Project report

NUTRITION ASSISTANT APPLICATION

PROJECT ID: PNT2022TMID24587

Submitted by -

TEAM MEMBERS:

BHUVANESHWAR N - 210419104029

GOWTHAM M – 210419104059

VIVEK ILINDRA- 210419104067

SAI KAILASH JAMPURAM-210419104070

BOLISETTI CHIRANJEEVI ADINARAYANA-210419104035

INDEX

1.INTRODUCTION:

1.1PROJECT OVERVIEW:

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 and provide it's nutritional values. Our method employs "Clarifai's AI-Driven Food Detection Model" for accurate food identification and "Spoonacular Nutrition API" to give the nutritional value of the identified food. Clarifai AI-Driven Food Detection Model is an API that classifies the ingredients of the meal and provide the name of the meal. That name will be provided as an input to the

Spoonacular API which provides the nutritional value of the identified food.

1.2PURPOSE:

It helps user in monitoring the calorie intake, to keep track of daily intake, and to get the nutritional value of the uploaded image.

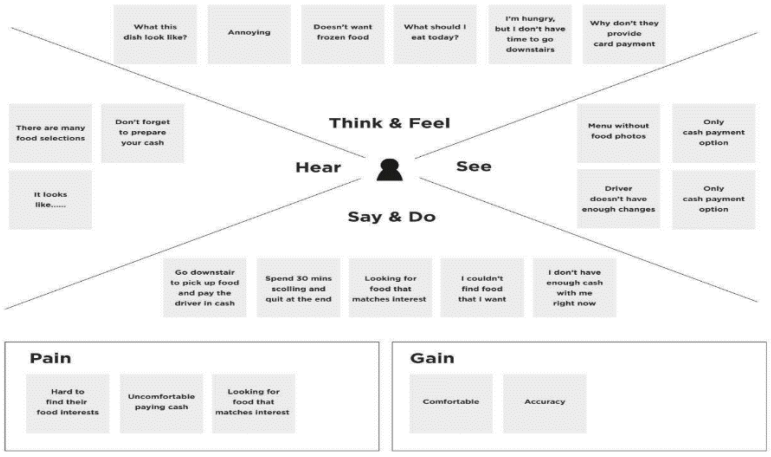
2.LITERATURE SURVEY:

Self-monitoring is the centrepiece of behavioural weight loss intervention programs. This article presents a systematic review of the literature on three components of self-monitoring in behavioural weight loss studies: diet, exercise and self-weighing. This review included articles that were published between 1993 and 2009 that reported on the relationship between weight loss and these self-monitoring strategies. Of the 22 studies identified, 14 focused on dietary self-monitoring, one on self-monitoring exercise and six on self-weighing. A wide array of methods was used to perform self-monitoring; the paper diary was used most often. Adherence to self-monitoring was reported most frequently as the number of diaries completed or the frequency of log-ins or reported weights. The use of technology, which included the Internet, personal digital

assistants and electronic digital scales were reported in five studies. Descriptive designs were used in the earlier studies while more recent reports involved prospective studies and randomized trials that examined the effect of self-monitoring on weight loss. A significant association between self-monitoring and weight loss was consistently found; however, the level of evidence was weak because of methodological limitations. The most significant limitations of the reviewed studies were the homogenous samples and reliance on self-report. In all but two studies, the samples were predominantly White and female. This review highlights the need for studies in more diverse populations, for objective measures of adherence to self-monitoring, and for studies that establish the required dose of self-monitoring for successful outcomes.

3.IDEATION PHASE AND PROPOSED SYSTEM:

3.1EMPATHY MAP:



3.2BRAINSTORMING AND PROPOSED SOLUTION:

Brainstorm & idea prioritization

Use this template in your own brainstorming sessions so your team can unleash their imagination and start shaping concepts even if you're not sitting in the same room.

10 minutes to prepare
1 hour to collaborate
2-8 people recommended

Share template feedback

Before you collaborate

A little bit of preparation goes a long way with this session. Here's what you need to do to get going.

10 minutes

Team gathering
Define who should participate in the session and send an invite. Share relevant information or pre-work ahead.

Set the goal
Think about the problem you'll be focusing on solving in the brainstorming session.

Learn how to use the facilitation tools
Use the Facilitation Superpowers to run a happy and productive session.

Open article

1 Define your problem statement

NUTRITION ASSISTANT APPLICATION

What problem are you trying to solve? Frame your problem as a How Might We statement. This will be the focus of your brainstorm.

5 minutes

How might we take health issues into account?

How might we provide health warnings?

How might we get inputs from the user?

How might we enhance Service and Support?

Key rules of brainstorming

To run an smooth and productive session

- Stay in topic.
- Encourage wild ideas.
- Defer judgment.
- Listen to others.
- Go for volume.
- If possible, be visual.

2 Brainstorm

Write down any ideas that come to mind that address your problem statement.

10 minutes

MINERALS
VITAMINS
FIBRE
Types of nutrients
Healthy ingredients
Ways to get input from the user
Warnings should be issued on
Pros & Cons for
Unhealthy ingredients to be reduced
White sugar
Support & Service

3 Group ideas

Take turns sharing your ideas while clustering similar or related notes as you go. In the last 15 minutes, give each cluster a sentence-like label. If a cluster is bigger than six sticky notes, try and see if you and break it up into smaller sub-groups.

20 minutes

APPLICATION
ADVANATGES
DISADVANATGES

4 Prioritize

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

20 minutes

Considering health issues of the user
Personal Pros for
Unhealthy ingredients to be reduced
Ways to get input from the user
Feasibility

Define CS, fit into CC

1. CUSTOMER SEGMENT(S) CS

- People who wish to analyze and track the nutritional information of their food
- People who want to monitor their diet routine
- People with some health issues who has to intake food according to the prescribed level

6. CUSTOMER CONSTRAINTS CC

- Internet connectivity
- Lack of serious interest
- Busy schedule

5. AVAILABLE SOLUTIONS AS

Web searching about the food in internet

- Pros:

The results would be quicker depending upon the internet connectivity

- Cons:

Inaccurate results, lack of specified and exact results

Explore AS, differentiate

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

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

- How might we get inputs from the user?
- How might we take health issues into account?
- How might we provide health warnings?
- How might we enhance Service and Support?

9. PROBLEM ROOT CAUSE RC

The root cause of this problem

- Is the unhealthy body condition of the user
- Is the desire of the user to stay fit and healthy

7. BEHAVIOUR BE

- Users start to analyze the nutrition present in their food
- Users tend to practice healthy dietary habits and fitness activities

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

Identify strong TR & EM

3. TRIGGERS TR

User gets to act if any of his friends or peers started using the nutrition assistance service.

4. EMOTIONS: BEFORE / AFTER EM

Unhealthy, Lazy, Improper eating habits> Healthy, Active, Proper eating practice

10. YOUR SOLUTION SL

To get accurate results user can upload the image of the food, which then gets matched with the database and display the exact result
Fascinating diet plans and food routines are made available to the user
Providing exclusive notification to the user about diet routines

8. CHANNELS of BEHAVIOUR CH

8.1 ONLINE

Using internet to browse the details of the nutrition present in a particular food

8.2 OFFLINE

Taking part in various fitness activities and avoiding unhealthy foods at all costs

Identify strong TR & EM

PROPOSED SOLUTION:

PARAMETERS	DESCRIPTION
Problem Statement	Providing the nutrition assistance to users from the image of the food they uploaded as an input.
Idea	The solution can be brought by using Clarifai's AI Driven Food Detection Model for getting accurate food identification and Food API to give the nutrition value of the identified food.
Uniqueness	Providing a user-friendly environment to access the nutritional information about the food by ▪ Taking the image of the food. ▪ Uploading from the gallery. ▪ Entering manually. Choosing from the provided list.

Customer Satisfaction	By giving periodic push notifications, diet plans or user goals , and getting feedback from the users for improvement.
Business Model	By introducing Paid membership in app advertisements, eCommerce (sponsoring the products of a brand associated with food industry).
Scalability of the solution	▪ Being very adaptive to the inputs. ▪ Providing regular updates and upgradation. Ensuring easy accessibility to the database.

4.REQUIRNMENT ANALYSIS:

Functional Requirements:

Following are the functional requirements of the proposed solution.

