Nutrition Assistant Application

DOMAIN: CLOUD APP DEVELOPMENT

TEAM ID: PNT2022TMID53109

BATCH: B7-1A3E

TEAM MEMBERS:

1.	Ranjani A	: 195001089
2.	Sanjana A	: 195001097
3.	Varsha S	: 195001121
4.	Pooja M	: 195001076

CONTENTS

1. INTRODUCTION

a. **Project 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. 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.

b. **Purpose**

One of the most basic functions of such an app 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.

2. LITERATURE SURVEY

a. **Existing problem**

Due to the improvement in people's standards of living, obesity rates are increasing at an alarming speed, and this is reflective to the risks in 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.

● App-based nutrient dashboard systems which can analyze real time images of meal and analyze it for nutritional content can be very handy and improve the dietary habit

b. References

https://www.researchgate.net/publication/348459867_Dietitian_assistant_opportunities_within_the_nutrition_care_process_for_patients_with_or_at_risk_of_malnutrition_a_systematic_review

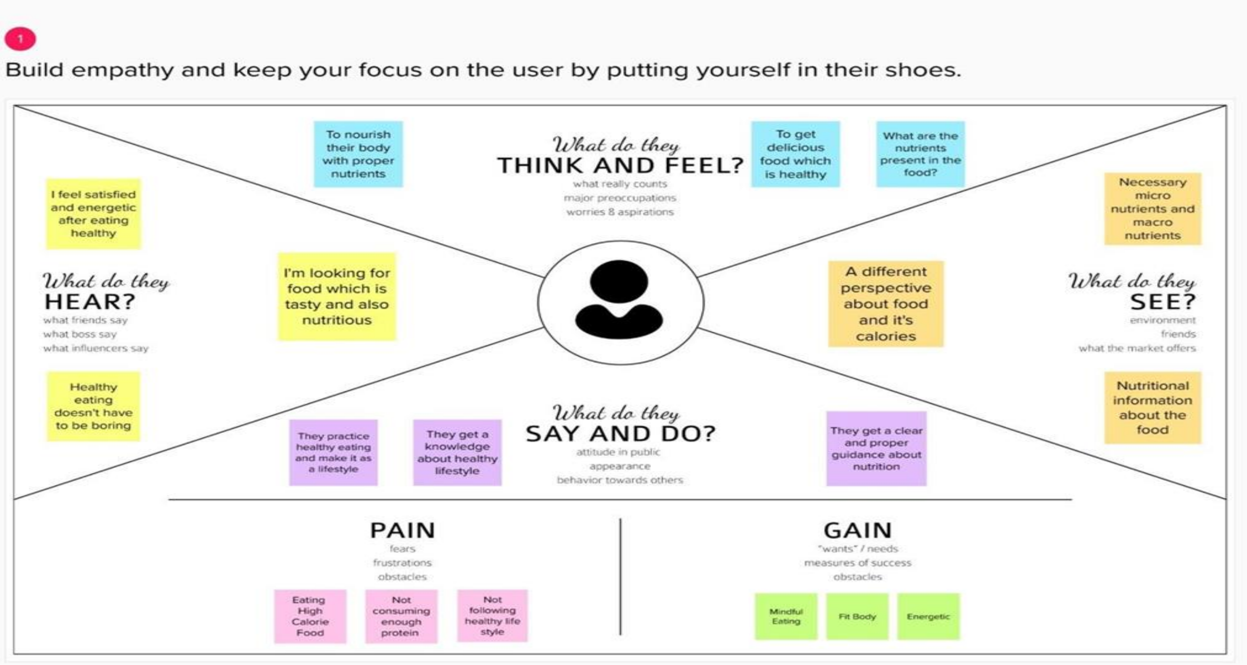
c. Problem Statement Definition

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. We know that in all food packets there is a label with its ingredients with its ratio and their calorie values. This is not much clear to make sense about our healthy diet calorie. This food detection model is introduced to understand about the calorie and nutrition value of a food that we are taking in our diet.

The initial step in this food detection model is to load a food image in a web. The image is passed to the server application, which uses Clarifai's AI-Driven Food Detection Model Service to analyze the images and Nutrition API to provide nutritional information about the analyzed Image which is stored in the database. The nutritional information of the analyzed image is displayed in the page. This is helpful in identifying the exact information regarding the nutrition values.

3. IDEATION & PROPOSEDSOLUTION

a. Empathy Map Canvas



1

Build empathy and keep your focus on the user by putting yourself in their shoes.

To nourish their body with proper nutrients

To get delicious food which is healthy

What are the nutrients present in the food?

I feel satisfied and energetic after eating healthy

I'm looking for food which is tasty and also nutritious

Healthy eating doesn't have to be boring

Necessary micro nutrients and macro nutrients

A different perspective about food and it's calories

Nutritional information about the food

They practice healthy eating and make it as a lifestyle

They get a knowledge about healthy lifestyle

They get a clear and proper guidance about nutrition

Eating High Calorie Food

Not consuming enough protein

Not following healthy life style

Mindful Eating

Fit Body

Energetic

b. **Ideation & Brainstorming**

Health nutrition prevents and controls diet related diseases.
Diet plans generated based on information gathered by user
There will be a dashboard for sharing health tips

c. **Proposed Solution**

Problem Statement

Many people are affected by anxiety and health issues. Unhealthy food Habits might increase cholesterol and lead to even heart attack. Due to the ignorance Of healthy food habits, obesity rates are increasing at an alarming speed. So, the Affected people should maintain the food calory value and diet plan to shift to a Healthy life style.

Idea Solution/Description

In this web application user will be able to know about the nutrition value of the Food.

- User have to take the picture of the food to know its nutritional value.
- Upload the picture in the web page.
- Then the user can view the nutritional value of the food picture in the result Page.

Novelty/Uniqueness

We will deliver a diet chart for individual based on their BMI (Body Mass Index), which is calculated at the BMI check point.

Social Impact/ Customer Satisfaction

Those who visit our site and properly follow their diet chart designed for them Will be able to easily overcome the obesity and make their body good and healthy .Because our site's information is authentic and a trusted one.

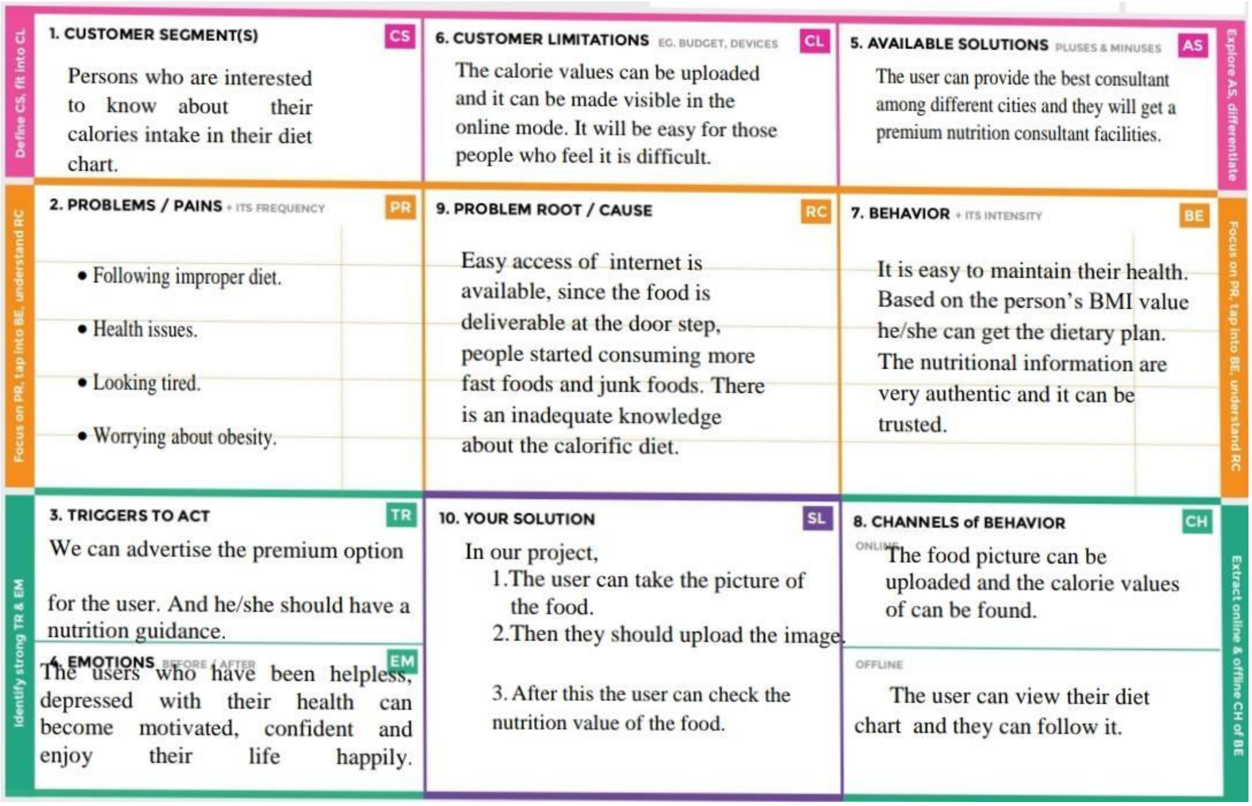
Business Model

We will advertise a premium option for the user. In premium, the user can Unlock a nutrition consultant page and they will get the consultant session with their Individual consultant where the user is allotted to one of the best consultants among the cities.

Scalability of Solution

Our solution that we provide is based on web application using client And server model. So, here the solution scalability is about how long the server runs And how long it takes for the client to get a response.

4. Problem Solution fit:



Define CS, fit into CL

1. CUSTOMER SEGMENT(S) CS

Persons who are interested to know about their calories intake in their diet chart.

6. CUSTOMER LIMITATIONS EG. BUDGET, DEVICES CL

The calorie values can be uploaded and it can be made visible in the online mode. It will be easy for those people who feel it is difficult.

5. AVAILABLE SOLUTIONS PLUSES & MINUSES AS

The user can provide the best consultant among different cities and they will get a premium nutrition consultant facilities.

Explore AS, differentiate

Focus on PR, tap into BE, understand RC

2. PROBLEMS / PAINS + ITS FREQUENCY PR

- Following improper diet.
- Health issues.
- Looking tired.
- Worrying about obesity.

9. PROBLEM ROOT / CAUSE RC

Easy access of internet is available, since the food is deliverable at the door step, people started consuming more fast foods and junk foods. There is an inadequate knowledge about the calorific diet.

