety of medical problems can affect your appetite. Your illness, medicines or surgery can cause these problems. Many people become frustrated when they know they need to eat to get well but they aren't hungry, or when they gain weight because they are fatigued and unable to exercise. Each of the following sections describes a nutritional problem and suggests possible solutions. Not all solutions will work for everyone.

### **SOLUTIONS:**

#### **Meal guidelines:**

- Drink beverages after a meal instead of before or during a meal so you do not feel as full.
- Plan meals to include your favorite foods.
- Try eating the high-calorie foods in your meal first.
- Use your imagination to increase the variety of food you're eating.
- Take advantage of times of the day when your appetite is best. For example, some people have a better appetite in the morning and can eat a larger breakfast.

#### **Snack guidelines:**

- Don't waste your energy eating foods that provide little or no nutritional value (such as potato chips, candy bars, colas and other snack foods).
- Choose high-protein and high-calorie snacks.



**Dining guidelines:**

- Make food preparation an easy task. Choose foods that are easy to prepare and eat.
- Make eating a pleasurable experience, not a chore.
  - Liven up your meals by using colorful place settings.
  - Play background music during meals.
- Eat with others. Invite a guest to share your meal or go out to dinner.
- Use colorful garnishes such as parsley and red or yellow peppers to make food look more appealing and appetizing.

**Weight loss:**

If your doctor tells you that you have lost too much weight, or if you are having difficulty maintaining a healthy weight, here are some tips:

- Drink milk or try one of the "high-calorie recipes" listed below instead of drinking low-calorie beverages.
- Avoid low-fat or low-calorie products unless you have been given other dietary guidelines. Use whole milk, whole milk cheese and yogurt.

Use the "Calorie Boosters" listed below to add calories to your favorite foods.

**High-calorie snacks:** • Ice cream.

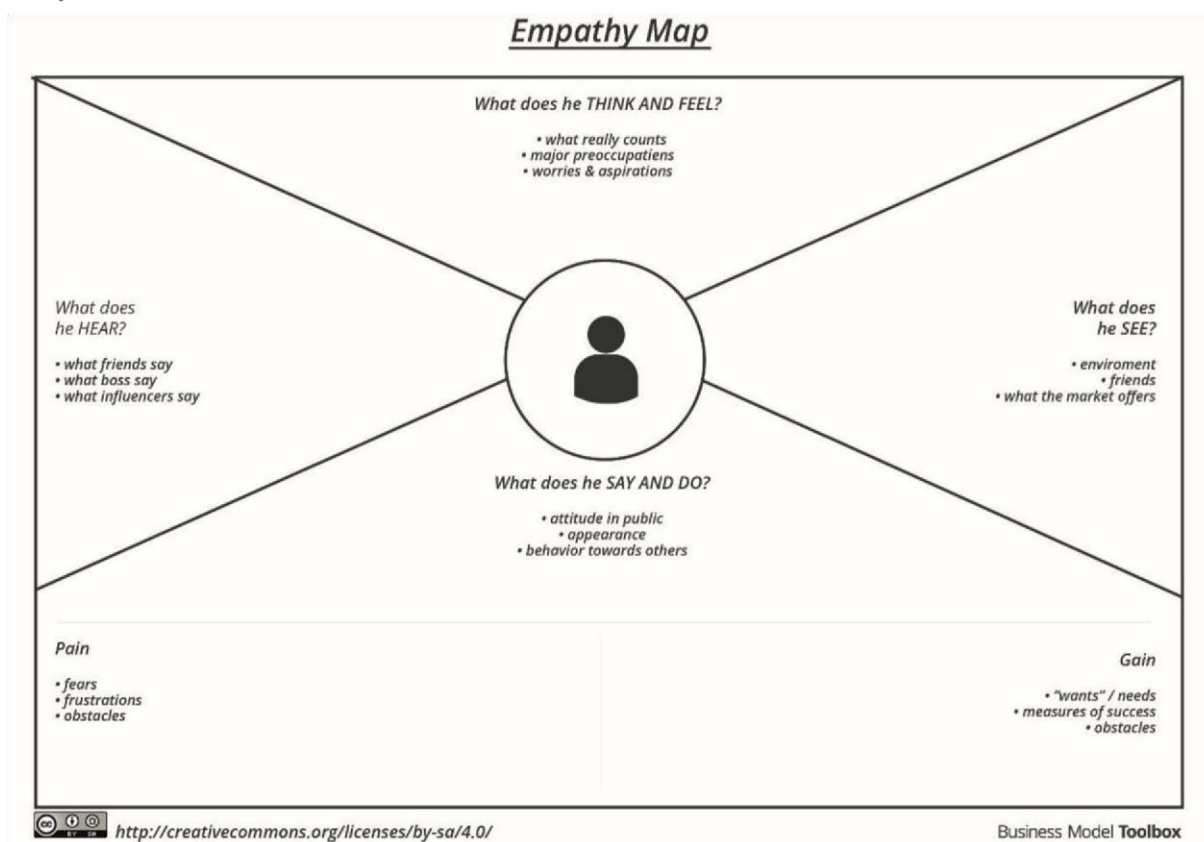
- Cookies.
- Pudding.
- Cheese.
- Granola bars.
- Custard.
- Sandwiches.
- Nachos with cheese.
- Eggs.
- Crackers with peanut butter.
- Bagels with peanut butter or cream cheese.
- Cereal with half and half.
- Fruit or vegetables with dips.
- Yogurt with granola.
- Popcorn with margarine and parmesan cheese.
- Bread sticks with cheese sauce.

### 3. IDEATION AND PROPOSED SOLUTION

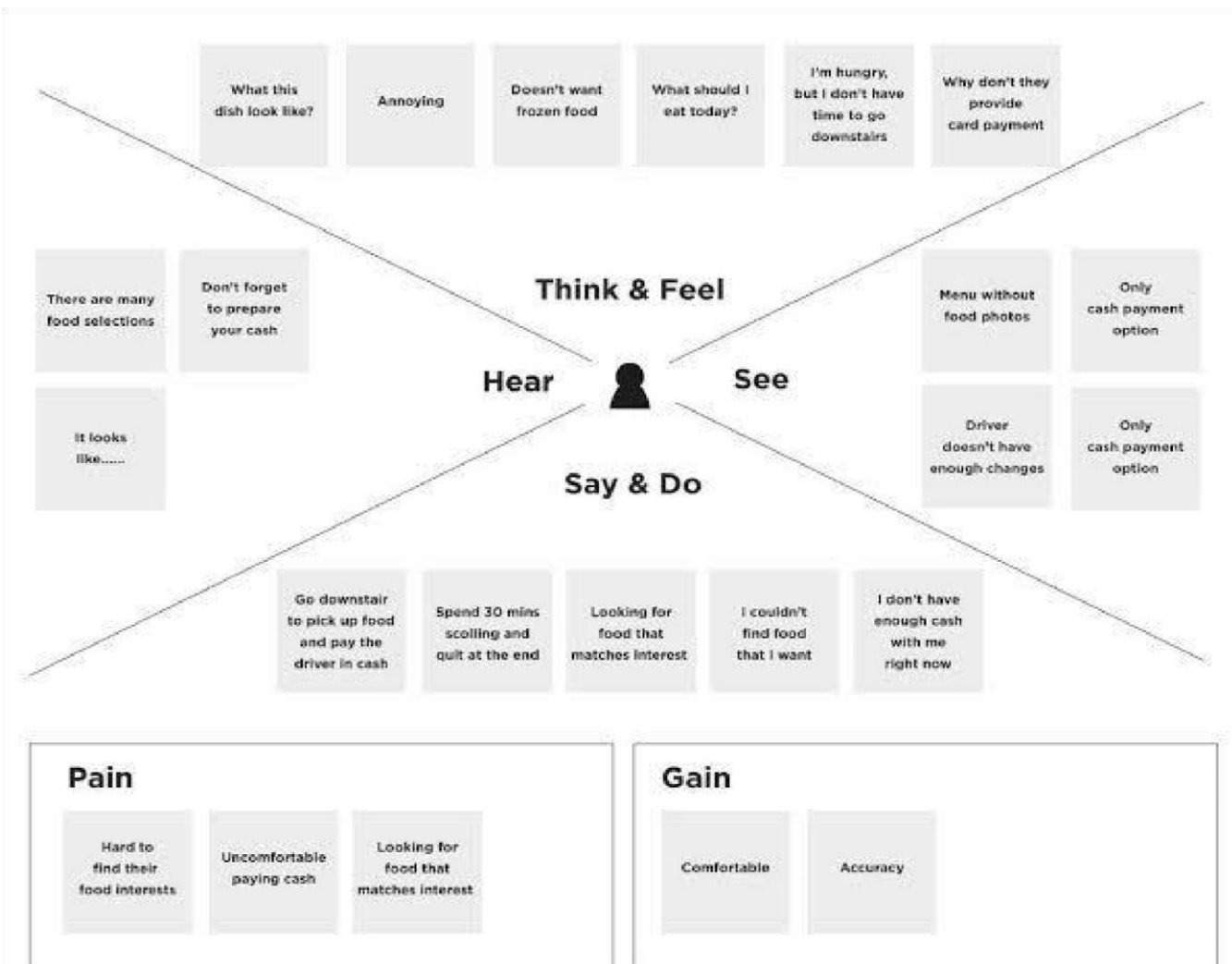
#### 3.1 Empathy Map Canvas :

- An empathy map is a simple, easy-to-digest visual that captures knowledge about a user's behaviours and attitudes.
- It is a useful tool to help teams better understand their users.
- Creating an effective solution requires understanding the true problem and the person who is experiencing it.
- The exercise of creating the map helps participants consider things from the user's perspective along with his or her goals and challenges.

Example:



**Empathy Map for Nutrition Assistant application:**



### 3.2 Ideation and Proposed Solution:







### 3.3 Proposed Solution :

#### Proposed Solution Template:

The project aims at developing an application that helps people to lead a healthy lifestyle by providing information about the ingredients and their nutritional content in the food they are consuming. By this people can avoid various health-related issues like obesity, heart attack, diabetics etc. Monitoring and tracking of goal and diet plans will be provided for the user based on the data collected from them.