FR No.	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)
FR-1	User Registration	Registration through Form Registration through Gmail
FR-2	User Confirmation	Confirmation via Email Confirmation via OTP
FR-3	User Links social media	Confirmation via in-app notifications
FR-4	User tracks calories consumed	Working calorie tracker, both manual and AI driven
FR-5	User upgrades to premium	Working Payment gateway, multiple modes of payment
FR-6	User finds and competes with friends	Working in app social tab, with achievements and medals or badges

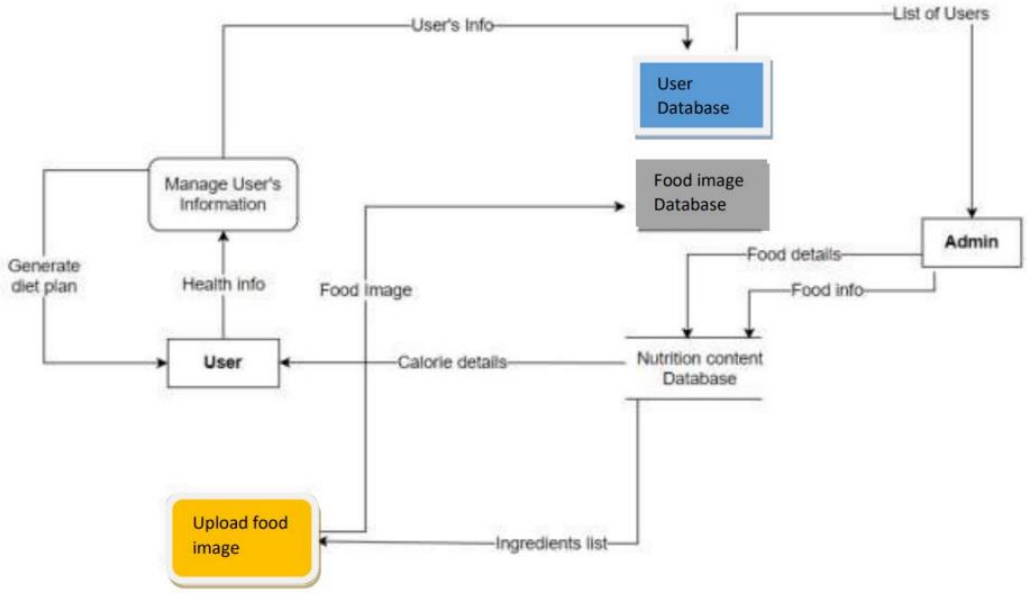
Non-functional Requirements:

Following are the non-functional requirements of the proposed solution.

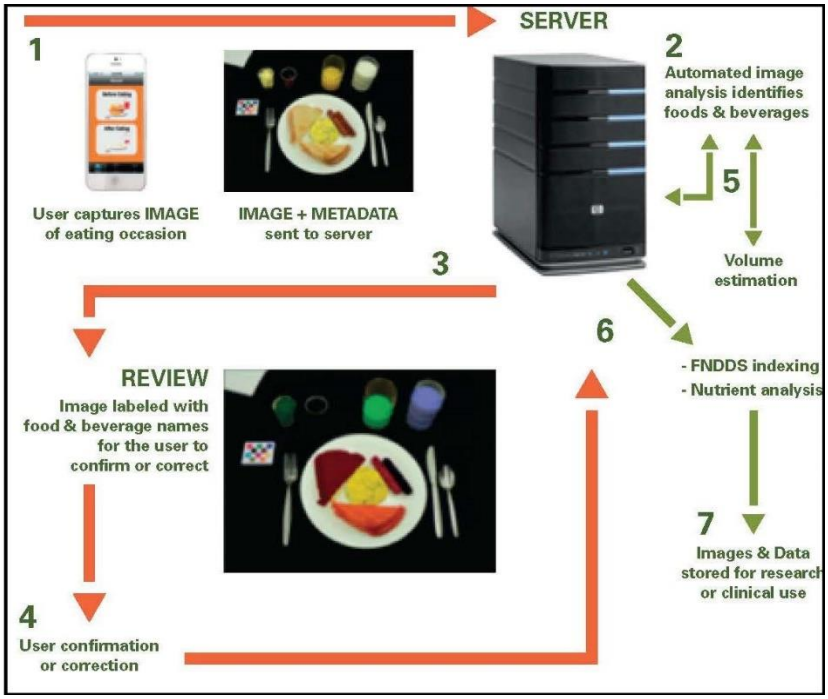
FR No.	Non-Functional Requirement	Description
NFR-1	**Usability**	Widely usable for people of all ages, and easily accessible through any browser.
NFR-2	**Security**	A layer of security to protect the confidential passwords used to login to social media and Gmail.
NFR-3	**Reliability**	Highly reliable, with the app being a browser-based application.
NFR-4	**Performance**	
NFR-5	**Availability**	Available all the time, since it is on the internet. There may be short maintenance breaks for updating or fixing bugs, that will be notified in advance.
NFR-6	**Scalability**	Highly scalable, with scaling depending on the foot traffic of the website application itself.

5.PROJECT DESIGN:

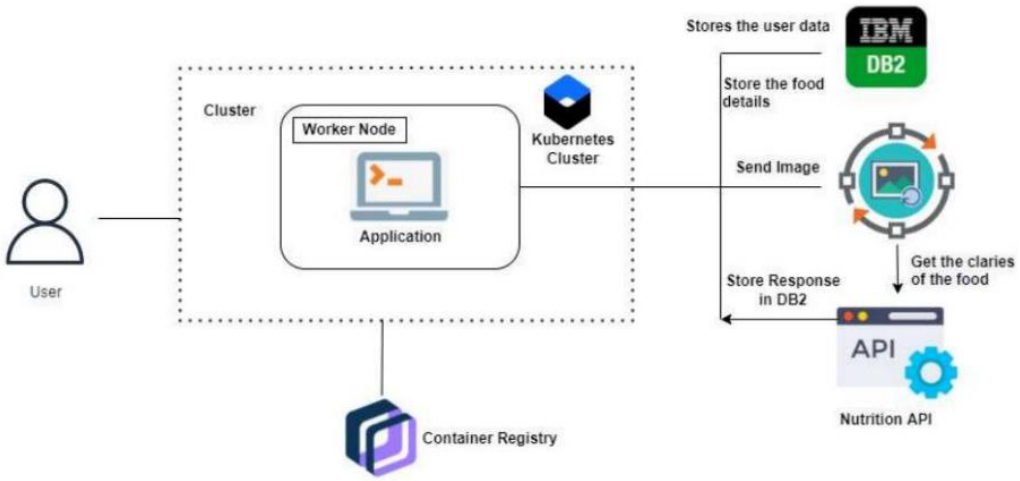
5.1DATA FLOW DIAGRAMS:



5.2SOLUTION ARCHITECTURE:



5.3TECHNICAL ARCHITECTURE:

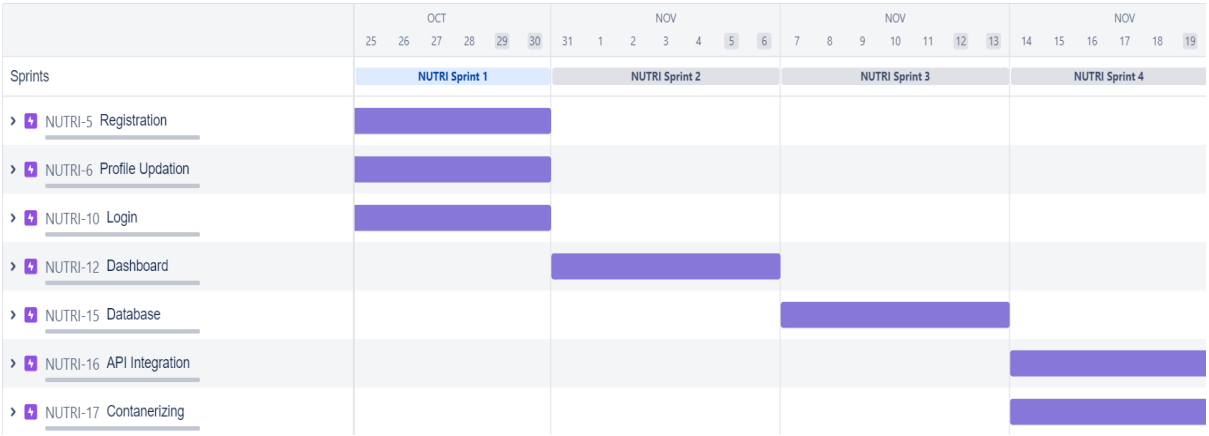


6.1PROJECT PLANNING AND SCHEDULING:

SPRINT NO	FUNCTIONAL REQUIRNMENT	USER STORY NO	TASK	STORY POINTS	PRIORITY
SPRINT-1	USER PANEL	USN -1	User will register and login to the website and start using the application functionality	20	high
SPRINT-2	CORE FUNCTIONALITY	USN-2	User will upload the food image and fetch the food nutrition and calorie contents from clarifai's food detection model.	20	high

SPRINT-3	USER HISTORY AND ACTIVITY STATISTICS	USN-3	User history will be stored and activity statistics can be accessed by users	20	High
SPRINT-4	Final Delivery	USN-4	Containerize the application using docker and Kubernetes and deployment of the application and document the application.	20	High

6.2JIRA FILE:



7.CODING AND SOLUTIONING:

home.html

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8" />
    <meta http-equiv="X-UA-Compatible" content="IE=edge" />
    <meta name="viewport" content="width=device-width, initial-scale=1.0" />
    <link
      rel="stylesheet"
      href="home-style.css"
    />
    <title>HOME</title>
    <script src="https://kit.fontawesome.com/a076d05399.js"></script>
  </head>
  <body>
    <header>
      <img
        class="logo"
        src="cardiogram.png"
        alt="logo"
      />
      <nav>
        <input type="checkbox" id="check" />
        <label for="check" class="checkbtn">
          <i class="fas fa-bars"></i>
        </label>
        <label class="nutri">NutriAux</label>
        <ul>
          <li><a href="Registration.html">Register</a></li>
          <li><a href="login.html">Login</a></li>
          <li><a href="#">Support</a></li>
        </ul>
      </nav>
    </header>
    <div class="bg-text">
      <p>
        <span>NutritionAPP/span> is a web app that aims at automatically
estimating
        the food attributes such as ingredients and nutritional value by
        classifing the input image of the food. The person who wishes to use
the
        app must register before using the app. The user can login to their
        account to see thier respective dashboard with the details of height,
        weight, BMI that is automatically generated and the amount of calorie
```