7. BEHAVIOR + ITS INTENSITY BE

It is easy to maintain their health. Based on the person's BMI value he/she can get the dietary plan. The nutritional information are very authentic and it can be trusted.

Focus on PR, tap into BE, understand RC

Identify strong TR & EM

3. TRIGGERS TO ACT TR

We can advertise the premium option

for the user. And he/she should have a nutrition guidance.

4. EMOTIONS BEFORE / AFTER EM

The users who have been helpless, depressed with their health can become motivated, confident and enjoy their life happily.

10. YOUR SOLUTION SL

In our project,

1.The user can take the picture of the food.

2.Then they should upload the image.

3. After this the user can check the nutrition value of the food.

8. CHANNELS of BEHAVIOR CH

ONLINE

The food picture can be uploaded and the calorie values of can be found.

OFFLINE

The user can view their diet chart and they can follow it.

Extract online & offline CH of BE

5.REQUIREMENT ANALYSIS

a. 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

FR-2	User Confirmation	Confirmation via Email
FR-3	Search for donor	Search result can be viewed in a list. Each element in the list represents a specific donorwith the donordetails.
FR-4	Administrator	Monitor the overall functionalities of theapplication and ensure quality of service
FR-5	Send Request	If plasma is required, the requester places a requestto the admin.
FR-6	User Credentials	Administrator has registered then the software administrator should be able to login to the web application. The login information will be stored on the database for future use.

b. Non-Functional Requirements

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

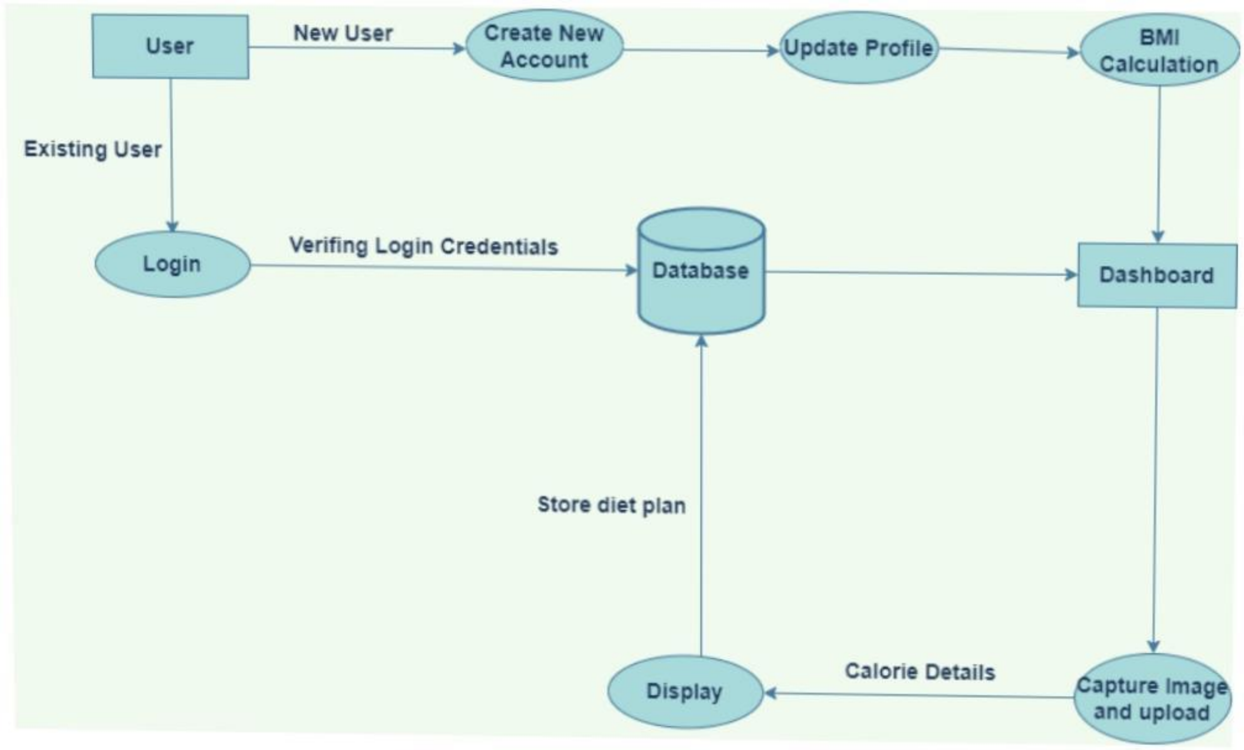
FR No.	Non-Functional Requirement	Description

NFR-1	Usability	Plasma donor Application is very usefulto the emergency situation patient, becausethat application givesthe information of the nearbyplasma donors and request to donate theirplasmato patientvia email, SMS etc.
NFR-2	Security	Secured website and application that provides varioussecurity features EmailVerification, password login etc.
NFR-3	Reliability	The plasma donorapplication should workproperly,even when faults occur.
NFR-4	Performance	The plasma donor application must perform well in different scenarios. The LoadTime should be lessand there needs to be quick response to the request saying thatdelay is less.
NFR-5	Availability	The system shouldbe available all times, meaning the user can access it using application. In case if a hardware failure or database corruption ,a replacement page will be shown. Also, in case of a hardware failure or database corruption, backups ofthe database should be retrieved from the application data folder and saved by the administrator. Must be available 24/7.

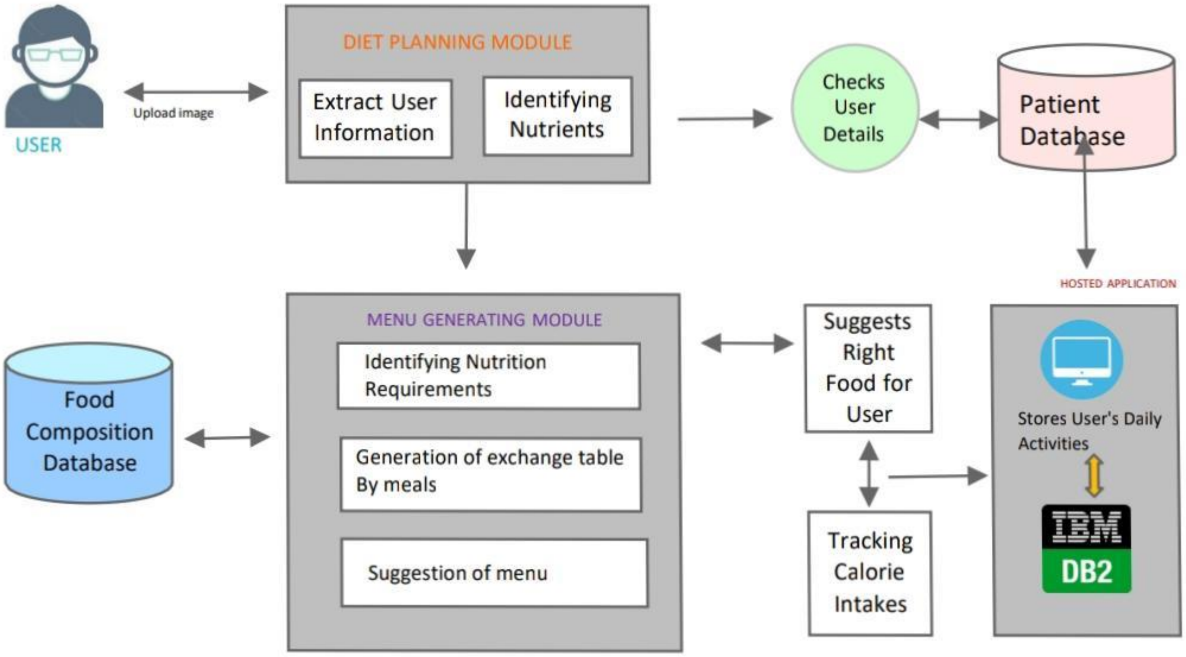
NFR-6	Scalability	In the application to handle an increase in workload withoutperformance degradation, or its abilityto quickly enlarge. The solution must allow the hardware end of the deployed software services and components to be scaledhorizontally as well as vertically.

5.Project Design :

5.1 Data Flow Diagram:



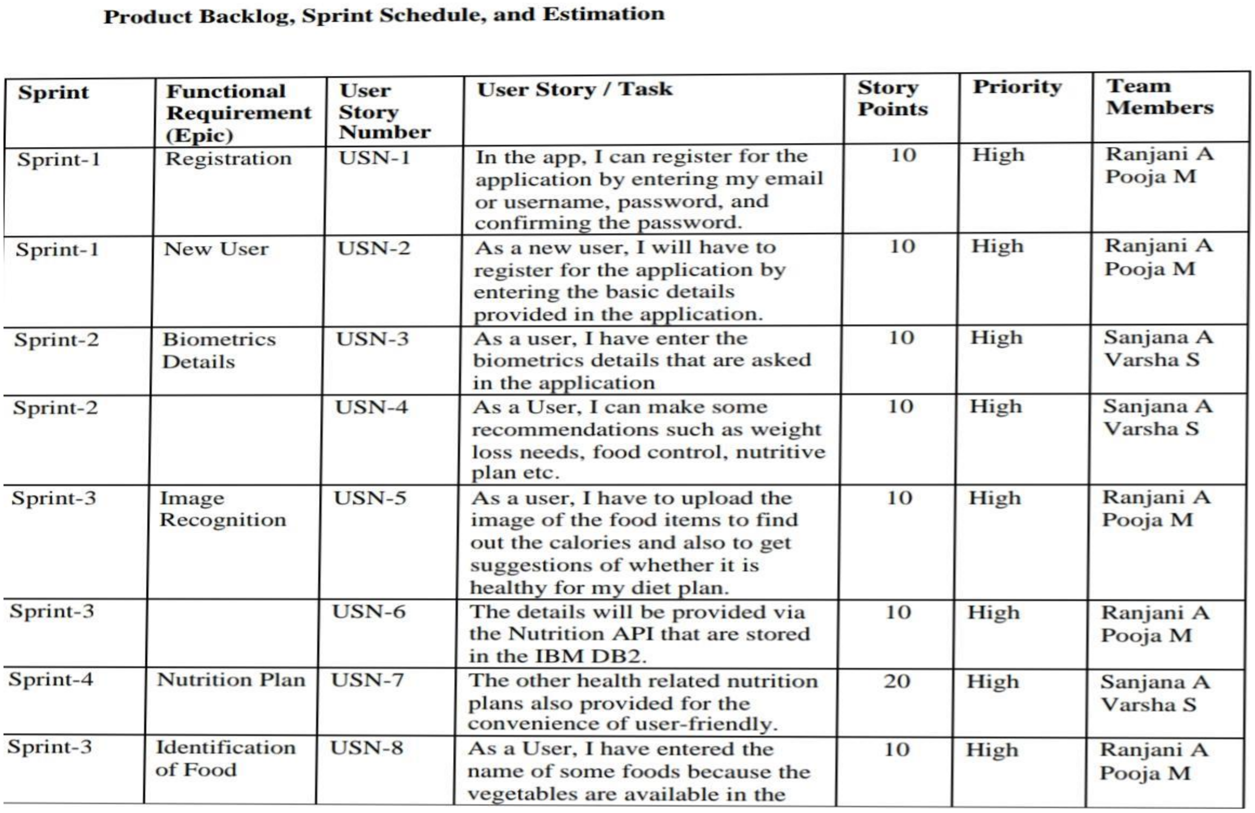
5.2 Solution and technical Architecture:



<u>6.Project Planning and Scheduling :</u>

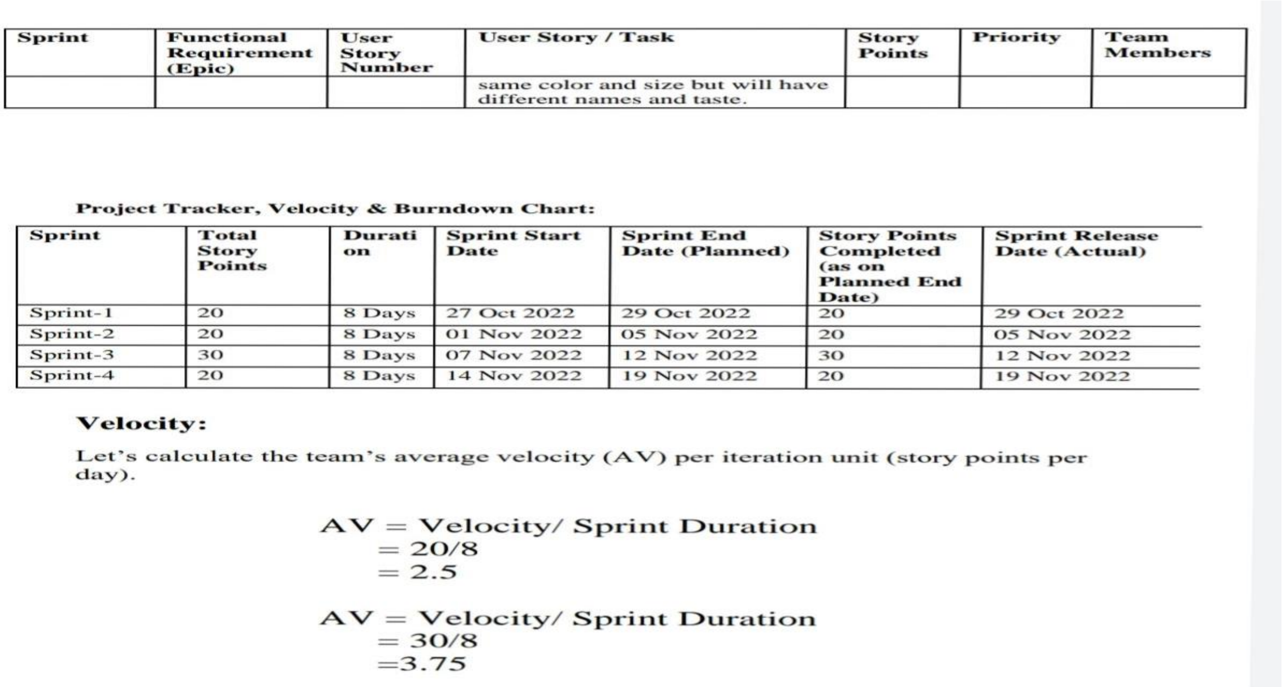
<u>6.1 Sprint Planning and Estimation :</u>

6.2 Sprint Delivery Schedule:



Product Backlog, Sprint Schedule, and Estimation

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-1	Registration	USN-1	In the app, I can register for the application by entering my email or username, password, and confirming the password.	10	High	Ranjani A Pooja M
Sprint-1	New User	USN-2	As a new user, I will have to register for the application by entering the basic details provided in the application.	10	High	Ranjani A Pooja M
Sprint-2	Biometrics Details	USN-3	As a user, I have enter the biometrics details that are asked in the application	10	High	Sanjana A Varsha S
Sprint-2		USN-4	As a User, I can make some recommendations such as weight loss needs, food control, nutritive plan etc.	10	High	Sanjana A Varsha S
Sprint-3	Image Recognition	USN-5	As a user, I have to upload the image of the food items to find out the calories and also to get suggestions of whether it is healthy for my diet plan.	10	High	Ranjani A Pooja M
Sprint-3		USN-6	The details will be provided via the Nutrition API that are stored in the IBM DB2.	10	High	Ranjani A Pooja M
Sprint-4	Nutrition Plan	USN-7	The other health related nutrition plans also provided for the convenience of user-friendly.	20	High	Sanjana A Varsha S
Sprint-3	Identification of Food	USN-8	As a User, I have entered the name of some foods because the vegetables are available in the same color and size but will have different names and taste.	10	High	Ranjani A Pooja M



Project Tracker, Velocity & Burndown Chart:

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	8 Days	27 Oct 2022	29 Oct 2022	20	29 Oct 2022
Sprint-2	20	8 Days	01 Nov 2022	05 Nov 2022	20	05 Nov 2022
Sprint-3	30	8 Days	07 Nov 2022	12 Nov 2022	30	12 Nov 2022
Sprint-4	20	8 Days	14 Nov 2022	19 Nov 2022	20	19 Nov 2022

Velocity:

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

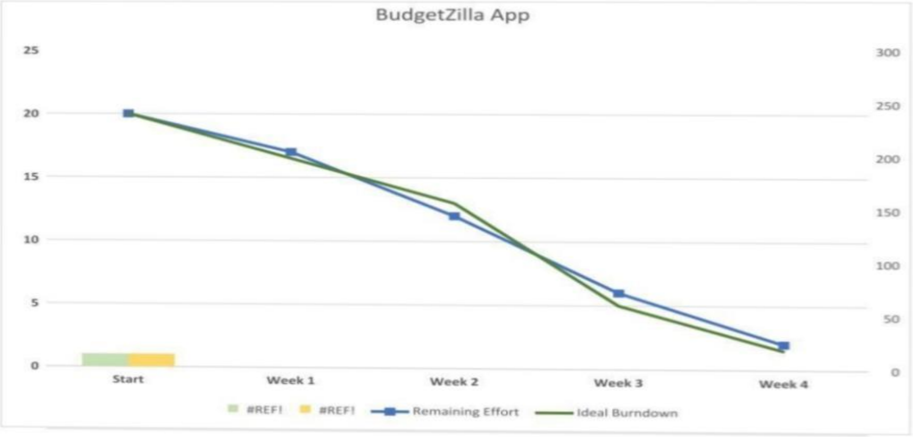
$$AV = \text{Velocity}/\ \text{Sprint Duration} = 20/8 = 2.5$$

$$AV = \text{Velocity}/\ \text{Sprint Duration} = 30/8 = 3.75$$

Burndown Chart:

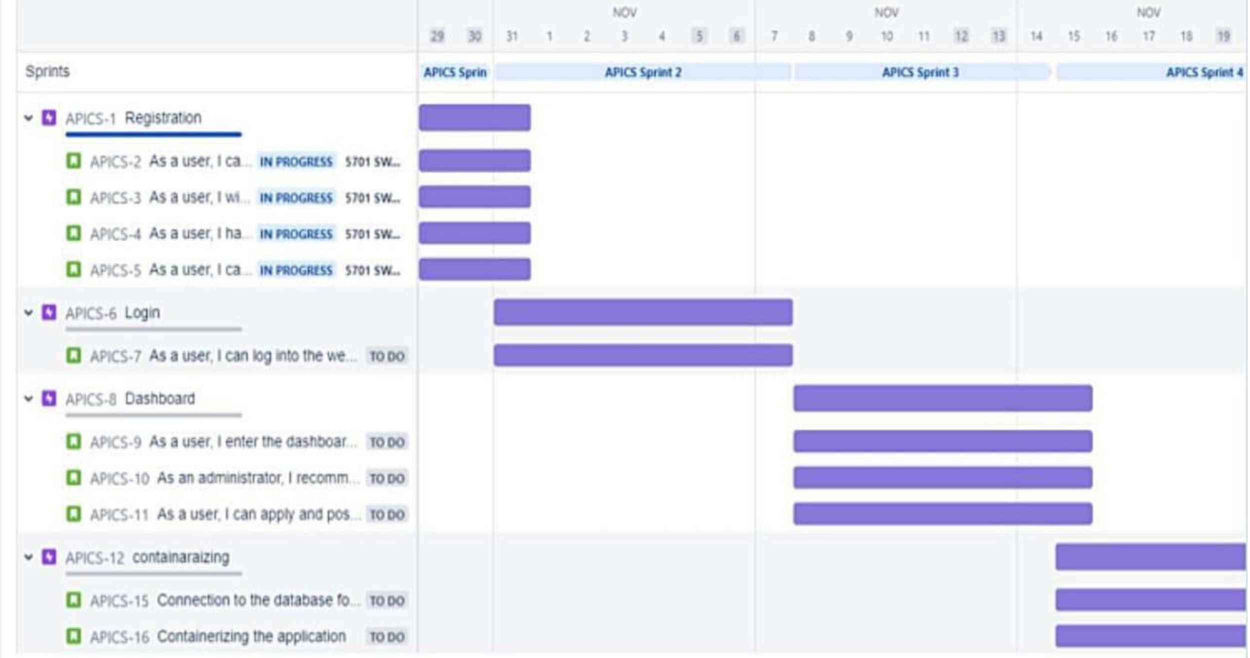
X-axis - Day

Y-axis - Story Points



6.3 Report From JIRA :

7.Coding and Solutioning:

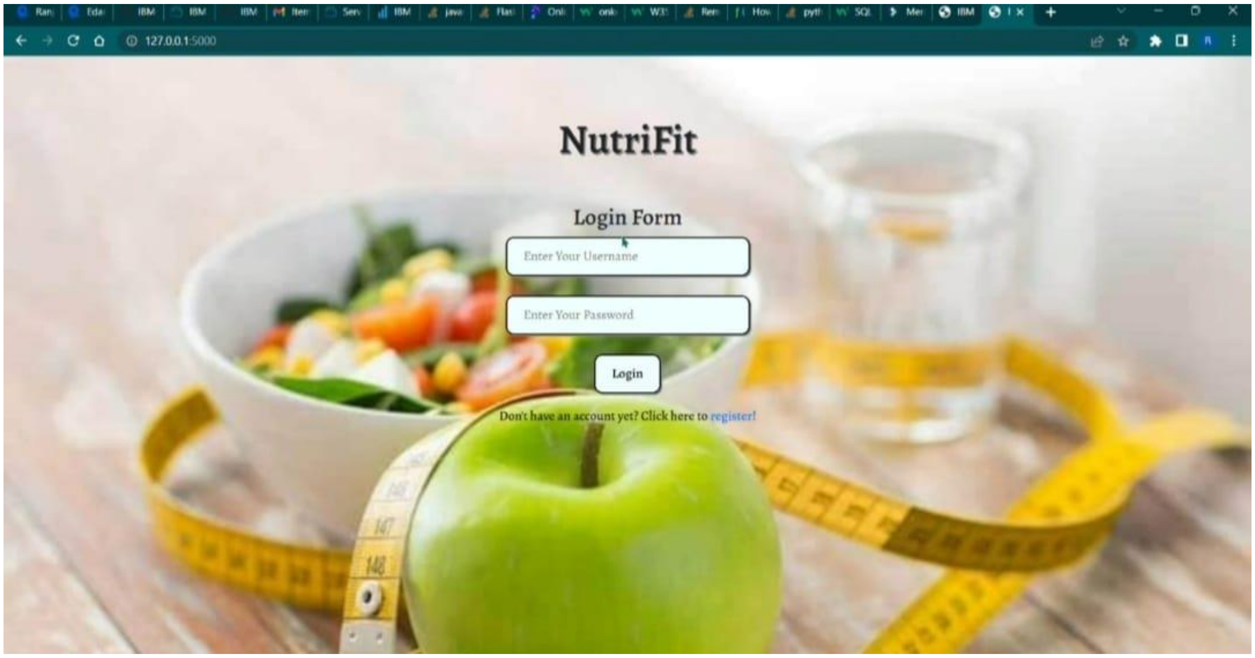


Sprints		
APICS-1 Registration		
APICS-2 As a user, I ca...	IN PROGRESS	5701 SW...
APICS-3 As a user, I wi...	IN PROGRESS	5701 SW...
APICS-4 As a user, I ha...	IN PROGRESS	5701 SW...
APICS-5 As a user, I ca...	IN PROGRESS	5701 SW...
APICS-6 Login		
APICS-7 As a user, I can log into the we...	TO DO	
APICS-8 Dashboard		
APICS-9 As a user, I enter the dashboar...	TO DO	
APICS-10 As an administrator, I recomm...	TO DO	
APICS-11 As a user, I can apply and pos...	TO DO	
APICS-12 containaraizing		
APICS-15 Connection to the database fo...	TO DO	
APICS-16 Containerizing the application	TO DO	

7.1 Feature 1 :

- **Login/signup**

```
<!DOCTYPE html>
<html lang="en">
<head></br></br></br>
    <h1 style="font-size: 50px; text-shadow: 2px 2px grey; font-weight: bold; font-family: 'Alegreya', serif !important;"><center>NutriFit</center></h1>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Nutrtion Assistant Application</title>
        <!-- favicon -->
        <!-- <link rel="shortcut icon" href="/assets/img/favicon.ico" type="image/x-icon"> -->
        <!-- <link rel="icon" href="/assets/img/favicon.ico" type="image/x-icon"> -->
        <link rel="icon" type="image/png" sizes="16x16" href="/assets/img/favicon-32x32.png">
        <!-- bootstrap css cdn -->
        <link rel="stylesheet" href="https://stackpath.bootstrapcdn.com/bootstrap/4.5.2/css/bootstrap.min.css" integrity="sha384-JcKb8q3iqJ61gNV9KGb8thSsNjpSL0n8PARn9HuZOnIxN0hoP+VmmDGMN5t9UJ0Z" crossorigin="anonymous">
        <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/4.7.0/css/font-awesome.css">
        <!-- css stylesheet -->
        <link rel="stylesheet" href="../static/style.css">
        <!-- font styles cdn -->
        <link rel="preconnect" href="https://fonts.gstatic.com">
        <link href="https://fonts.googleapis.com/css2?family=Alegreya&display=swap" rel="stylesheet">
        <link
href="https://fonts.googleapis.com/css2?family=Alegreya:wght@6
00&display=swap" rel="stylesheet">
        <style>
            body {
              background-image:
url('https://images.jdmagicbox.com/comp/bangalore/h3/080pxx80.
xx80.111217105334.y7h3/catalogue/swastha-nutrition-and-diet-
counseling-jp-nagar-3rd-phase-bangalore-dietitians-
lfbo8fib6y.jpg');
              background-repeat: no-repeat;
              background-attachment: fixed;
              background-size: cover;
            }
            #button
            {
                background-color:azure;
                padding: 10px 20px;
                border: 2px solid black;
                box-shadow: 2px 2px grey;
                color:black;
                font-weight: bold;
            }
            input{
                background-color:azure;
                width:300px;
                border: 2px solid black;
                box-shadow: 2px 2px grey;
                border-radius: 10px;
            }
            h3{
                font-weight: bold;
            }
            p{
                font-weight: bold;
            }
            </style>
    </head>
<body>

      <!-- navbar ends -->
    <!-- Login form -->

    <div class="login text-center mt-5">
        <h3> Login Form </h3>
        <form action="/" method="post">
        <div class="msg">{{ msg }}</div>
            <input type="text" name="username" placeholder="Enter
Your Username"  id="username" required></br></br>
                          <input type="password" name="password"
placeholder="Enter Your Password" id="password"
required></br></br>
        <button type="submit" id="button" class="btn btn-primary">
Login </button>
        </form>
    </div>

    <div class="note mt-3 text-center"> <!--Register form -->
    <p> Don't have an account yet? Click here to <a
href="register">register! </a> </p>
    </div>
</body>
</html>
```



NutriFit

Login Form

Enter Your Username

Enter Your Password

Login

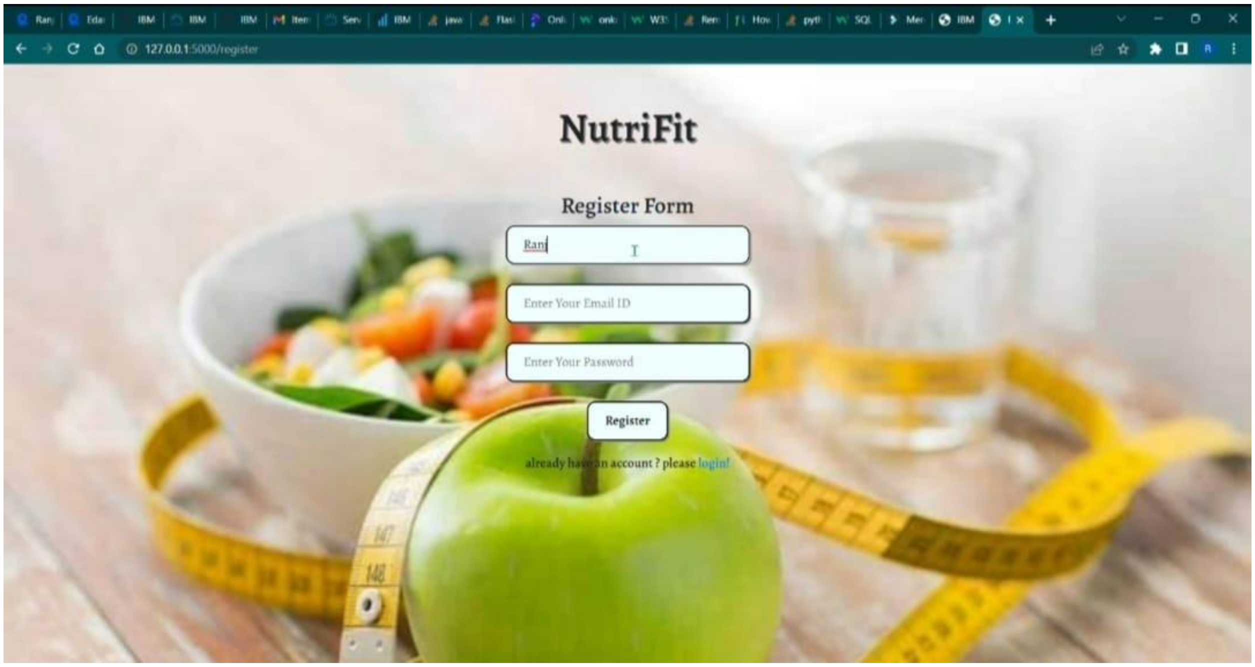
Don't have an account yet? Click here to register!