S. No.	Parameter	Description
1.	Problem Statement (Problem to be solved)	<ul style="list-style-type: none"><li>• It is easy to fall into a trap of eating unhealthy foods which is heavy in calories. Once the nutritional value is replaced by foods high in sugar, bad fats and salt it leads to various health issues so users need to control their daily calorie intake to lead a healthy lifestyle.</li></ul>
2.	Idea / Solution description	<ul style="list-style-type: none"><li>• The solution is a responsive Web application that can be used in any PC devices.</li><li>• The website provides a userfriendly interface and accepts multiple samples predicting them simultaneously.</li><li>• Our method uses Clarifai's AI- driven food recognition model to accurately identify food suggestions.</li><li>• A detailed report of the concerned person's health will be generated.</li></ul>
3.	Novelty / Uniqueness	<ul style="list-style-type: none"><li>• Keep a food journal.</li><li>• Providing individual diet charts for users based on their BMI and medical condition if any.</li><li>• Provides recipes according to their diet.</li><li>• Providing a user-friendly environment.</li></ul>
4.	Social Impact / Customer Satisfaction	<ul style="list-style-type: none"><li>• Getting feedback from the users for enhancement and giving notification on their diet plans and goal tracking.</li><li>• Nutrition focused food banking &amp; targeted in-depth reporting reviews that</li></ul>







		paid subscriptions the best.
5.	Business Model (Revenue Model)	<ul style="list-style-type: none"> <li>• Advertising membership option for users to get more benefits like diet- plans or consultation from experts and In-app advertisements.</li> <li>• Revenue is generated on a subscription basis, with big data processing and targeted in-depth reporting reviews that paid subscriptions the best.</li> </ul>
6.	Scalability of the Solution	<ul style="list-style-type: none"> <li>• Providing regular updates</li> <li>• Efficient goal tracking assistance •</li> <li>• The additional features such that sleep tracking, mensuration tracking can be done.</li> </ul>





# 3.4 Problem Solution Fit :

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

ONE /

## 2. JOBS-TO- J&P

BE-

**PROBLEMS**Which jobs -to -be- done (or problems) do you address for your customers? There could be more than one; explore different sides.

- 1.To calculate calories and
- nutrients present.
- 2.
- M

## 9. PROBLEM ROOT CAUSE

RC

What is the real reason that this problem exists?  
What is the back story behind the need to do this job?  
i.e. customers have to do it because of the change in

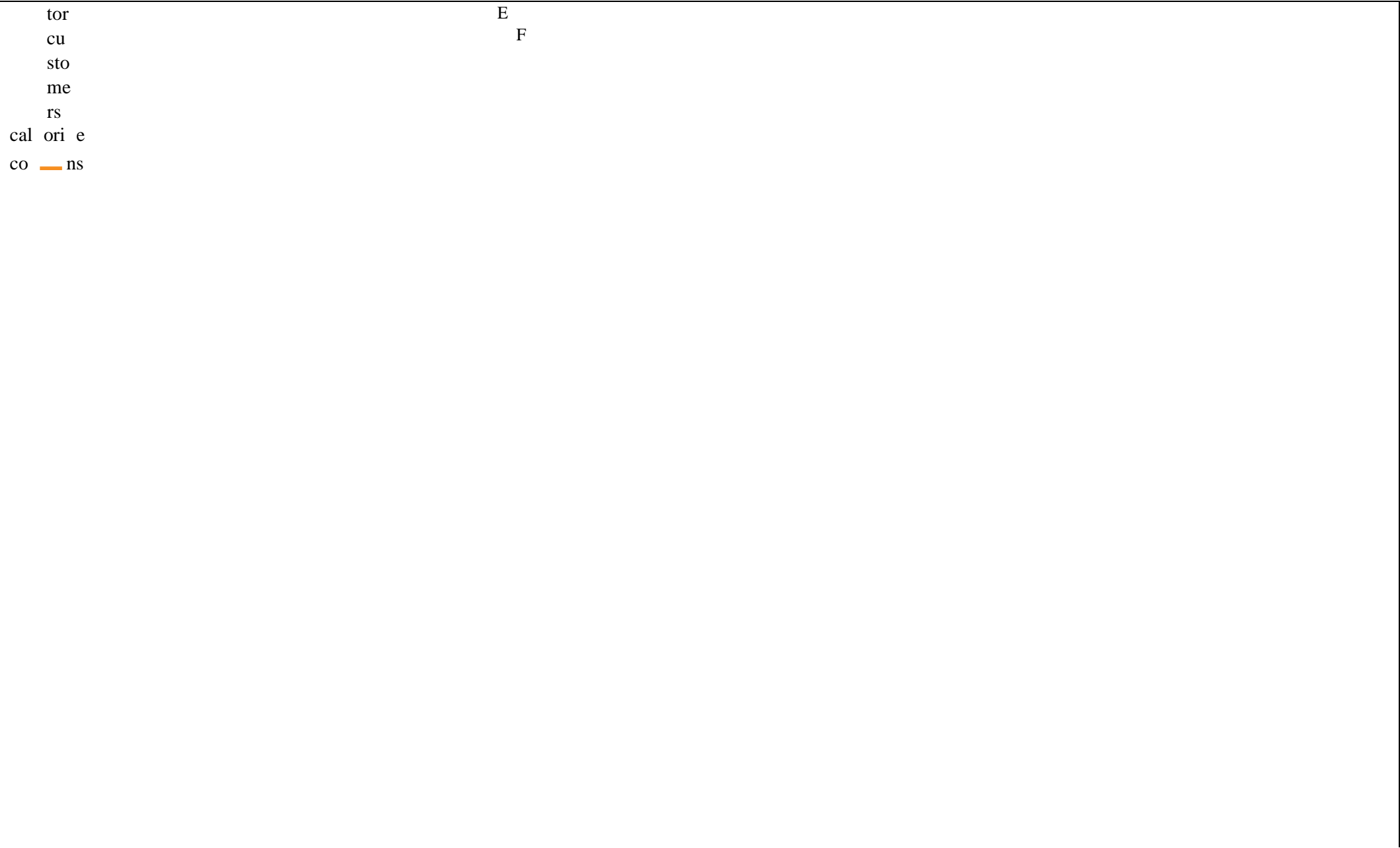
- 1. Due to shortage of time, preparation of healthy home food is replaced by consuming unhealthy fast food.
- 2. Teenagers are addicted to fast food which leads to obesity

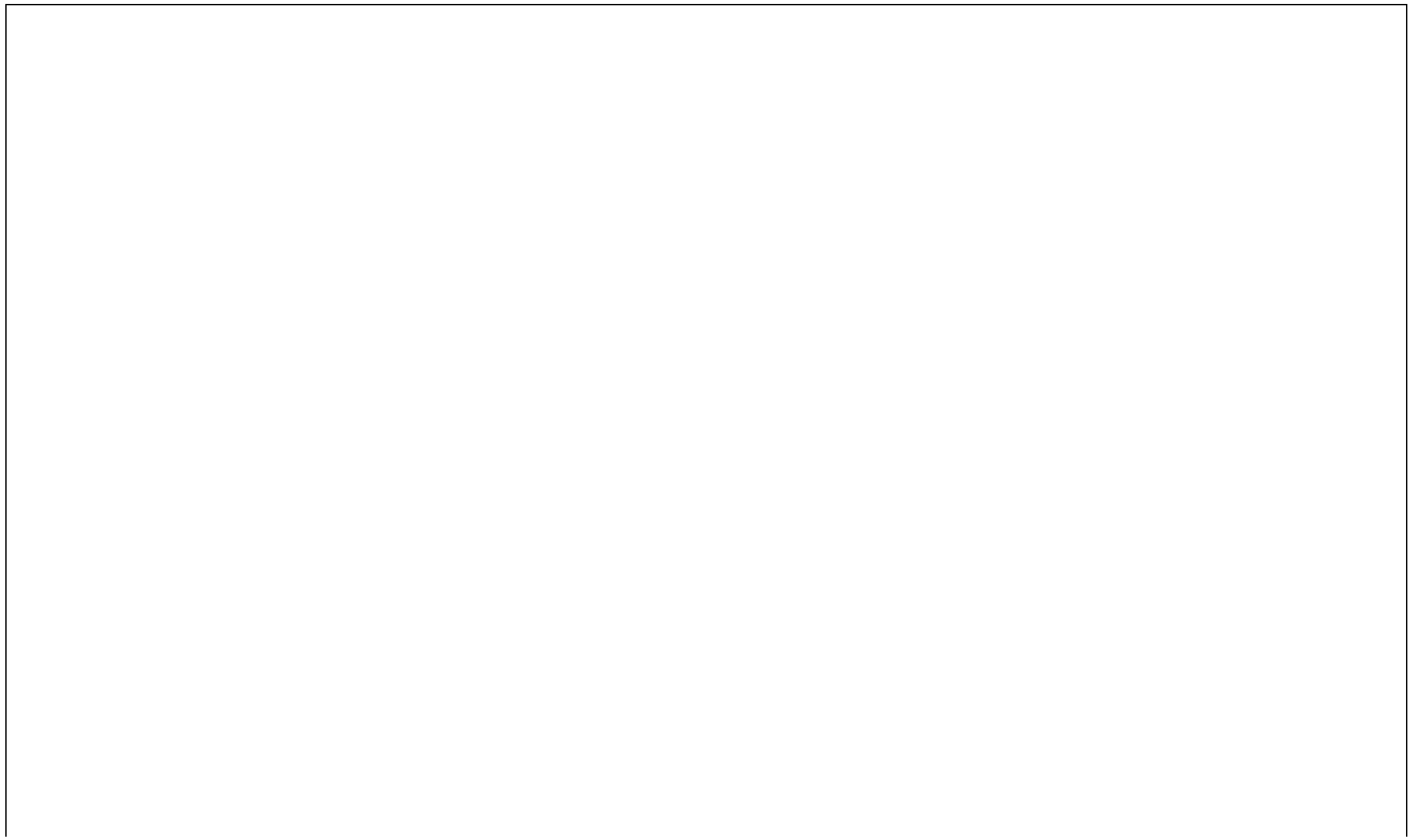
--	--











<b>6. CUSTOMER CONSTRAINTS</b>	
What constraints prevent your customers from taking action or limit the effectiveness of their action?	

of solutions? i.e. spending power, budget, no cash, network connected devices.

## Workin SOLUTION

## Workin solution

g out or available to th

healthy <sup>ma</sup>way.