```
        that can be taken according to the BMI of the user. The user can
upload
        the image of the food that he has taken. The app will then display the
        nutritional value of the food. The user can take a note of it and can
        alter his food habits accoring to his BMI and amount of calories that
        can be intaken. The user can view the history of the food and the
        calorie intake details in the histroy page. The user can make use of
the
        support in case of any queries.
      </p>
    </div>
    <section></section>
  </body>
</html>
```

home-style.css

```
* {
  padding: 0;
  margin: 0;
  text-decoration: none;
  list-style: none;
  font-family: verdana;
  box-sizing: border-box;
}
body {
  background-color: #fdc100;
}
span {
  font-size: 30px;
  text-align: center;
  font-weight: bold;
}
img.logo {
  height: 60px;
  width: 80px;
  margin: 10px;
}
header {
  display: flex;
  justify-content: space-between;
  align-items: center;
}
nav {
  background-color: #fdc100;
  height: 80px;
  width: 100%;
}
label.nutri {
```

```
  padding: 0 30px;
  color: black;
  font-size: 25px;
  line-height: 80px;
  font-weight: bold;
}
nav ul {
  float: right;
  margin-right: 20px;
}
nav ul li {
  display: inline-block;
  line-height: 80px;
  margin: 0 5px;
}
nav ul li a {
  color: black;
  font-size: 17px;
  border: 1px solid transparent;
  padding: 7px 13px;
  border-radius: 3px;
}
a.active,
a:hover {
  background: #80cc66;
  border: 1px solid white;
  transition: 0.5s;
}
.checkbtn {
  font-size: 30px;
  color: crimson;
  float: right;
  line-height: 80px;
  margin-right: 40px;
  cursor: pointer;
  display: none;
}
#check {
  display: none;
}
.bg-text {
  background-color: rgb(0, 0, 0); /* Fallback color */
  background-color: rgba(0, 0, 0, 0.7); /* Black w/opacity/see-through */
  color: white;
  border: 3px solid #f1f1f1;
  position: absolute;
  top: 50%;
  left: 50%;
```

```
    transform: translate(-50%, -50%);
    z-index: 2;
    width: 80%;
    padding: 20px;
    text-align: justify;
    font-size: 22px;
  }

  @media (max-width: 992px) {
    label.nutri {
      font-size: 30px;
      padding-left: 50px;
    }
    nav ul li a {
      font-size: 16px;
    }
    .bg-text {
      margin-top: 50px;
      top: 50%;
      left: 50%;
    }
  }
  @media (max-width: 858px) {
    .checkbtn {
      display: block;
    }
    label.nutri {
      display: none;
    }
    .bg-text {
      font-size: 15px;
      top: 40%;
      left: 50%;
    }
    ul {
      position: fixed;
      width: 100%;
      height: 100vh;
      background: #ffee99;
      z-index: 20;
      top: 80px;
      left: -100%;
      text-align: center;
      transition: all 0.5s;
    }
    nav ul li {
      display: block;
      margin: 50px 0;
```

```
    line-height: 30px;
  }
  nav ul li a {
    font-size: 20px;
    color: black;
  }
  a.active,
  a:hover {
    background: none;
    border: none;
    color: #f26d1b;
  }
  #check:checked ~ ul {
    left: 0;
  }
}
section {
  width: 100%;
  background: url(food2.jpg);
  background-position: center;
  background-size: cover;
  height: calc(100vh - 80px);
  background-repeat: no-repeat;
}
```

REGISTRATION.HTML

```
* {
  padding: 0;
  margin: 0;
  text-decoration: none;
  list-style: none;
  font-family: verdana;
  box-sizing: border-box;
}
body {
  background-color: #fdc100;
}
span {
  font-size: 30px;
  text-align: center;
  font-weight: bold;
}
img.logo {
  height: 60px;
  width: 80px;
  margin: 10px;
}
header {
```

```
  display: flex;
  justify-content: space-between;
  align-items: center;
}
nav {
  background-color: #fdc100;
  height: 80px;
  width: 100%;
}
label.nutri {
  padding: 0 30px;
  color: black;
  font-size: 25px;
  line-height: 80px;
  font-weight: bold;
}
nav ul {
  float: right;
  margin-right: 20px;
}
nav ul li {
  display: inline-block;
  line-height: 80px;
  margin: 0 5px;
}
nav ul li a {
  color: black;
  font-size: 17px;
  border: 1px solid transparent;
  padding: 7px 13px;
  border-radius: 3px;
}
a.active,
a:hover {
  background: #80cc66;
  border: 1px solid white;
  transition: 0.5s;
}
.checkbtn {
  font-size: 30px;
  color: crimson;
  float: right;
  line-height: 80px;
  margin-right: 40px;
  cursor: pointer;
  display: none;
}
#check {
```

```
    display: none;
  }
  .bg-text {
    background-color: rgb(0, 0, 0); /* Fallback color */
    background-color: rgba(0, 0, 0, 0.7); /* Black w/opacity/see-through */
    color: white;
    border: 3px solid #f1f1f1;
    position: absolute;
    top: 50%;
    left: 50%;
    transform: translate(-50%, -50%);
    z-index: 2;
    width: 80%;
    padding: 20px;
    text-align: justify;
    font-size: 22px;
  }

  @media (max-width: 992px) {
    label.nutri {
      font-size: 30px;
      padding-left: 50px;
    }
    nav ul li a {
      font-size: 16px;
    }
    .bg-text {
      margin-top: 50px;
      top: 50%;
      left: 50%;
    }
  }
  @media (max-width: 858px) {
    .checkbtn {
      display: block;
    }
    label.nutri {
      display: none;
    }
    .bg-text {
      font-size: 15px;
      top: 40%;
      left: 50%;
    }
    ul {
      position: fixed;
      width: 100%;
      height: 100vh;
```

```
    background: #ffee99;
    z-index: 20;
    top: 80px;
    left: -100%;
    text-align: center;
    transition: all 0.5s;
  }
  nav ul li {
    display: block;
    margin: 50px 0;
    line-height: 30px;
  }
  nav ul li a {
    font-size: 20px;
    color: black;
  }
  a.active,
  a:hover {
    background: none;
    border: none;
    color: #f26d1b;
  }
  #check:checked ~ ul {
    left: 0;
  }
}
section {
  width: 100%;
  background: url(food2.jpg);
  background-position: center;
  background-size: cover;
  height: calc(100vh - 80px);
  background-repeat: no-repeat;
}
```

Personaldetails.html

```
* {
  padding: 0;
  margin: 0;
  text-decoration: none;
  list-style: none;
  font-family: verdana;
  box-sizing: border-box;
}
body {
  background-color: #fdc100;
}
span {
```

```
  font-size: 30px;
  text-align: center;
  font-weight: bold;
}
img.logo {
  height: 60px;
  width: 80px;
  margin: 10px;
}
header {
  display: flex;
  justify-content: space-between;
  align-items: center;
}
nav {
  background-color: #fdc100;
  height: 80px;
  width: 100%;
}
label.nutri {
  padding: 0 30px;
  color: black;
  font-size: 25px;
  line-height: 80px;
  font-weight: bold;
}
nav ul {
  float: right;
  margin-right: 20px;
}
nav ul li {
  display: inline-block;
  line-height: 80px;
  margin: 0 5px;
}
nav ul li a {
  color: black;
  font-size: 17px;
  border: 1px solid transparent;
  padding: 7px 13px;
  border-radius: 3px;
}
a.active,
a:hover {
  background: #80cc66;
  border: 1px solid white;
  transition: 0.5s;
}
```

```css
.checkbtn {
  font-size: 30px;
  color: crimson;
  float: right;
  line-height: 80px;
  margin-right: 40px;
  cursor: pointer;
  display: none;
}
#check {
  display: none;
}
.bg-text {
  background-color: rgb(0, 0, 0); /* Fallback color */
  background-color: rgba(0, 0, 0, 0.7); /* Black w/opacity/see-through */
  color: white;
  border: 3px solid #f1f1f1;
  position: absolute;
  top: 50%;
  left: 50%;
  transform: translate(-50%, -50%);
  z-index: 2;
  width: 80%;
  padding: 20px;
  text-align: justify;
  font-size: 22px;
}

@media (max-width: 992px) {
  label.nutri {
    font-size: 30px;
    padding-left: 50px;
  }
  nav ul li a {
    font-size: 16px;
  }
  .bg-text {
    margin-top: 50px;
    top: 50%;
    left: 50%;
  }
}
@media (max-width: 858px) {
  .checkbtn {
    display: block;
  }
  label.nutri {
    display: none;
```