Sign up

```
<!DOCTYPE html>
<html lang="en">
<head></br></br>
   <h1 style="font-size: 50px; text-shadow: 2px 2px grey; font-weight: bold; font-family: 'Alegreya', serif !important;"><center>NutriFit</center></h1>
   <meta charset="UTF-8">
   <meta http-equiv="X-UA-Compatible" content="IE=edge">
   <meta name="viewport" content="width=device-width, initial-scale=1.0">
   <title>Nutrtion Assistant Application</title>
      <!-- favicon -->
      <!-- <link rel="shortcut icon" href="/assets/img/favicon.ico" type="image/x-icon"> -->
      <!-- <link rel="icon" href="/assets/img/favicon.ico" type="image/x-icon"> -->
      <link rel="icon" type="image/png" sizes="16x16" href="/assets/img/favicon-32x32.png">
      <!-- bootstrap css cdn -->
```

```
        <link rel="stylesheet"
href="https://stackpath.bootstrapcdn.com/bootstrap/4.5.2/css/boots
trap.min.css" integrity="sha384-
JcKb8q3iqJ61gNV9KGb8thSsNjpSL0n8PARn9HuZOnIxN0hoP+
VmmDGMN5t9UJ0Z" crossorigin="anonymous">
        <link rel="stylesheet"
href="https://cdnjs.cloudflare.com/ajax/libs/font-
awesome/4.7.0/css/font-awesome.css">
        <!-- css stylesheet -->
        <link rel="stylesheet" href="../static/style.css">
        <!-- font styles cdn -->
        <link rel="preconnect" href="https://fonts.gstatic.com">
        <link
href="https://fonts.googleapis.com/css2?family=Alegreya&display
=swap" rel="stylesheet">
        <link
href="https://fonts.googleapis.com/css2?family=Alegreya:wght@6
00&display=swap" rel="stylesheet">
        <style>
            body {
              background-image:
url('https://images.jdmagicbox.com/comp/bangalore/h3/080pxx80.
xx80.111217105334.y7h3/catalogue/swastha-nutrition-and-diet-
counseling-jp-nagar-3rd-phase-bangalore-dietitians-
lfbo8fib6y.jpg');
              background-repeat: no-repeat;
              background-attachment: fixed;
              background-size: cover;
            }
            #button
            {
                background-color:azure;
                padding: 10px 20px;
                border: 2px solid black;
                box-shadow: 2px 2px grey;
            color:black;
            font-weight: bold;
         }
         input{
            background-color:azure;
            width:300px;
            border: 2px solid black;
            box-shadow: 2px 2px grey;
            border-radius: 10px;
         }
         h3{
            font-weight: bold;
         }
         p{
            font-weight: bold;
         }
         </style>
   </head>
<body>

     <!-- navbar ends -->
   <!-- Login form -->
   <div class="login text-center mt-5">
      <h3> Register Form </h3>
      <form action="/register" method="post">
      <div class="msg">{{ msg }}</div>
         <!-- <input type="text" placeholder="fullname"
id="fullname"> </br></br> -->

         <input type="text" name="username" placeholder="Enter
Your Username"  id="username" required></br></br>
                    <input type="email" name="email"
placeholder="Enter Your Email ID"  id="email"
required></br></br>
```

```
                        <input type="password" name="password"
placeholder="Enter Your Password" id="password"
required></br></br>

       <button type="submit" id="button" class="btn btn-primary">
Register </button>
       </form>
    </div>

    <div class="note mt-3 text-center"> <!--Register form -->
    <p> already have an account ? please  <a href="/">login! </a>
</p>

    </div>
</body>
</html>
```



127.0.0.1:5000/register

NutriFit

Register Form

Ran

Enter Your Email ID

Enter Your Password

already have an account ? please login!

- **Personal information**

<html>

```
<head>
        <link rel="stylesheet" href="../static/personal.css">
    <style>
        body {
          background-image:
url('https://images.jdmagicbox.com/comp/bangalore/h3/080pxx80.xx80.
111217105334.y7h3/catalogue/swastha-nutrition-and-diet-counseling-
jp-nagar-3rd-phase-bangalore-dietitians-lfbo8fib6y.jpg');
          background-repeat: no-repeat;
          background-attachment: fixed;
          background-size: cover;
        }
        </style>
</head>
<style>
        #navbar button{

                            border: 3px solid grey;
                            background-image:linear-gradient(to right,rgb(250,
234, 154) ,lightgreen);
                            border-radius: 10px;
                            width: 250px;
                            padding: 5px;
                            margin-left: 15px;
                            margin-top: 20px;
                            margin-right: 15px;
                            cursor:pointer;
                            height: 50px;
                            font-weight: bold;
                            font-size: 20px;
                            box-shadow: 2px 2px grey;
                          }
        button:hover
        {
          color: black;
        }
        </style>
</head>
<body>
  <div class="main">
        <div id="navbar">
          <br>
          <nav>
                  <a href="daily_update"><button  style="margin-left:
80px;"><span>Daily Update</span></button></a>
                  <a href="uploadimage"><button><span>Upload
Image</span></button></a>
                  <a
href="premium"><button><span>Premium</span></button></a>
                  <a href="user_diet_info"><button ><span>Personal
Information</span></button></a>
                  <a href="logout"><button ><span>Log
Out</span></button></a>

          </nav>
  </div>
<body>
        <div id="box1">

                <form action="none" method="post" name="bmiform">
                        <fieldset>
                        <legend>Height and Weight</legend>
                                <label for="pounds">Weight (pounds):</label>
                                <input type="text" name="pounds" id="pounds"
required>
                                <br>
                                <label for="inches">Height (inches):</label>
                                <input type="text" name="inches" id="inches"
required>
```

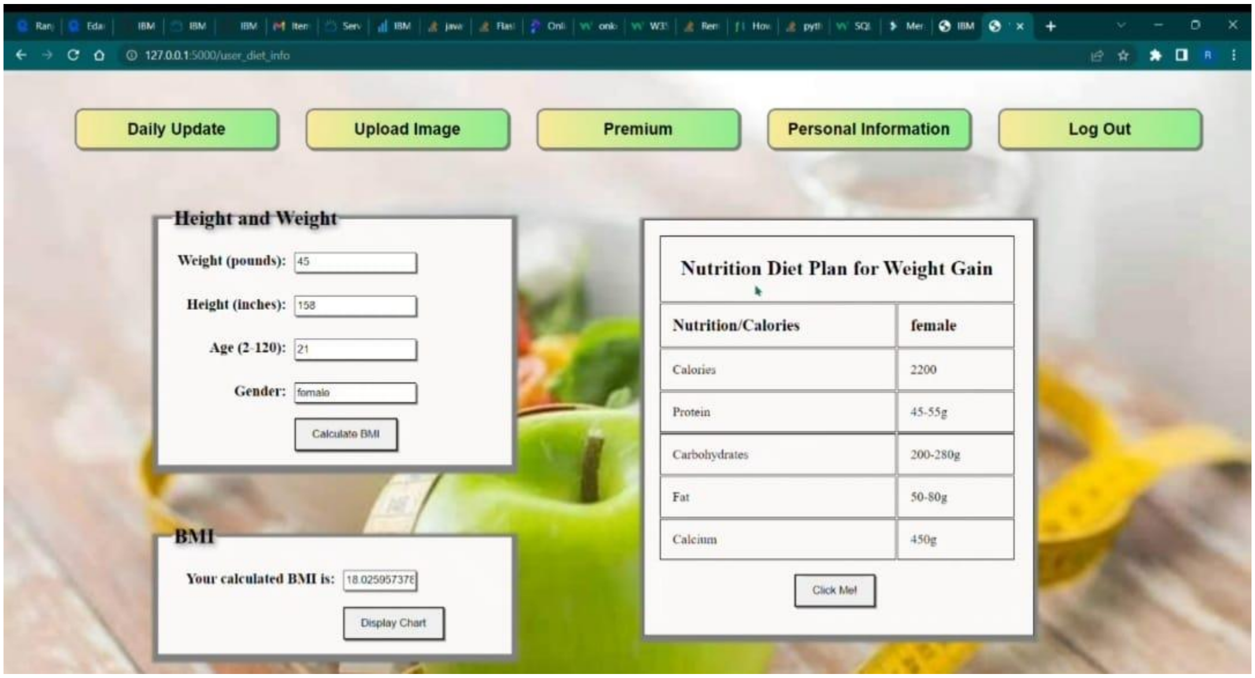
```
                                <br>
              <label for="age">Age (2-120):</label>
                                <input type="text" name="age" id="age"
required>
                                <br>
              <label for="gender">Gender:</label>
                                <input type="text" name="gender" id="gender"
required>
                                <br>

                                <button type="button" name="calculate"
value="Calculate your BMI" onclick="calcBMI()">Calculate
BMI</button>
                        </fieldset>

                        <fieldset>
                        <legend>BMI</legend>
                                <label for="bmi" id="bmi_label">Your calculated
BMI is:</label>
                                <input type="text" name="bmi" id="bmi"
readonly="readonly">
                                <button type="button" id="btn1" name="chart"
value="chart display" onClick="myfun()">Display Chart</button>
                        </fieldset>
                </form>
        </div>
                <!-- <div class="msg">{{ msg }}</div> -->



        <table id="myTable1" class="hidden">
                <tr>
                        <th colspan="3" style="padding:25px;" id="weight-
info"></th>
                </tr>
                <tr style="font-size:20px; font-weight:bold;">
```

```
               <td>Nutrition/Calories</td>
               <td id="gender-info"></td>
          </tr>
    <tr">
    <td>Calories</td>
    <td id="calories"></td>
   </tr>
   <tr>
    <td>Protein</td>
    <td id="protein"></td>
   </tr>
   <tr>
    <td>Carbohydrates</td>
    <td id="carbohydrates"></td>
   </tr>
   <tr>
    <td>Fat</td>
    <td id="fat"></td>
   </tr>
   <tr>
    <td>Calcium</td>
    <td id="calcium"></td>
   </tr>
      <tr>


      <form action="/user_diet_info" method="post">
          <input type="hidden" id="protein1" name="protein1">
          <input type="hidden"  id="calories1" name="calories1">
          <input type="hidden"  id="carbohydrates1"
name="carbohydrates1">
          <input type="hidden" id="fat1" name="fat1">
          <input type="hidden" id="calcium1" name="calcium1">
```

```
                <td colspan="3" style="border-color: white;"><button type="submit" onclick="alert('Successfully Updated !')">Click Me!</button></td>

        </form>
</tr>
        </table>

     <script >
                function calcBMI() {
    var weight = document.bmiform.pounds.value, height = document.bmiform.inches.value;
    document.bmiform.bmi.value = parseFloat((weight) / ((height/100) * (height/100)));
  }

  function myfun()
  {
    bmi = document.getElementById("bmi").value;
    gender = document.getElementById("gender").value;
    document.getElementById("gender-info").innerText = gender;
    var dietplan ="";
    var calo = 0
    var pro = ""
    var carbo = ""
    var fat = ""
    var calci = ""

    if(gender == "male" || gender == "Male")
    {
        if(bmi < 18.5)
        {
            // document.getElementById("table1");
            // toggleTable1();
            // return false;
            dietplan = "Nutrition Diet Plan for Weight Gain"
            calo = 2800
            pro = "55-60g"
            carbo = "240-300g"
            fat = "40-100g"
            calci = "450g"

        }
        if(bmi > 18.5 && bmi < 22.9)
        {
            // document.getElementById("table2");
            // toggleTable2();
            // return false;
            dietplan = "Nutrition Diet Plan for Normal Weight"
            calo = 2400
            pro = "55g"
            carbo = "240-300g"
            fat = "40-80g"
            calci = "420g"
        }
        if(bmi > 22.9)
        {
            // document.getElementById("table3");
            // toggleTable3();
            // return false;
            dietplan = "Nutrition Diet Plan for Weight Loss"
            calo = 1800
            pro = "30-130g"
            carbo = "200-240g"
            fat = "30-70g"
            calci = "450mg"
        }
    }
    if(gender == "female" || gender == "Female")
    {
```

```
if(bmi < 18.5)
{
   // document.getElementById("table4");
   // toggleTable4();
   // return false;
   dietplan = "Nutrition Diet Plan for Weight Gain"
   calo = 2200
   pro = "45-55g"
   carbo = "200-280g"
   fat = "50-80g"
   calci = "450g"
}
if(bmi > 18.5 && bmi < 22.9)
{
   // document.getElementById("table5");
   // toggleTable5();
   // return false;
   dietplan = "Nutrition Diet Plan for Normal Weight"
   calo = 2200
   pro = "45g"
   carbo = "200-280g"
   fat = "50-80g"
   calci = "420g"
}
if(bmi > 22.9)
{
   // document.getElementById("table6");
   // toggleTable6();
   // return false;
   dietplan = "Nutrition Diet Plan for Weight Loss"
   calo = 1500
   pro = "60-120g"
   carbo = "180-220g"
   fat = "25-50g"
   calci = "450g"
```

```
        }
      }
      document.getElementById("myTable1").style.display ='block';
      document.getElementById('weight-info').innerText = dietplan;
      document.getElementById('calories').innerText = calo;
      document.getElementById('protein').innerText = pro;
      document.getElementById('carbohydrates').innerText = carbo;
      document.getElementById('fat').innerText = fat;
      document.getElementById('calcium').innerText = calci;

      document.getElementById('calories1').value  = calo;
      document.getElementById('protein1').value = pro;
      document.getElementById('carbohydrates1').value = carbo;
      document.getElementById('fat1').value = fat;
      document.getElementById('calcium1').value = calci;
      console.log(dietplan + "\n" +
document.getElementById('calcium1').value);
      // document.getElementById("myTable1").classList.toggle("hidden");
    }

    function toggleTable1() {
      document.getElementById("myTable1").classList.toggle("hidden");
    }

          </script>
</body>
</html>
```