Everyone int

who feels to

stay

ain fit and healthy

by

die t

consuming

nutritious food

and following

calorie conscious

diet.

physica<sup>notetaking</sup>

l fitness

1.

What <sup>pros</sup>**involve**  
& cons do these solutions have? i.e.

pen **S** and paper is an alternative to

digital

**Pers onal**  
**diet trac**



kin  
g  
app  
whi  
ch  
help  
s to  
mai  
ntai  
n  
diet.  
2. Pers  
onal  
nutr  
itio  
nist  
or  
trai  
ner

d e n t i f y

to  
sug  
gest  
corr  
ect  
sch  
edul  
e  
acc  
ordi  
ng  
to  
cust  
ome  
r  
req  
uire  
men  
t.





### 3. TRIGGERS

TR

What triggers customers to act? i.e. seeing their neighbour installing solar panels, reading about a more efficient solution in the news.

1. When people around us bully.
2. Peer pressure, beauty standards, society point of view etc.,
3. When obesity and consumption of unhealthy foods leads to health issues

### 4. EMOTIONS: BEFORE / AFTER

EM

How do customers feel when they face a problem or a job and afterwards?

i.e. lost, insecure > confident, in control - use it in your communication strategy & design.

They scared of declining health, so they get motivated towards eating healthy foods and move to healthy lifestyle.

### 10. YOUR SOLUTION

SL

If you are working on an existing business, write down your current solution first, fill in the canvas, and check how much it fits reality.

If you are working on a new business proposition, then keep it blank until you fill in the canvas and come up with a solution that fits within customer limitations, solves a problem and matches customer behaviour.

1. Follow the correct diet plan and consume suggested calories per day.
2. Try to involve yourself in physical fitness like sports, gym, yoga etc.,

### 8. CHANNELS of BEHAVIOUR

CH

#### 8.1 ONLINE

What kind of actions do customers take online? Extract online channels from #7

1. follow people who give healthy and nutritious food recipes.
2. Keep track of fitness freaks in social media and follow their fitness tips

#### 8.2 OFFLINE

What kind of actions do customers take offline? Extract offline channels from #7 and use them for customer development.

Notice people around you who follows healthy habits in both consumption of food and workouts.





## **4. REQUIREMENT ANALYSIS**

### **4.1 FUNCTIONAL REQUIREMENT**

- Introduction
- Login
- Upload Image
- Submit Image
- Recognize Image
- Report

### **4.2 NON FUNCTIONAL REQUIREMENTS**

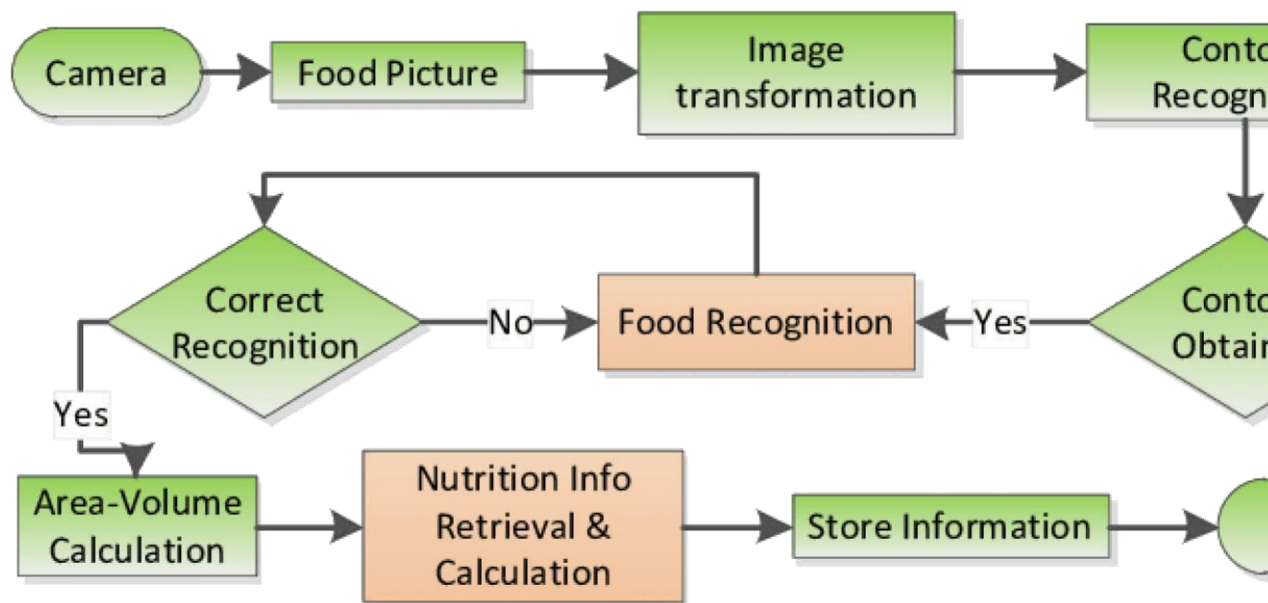
- Usability
- Security
- Reliability
- Performance
- Availability
- Scalability



## 5. PROJECT DESIGN

### 5.1 Data Flow Diagrams:

A Data Flow Diagram (DFD) is a traditional visual representation of the information flows within a system. It shows how data enters and leaves the system, what data is stored.



## 5.2 User Stories

Use the below template to list all the user stories for the product.

User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance
Customer	Registration	USN-1	As a user, I can register for the application by entering my name, age, gender, e-mail, password and confirming my password	I can access dashboard
	Registration	USN-2	As a user, I will receive confirmation email once I have registered for the application	I can receive email & click
	Profile Updating	USN-3	As a user, I have to enter my height, weight and daily activity details	I can information o
	Login	USN-4	As a user, I can login to the application through Gmail with login credentials	I can access dashboard
	Database	USN-5	As a user, I can upload or capture live image of the meal	I can get the value of that meal
	Dashboard	USN-6	As a user, I can track my daily calories intake	I can access dashboard

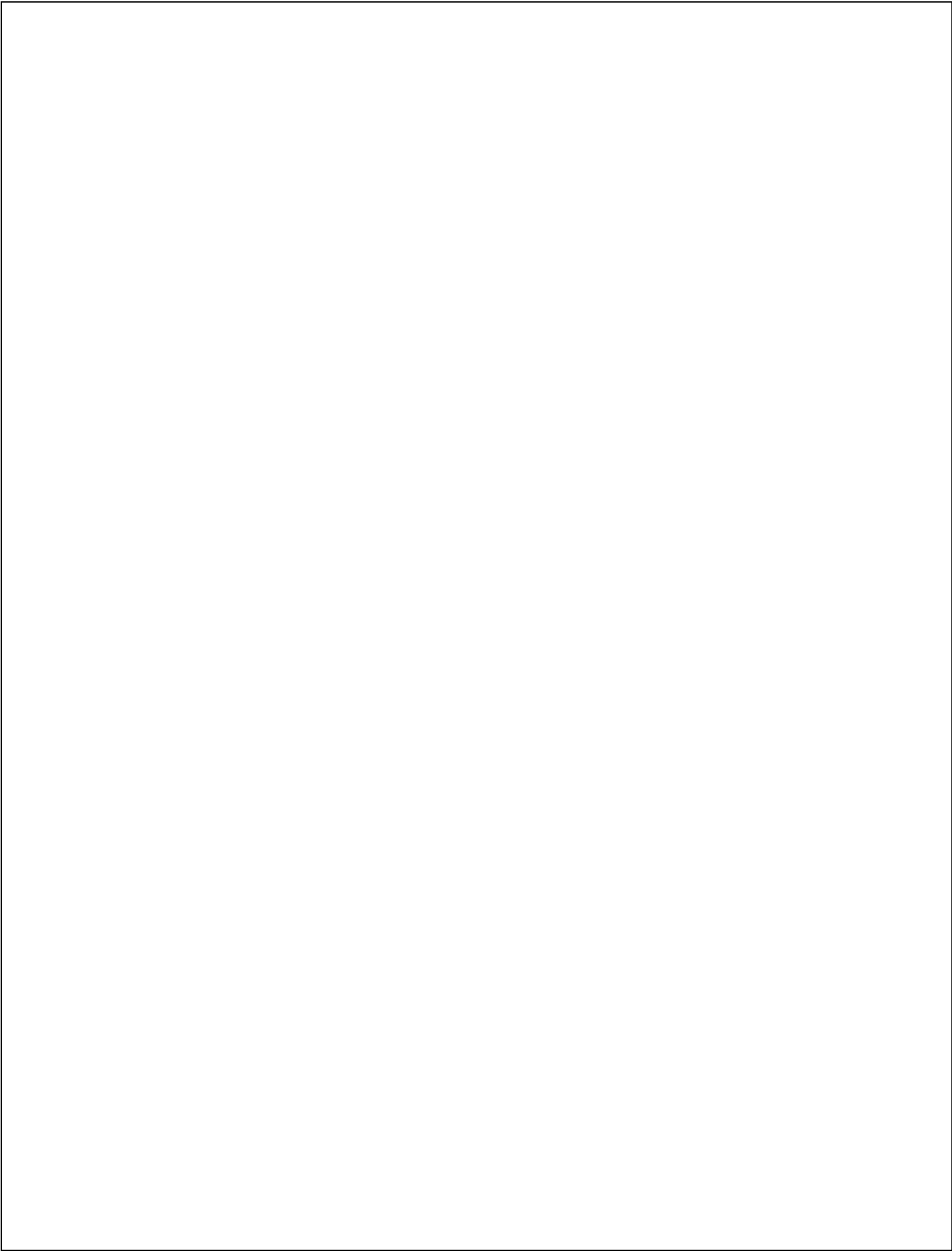


Administrator	Maintaining details for users	USN-7	Maintaining details for users	I can access
	Security	USN-8	As a user, I feel the site is very secure	I can access n with my login o