```
  }
  .bg-text {
    font-size: 15px;
    top: 40%;
    left: 50%;
  }
  ul {
    position: fixed;
    width: 100%;
    height: 100vh;
    background: #ffee99;
    z-index: 20;
    top: 80px;
    left: -100%;
    text-align: center;
    transition: all 0.5s;
  }
  nav ul li {
    display: block;
    margin: 50px 0;
    line-height: 30px;
  }
  nav ul li a {
    font-size: 20px;
    color: black;
  }
  a.active,
  a:hover {
    background: none;
    border: none;
    color: #f26d1b;
  }
  #check:checked ~ ul {
    left: 0;
  }
}
section {
  width: 100%;
  background: url(food2.jpg);
  background-position: center;
  background-size: cover;
  height: calc(100vh - 80px);
  background-repeat: no-repeat;
}
```

Login.html

```
* {
  padding: 0;
```

```
  margin: 0;
  text-decoration: none;
  list-style: none;
  font-family: verdana;
  box-sizing: border-box;
}
body {
  background-color: #fdc100;
}
span {
  font-size: 30px;
  text-align: center;
  font-weight: bold;
}
img.logo {
  height: 60px;
  width: 80px;
  margin: 10px;
}
header {
  display: flex;
  justify-content: space-between;
  align-items: center;
}
nav {
  background-color: #fdc100;
  height: 80px;
  width: 100%;
}
label.nutri {
  padding: 0 30px;
  color: black;
  font-size: 25px;
  line-height: 80px;
  font-weight: bold;
}
nav ul {
  float: right;
  margin-right: 20px;
}
nav ul li {
  display: inline-block;
  line-height: 80px;
  margin: 0 5px;
}
nav ul li a {
  color: black;
  font-size: 17px;
```

```
  border: 1px solid transparent;
  padding: 7px 13px;
  border-radius: 3px;
}
a.active,
a:hover {
  background: #80cc66;
  border: 1px solid white;
  transition: 0.5s;
}
.checkbtn {
  font-size: 30px;
  color: crimson;
  float: right;
  line-height: 80px;
  margin-right: 40px;
  cursor: pointer;
  display: none;
}
#check {
  display: none;
}
.bg-text {
  background-color: rgb(0, 0, 0); /* Fallback color */
  background-color: rgba(0, 0, 0, 0.7); /* Black w/opacity/see-through */
  color: white;
  border: 3px solid #f1f1f1;
  position: absolute;
  top: 50%;
  left: 50%;
  transform: translate(-50%, -50%);
  z-index: 2;
  width: 80%;
  padding: 20px;
  text-align: justify;
  font-size: 22px;
}

@media (max-width: 992px) {
  label.nutri {
    font-size: 30px;
    padding-left: 50px;
  }
  nav ul li a {
    font-size: 16px;
  }
  .bg-text {
    margin-top: 50px;
```

```
    top: 50%;
    left: 50%;
  }
}
@media (max-width: 858px) {
  .checkbtn {
    display: block;
  }
  label.nutri {
    display: none;
  }
  .bg-text {
    font-size: 15px;
    top: 40%;
    left: 50%;
  }
  ul {
    position: fixed;
    width: 100%;
    height: 100vh;
    background: #ffee99;
    z-index: 20;
    top: 80px;
    left: -100%;
    text-align: center;
    transition: all 0.5s;
  }
  nav ul li {
    display: block;
    margin: 50px 0;
    line-height: 30px;
  }
  nav ul li a {
    font-size: 20px;
    color: black;
  }
  a.active,
  a:hover {
    background: none;
    border: none;
    color: #f26d1b;
  }
  #check:checked ~ ul {
    left: 0;
  }
}
section {
  width: 100%;
```

```
  background: url(food2.jpg);
  background-position: center;
  background-size: cover;
  height: calc(100vh - 80px);
  background-repeat: no-repeat;
}
```

Dashboard.html

```
* {
  padding: 0;
  margin: 0;
  text-decoration: none;
  list-style: none;
  font-family: verdana;
  box-sizing: border-box;
}
body {
  background-color: #fdc100;
}
span {
  font-size: 30px;
  text-align: center;
  font-weight: bold;
}
img.logo {
  height: 60px;
  width: 80px;
  margin: 10px;
}
header {
  display: flex;
  justify-content: space-between;
  align-items: center;
}
nav {
  background-color: #fdc100;
  height: 80px;
  width: 100%;
}
label.nutri {
  padding: 0 30px;
  color: black;
  font-size: 25px;
  line-height: 80px;
  font-weight: bold;
}
nav ul {
  float: right;
```

```
  margin-right: 20px;
}
nav ul li {
  display: inline-block;
  line-height: 80px;
  margin: 0 5px;
}
nav ul li a {
  color: black;
  font-size: 17px;
  border: 1px solid transparent;
  padding: 7px 13px;
  border-radius: 3px;
}
a.active,
a:hover {
  background: #80cc66;
  border: 1px solid white;
  transition: 0.5s;
}
.checkbtn {
  font-size: 30px;
  color: crimson;
  float: right;
  line-height: 80px;
  margin-right: 40px;
  cursor: pointer;
  display: none;
}
#check {
  display: none;
}
.bg-text {
  background-color: rgb(0, 0, 0); /* Fallback color */
  background-color: rgba(0, 0, 0, 0.7); /* Black w/opacity/see-through */
  color: white;
  border: 3px solid #f1f1f1;
  position: absolute;
  top: 50%;
  left: 50%;
  transform: translate(-50%, -50%);
  z-index: 2;
  width: 80%;
  padding: 20px;
  text-align: justify;
  font-size: 22px;
}
```

```
@media (max-width: 992px) {
  label.nutri {
    font-size: 30px;
    padding-left: 50px;
  }
  nav ul li a {
    font-size: 16px;
  }
  .bg-text {
    margin-top: 50px;
    top: 50%;
    left: 50%;
  }
}
@media (max-width: 858px) {
  .checkbtn {
    display: block;
  }
  label.nutri {
    display: none;
  }
  .bg-text {
    font-size: 15px;
    top: 40%;
    left: 50%;
  }
  ul {
    position: fixed;
    width: 100%;
    height: 100vh;
    background: #ffee99;
    z-index: 20;
    top: 80px;
    left: -100%;
    text-align: center;
    transition: all 0.5s;
  }
  nav ul li {
    display: block;
    margin: 50px 0;
    line-height: 30px;
  }
  nav ul li a {
    font-size: 20px;
    color: black;
  }
  a.active,
  a:hover {
```

```
    background: none;
    border: none;
    color: #f26d1b;
  }
  #check:checked ~ ul {
    left: 0;
  }
}
section {
  width: 100%;
  background: url(food2.jpg);
  background-position: center;
  background-size: cover;
  height: calc(100vh - 80px);
  background-repeat: no-repeat;
}
```

Dashboard.css

```
* {
  padding: 0;
  margin: 0;
  text-decoration: none;
  list-style: none;
  font-family: verdana;
  box-sizing: border-box;
}
body {
  background-color: #fdc100;
}
span {
  font-size: 30px;
  text-align: center;
  font-weight: bold;
}
img.logo {
  height: 60px;
  width: 80px;
  margin: 10px;
}
header {
  display: flex;
  justify-content: space-between;
  align-items: center;
}
nav {
  background-color: #fdc100;
  height: 80px;
  width: 100%;
```

```
}
label.nutri {
  padding: 0 30px;
  color: black;
  font-size: 25px;
  line-height: 80px;
  font-weight: bold;
}
nav ul {
  float: right;
  margin-right: 20px;
}
nav ul li {
  display: inline-block;
  line-height: 80px;
  margin: 0 5px;
}
nav ul li a {
  color: black;
  font-size: 17px;
  border: 1px solid transparent;
  padding: 7px 13px;
  border-radius: 3px;
}
a.active,
a:hover {
  background: #80cc66;
  border: 1px solid white;
  transition: 0.5s;
}
.checkbtn {
  font-size: 30px;
  color: crimson;
  float: right;
  line-height: 80px;
  margin-right: 40px;
  cursor: pointer;
  display: none;
}
#check {
  display: none;
}
.bg-text {
  background-color: rgb(0, 0, 0); /* Fallback color */
  background-color: rgba(0, 0, 0, 0.7); /* Black w/opacity/see-through */
  color: white;
  border: 3px solid #f1f1f1;
  position: absolute;
```

```
    top: 50%;
    left: 50%;
    transform: translate(-50%, -50%);
    z-index: 2;
    width: 80%;
    padding: 20px;
    text-align: justify;
    font-size: 22px;
  }

  @media (max-width: 992px) {
    label.nutri {
      font-size: 30px;
      padding-left: 50px;
    }
    nav ul li a {
      font-size: 16px;
    }
    .bg-text {
      margin-top: 50px;
      top: 50%;
      left: 50%;
    }
  }
  @media (max-width: 858px) {
    .checkbtn {
      display: block;
    }
    label.nutri {
      display: none;
    }
    .bg-text {
      font-size: 15px;
      top: 40%;
      left: 50%;
    }
    ul {
      position: fixed;
      width: 100%;
      height: 100vh;
      background: #ffee99;
      z-index: 20;
      top: 80px;
      left: -100%;
      text-align: center;
      transition: all 0.5s;
    }
    nav ul li {
```