Height and Weight

Weight (pounds): 45

Height (inches): 158

Age (2-120): 21

Gender: femalo

BMI

Your calculated BMI is: 18.025957378

Nutrition Diet Plan for Weight Gain

Nutrition/Calories	female
Calories	2200
Protein	45-55g
Carbohydrates	200-280g
Fat	50-80g
Calcium	450g

7.2 Feature 2

- Daily update

```
<!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” />

    <title>Nutrtion Assistant Application</title>

    <link rel=”preconnect” href=https://fonts.googleapis.com>
    <link rel=”preconnect” href=https://fonts.gstatic.com
crossorigin>

    <link
href=https://fonts.googleapis.com/css2?family=Montserrat:ital,wg
ht@0,100;0,200;0,300;0,400;0,500;0,600;0,700;0,800;1,400;1,600
&display=swap rel=”stylesheet”>

    <link rel=”stylesheet” href=”../static/stylesheet.css”/>




  </head>

  <style>

    #navbar button{



              Border: 3px solid grey;

              Background-image:linear-gradient(to right,rgb(250, 234,
154) ,lightgreen);

              Border-radius: 10px;

              Width: 250px;

              Padding: 5px;

              Margin-left: 15px;
```

```
          Margin-top: 20px;

          Margin-right: 15px;

          Cursor:pointer;

          Height: 50px;

          Font-weight: bold;

          Font-size: 20px;

          Box-shadow: 2px 2px grey;

        }

Button:hover

{

  Color: #fff;



}

Table

    {

        Margin-top: 70px;

        Margin-left:400px;

        Border: 3px solid grey;
        Box-shadow: 0px 3px 3px 5px  grey;
        Padding: 17px;
        Width: 700px;
        Background-color:rgb(252, 249, 249);
    }

    Td{
        Padding:20px;

    }

    Table, th, td {
        Border: 1px solid black;

    }

    Th{
      Font-size: 15px;

      Font-weight: bold;

    }

  </style>

</head>

<body>

 <div class=”main”>

  <div id=”navbar”>

   <br>

   <nav>

     <a href=”daily_update”><button  style=”margin-left: 80px;”><span>Daily Update</span></button></a>

     <a href=”uploadimage”><button><span>Upload Image</span></button></a>

     <a href=”premium”><button><span>Premium</span></button></a>

     <a href=”user_diet_info”><button ><span>Personal Information</span></button></a>

     <a href=”logout”><button ><span>Log Out</span></button></a>
```

```
      </nav>

  </div>

  <br>

    <div class=”main”>

        <div id=”continer”>



            <div id=”form-search”>

                <FORM NAME=”myform” action=”/daily_update”
id=”cointiner-inline” METHOD=”post”>

                        <INPUT TYPE=”text” placeholder=”Enter
Product” id=”product-name” NAME=”product-name”
VALUE=””>

                          <INPUT TYPE=”button” style=”margin-left:
10px;margin-top:0px;width: 250px;” class=”button”
NAME=”button1” Value=”Get Available Nutrients”
onClick=”getNutrients()”>

                       <br>

                      <input type=”hidden” value=0 id=”protein1”
name=”protein1”>

                      <input type=”hidden” value=0 id=”calories1”
name=”calories1”>
```

```
          <input type=”hidden” value=0 id=”carbohydrates1”
name=”carbohydrates1”>

          <input type=”hidden” value=0 id=”fat1”
name=”fat1”>

          <input type=”hidden” value=0 id=”calcium1”
name=”calcium1”>

          <INPUT TYPE=”submit” style=”margin-left: -
250px;margin-top:70px;width: 250px;” class=”button”
NAME=”button1” Value=”Remaining Intake Nutrients”>

           <br>

         </form>

         <div id=”cointiner-inline” style=”margin-top:-70px”>

         <input type=”file”  accept=”image/*”
id=”imageInput” >

           </div>

       </div>



      </div>

      <div class=”msg” id = “msg1”>

       <div class=”bcol”>
```

```
         <table id=”mytable1”>



          <!--  msg = [‘Calories’, str(calo) , ‘Proteins’,
str(protein) + “ g”, ‘Carbohydrates’, str(carbo) + “ g”, ‘Fat’, str(fat)
+ “ g”, ‘Calcium’, str(calci) + “ g”]

 →

          <th>{{msg[2]}}</th>

          <th>{{msg[3]}}</th>

        </tr>

        <tr>

          <th>{{msg[0]}}</th>

          <th>{{msg[1]}}</th>

         </tr>

        <tr>

          <th>{{msg[4]}}</th>

          <th>{{msg[5]}}</th>

        </tr>

        <tr>

          <th>{{msg[6]}}</th>
```

```
    <th>{{msg[7]}}</th>
  </tr>

    <tr>
      <th>{{msg[8]}}</th>
      <th>{{msg[9]}}</th>
    </tr>
    </table>
 </div>

 </div>
<div class=”box2”>

  <div class=”bcol”>
    <table id=”mytable2” >
     <tr>
```

```
                      <th colspan=”2” style=”background-color:#ffffff
;margin-top: 25px;font-weight: bold;padding-left:
15px;”>Nutrition Facts</th>

                    </tr>

                      <tr>

                          <th>Calories</th>

                          <th id=”calories”></th>

                        </tr>

                      <tr>

                          <th>Total Fat</th>

                          <th id=”totalfat”></th>

                      </tr>

                      <tr>

                          <th>Cholesterol</th>

                          <th id=”cholesterol”></th>

                      </tr>

                      <tr>

                          <th>Carbohydrates</th>

                          <th id=”carbohydrates”></th>
```

```
  </tr>

<tr>

    <th>Calcium</th>

    <th id=”calcium”></th>

</tr>

<tr>

    <th>Sodium</th>

    <th id=”sodium”></th>

</tr>

<tr>

    <th>Potassium</th>

    <th id=”potassium”></th>

</tr>

<tr>

    <th>Sugar</th>

    <th id=”sugar”></th>

  </tr>
```

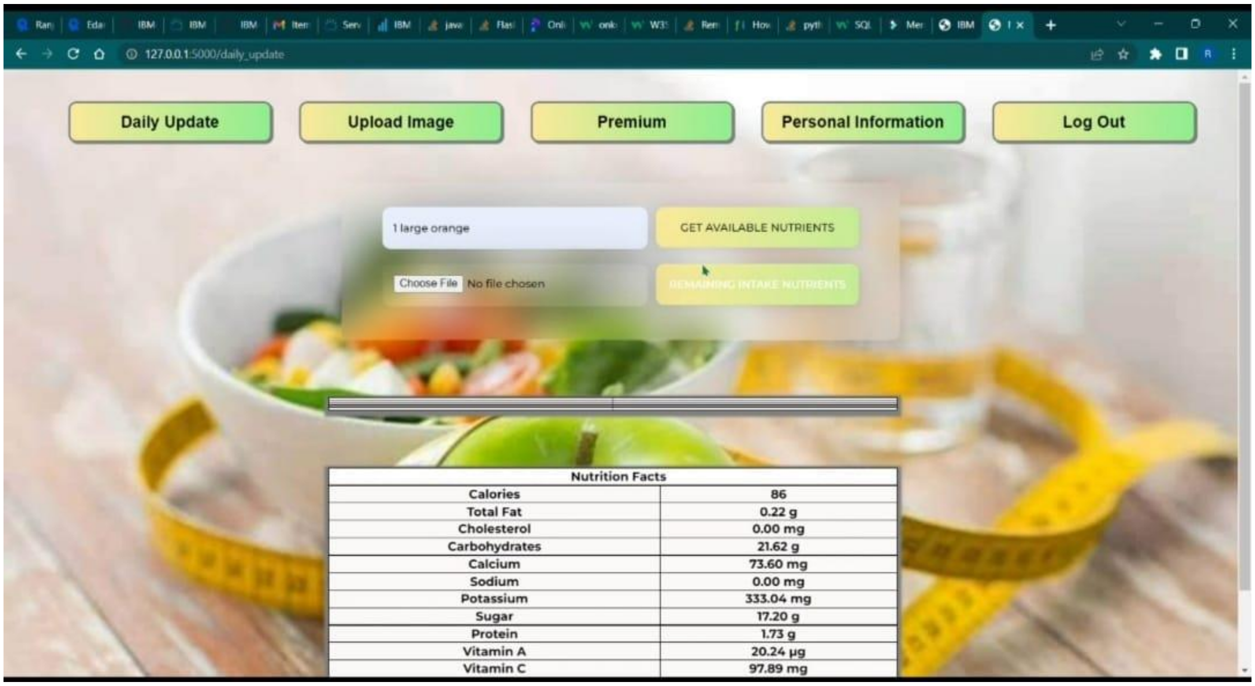
```
           <tr>
            <th>Protein</th>
            <th id=”protein”></th>
           </tr>

           <tr>
            <th>Vitamin A</th>
            <th id=”vitaminA”></th>
           </tr>
           <tr>
            <th>Vitamin C</th>
            <th id=”vitaminC”></th>
           </tr>
         </table>
      </div>
    </div>
</div>
```

```
    <script>

        Function getNutrients(){

            Const options = {

                 Method: ‘GET’,

                 Headers: {

                ‘X-RapidAPI-Key’:
‘e265f441b6mshb2c32e392137c4fp180d62jsnd464b9c5a77b’,

                ‘X-RapidAPI-Host’: ‘edamam-edamam-nutrition-
analysis.p.rapidapi.com’

                 }

        };

        Let data = document.getElementById(“product-name”).value

        Fetch(‘https://edamam-edamam-nutrition-
analysis.p.rapidapi.com/api/nutrition-data?ingr=’ + data, options)

             .then(response => response.json())

             .then(response => {result = response;

        Console.log(result)

            Document.getElementById(“totalfat”).innerText =
result.totalNutrients.FAT.quantity.toFixed(2) + “ “ +
result.totalNutrients.FAT.unit;
```

```
        Document.getElementById(“calories”).innerText =
result.calories;

        Document.getElementById(“cholesterol”).innerText =
result.totalNutrients.CHOLE.quantity.toFixed(2) + “ “ +
result.totalNutrients.CHOLE.unit;

        Document.getElementById(“calcium”).innerText =
result.totalNutrients.CA.quantity.toFixed(2) + “ “ +
result.totalNutrients.CA.unit;

        Document.getElementById(“sodium”).innerText =
result.totalNutrients.NA.quantity.toFixed(2) + “ “ +
result.totalNutrients.NA.unit;

        Document.getElementById(“potassium”).innerText =
result.totalNutrients.K.quantity.toFixed(2) + “ “ +
result.totalNutrients.K.unit;

        Document.getElementById(“sugar”).innerText =
result.totalNutrients.SUGAR.quantity.toFixed(2) + “ “ +
result.totalNutrients.SUGAR.unit;

        Document.getElementById(“protein”).innerText =
result.totalNutrients.PROCNT.quantity.toFixed(2) + “ “ +
result.totalNutrients.PROCNT.unit;

        Document.getElementById(“carbohydrates”).innerText =
result.totalNutrients.CHOCDF.quantity.toFixed(2) + “ “ +
result.totalNutrients.CHOCDF.unit;

        Document.getElementById(“vitaminA”).innerText =
result.totalNutrients.VITA_RAE.quantity.toFixed(2) + “ “ +
result.totalNutrients.VITA_RAE.unit;
```