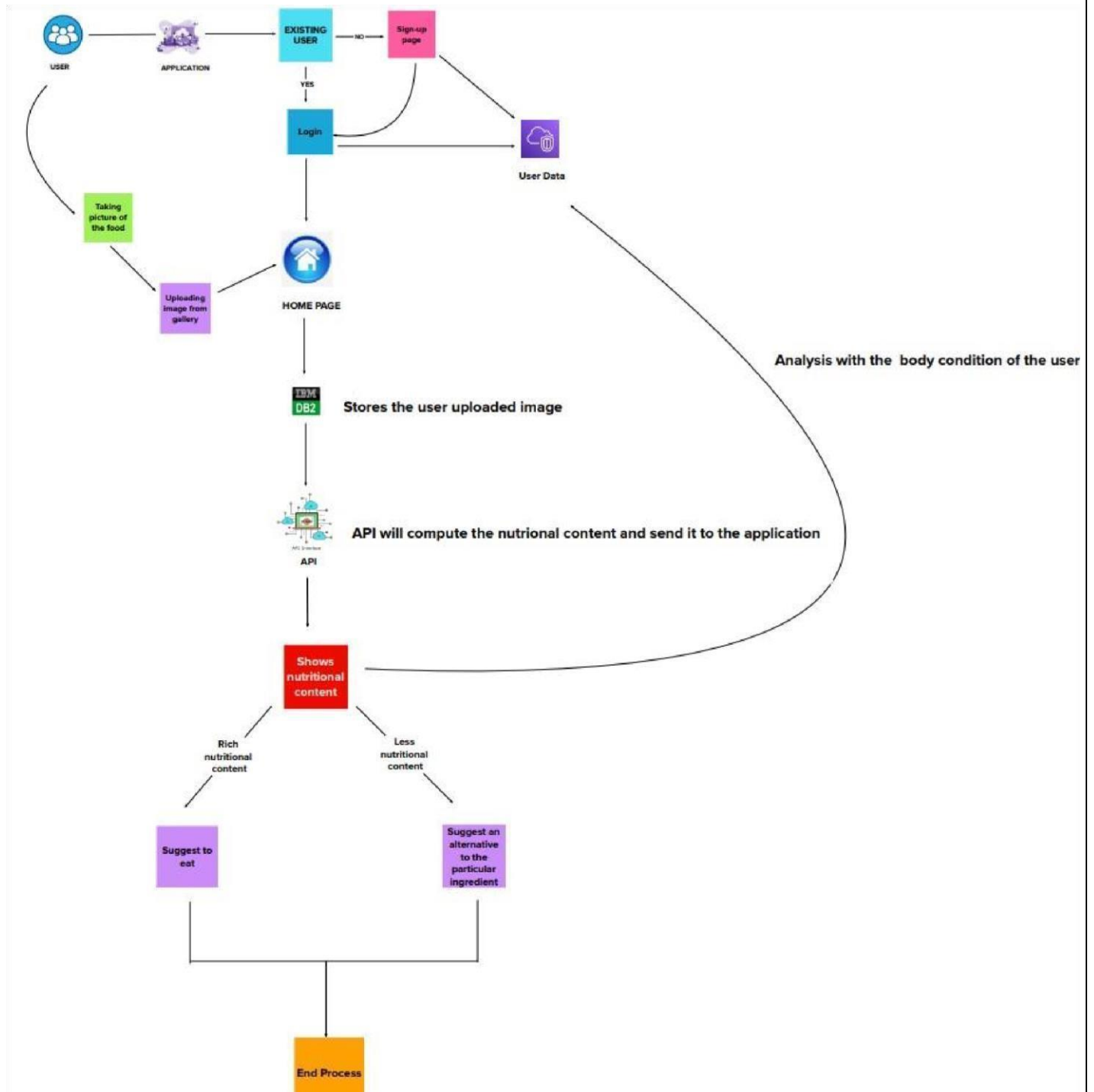
### 5.3 SOLUTION AND TECHNICAL ARCHITECTURE

Due to the ignorance of healthy food habits, obesity rates are increasing at an alarming speed, and this is reflective of the risks to people's health. People need to control their daily calorie intake by eating healthier foods, which is the most basic method to avoid obesity. However, although food packaging comes with nutrition (and calorie) labels, it's still not very convenient for people to refer to App-based nutrient dashboard systems which can analyze real-time images of a meal and analyze it for nutritional content which can be very handy and improves the dietary habits, and therefore, helps in maintaining a healthy lifestyle.

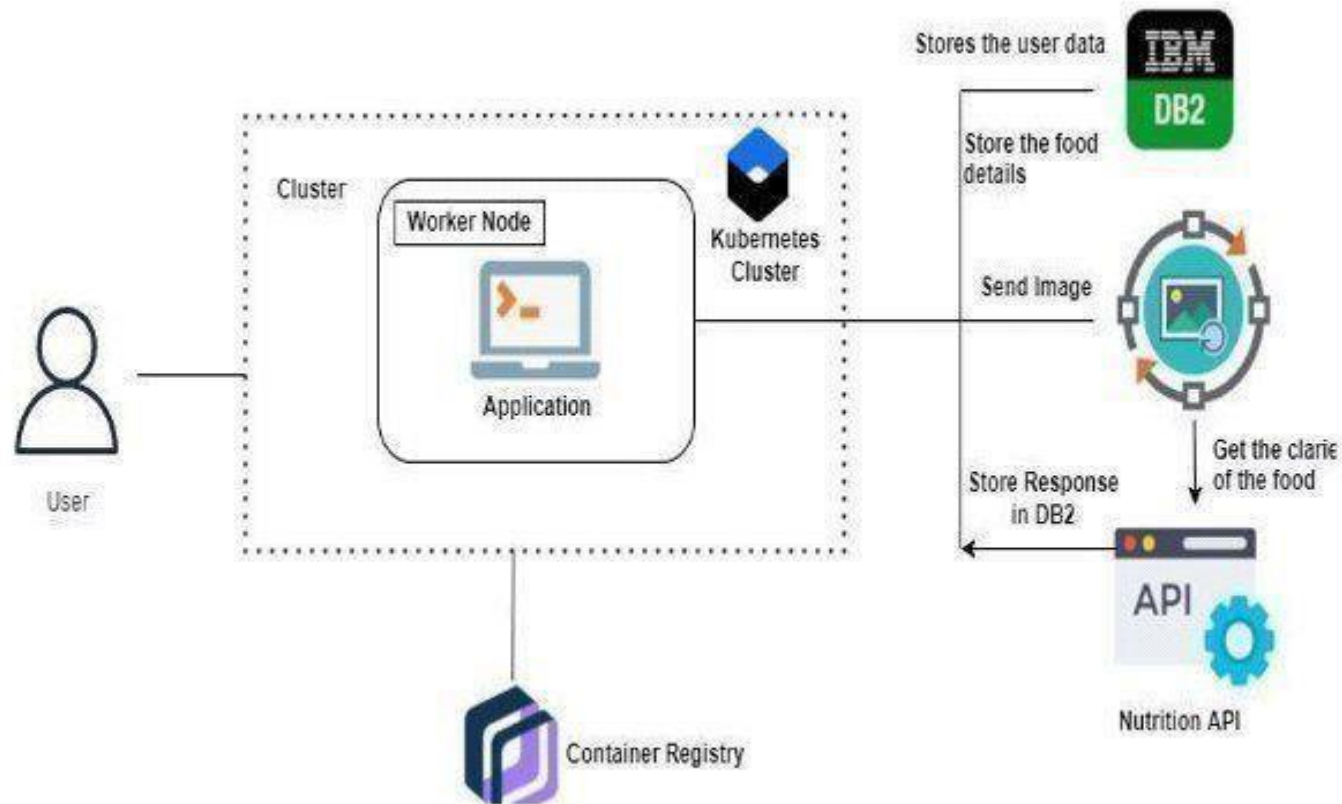
This project aims at building a web App that automatically estimates food attributes such as ingredients and nutritional value by classifying the input image of food. Our method employs **Clarifai's AI-Driven Food Detection Model** for accurate food identification and Food API's to give the nutritional value of the identified food.



## Solution Architecture Diagram.



**TECHNICAL ARCHITECTURE**









## 6. PROJECT PLANNING AND SCHEDULING

### 6.1 Sprint Planning And Estimation

TITLE	DESCRIPTION	DATE
Literature Survey & Information Gathering	Literature survey on the selected project & gathering information by referring the, technical papers, research publications etc.	07 NOVEMBER 2022
Prepare Empathy Map	Prepare Empathy Map Canvas to capture the user Pains & Gains, Prepare list of problem statements	07 NOVEMBER 2022
Ideation	List the by organizing the brainstorming session and prioritize the top 3 ideas based on the feasibility & importance.	07 NOVEMBER 2022
Proposed Solution	Prepare the proposed solution document, which includes the novelty, feasibility of idea, business model, social impact, scalability of solution, etc.	07 NOVEMBER 2022
Problem Solution Fit	Prepare problem - solution fit document.	07 NOVEMBER 2022

<b>Solution Architecture</b>	Prepare solution architecture document.	01 SEPTEMBER 2022
------------------------------	---	-------------------

<b>Customer Journey</b>	Prepare the customer journey maps to understand the user interactions & experiences with the application (entry to exit).	16 OCTOBER 2022
<b>Functional Requirement</b>	Prepare the functional requirement document.	16 OCTOBER 2022
<b>Data Flow Diagrams</b>	Draw the data flow diagrams and submit for review.	16 OCTOBER 2022
<b>Technology Architecture</b>	Prepare the technology architecture diagram.	16 OCTOBER 2022
<b>Prepare Milestone &amp; Activity List</b>	Prepare the milestones & activity list of the project.	23 OCTOBER 2022
<b>Project Development - Delivery of Sprint-1, 2, 3 &amp; 4</b>	Develop & submit the developed code by testing it.	IN PROGRESS...