```
    display: block;
    margin: 50px 0;
    line-height: 30px;
  }
  nav ul li a {
    font-size: 20px;
    color: black;
  }
  a.active,
  a:hover {
    background: none;
    border: none;
    color: #f26d1b;
  }
  #check:checked ~ ul {
    left: 0;
  }
}
section {
  width: 100%;
  background: url(food2.jpg);
  background-position: center;
  background-size: cover;
  height: calc(100vh - 80px);
  background-repeat: no-repeat;
}
```

Upload.html

```
<!DOCTYPE html>
<html>
    <head>
        <title>Upload image</title>
        <link
href="https://cdn.jsdelivr.net/npm/bootstrap@5.0.2/dist/css/bootstrap.min.css"
rel="stylesheet" integrity="sha384-
EVSTQN3/azprG1Anm3QDgpJLIm9Nao0Yz1ztcQTwFspd3yD65VohhpuuCOmLASjC"
crossorigin="anonymous">
        <link rel="stylesheet"
href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/4.7.0/css/font-
awesome.min.css">
        <link rel="stylesheet" href="upload.css">
    </head>
    <body>
        <script
src="https://cdn.jsdelivr.net/npm/bootstrap@5.2.2/dist/js/bootstrap.bundle.min
.js" integrity="sha384-
OERcA2EqjJCMA+/3y+gxIOqMEjwtxJY7qPCqsdltbNJuaOe923+mo//f6V8Qbsw3"
crossorigin="anonymous"></script>
```

```
        <form action="/getnutri" method="post" enctype="multipart/form-data">
        <div class="container uploadOuter ">
            <table class="uploadimage">
                <tr>
            <th><label for="uploadFile" id="file" ><h2>Choose your image to
upload</h2><i class="fa fa-file-photo-o" style="font-
size:48px;color:red"></i></i>

            </label></th>
            <th>    </th>
            <th><h3>OR</h3></th>

          <th><span class="dragBox" >
            <h2>Drag and Drop image here</h2><i class="fa fa-file-photo-o"
style="font-size:48px;color:red"></i>
          <input name="file" type="file"
onChange="dragNdrop(event)"  ondragover="drag()" ondrop="drop()"
id="uploadFile" required/>
        </span></th>
      </tr>
      </table>

          <button class="btn btn-primary btn-lg" id="b1"
onclick="window.history.back()">Go Back </button>
          <button class="btn btn-primary btn-lg" id="b1" type="submit" >Submit
</button>

          </div>
        </form>
            </div>
          </div>
          <div>
            <table class="table table-bordered"><br/><br/>
                <tr>
                  <th scope="col" style="background-
color:#050a30;color:white;">Calories</th>
                  <td style="background-color:white;">{{calories}}</td>
                </tr>
                <tr>
                  <th scope="col" style="background-
color:#050a30;color:white;">Protein</th>
                  <td style="background-color:white;">{{protein}}</td>

                </tr>
                <tr>
                  <th scope="col" style="background-
color:#050a30;color:white;">Fat</th>
                  <td style="background-color:white;">{{fat}}</td>
```

```
                </tr>
                <tr>
                  <th scope="col" style="background-
color:#050a30;color:white;">Carbs</th>
                  <td style="background-color:white;">{{carbs}}</td>
                </tr>

            </table>
          </div>

            <div id="preview" style="height:105px">  </div>

          </div>
          </div>
          <script>
            "use strict";
function dragNdrop(event) {
    var fileName = URL.createObjectURL(event.target.files[0]);
    var preview = document.getElementById("preview");
    var previewImg = document.createElement("img");
    previewImg.setAttribute("src", fileName);
    previewImg.setAttribute("width", "100px");
    previewImg.setAttribute("height", "100px");
    preview.innerHTML = "";
    preview.appendChild(previewImg);
}
function drag() {
    document.getElementById('uploadFile').parentNode.className = 'draging
dragBox';
}
function drop() {
    document.getElementById('uploadFile').parentNode.className = 'dragBox';
}
          </script>
    </body>
</html>
```

Upload.css

```
body {
  background-image: url("calorie_calc_bg.png");
  background-color: #f1f375d7;
  background-blend-mode: color-burn;
  background-size: 430px 400px;
}
.uploadOuter {
  text-align: center;
  padding: 30px;
```

```
}
h2,
h3 {
  color: #000c66;
}
table {
  display: flex;
  flex-direction: row;
  justify-content: center;
  margin: 10px;
  margin-bottom: auto;
  padding: 20px;
}
#file {
  padding: 15px;
  width: 300px;
  height: 180px;
  text-align: center;
  border: 2px dotted #050a30;
  border-radius: 7px;
  cursor: pointer;
  position: relative;
  color: #000c66;
}
.dragBox {
  padding: 15px;
  width: 300px;
  height: 180px;
  border-radius: 7px;
  margin: 0 auto;
  position: relative;
  text-align: center;
  font-weight: bold;
  line-height: 95px;
  border: 2px dotted #050a30;
  display: inline-block;
  transition: transform 0.3s;
}
input[type="file"] {
  position: absolute;
  height: 100%;
  width: 100%;
  opacity: 0;
  top: 0;
  left: 0;
}
.draging {
  transform: scale(1.1);
```

```
}
#preview {
  text-align: center;
}
img {
  max-width: 100%;
}
#b1 {
  background-color: #000c66;
  color: white;
  border: none;
}
```

History.html

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8" />
    <meta name="viewport" content="width=device-width, initial-scale=1.0" />
    <meta http-equiv="X-UA-Compatible" content="ie=edge" />
    <title>Track history</title>

    <link
href="https://cdn.jsdelivr.net/npm/bootstrap@5.2.2/dist/css/bootstrap.min.css"
rel="stylesheet" integrity="sha384-
Zenh87qX5JnK2Jl0vWa8Ck2rdkQ2Bzep5IDxbcnCeuOxjzrPF/et3URy9Bv1WTRi"
crossorigin="anonymous"/>
    <!--Navbar fonts-->
    <link rel="stylesheet"
href="https://cdn.jsdelivr.net/npm/@fortawesome/fontawesome-
free@6.2.0/css/fontawesome.min.css" integrity="sha384-
z4tVnCr80ZcL0iufVdGQSUzNvJsKjEtqYZjiQrrYKlpGow+btDHDfQWkFjoaz/Zr"
crossorigin="anonymous">

    <!--Adding icons-->
    <!--Date icon and bootstrap alignment-->
    <link rel="stylesheet"
href="https://maxcdn.bootstrapcdn.com/bootstrap/3.4.0/css/bootstrap.min.css"/>
    <!--Burger icon-->
    <script src='https://kit.fontawesome.com/a076d05399.js'
crossorigin='anonymous'></script>
    <!--Bolt icon-->
    <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-
awesome/4.7.0/css/font-awesome.min.css">
    <!--Nav Icon-->
    <script src="https://kit.fontawesome.com/a076d05399.js"
crossorigin="anonymous"></script>
```

```
    <!--Link to trackhistorystyle.css file-->
    <link rel="stylesheet" href="trackhistorystyle.css"/>

  </head>
  <body>
    <!--Navbar-->
    <div class="bg">

      <nav>
        <img src="{{url_for('static',filename='NutriAux.png')}}"
style="width:60px; height:55px;margin-right:15px;margin-bottom: 17px;margin-
left:15px;"/>
      <input type="checkbox" id="check">
      <label for="check" class="checkbtn">
        <i class="fas fa-bars"></i>
      </label>
      <label class="logo"><span style="margin-top:5px;">Track
History</span></label>
        <ul>
          <li>
            <button class="navbutton" onclick="window.history.back()" ><a >Go
Back</a></button>
          </li>
          <li>
            <a href="#">Upload Image</a>
          </li>
          <li class="nav-item">
            <a href="{{url_for('support')}}">Support</a>
          </li>
          <li class="nav-item">
            <a href="{{url_for('home')}}">Log out</a>
          </li>
        </ul>
    </nav>

    <div class="container" style="margin-top: 50px;">
      <form action='/addhistory' method="POST">
        <div class="row">
          <div class="col-md-3 mt-3">
            <!--component-->
            <span style="font-size:17px;">Date:</span>
              <div class="input-group date" id="datePicker"  >
                <span class="input-group-addon">
                  <i class="glyphicon glyphicon-calendar"
style="color:#0b3c49"></i>
                </span>
                <input
                  type="text"
```

```
                    class="form-control"
                    name="date"
                    id="date"
                    readonly
                    style="background-color:white"
                  />
                </div>
              </div>

              <div class="col-md-3 mt-3">
                <span style="font-size:17px;">Enter the food name:</span>

                  <div class="input-group date" id="datePicker" >
                    <span class="input-group-addon" >
                      <i class='fas fa-hamburger ' style="color:#0b3c49"></i>
                    </span>
                    <input
                      type="text"
                      class="form-control"
                      name="meal_name"
                      id="meal_name"
                    />
                  </div>
                </div>
                <div class="col-md-3 mt-3">
                  <span style="font-size:17px;">Enter Calories:</span>
                     <div class="input-group date" id="datePicker" >
                       <span class="input-group-addon">
                        <i class="fa fa-bolt" style=" color:#0b3c49"></i>
                       </span>
                       <input
                         type="text"
                         class="form-control"
                         name="calories"
                         id="calories"
                       />
                     </div></div>
                <div class="col-md-3 mt-3" style="margin-left:auto">
                  <button type="submit" class="btn btn-default" width="50px"
style="margin-top:20px ;background-color:white;color: #0b3c49;border-color:
#0b3c49;" ><b>Add +</b></button>
                </div>

            </div>
            <div class="row">
                  <div class="col-md-12">
                    <table class="table table-bordered"><br/><br/>
```

```
                    <thead class="thead" style="background-
color:#0b3c49;color:white;" >
                      <tr>
                        <th scope="col">Date</th>
                        <th scope="col">Food Name</th>
                        <th scope="col">Calories</th>
                      </tr>
                    </thead>
                    <tbody>
                      {%set count = namespace(value=0)%}
                        {%for i in range(no_of_rows)%}
                          <tr>
                            {%for j in range(3)%}
                              <td>{{history[count.value]}}{%set count.value =
count.value + 1%}</td>
                            {%endfor%}
                          </tr>
                        {%endfor%}
                    </tbody>
                  </table>
                </div>
            </div>
          </div>
        </form>
      </div>
    </div>
      <!--Link to trackhistory Javascript-->
      <script src="{{url_for('static',filename='trackhistory.js')}}"></script>

      <!-- jQuery first, then Popper.js, then Bootstrap JS -->
      <script src="https://code.jquery.com/jquery-3.2.1.slim.min.js"
integrity="sha384-
KJ3o2DKtIkvYIK3UENzmM7KCkRr/rE9/Qpg6aAZGJwFDMVNA/GpGFF93hXpG5KkN"
crossorigin="anonymous"></script>
      <script
src="https://cdn.jsdelivr.net/npm/popper.js@1.12.9/dist/umd/popper.min.js"
integrity="sha384-
ApNbgh9B+Y1QKtv3Rn7W3mgPxhU9K/ScQsAP7hUibX39j7fakFPskvXusvfa0b4Q"
crossorigin="anonymous"></script>
      <script
src="https://cdn.jsdelivr.net/npm/bootstrap@4.0.0/dist/js/bootstrap.min.js"
integrity="sha384-
JZR6Spejh4U02d8jOt6vLEHfe/JQGiRRSQQxSfFWpi1MquVdAyjUar5+76PVCmYl"
crossorigin="anonymous"></script>
    </body>
</html>
```