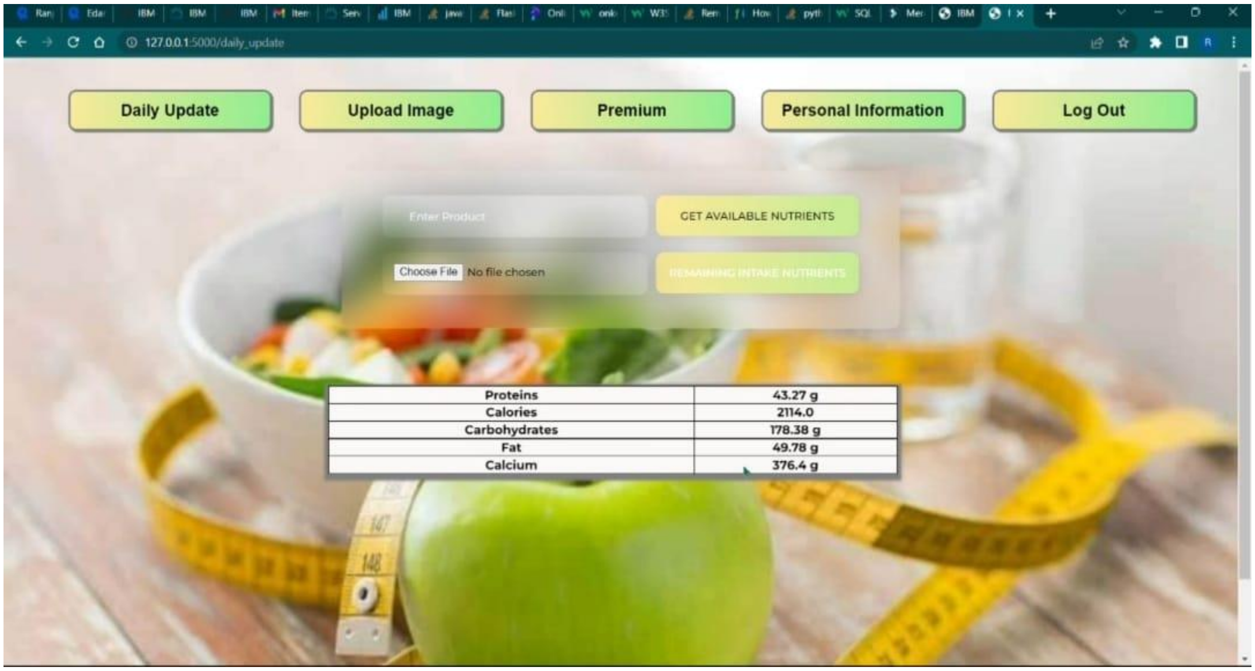
```
            Document.getElementById(“vitaminC”).innerText =
result.totalNutrients.VITC.quantity.toFixed(2) + “ “ +
result.totalNutrients.VITC.unit;



            Document.getElementById(‘calories1’).value  =
result.calories;

            Document.getElementById(‘protein1’).value =
result.totalNutrients.PROCNT.quantity.toFixed(2)

            Document.getElementById(‘carbohydrates1’).value =
result.totalNutrients.CHOCDF.quantity.toFixed(2)

            Document.getElementById(‘fat1’).value =
result.totalNutrients.FAT.quantity.toFixed(2)

            Document.getElementById(‘calcium1’).value =
result.totalNutrients.CA.quantity.toFixed(2)


Document.getElementsByClassName(“box2”)[0].style.display =
‘block’;

            Console.log(document.getElementById(‘fat1’).value)

            })

             .catch(err => console.error(err));



        }
```

```
    </script>

  </body>

</html>
```



1 large orange

Nutrition Facts	
Calories	86
Total Fat	0.22 g
Cholesterol	0.00 mg
Carbohydrates	21.62 g
Calcium	73.60 mg
Sodium	0.00 mg
Potassium	333.04 mg
Sugar	17.20 g
Protein	1.73 g
Vitamin A	20.24 μg
Vitamin C	97.89 mg



Proteins	43.27 g
Calories	2114.0
Carbohydrates	178.38 g
Fat	49.78 g
Calcium	376.4 g

- **Premium features**

```
<!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" />
    <title>Nutrtion Assistant Application</title>
    <link rel="preconnect" href="https://fonts.googleapis.com">
    <link rel="preconnect" href="https://fonts.gstatic.com" crossorigin>
    <link href="https://fonts.googleapis.com/css2?family=Montserrat:ital,wght@0,100;0,200;0,300;0,400;0,500;0,600;0,700;0,800;1,400;1,600&display=swap" rel="stylesheet">
    <link rel="stylesheet" href="../static/stylesheet.css"/>
  </head>
  <style>
    #navbar button{

              border: 3px solid grey;
              background-image:linear-gradient(to right,rgb(250, 234,
154) ,lightgreen);
              border-radius: 10px;
              width: 250px;
              padding: 5px;
              margin-left: 15px;
              margin-top: 20px;
              margin-right: 15px;
              cursor:pointer;
              height: 50px;
              font-weight: bold;
              font-size: 20px;
              box-shadow: 2px 2px grey;
            }
    button:hover
    {
      color: #fff;

    }
    table
        {
            margin-top: 70px;
            margin-left:400px;
            border: 3px solid grey;
            box-shadow: 0px 3px 3px 5px  grey;
            padding: 17px;
            width: 700px;
            background-color:rgb(252, 249, 249);
    }

    td{
      padding:20px;

    }

    table, th, td {
      border: 1px solid black;

    }

    th{
      font-size: 15px;
      font-weight: bold;
    }
  </style>
</head>
<body>
 <div class="main">
  <div id="navbar">
   <br>
   <nav>
     <a href="daily_update"><button  style="margin-left:
80px;"><span>Daily Update</span></button></a>
     <a href="uploadimage"><button><span>Upload
Image</span></button></a>
     <a
href="premium"><button><span>Premium</span></button></a>
     <a href="user_diet_info"><button ><span>Personal
Information</span></button></a>
     <a href="logout"><button ><span>Log
Out</span></button></a>

   </nav>
 </div>
 <br>
  <div class="main">
    <div id="continer">

      <div id="form-search">
        <FORM NAME="myform" action="/daily_update"
id="cointiner-inline" METHOD="post">
            <INPUT TYPE="text" placeholder="Enter Product"
id="product-name" NAME="product-name" VALUE="">
             <INPUT TYPE="button" style="margin-left:
10px;margin-top:0px;width: 250px;" class="button"
NAME="button1" Value="Get Available Nutrients"
onClick="getNutrients()">
            <br>
           <input type="hidden" value=0 id="protein1"
name="protein1">
           <input type="hidden" value=0 id="calories1"
name="calories1">
           <input type="hidden" value=0 id="carbohydrates1"
name="carbohydrates1">
           <input type="hidden" value=0 id="fat1" name="fat1">
           <input type="hidden" value=0 id="calcium1"
name="calcium1">
           <INPUT TYPE="submit" style="margin-left: -
250px;margin-top:70px;width: 250px;" class="button"
NAME="button1" Value="Remaining Intake Nutrients">
            <br>
          </form>
          <div id="cointiner-inline" style="margin-top:-70px">
          <input type="file"  accept="image/*" id="imageInput" >
          </div>
        </div>

       </div>
```