## 6.2 SPRINT DELIVERY SCHEDULE

Use the below template to create product backlog and sprint schedule

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Member s	
Sprint-1	Setting Up Application Environment	USN-1	To create lots of environment. Create or Enrolment to the IBM cloud, Docker CLI installation, create an account in SendGrid and Nutrition API, etc.,	20	High	SRITHAR HARIPRASAD DHILIPKUMAR KAMALESH	n
Sprint-2	Implementing Web Application	USN-2	We create a UI to interact with application. Create database system DB2 and connect it with python and integrate with Nutrition API.	20	High	SRITHAR HARIPRASAD DHILIPKUMAR KAMALESH	n
Sprint-3	Integrating SendGrid Service	USN-3	SendGrid integration with python code for include some RestAPI services for to give a Nutrition and calorie value.	20	High	SRITHAR HARIPRASAD DHILIPKUMAR KAMALESH	n
Sprint-4	Deployment of App in IBM Cloud	USN-4	In the deploy process, the deployment in Kubernetes cluster is the major task before that we need to containerize the app and upload image to IBM container Registry	20	High	SRITHAR HARIPRASAD DHILIPKUMAR KAMALESH	n

### Project Tracker, Velocity & Burndown Chart: (4 Marks)

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		05 Nov 2022	
Sprint-3	20	6 Days	07 Nov 2022	12 Nov 2022		12 Nov 2022	
Sprint-4	20	6 Days	14 Nov 2022	19 Nov 2022		19 Nov 2022	

**Velocity:**

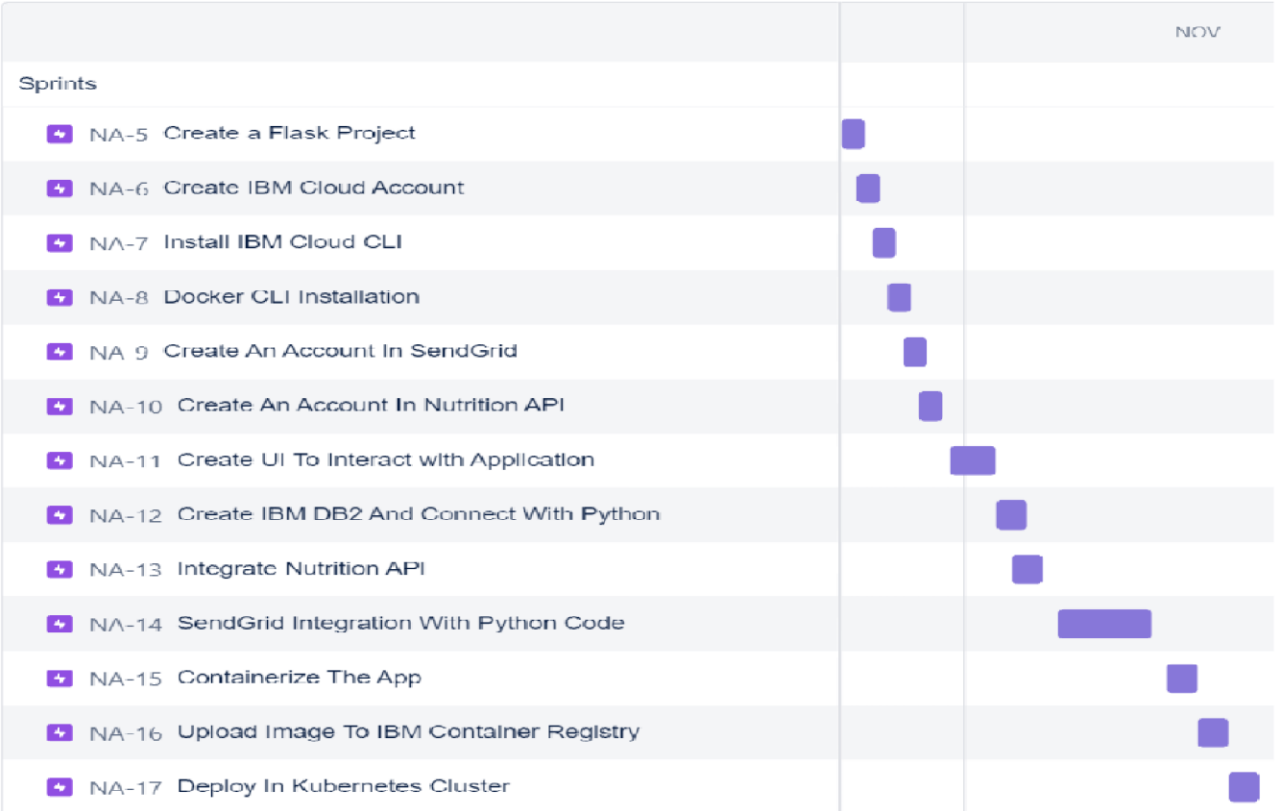
Imagine we have a 10-day sprint duration, and the velocity of the team is 20 (points per sprint).

Let's calculate the team's average velocity (AV) per iteration unit (story points per day)

$$AV = \frac{\text{sprint duration}}{\text{velocity}} = \frac{20}{10} = 2$$

Burndown Chart:

A burn down chart is a graphical representation of work left to do versus time. It is often used in agile software development methodologies such as Scrum. However, burn down charts can be applied to any project containing measurable progress over time.



## 7. CODING AND SOLUTIONING

### 7.1 Feacture 1

```
<!DOCTYPE HTML>
```

```
<html>
```

```
  <head>
```

```
    <title>Nutrition Viusalization System</title>
```

```
    <meta charset="utf-8">
```

```
    <meta name="viewport" content="width=device-width, initialscale=1,
  userscalable=no">
```

```
    <link rel="stylesheet" href="{ { url_for('static', filename='css/main.css')
  }}">
```

```
    <link rel="stylesheet" href="{ { url_for('static',
  filename='css/noscript.css') }}">
```

```
  </head>
```

```
  <body class="is-preload">
```

```
    <!-- Wrapper -->
```

```
      <div id="wrapper">
```

```
        <!-- Header -->
```

```
          <header id="header">
```

```
            <div class="logo">
```

```
              <span class="icon fagem"></span>
```

```
            </div>
```

```
            <div class="content">
```

```
              <div class="inner">
```

```
                <h1>Food Image
```

```
                Recognition And Nutrition Visualization System</h1>
```

<p>In this Project, An user-  
interactive system that takes food images as an input,  
recognizes the food  
automatically and gives the nutritional-facts as  
an output will save a  
lot of time.</p>

</div>

</div>

<nav>

<ul>

<li><a

href="/test">LOGIN</a></li>

<li><a

href="#model">INTRO</a></li>

</ul>

</nav>

</header>

<!-- Main -->

<div id="main">

<!-- Work -->

<article id="model">

<h2 class="major">Work OF Project

Model</h2>

<span class="image main">

</span>

<p> Each year, approximately 6,78,000 deaths are caused in the world due to unhealthy diet. A typical American diet is too high in calories, fat, sugars, sodium, etc. Hence, people have become more proactive when it comes to health matters. Services like eating habit recorder and calorie/nutrition calculator have become extremely popular. They can make users aware of problems like obesity, cancer, diabetes, heart-disease, etc. that can be caused by unhealthy diets. Most of these services require the users to manually select a food item from a hierarchical menu which is a time consuming process and not so user friendly.</p>

</article>

</div>

<!-- Footer -->

<footer id="footer">

<p>Amrita\_College\_Of\_Engineering\_And\_Technology</p>

</footer>



```

        </div>

<!-- BG -->
        <div id="bg"></div>

<!-- Scripts -->
        <script type=text/javascript src="{{ url_for('static',
filename='js/jquery.min.js') }}"></script>

        <script type=text/javascript src="{{ url_for('static',
filename='js/browser.min.js') }}"></script>

        <script type=text/javascript src="{{ url_for('static',
filename='js/breakpoints.min.js') }}"></script>

        <script type=text/javascript src="{{ url_for('static',
filename='js/util.js') }}"></script>

        <script type=text/javascript src="{{ url_for('static', filename='js/main.js')
        }}"></script>

</body>
</html>

```

## 7.2 FEATURE 2

### App.py

```

import tensorflow  from flask import Flask, request, render_template import  csv
import math import os import numpy as np from tensorflow.keras.preprocessing
import image from tensorflow.python.keras.models import load_model from
werkzeug.utils import secure_filename

```

```
tmpl_dir = os.path.join(os.path.dirname(os.path.abspath(__file__)), 'templates') app
= Flask(__name__, template_folder=tmpl_dir)
```

```
UPLOAD_FOLDER = 'uploads' app.config['uploads']
= UPLOAD_FOLDER
```

```
# define label meaning
```

```
label = ['apple pie',
'baby back ribs',
    'baklava',
    'beef carpaccio',
    'beef tartare',
    'beet salad',
    'beignets',
    'bibimbap',
    'bread pudding',
    'breakfast burrito',
    'bruschetta',
    'caesar salad',
    'cannoli',
    'caprese salad',
    'carrot cake',
    'ceviche',
    'cheese plate',
    'cheesecake',
    'chicken curry',
    'chicken quesadilla',
    'chicken wings',
```

'chocolate cake',  
'chocolate mousse',  
'churros',  
'clam chowder',  
'club sandwich',  
'crab cakes',  
'creme brulee',  
'croque madame',  
'cup cakes',  
'deviled eggs',  
'donuts',  
'dumplings',  
'edamame',  
'eggs benedict',  
'escargots',  
'falafel',  
'filet mignon',  
'fish and\_chips',  
'foie gras',  
'french fries',  
'french onion soup',  
'french toast',  
'fried calamari',  
'fried rice',  
'frozen yogurt', 'garlic  
bread',  
'gnocchi',  
'greek salad',