Trackhistorystyle.css

```css
* {
  margin: 0;
  padding: 0;
  text-decoration: none;
  list-style: none;
  box-sizing: border-box;
}
body {
  font-family: montserrat;
  background: #dcf1f1;
}
.bg {
  height: 100vh;
  width: 100%;
  background-image: linear-gradient(#dcf1f16a, #dcf1f16a),
    url("track_history.png");
}
button.navbutton {
  background: none;
  color: white;
  border: none;
  padding: 0;
  font: inherit;
  cursor: pointer;
  outline: inherit;
}

nav {
  background: #0b3c49;
  height: 80px;
  width: 100%;
}
label.logo {
  color: white;
  font-size: 25px;
  line-height: 80px;
  padding: 0 10px;
  font-weight: bold;
}
nav ul {
  float: right;
  margin-right: 10px;
}
nav ul li {
  display: inline-block;
  line-height: 80px;
  margin: 0 5px;
}
```

```
nav ul li a {
  color: white;
  font-size: 17px;
  padding: 7px 13px;
  border-radius: 3px;
  text-transform: uppercase;
}
a:hover {
  color: white;
  background: #02cecb87;
  text-decoration: none;
}
.checkbtn {
  font-size: 30px;
  color: white;
  float: right;
  line-height: 80px;
  margin-right: 40px;
  cursor: pointer;
  display: none;
}
#check {
  display: none;
}
@media (max-width: 952px) {
  label.logo {
    font-size: 25px;
  }
  nav ul li a {
    font-size: 16px;
  }
}
@media (max-width: 858px) {
  .checkbtn {
    display: block;
  }
  ul {
    position: fixed;
    width: 100%;
    height: 100vh;
    background: #07262f;
    top: 80px;
    left: -100%;
    text-align: center;
    z-index: 20;
    transition: all 0.5s;
  }
  nav ul li {
```

```
    display: block;
    margin: 50px;
    line-height: 30px;
  }
  nav ul li a {
    font-size: 20px;
  }
  a:hover {
    background: none;
    color: #90f1bab6;
  }
  button.navbutton:hover {
    color: #90f1bab6;
  }
  #check:checked ~ ul {
    left: 0;
  }
}
.container {
  background: white;
  margin-top: 30px;
  margin-bottom: 30px;
  border: 1px solid rgba(0, 0, 0, 0.4);
  padding: 30px;
  border-radius: 6px;
}
```

Trackhistory.js

```
var todaydate = new Date();
var day = todaydate.getDate();
var month = todaydate.getMonth() + 1;
var year = todaydate.getFullYear();
var datestring = day + "." + month + "." + year;
document.getElementById("date").value = datestring;
```

support.html

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8" />
    <meta name="viewport" content="width=device-width, initial-scale=1.0" />
    <meta http-equiv="X-UA-Compatible" content="ie=edge" />
    <title>Support</title>
    <!-- CSS only -->
    <link
      href="https://cdn.jsdelivr.net/npm/bootstrap@5.2.2/dist/css/bootstrap.mi
n.css"
      rel="stylesheet"
```

```
      integrity="sha384-
Zenh87qX5JnK2Jl0vWa8Ck2rdkQ2Bzep5IDxbcnCeuOxjzrPF/et3URy9Bv1WTRi"
      crossorigin="anonymous"
    />
    <link
      rel="stylesheet"
      href="https://stackpath.bootstrapcdn.com/font-awesome/4.7.0/css/font-
awesome.min.css"
    />
    <!-- JavaScript Bundle with Popper Bootstrap -->
    <script
      src="https://cdn.jsdelivr.net/npm/bootstrap@5.2.2/dist/js/bootstrap.bund
le.min.js"
      integrity="sha384-
OERcA2EqjJCMA+/3y+gxIOqMEjwtxJY7qPCqsdltbNJuaOe923+mo//f6V8Qbsw3"
      crossorigin="anonymous"
    ></script>
    <link
      rel="stylesheet"
      href="supportstyle.css"
    />
  </head>
  <body>
    <img
      src="cardiogram.png"
      style="width: 60px; height: 60px; margin-left: 15px; margin-top: 5px"
    />
    <h2 class="support-message">
      <b>Feeling stuck ? We are here to help you :)</b>
    </h2>
    <section>
      <div class="container">
        <div class="row">
          <!--FAQ column-->
          <div class="col-md-6" style="margin-top: 30px">
            <div class="accordion" id="accordionExample">
              <div class="accordion-item">
                <h2 class="accordion-header" id="headingOne">
                  <button
                    class="accordion-button collapsed"
                    type="button"
                    data-bs-toggle="collapse"
                    data-bs-target="#collapseOne"
                    aria-expanded="false"
                    aria-controls="collapseOne"
                    style="
                      --bs-accordion-btn-focus-box-shadow: none;
                      --bs-accordion-active-color: #000;
```

```
                    --bs-accordion-active-bg: #fdc100;
                    --bs-accordion-border-color: none;
                  "
                >
                  How can I use this app?
                </button>
              </h2>
              <div
                id="collapseOne"
                class="accordion-collapse collapse"
                aria-labelledby="headingOne"
                data-bs-parent="#accordionExample"
              >
                <div class="accordion-body">
                  <ul>
                    <li>
                      Complete the <code>registration</code> process by
                      clicking the register button present in the home page
                    </li>
                    <li>
                      After successful registration, <code>login</code> into
                      your account.
                    </li>
                    <li>
                      After login successfully you will be led to your
                      <code>dashboard</code>, where you can see your
personal
                      details like height, weight, BMI and daily calorie
                      intake needed (i.e, how much calorie you should
consume
                      per day). Also you can select various options like
                      Upload image, track history
                    </li>
                  </ul>
                </div>
              </div>
            </div>
            <div class="accordion-item">
              <h2 class="accordion-header" id="headingTwo">
                <button
                  class="accordion-button collapsed"
                  type="button"
                  data-bs-toggle="collapse"
                  data-bs-target="#collapseTwo"
                  aria-expanded="false"
                  aria-controls="collapseTwo"
                  style="
                    --bs-accordion-btn-focus-box-shadow:: none;
```