'grilled cheese sandwich',  
'grilled salmon',  
'guacamole',  
'gyoza',  
'hamburger',  
'hot and sour soup',  
'hot dog',  
'huevos rancheros',  
'hummus',  
'ice cream',  
'lasagna',  
'lobster bisque',  
'lobster roll sandwich',  
'macaroni and cheese',  
'macarons',  
'miso soup', 'mussels',  
'nachos',  
'omelette',  
'onion rings',  
'oysters',  
'pad thai',  
'paella',  
'pancakes',  
'panna cotta',  
'peking duck',  
'pho',  
'pizza',  
'pork chop',

```
'poutine',
'prime rib',
'pulled pork sandwich',
'ramen',
'ravioli',
'red velvet cake',
'risotto',
'samosa',
'sashimi',
'scallops',
'seaweed salad',
'shrimp and grits',
'spaghetti bolognese',
'spaghetti carbonara', 'spring rolls',
'steak',
'strawberry shortcake',
'sushi',
'tacos',
'octopus balls',
'tiramisu',
'tuna tartare',
'waffles']
```

```
nu_link = 'https://www.nutritionix.com/food/'
```

```
# Loading the best saved model to make predictions.
tensorflow.keras.backend.clear_session() model_best =
load_model('best_model_101class.h5') print('model successfully loaded!')
```

```
start = [0] passed =
```

```
[0] pack = [[]]
```

```
num
```

```
= [0]
```

```
nutrients = [
```

```
    {'name': 'protein', 'value': 0.0},
```

```
    {'name': 'calcium', 'value': 0.0},
```

```
    {'name': 'fat', 'value': 0.0},
```

```
    {'name': 'carbohydrates', 'value': 0.0},
```

```
    {'name': 'vitamins', 'value': 0.0}
```

```
]
```

```
with open('nutrition101.csv', 'r') as file:
```

```
    reader = csv.reader(file)
```

```
nutrition_table = dict()    for i,
```

```
row in enumerate(reader):    if i
```

```
== 0:        name = "
```

```
continue    else:
```

```
        name = row[1].strip()    nutrition_table[name]
```

```
= [        {'name': 'protein',
```

```
'value': float(row[2])},
```

```
        {'name': 'calcium', 'value': float(row[3])},
```

```
        {'name': 'fat', 'value': float(row[4])},
```

```
        {'name': 'carbohydrates', 'value': float(row[5])},    {'name': 'vitamins',  
'value': float(row[6])}
```

```
]
```

```
@app.route('/') def index():    img = 'static/profile.jpg'    return  
render_template('home_new.html', img=img)
```

```
@app.route('/test') def test():  
    img = 'static/profile.jpg'    return render_template('index.html', img=img)
```

```
@app.route('/recognize') def magic():    return  
render_template('recognize.html', img=file)
```

```
@app.route('/upload', methods=['POST']) def upload():  
file = request.files.getlist("img")    print(file)    for f  
in file:  
    filename = secure_filename(str(num[0] + 500) + '.jpg')    num[0]  
+= 1    name =  
os.path.join(app.config['uploads'], filename)    print('save  
name', name)    f.save(name)
```

```
    pack[0] = []    return render_template('recognize.html',  
img=file) @app.route('/predict') def predict():  
    result = []    # pack = []  
print('total image', num[0])    for i  
in range(start[0], num[0]):  
    pa = dict()
```

```

        filename = f{"uploads"}/{i + 500}.jpg'    print('image filepath',
filename)    pred_img = filename    pred_img =
image.load_img(pred_img, target_size=(200, 200))    pred_img =
image.img_to_array(pred_img)    pred_img = np.expand_dims(pred_img,
axis=0)    pred_img = pred_img / 255.

```

```

        pred = model_best.predict(pred_img)    print("Pred")
print(pred)

```

```

        if math.isnan(pred[0][0]) and math.isnan(pred[0][1]) and \
math.isnan(pred[0][2]) and math.isnan(pred[0][3]):
            pred = np.array([0.05, 0.05, 0.05, 0.07, 0.09, 0.19, 0.55, 0.0, 0.0, 0.0,
0.0])

```

```

        top = pred.argsort()[0][-3:]    label.sort()
        _true = label[top[2]]    pa['image'] = f{"uploads"}/{i
+ 500}.jpg'    print(pa['image']
)
        x = dict()    x[_true] = float("{:.2f}".format(pred[0][top[2]] *
100))    x[label[top[1]]] = float("{:.2f}".format(pred[0][top[1]] *
100))    x[label[top[0]]] = float("{:.2f}".format(pred[0][top[0]] *
100))    pa['result'] = x    #print(x)    pa['nutrition'] =
nutrition_table[_true]    pa['food'] = f{nu_link}{_true}'    pa['idx']
= i - start[0]    pa['quantity'] = 100

```

```

        pack[0].append(pa)    passed[0]
+= 1

```



```

start[0] = passed[0]    print('successfully
packed')

# compute the average source of calories    for p
in pack[0]:

    nutrients[0]['value'] = (nutrients[0]['value'] +
p['nutrition'][0]['value'])    nutrients[1]['value'] = (nutrients[1]['value'] +
p['nutrition'][1]['value'])    nutrients[2]['value'] = (nutrients[2]['value'] +
p['nutrition'][2]['value'])    nutrients[3]['value'] = (nutrients[3]['value'] +
p['nutrition'][3]['value'])    nutrients[4]['value'] = (nutrients[4]['value'] +
p['nutrition'][4]['value'])    nutrients[0]['value'] = nutrients[0]['value'] /
num[0]    nutrients[1]['value'] = nutrients[1]['value'] / num[0]
nutrients[2]['value'] = nutrients[2]['value'] / num[0] nutrients[3]['value'] =
nutrients[3]['value'] / num[0]    nutrients[4]['value'] = nutrients[4]['value'] /
num[0]

return render_template('results.html', pack=pack[0], whole_nutrition=nutrients)

@app.route('/update', methods=['POST']) def update():
    return render_template('index.html', img='static/P2.jpg')

if __name__ == "__main__":
import click

    @click.command()
    @click.option('--debug', is_flag=True)
    @click.option('--threaded', is_flag=True)

```

```

@click.argument('HOST', default='127.0.0.1')
@click.argument('PORT', default=5000, type=int) def run(debug, threaded,
host, port):
    """
    This function handles command line parameters.
    Run the server using          python server.py
    Show the help text using      python server.py
    help
    """
    HOST, PORT = host, port      app.run(host=HOST, port=PORT,
debug=debug, threaded=threaded) run()

```

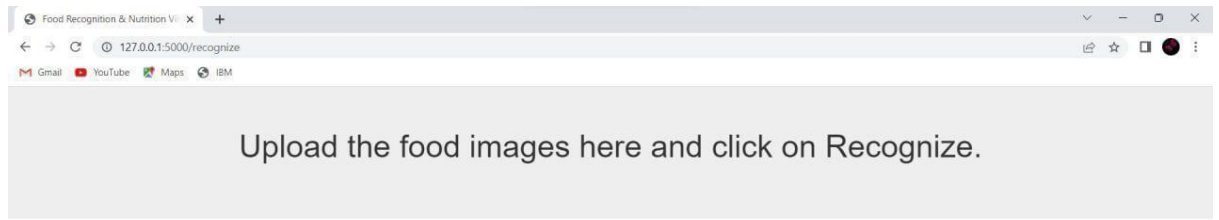
## 8. Testing

### 8.1 Test Cases

- 1.our code was tested on various food to check whether it gives the correct output.
- 2.To satisfy the cutomers expectations we tested fully.

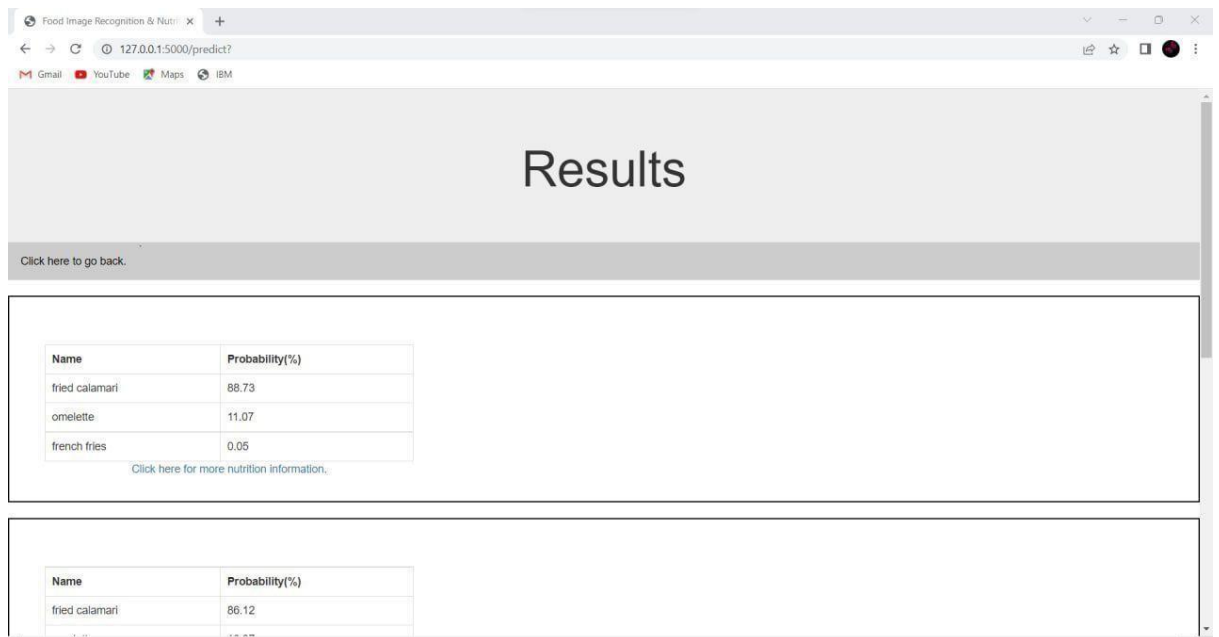
### 8.2 User Acceptance Testing

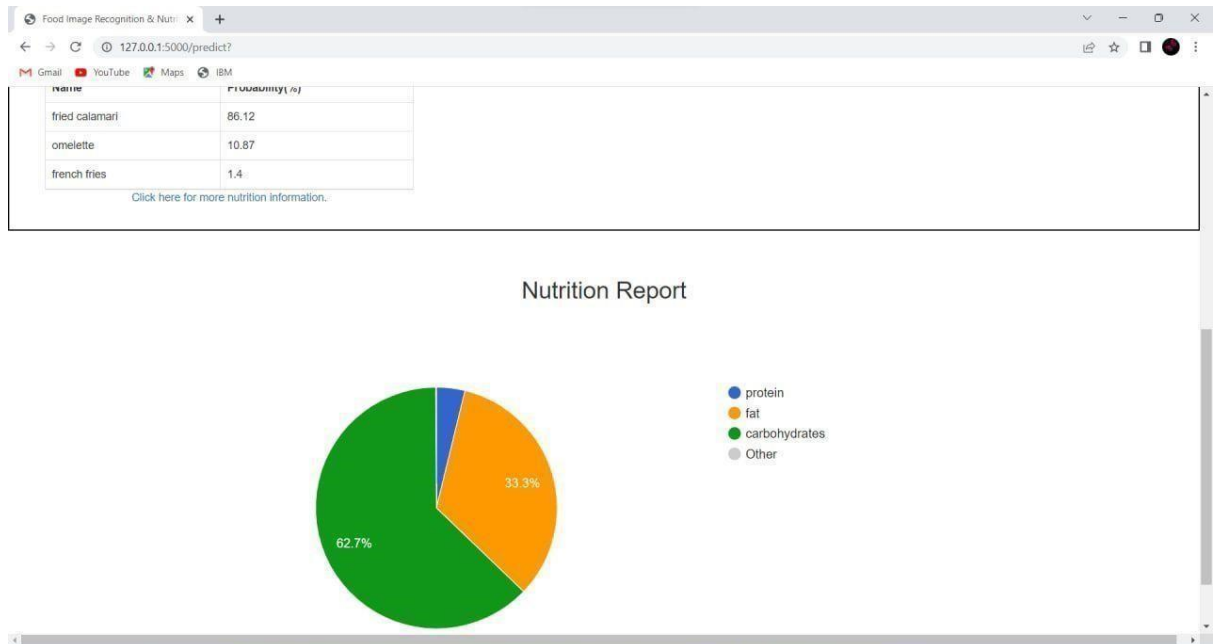
Our project was tested an end user to verify that it is working correctly.



Upload food images.

No file chosen

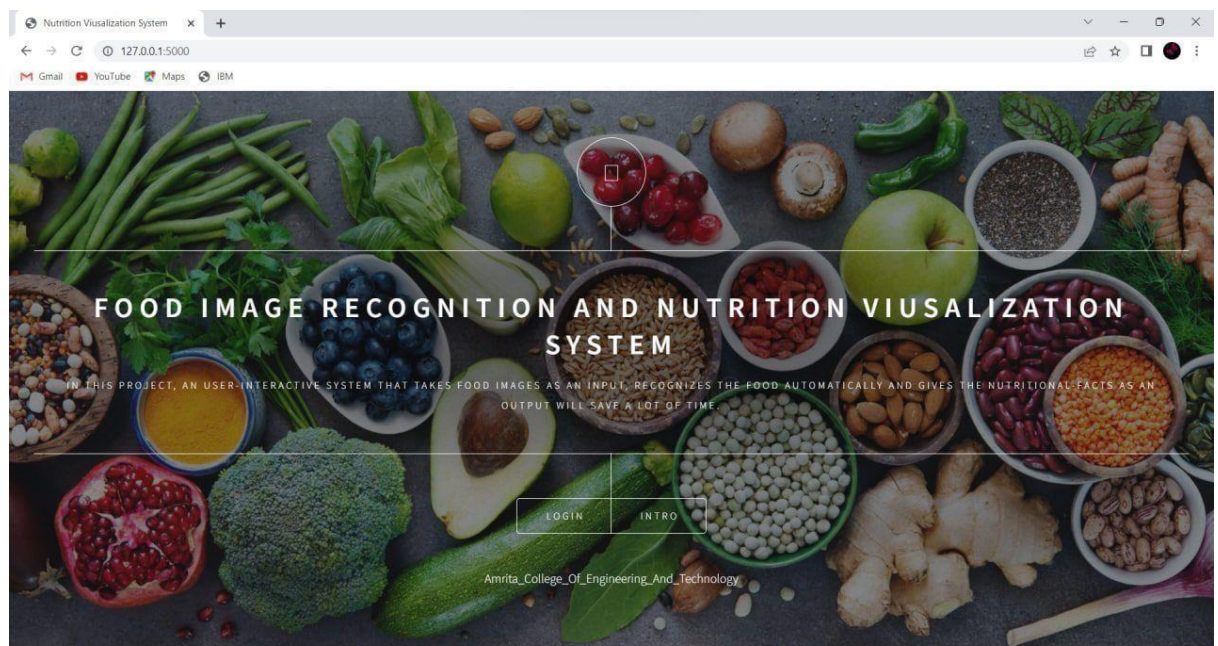


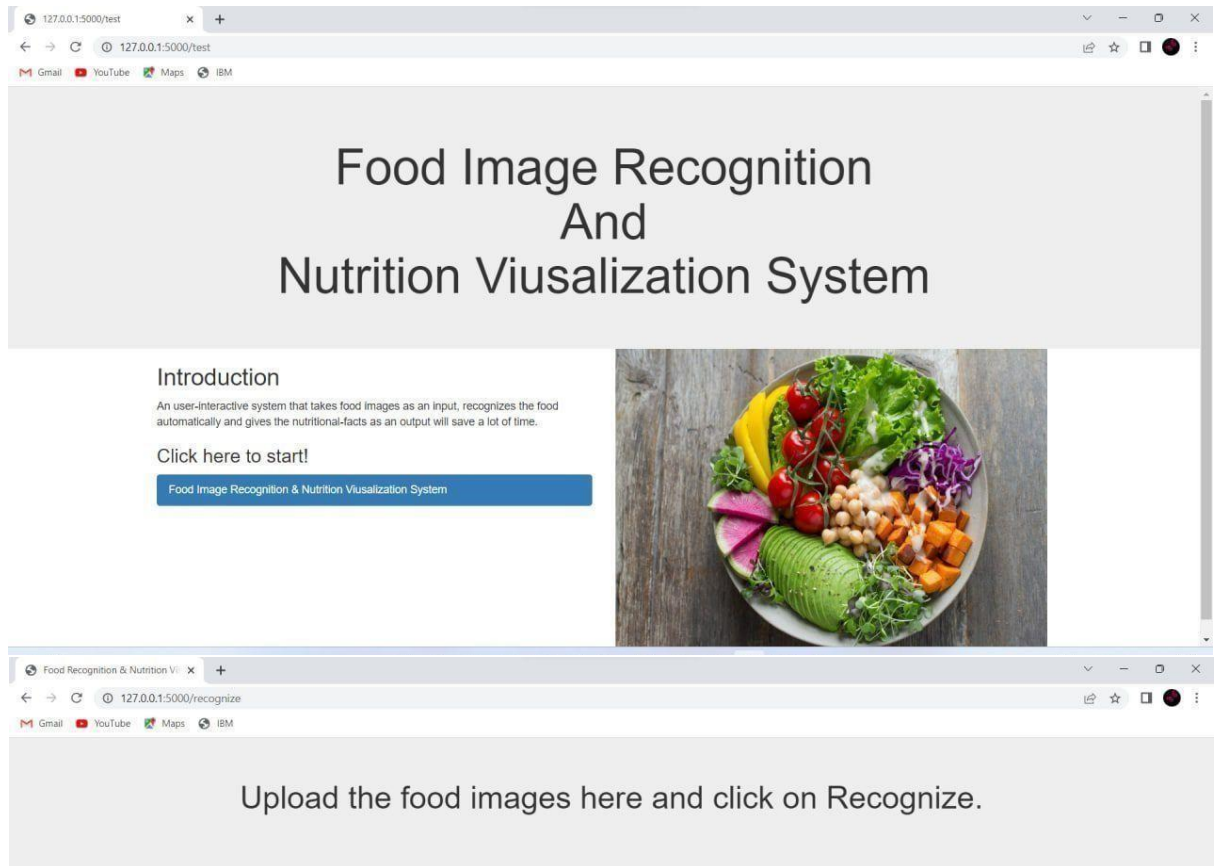


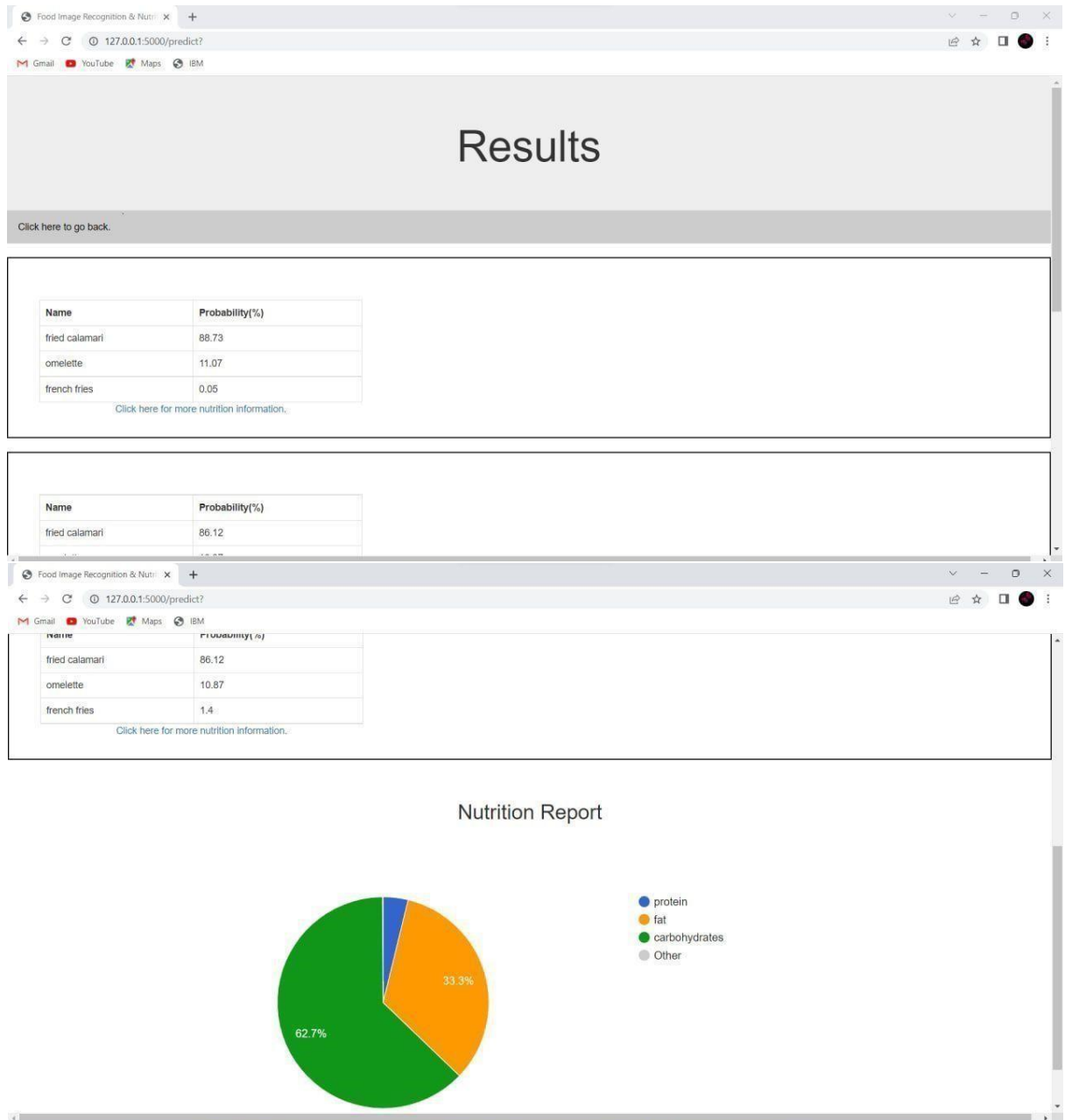
## 9. RESULTS

### 9.1 Performance Metrics

The proposed procedure was implemented and tested set of images. The training database consists of various images of food items. Once a food is recognized the equivalent Nutrition is shown on the screen.