```
                    --bs-accordion-active-color: #000;
                    --bs-accordion-active-bg: #fdc100;
                    --bs-accordion-border-color: none;
                  "
                >
                  How can I Register?
                </button>
              </h2>
              <div
                id="collapseTwo"
                class="accordion-collapse collapse"
                aria-labelledby="headingTwo"
                data-bs-parent="#accordionExample"
              >
                <div class="accordion-body">
                  Inorder to use our application , registration process is
                  mandatory. This is because we have to keep track of your
                  personal details like height, weight etc.Click on the
                  <code>register</code> button present inside the home page.
                  You will be asked to provide your personal details for
                  providing a personalised experience. Once you have
                  successfully registered you will receive a confirmation
mail
                  and you will be redirected to the home page. From there
you
                  can login and enter into your Dashboard
                </div>
              </div>
            </div>
            <div class="accordion-item">
              <h2 class="accordion-header" id="headingThree">
                <button
                  class="accordion-button collapsed"
                  type="button"
                  data-bs-toggle="collapse"
                  data-bs-target="#collapseThree"
                  aria-expanded="false"
                  aria-controls="collapseThree"
                  style="
                    --bs-accordion-btn-focus-box-shadow: none;
                    --bs-accordion-active-color: #000;
                    --bs-accordion-active-bg: #fdc100;
                    --bs-accordion-border-color: none;
                  "
                >
                  How can I login?
                </button>
              </h2>
```

```
                <div
                  id="collapseThree"
                  class="accordion-collapse collapse"
                  aria-labelledby="headingThree"
                  data-bs-parent="#accordionExample"
                >
                  <div class="accordion-body">
                    It is necessary
                    <strong>complete your registration process</strong> before
                    you login. Once you have succesfully registered, you can
                    click on the <code>login</code> button present inside the
                    home page. You will be led to the Login page where you
will
                    be asked to enter your E-Mail ID and password that you
                    entered in the registration process. After successful
login
                    you will be led to your Dashboard.
                  </div>
                </div>
              </div>
              <div class="accordion-item">
                <h2 class="accordion-header" id="headingFour">
                  <button
                    class="accordion-button collapsed"
                    type="button"
                    data-bs-toggle="collapse"
                    data-bs-target="#collapseFour"
                    aria-expanded="false"
                    aria-controls="collapseFour"
                    style="
                      --bs-accordion-btn-focus-box-shadow: none;
                      --bs-accordion-active-color: #000;
                      --bs-accordion-active-bg: #fdc100;
                      --bs-accordion-border-color: none;
                    "
                  >
                    How can I get the nutritional value of the food that I am
                    eating now?
                  </button>
                </h2>
                <div
                  id="collapseFour"
                  class="accordion-collapse collapse"
                  aria-labelledby="headingFour"
                  data-bs-parent="#accordionExample"
                >
                  <div class="accordion-body">
                    With the help of our <code>Upload Image</code> feature you
```

```
                    can upload an image of the meal that you would like to
have.
                    Once the image has been uploaded you will be provided with
                    the nutrional content present inside that particular dish.
                    You can find this feature in the <code>menu bar</code>.
                  </div>
                </div>
              </div>
              <div class="accordion-item">
                <h2 class="accordion-header" id="headingFive">
                  <button
                    class="accordion-button collapsed"
                    type="button"
                    data-bs-toggle="collapse"
                    data-bs-target="#collapseFive"
                    aria-expanded="false"
                    aria-controls="collapseFive"
                    style="
                      --bs-accordion-btn-focus-box-shadow: none;
                      --bs-accordion-active-color: #000;
                      --bs-accordion-active-bg: #fdc100;
                      --bs-accordion-border-color: none;
                    "
                  >
                    How can I track my daily calorie intake?
                  </button>
                </h2>
                <div
                  id="collapseFive"
                  class="accordion-collapse collapse"
                  aria-labelledby="headingFive"
                  data-bs-parent="#accordionExample"
                >
                  <div class="accordion-body">
                    With the help of our <code>Track History</code> feature
you
                    can track your daily calorie intake which will help you to
                    know your current progress and achieve your fitness goal.
                    You can find this feature in your <code>Dashboard</code>.
                  </div>
                </div>
              </div>
              <div class="accordion-item">
                <h2 class="accordion-header" id="headingSix">
                  <button
                    class="accordion-button collapsed"
                    type="button"
                    data-bs-toggle="collapse"
```

```
                    data-bs-target="#collapseSix"
                    aria-expanded="false"
                    aria-controls="collapseSix"
                    style="
                      --bs-accordion-btn-focus-box-shadow: none;
                      --bs-accordion-active-color: #000;
                      --bs-accordion-active-bg: #fdc100;
                      --bs-accordion-border-color: none;
                    "
                  >
                    How much calorie should I consume per day?
                  </button>
                </h2>
                <div
                  id="collapseSix"
                  class="accordion-collapse collapse"
                  aria-labelledby="headingSix"
                  data-bs-parent="#accordionExample"
                >
                  <div class="accordion-body">
                    With the help of the data that you provide during the
                    registration process like Height, Weight, Daily
activities,
                    etc., we will provide you how much calorie you should
                    consume daily which will be shown as
                    <code>Daily calorie Intake</code> in your dashboard.
                  </div>
                </div>
              </div>
            </div>
          </div>
          <!--FAQ column-->
          <div class="col-md-6 text-center">
            <img
              src="supportdesk.png"
              class="support-desk"
            />
            <br /><br /><br />
            <button onclick="window.history.back()" class="go-back-button">
              Go back
            </button>
          </div>
        </div>
      </div>
    </section>
  </body>
</html>
```

Supportstyle.css

```
* {
  margin: 0;
  padding: 0;
}
body {
  background-color: #fff6bf;
}
.support-message {
  text-align: center;
  margin: 20px 0 !important;
  font-family: sans-serif;
  background-color: transparent;
}
.accordion-item,
.accordion-header {
  background-color: transparent !important;
  border: none !important;
  padding: 0 !important;
  margin-bottom: 20px;
}

.accordion-button {
  width: 100%;
  height: 60px;
  background-color: #fdc100;
  font-weight: bold;
  text-align: left;
  text-decoration: none;
}

.accordion-button:active {
  background-color: #fdc100;
}
.support-desk {
  width: 490px;
  height: 340px;
  margin-top: 20px;
}
.go-back-button {
  color: white;
}
.go-back-button {
  background: lighten(#06837f, 3%);
  border: 1px solid darken(#06837f, 4%);
  box-shadow: 0px 2px 0 darken(#06837f, 5%), 2px 4px 6px darken(#06837f, 2%);
  font-weight: 900;
  letter-spacing: 1px;
```

```
  transition: all 150ms linear;
}

.go-back-button:hover {
  background: darken(#06837f, 1.5%);
  border: 1px solid rgba(#000, 0.05);
  box-shadow: 1px 1px 2px rgba(#fff, 0.2);
  color: lighten(#06837f, 18%);
  text-decoration: none;
  text-shadow: -1px -1px 0 darken(#06837f, 9.5%);
  transition: all 250ms linear;
}

/* CSS */
.go-back-button {
  background-color: #06837f;
  background-image: linear-gradient(#06837f, #329ca0);
  border: 1px solid #15aca7;
  border-radius: 4px;
  box-shadow: rgba(0, 0, 0, 0.12) 0 1px 1px;
  color: #ffffff;
  cursor: pointer;
  font-family: -apple-system, ".SFNSDisplay-Regular", "Helvetica Neue",
    Helvetica, Arial, sans-serif;
  font-size: 15px;
  margin: 0;
  outline: 0;
  padding: 11px 15px 12px;
  text-align: center;
  transition: box-shadow 0.05s ease-in-out, opacity 0.05s ease-in-out;
  user-select: none;
  -webkit-user-select: none;
  touch-action: manipulation;
  width: 30%;
}

.go-back-button:hover {
  box-shadow: rgba(24, 230, 127, 0.995) 0 0 2px inset,
    rgba(5, 197, 235, 0.818) 0 1px 2px;
  text-decoration: none;
  transition-duration: 0.15s, 0.15s;
}

.go-back-button:active {
  box-shadow: rgba(0, 0, 0, 0.15) 0 2px 4px inset, rgba(0, 0, 0, 0.4) 0 1px
1px;
}
```

Nutritionapp.py

```
from flask import Flask,render_template,request,url_for,redirect,session
import ibm_db
import os
from sendgrid import SendGridAPIClient
from sendgrid.helpers.mail import Mail
import requests

app=Flask(__name__)
app.secret_key='a'
try:
   conn=ibm_db.connect("DATABASE=bludb;HOSTNAME=ba99a9e6-
d59e-4883-8fc0-
d6a8c9f7a08f.c1ogj3sd0tgtu0lqde00.databases.appdomain.cloud;PORT=31
321;SECURITY=SSL;SSLServerCertificate=DigiCertGlobalRootCA.crt;U
ID=gkr16989;PWD=WvN7xr79Kp6YfdL7","","")
except:
   print("Unable to connect: ",ibm_db.conn_error())


@app.route("/")
def home():
   session['status_msg']=' '

   return render_template('Home.html')

@app.route("/reg")
def reg():
   return render_template('Registration.html')

@app.route("/register",methods=["POST","GET"])
def register():
    if request.method == 'POST' :
      firstName = request.form['firstName']
      lastName = request.form['lastName']

      session['email'] = request.form['email']
      phoneNumber = request.form['phoneNumber']
      password = request.form['password']
      sql = "SELECT * FROM registration WHERE EMAIL_ID=?"
      stmt = ibm_db.prepare(conn, sql)
      ibm_db.bind_param(stmt,1,session['email'])
```

```
        ibm_db.execute(stmt)
        account = ibm_db.fetch_assoc(stmt)
        print(account)
        message =
Mail(from_email='nutriaux04@gmail.com',to_emails=session['email'],subj
ect="NutriAux - Registration",html_content='<b>NutriAux welcomes
you</b><br/><p>Your account has been registered successfully</p>')
        try:

sg=SendGridAPIClient('SG.QMrCVkeuQKmODkjr39Y5bQ.O1kEThoHT
OOBXz8KJRgjauH3slyG_KsbQ4_yuIzQ0jY')
            response=sg.send(message)
            print(response.status_code)
            print(response.body)
            print(response.headers)
        except Exception as e:
            print(e)


        if account:


            session['status_msg']= 'Account already exists ! Kindly login'
            return redirect(url_for('login'))
        else :
            insert_sql = "INSERT INTO registration VALUES (?,?,?,?,?)"
            prep_stmt = ibm_db.prepare(conn, insert_sql)
            ibm_db.bind_param(prep_stmt, 1, firstName)
            ibm_db.bind_param(prep_stmt, 2, lastName)
            ibm_db.bind_param(prep_stmt, 3, session['email'])
            ibm_db.bind_param(prep_stmt, 4, phoneNumber)
            ibm_db.bind_param(prep_stmt, 5, password)
            ibm_db.execute(prep_stmt)
        print('You have successfully registered !')
        return redirect(url_for('personaldetails'))




@app.route("/personaldetails")
def personaldetails():
    return render_template("personaldetails.html")
```

```
@app.route("/addpersonaldetails",methods=["POST","GET"])
def addpersonaldetails():
   if request.method == 'POST' :
      age=float(request.form.get('age'))
      gender=request.form.get('Gender')
      weight=float(request.form.get('weight'))
      height=float(request.form.get('height'))
      activity=request.form.get('activity')
      print(age,gender,weight,height,activity)
      if(gender == 'male'and activity == "1"):
         totalCalories = 1.2 * (66.5 + (13.75 * weight) + (5.003 * height) -
(6.755 * age))
      elif(gender == 'male' and activity == "2"):
         totalCalories = 1.375 * (66.5 + (13.75 * weight) + (5.003 * height) -
(6.755 * age))
      elif (gender == 'male' and activity == "3"):
         totalCalories = 1.55 * (66.5 + (13.75 * weight) + (5.003 * height) -
(6.755 * age))
      elif(gender == 'male' and activity == "4"):
         totalCalories = 1.725 * (66.5 + (13.75 * weight) + (5.003 * height) -
(6.755 * age))
      elif(gender == 'male' and activity == "5"):
         totalCalories = 1.9 * (66.5 + (13.75 * weight) + (5.003 * height) -
(6.755 * age))
      elif(gender == 'female' and activity == "1"):
         totalCalories = 1.2 * (655 + (9.563 * weight) + (1.850 * height) -
(4.676 * age))
      elif(gender == 'female' and activity == "2"):
         totalCalories = 1.375 * (655 + (9.563 * weight) + (1.850 * height) -
(4.676 * age))
      elif(gender == 'female' and activity == "3"):
         totalCalories = 1.55 * (655 + (9.563 * weight) + (1.850 * height) -
(4.676 * age))
      elif(gender == 'female' and activity == "4"):
         totalCalories = 1.725* (655 + (9.563 * weight) + (1.850 * height) -
(4.676 * age))
      else:
         totalCalories = 1.9 * (655 + (9.563 * weight) + (1.850 * height) -
(4.676 * age))
      print(int(totalCalories))
      BMI = (weight / (height/100)**2 )
      if BMI <= 18.5:
         BMI_message="underweight"
```

```
        elif BMI <= 24.9:
            BMI_message="healthy"
        elif BMI <= 29.9:
            BMI_message="overweight"
        else:
            BMI_message="obese"
        print(BMI)
        insert_query="INSERT INTO personal_details
VALUES(?,?,?,?,?,?,?)"
        prep_stmt=ibm_db.prepare(conn,insert_query)
        ibm_db.bind_param(prep_stmt,1,session['email'])
        ibm_db.bind_param(prep_stmt,2,str(int(age)))
        ibm_db.bind_param(prep_stmt,3,gender)
        ibm_db.bind_param(prep_stmt,4,str(weight))
        ibm_db.bind_param(prep_stmt,5,str(height))
        ibm_db.bind_param(prep_stmt,6,str(totalCalories))
        ibm_db.bind_param(prep_stmt,7,str(BMI))
        ibm_db.execute(prep_stmt)
    return redirect(url_for('login'))


@app.route("/login")
def login():
    return render_template("login.html",message=session['status_msg'])


@app.route("/verify",methods=["POST","GET"])
def verify():
    session['email'] = request.form.get("email")
    password = request.form.get("password")
    get_query="SELECT * FROM registration WHERE EMAIL_ID=?
AND PASSWORD=?"
    prep=ibm_db.prepare(conn,get_query)
    ibm_db.bind_param(prep,1,session['email'])
    ibm_db.bind_param(prep,2,password)
    result=ibm_db.execute(prep)
    login = ibm_db.fetch_assoc(prep)
    if login:
        get_query="SELECT weight,height,daily_calorie_intake,BMI FROM
personal_details WHERE EMAIL_ID=?"
        prep=ibm_db.prepare(conn,get_query)
        ibm_db.bind_param(prep,1,session['email'])
        result=ibm_db.execute(prep)
```

```
        data = ibm_db.fetch_tuple(prep)
        global weight
        weight= data[0]
        global height
        height= data[1]
        global daily_calorie_intake
        daily_calorie_intake=data[2]
        daily_calorie_intake=daily_calorie_intake[0:7]
        global BMI
        BMI=data[3]
        BMI=BMI[0:4]
        return redirect((url_for('dashboard')))
    print("Wrong password" , session['email'],password)
    return render_template("login.html",message="Incorrect Email ID or
Password! Try again")

@app.route("/dashboard")
def dashboard():
    return
render_template('dashboard.html',weight=weight,height=height,daily_calo
rie_intake=daily_calorie_intake,BMI=BMI)

@app.route("/upload")
def upload():
    return
render_template('upload.html',calories="",fat="",protein="",carbs="")



@app.route("/history")
def history():
    get_query="SELECT CHAR(DATE_OF_CONSUMPTION,EUR)AS
DATE,MEAL_NAME,CALORIES FROM TRACKHISTORY WHERE
EMAIL_ID=?"
    prep=ibm_db.prepare(conn,get_query)
    ibm_db.bind_param(prep,1,session['email'])
    result=ibm_db.execute(prep)
    if result==False:
        print("not working")
    history=[]
    dictionary = ibm_db.fetch_assoc(prep)
    while dictionary != False:
        history.insert(0,dictionary["DATE"])
```

```
        history.insert(1,dictionary["MEAL_NAME"])
        history.insert(2,dictionary["CALORIES"])

        print("The date is : ", dictionary["DATE"])
        print("The name is : ", dictionary["MEAL_NAME"])
        print("The calories is : ", dictionary["CALORIES"])
        dictionary = ibm_db.fetch_assoc(prep)
    print(history)
    no_of_rows=len(history)//3
    print(no_of_rows)
    return
render_template("History.html",history=history,no_of_rows=no_of_rows)


@app.route("/addhistory",methods=["POST","GET"])
def addhistory():
    if request.method=='POST':
        date=request.form['date']
        meal_name=request.form['meal_name']
        calories=request.form['calories']
        insert_query="INSERT INTO trackhistory VALUES(?,?,?,?)"
        prep_stmt=ibm_db.prepare(conn,insert_query)
        ibm_db.bind_param(prep_stmt,1,session['email'])
        ibm_db.bind_param(prep_stmt,2,date)
        ibm_db.bind_param(prep_stmt,3,meal_name)
        ibm_db.bind_param(prep_stmt,4,calories)
        ibm_db.execute(prep_stmt)

    return redirect(url_for('history'))


@app.route("/support")
def support():
    return render_template("support.html")

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

8.ADVANTAGES & DISADVANTAGES

Advantages:

• By using our webapp, the user can know their BMI, which will lead the user to decide whether he has to gain weight or lose weight

• User can know their daily calorie intake, which can help them to know amount of calorie they can consume for that particular day.

• The user can upload the image of the meal which will provide them the nutritional value of that particular meal.

• The user can track the daily calorie intake which will help them to know their progress towards their fitness goal.

Disadvantages:

• It requires an active internet connection.

• Not all types of foods can be detected correctly by Clarifai Food Detection Model API.

• The user cannot update their personal details once it has been registered.

9. CONCLUSION

Since obesity rate has become a major problem in this decade, the diet management is very important. The information about the nutritional value of the food that has been printed in the food packages are not convenient to keep track of the daily calorie intake. This APP helps in finding the nutritional content present in the food with real time image processing using Clarifai Food Detection Model API and Spoonacular Nutrition API. The user can upload his daily meal image and get the nutritional value. They can also track their daily calorie intake.

10.FUTURE SCOPE :

This App will be upgraded in the following years with the feature of “Profile Updation”. The user can update his personal details like height, weight and age which will help them to keep track of the daily calorie intake and the BMI. “Dietary Recommendation” facility and “Water Reminder” facility will also be added in the future.