## 10. ADVANTAGES

- It provides a maintained strategy of healthy eating habits.
- It delivers information on the nutritional value of foods and how balanced and healthy eating habits are important for us.
- It limits the amount of unnecessary food such as fat that people consume a lot.

## DISADVANTAGES

- It does not provide any login options in this applications

It does not provide any option for taking food images only upload option is provided

## **11. CONCLUSION**

In conclusion, many people have become aware of their health. Moreover, They are also informed how to live a healthy lifestyle. Most of the research related to these themes aims to identify changes in healthy lifestyle behavior with web applications that are considered effective in dietary self-monitoring.

## **12. FUTURE SCOPE**

In future we can add various features according to our needs. Such as adding login to make whether the user is authenticated or not. We can add option of taking images which is not seen in this application and also exercise that may help the users to loss their weight using this application. We can add location option to see the location of the users etc.

## **13.APPENDIX**

```
App.py import tensorflow
```

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

```
import csv
```

```
import math
```

```
import os
```

```
import numpy as np
```

```
from tensorflow.keras.preprocessing import image
```

```
from tensorflow.python.keras.models import load_model
```

```
from werkzeug.utils import secure_filename
```

```
tmpl_dir = os.path.join(os.path.dirname(os.path.abspath(__file__)),  
                          'templates')
```

```
app = Flask(__name__, template_folder=tmpl_dir)
```

```
UPLOAD_FOLDER = 'uploads'
```

```
app.config['uploads'] = UPLOAD_FOLDER
```

```
# define label meaning
```

```
label = ['apple pie',
```



'baby back ribs',

'baklava',

'beef carpaccio',

'beef tartare',

'beet salad',

'beignets',

'bibimbap',

'bread pudding',

'breakfast burrito',

'bruschetta',

'caesar salad',

'cannoli',

'caprese salad',

'carrot cake',

'ceviche',

'cheese plate',

'cheesecake',

'chicken curry',

'chicken quesadilla',

'chicken wings',

'chocolate cake',

'chocolate mousse',

'churros',

'clam chowder',

'club sandwich',

'crab cakes',

'creme brulee',

'croque madame',

'cup cakes',

'deviled eggs',

'donuts',

'dumplings',

'edamame',

'eggs benedict',

'escargots',

'falafel',

'filet mignon',

'fish and\_chips',

'foie gras',

'french fries',

'french onion soup',

'french toast',

'fried calamari',

'fried rice',

'frozen yogurt',

'garlic bread',

'gnocchi',

'greek salad',

'grilled cheese sandwich',

'grilled salmon',

'guacamole', 'gyoza',

'hamburger',

'hot and sour soup',

'hot dog',

'huevos rancheros',

'hummus',

'ice cream',

'lasagna',

'lobster bisque',

'lobster roll sandwich',

'macaroni and cheese', 'macarons',

'miso soup',

'mussels',

'nachos',

'omelette',

'onion rings',

'oysters',

'pad thai',

'paella',

'pancakes',

'panna cotta',

'peking duck',

'pho',

'pizza',

'pork chop',

'poutine',

'prime rib',

'pulled pork sandwich',

'ramen',

'ravioli',

'red velvet cake',

'risotto',

'samosa',

'sashimi',

'scallops',

'seaweed salad',

'shrimp and grits',

'spaghetti bolognese', 'spaghetti carbonara',

'spring rolls',

'steak',

'strawberry shortcake',

'sushi',

'tacos',

'octopus balls',

'tiramisu',

'tuna tartare',



```
'waffles']
```

```
nu_link = 'https://www.nutritionix.com/food/' # Loading the best saved
```

```
model to make predictions. tensorflow.keras.backend.clear_session()
```

```
model_best = load_model('best_model_101class.h5')
```

```
print('model successfully loaded!')
```

```
start = [0]
```

```
passed = [0]
```

```
pack = [[]]
```

```
num = [0]
```

```
nutrients = [
```

```
    {'name': 'protein', 'value': 0.0},
```

```
{'name': 'calcium', 'value': 0.0},
```

```
{'name': 'fat', 'value': 0.0},
```

```
{'name': 'carbohydrates', 'value': 0.0},
```

```
{'name': 'vitamins', 'value': 0.0}
```

```
]
```

```
with open('nutrition101.csv', 'r') as file:
```

```
    reader = csv.reader(file)
```

```
    nutrition_table = dict()
```

```
    for i, row in enumerate(reader):
```

```
        if i == 0:
```

```
            name = "
```

```
        continue
```

```
    else:
```

```
        name = row[1].strip()
```

```
    nutrition_table[name] = [
```

```
        {'name': 'protein', 'value': float(row[2])},
```

```
        {'name': 'calcium', 'value': float(row[3])},
```

```
        {'name': 'fat', 'value': float(row[4])},
```

```
        {'name': 'carbohydrates', 'value': float(row[5])},
```

```
        {'name': 'vitamins', 'value': float(row[6])}
```

```
    ]
```

```
@app.route('/')
```

```
def index():
```

```
    img = 'static/profile.jpg'
```

```
    return render_template('home_new.html', img=img)
```

```
@app.route('/test')
```

```
def test():
```

```
    img = 'static/profile.jpg'
```

```
    return render_template('index.html', img=img) @app.route('/recognize')
```

```
def magic():
```

```
    return render_template('recognize.html', img=file)
```

```
@app.route('/upload', methods=['POST'])
```

```
def upload():    file =

request.files.getlist("img")

print(file)

for f in file:

    filename = secure_filename(str(num[0] + 500) + '.jpg')

    num[0] += 1

    name = os.path.join(app.config['uploads'], filename)

    print('save name', name)

    f.save(name)

pack[0] = []

return render_template('recognize.html', img=file)  @app.route('/predict')
```

```
def predict():

    result = []    #

    pack = []

    print('total image', num[0])

    for i in range(start[0], num[0]):

        pa = dict()

        filename = f{"uploads"}/{i + 500}.jpg'

        print('image filepath', filename)

        pred_img = filename

        pred_img = image.load_img(pred_img, target_size=(200, 200))

        pred_img = image.img_to_array(pred_img)

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

```
pred_img = pred_img / 255.
```

```
pred = model_best.predict(pred_img)
```

```
print("Pred")
```

```
print(pred)
```

```
if math.isnan(pred[0][0]) and math.isnan(pred[0][1]) and \
```

```
    math.isnan(pred[0][2]) and math.isnan(pred[0][3]):
```

```
    pred = np.array([0.05, 0.05, 0.05, 0.07, 0.09, 0.19, 0.55, 0.0, 0.0, 0.0,  
0.0])
```

```
top = pred.argsort()[0][-3:]
```

```
label.sort()
```

```
_true = label[top[2]]
```

```
pa['image'] = f{"uploads"}/{i + 500}.jpg'
```

```
print(pa['image'] )
```

```
x = dict()
```

```
x[_true] = float("{:.2f}".format(pred[0][top[2]] * 100))
```

```
x[label[top[1]]] = float("{:.2f}".format(pred[0][top[1]] * 100))  
x[label[top[0]]] = float("{:.2f}".format(pred[0][top[0]] * 100))
```

```
pa['result'] = x
```

```
#print(x)
```

```
pa['nutrition'] = nutrition_table[_true]
```

```
pa['food'] = f{nu_link}{_true}'
```

```
pa['idx'] = i - start[0]
```

```
pa['quantity'] = 100
```

```
pack[0].append(pa)
```



```
passed[0] += 1
```

```
start[0] = passed[0]
```

```
print('successfully packed')
```

```
# compute the average source of calories
```

```
for p in pack[0]:
```

```
    nutrients[0]['value'] = (nutrients[0]['value'] + p['nutrition'][0]['value'])
```

```
    nutrients[1]['value'] = (nutrients[1]['value'] + p['nutrition'][1]['value'])
```

```
    nutrients[2]['value'] = (nutrients[2]['value'] + p['nutrition'][2]['value'])
```

```
    nutrients[3]['value'] = (nutrients[3]['value'] + p['nutrition'][3]['value'])
```

```
    nutrients[4]['value'] = (nutrients[4]['value'] + p['nutrition'][4]['value'])  
    nutrients[0]['value'] = nutrients[0]['value'] / num[0]
```

```
    nutrients[1]['value'] = nutrients[1]['value'] / num[0]
```

---

```
nutrients[2]['value'] = nutrients[2]['value'] / num[0]
```

```
nutrients[3]['value'] = nutrients[3]['value'] / num[0]
```

```
nutrients[4]['value'] = nutrients[4]['value'] / num[0]
```





```
    return render_template('results.html',  
whole_nutrition=nutrients)
```

```
pack=pack[0],
```

```
@app.route('/update', methods=['POST'])
```

```
def update():
```

```
    return render_template('index.html', img='static/P2.jpg')
```

```
if __name__ == "__main__":
```

```
    import click
```

@click.command()



```
@click.option('--debug', is_flag=True)
```

```
@click.option('--threaded', is_flag=True)
```

```
@click.argument('HOST', default='127.0.0.1')
```

```
@click.argument('PORT', default=5000, type=int)
```

```
def run(debug, threaded, host, port):
```

```
    """
```

This function handles command line parameters.

Run the server using

```
python server.py Show
```

the help text using

```
python server.py --help
```

```
    """
```



HOST, PORT = host, port





```
app.run(host=HOST, port=PORT, debug=debug, threaded=threaded)
```

```
run()
```

## **SOURCE CODE**

<https://github.com/IBM-EPBL/IBM-Project-52749-1661146981>

## **GITHUB AND PROJECT DEMO LINK**

<https://drive.google.com/folderview?id=1mIr9JPNWJY7NOz00d4HbH138uc2JE00>