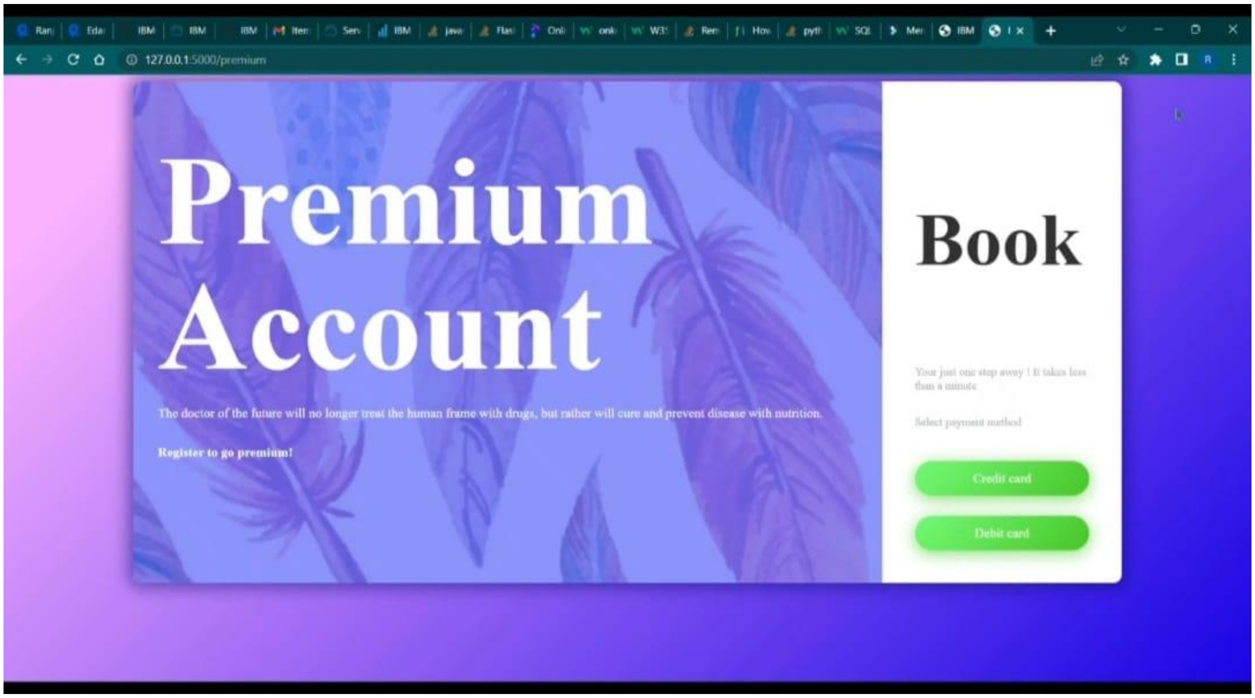
```
              <div class="msg" id = "msg1">
                <div class="bcol">
                  <table id="mytable1">

                    <!--  msg = ['Calories', str(calo) , 'Proteins', str(protein) +
" g", 'Carbohydrates', str(carbo) + " g", 'Fat', str(fat) + " g", 'Calcium',
str(calci) + " g"]
 -->
                    <th>{{msg[2]}}</th>
                    <th>{{msg[3]}}</th>
                </tr>
                <tr>
                    <th>{{msg[0]}}</th>
                    <th>{{msg[1]}}</th>
                  </tr>
                <tr>
                    <th>{{msg[4]}}</th>
                    <th>{{msg[5]}}</th>
                </tr>
                <tr>
                    <th>{{msg[6]}}</th>
                    <th>{{msg[7]}}</th>
                </tr>

                    <tr>
                        <th>{{msg[8]}}</th>
                        <th>{{msg[9]}}</th>
                    </tr>
                    </table>
              </div>

              </div>
            <div class="box2">

                <div class="bcol">
```

```
                    <table id="mytable2" >
                      <tr>
                        <th colspan="2" style="background-color:#ffffff
;margin-top: 25px;font-weight: bold;padding-left: 15px;">Nutrition
Facts</th>
                      </tr>
                        <tr>
                            <th>Calories</th>
                            <th id="calories"></th>
                          </tr>
                        <tr>
                            <th>Total Fat</th>
                            <th id="totalfat"></th>
                        </tr>
                        <tr>
                            <th>Cholesterol</th>
                            <th id="cholesterol"></th>
                        </tr>
                        <tr>
                            <th>Carbohydrates</th>
                            <th id="carbohydrates"></th>
                          </tr>
                        <tr>
                            <th>Calcium</th>
                            <th id="calcium"></th>
                        </tr>
                        <tr>
                            <th>Sodium</th>
                            <th id="sodium"></th>
                        </tr>
                        <tr>
                            <th>Potassium</th>
                            <th id="potassium"></th>
                        </tr>
                        <tr>
```

```
              <th>Sugar</th>
              <th id="sugar"></th>
             </tr>
             <tr>
              <th>Protein</th>
              <th id="protein"></th>
             </tr>

             <tr>
              <th>Vitamin A</th>
              <th id="vitaminA"></th>
             </tr>
             <tr>
              <th>Vitamin C</th>
              <th id="vitaminC"></th>
             </tr>
           </table>
        </div>
      </div>
  </div>

  <script>
    function getNutrients(){
      const options = {
      method: 'GET',
      headers: {
        'X-RapidAPI-Key':
'e265f441b6mshb2c32e392137c4fp180d62jsnd464b9c5a77b',
        'X-RapidAPI-Host': 'edamam-edamam-nutrition-
analysis.p.rapidapi.com'
      }
    };
    let data = document.getElementById("product-name").value
    fetch('https://edamam-edamam-nutrition-
analysis.p.rapidapi.com/api/nutrition-data?ingr=' + data, options)
```

```
        .then(response => response.json())
        .then(response => {result = response;
        console.log(result)
            document.getElementById("totalfat").innerText =
result.totalNutrients.FAT.quantity.toFixed(2) + " " +
result.totalNutrients.FAT.unit;
            document.getElementById("calories").innerText =
result.calories;
            document.getElementById("cholesterol").innerText =
result.totalNutrients.CHOLE.quantity.toFixed(2) + " " +
result.totalNutrients.CHOLE.unit;
            document.getElementById("calcium").innerText =
result.totalNutrients.CA.quantity.toFixed(2) + " " +
result.totalNutrients.CA.unit;
            document.getElementById("sodium").innerText =
result.totalNutrients.NA.quantity.toFixed(2) + " " +
result.totalNutrients.NA.unit;
            document.getElementById("potassium").innerText =
result.totalNutrients.K.quantity.toFixed(2) + " " +
result.totalNutrients.K.unit;
            document.getElementById("sugar").innerText =
result.totalNutrients.SUGAR.quantity.toFixed(2) + " " +
result.totalNutrients.SUGAR.unit;
            document.getElementById("protein").innerText =
result.totalNutrients.PROCNT.quantity.toFixed(2) + " " +
result.totalNutrients.PROCNT.unit;
            document.getElementById("carbohydrates").innerText =
result.totalNutrients.CHOCDF.quantity.toFixed(2) + " " +
result.totalNutrients.CHOCDF.unit;
            document.getElementById("vitaminA").innerText =
result.totalNutrients.VITA_RAE.quantity.toFixed(2) + " " +
result.totalNutrients.VITA_RAE.unit;
            document.getElementById("vitaminC").innerText =
result.totalNutrients.VITC.quantity.toFixed(2) + " " +
result.totalNutrients.VITC.unit;
```

```
            document.getElementById('calories1').value  = result.calories;
            document.getElementById('protein1').value =
result.totalNutrients.PROCNT.quantity.toFixed(2)
            document.getElementById('carbohydrates1').value =
result.totalNutrients.CHOCDF.quantity.toFixed(2)
            document.getElementById('fat1').value =
result.totalNutrients.FAT.quantity.toFixed(2)
            document.getElementById('calcium1').value =
result.totalNutrients.CA.quantity.toFixed(2)
            document.getElementsByClassName("box2")[0].style.display
= 'block';
            console.log(document.getElementById('fat1').value)
            })
        .catch(err => console.error(err));

        }


    </script>
  </body>
</html>
```



Premium Account

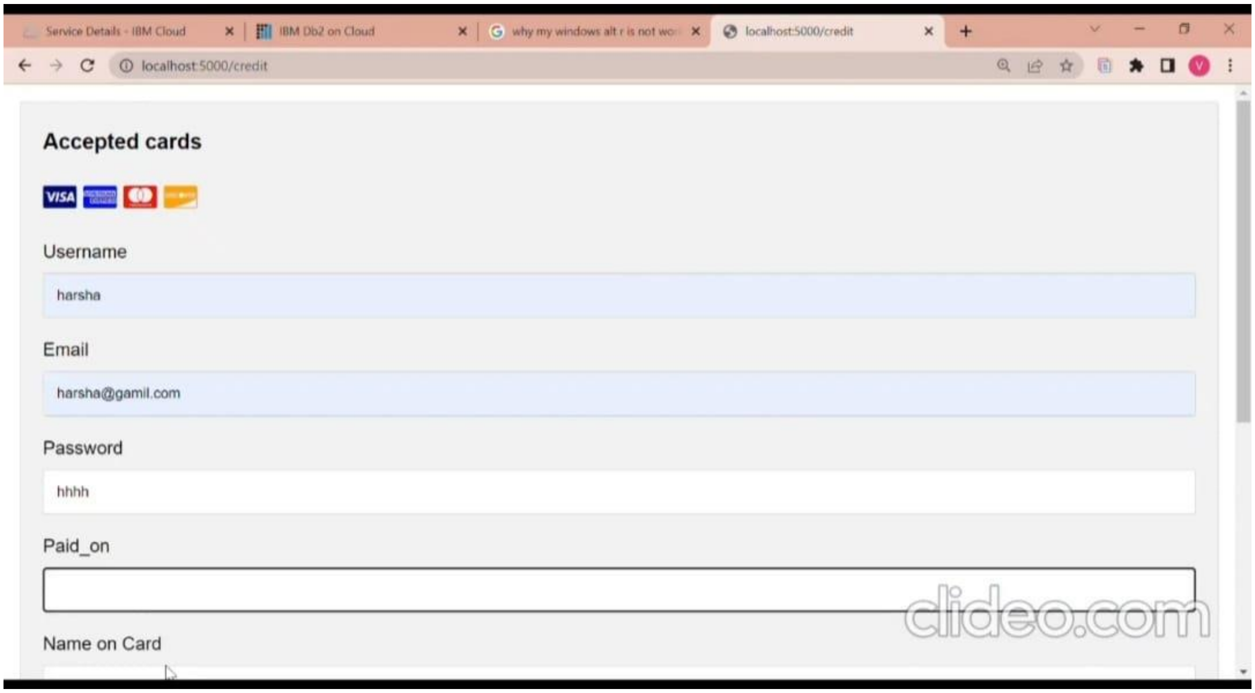
The doctor of the future will no longer treat the human frame with drugs, but rather will cure and prevent disease with nutrition.

Register to go premium!

Book

Your just one step away ! It takes less than a minute

Select payment method



Accepted cards

Username

harsha

Email

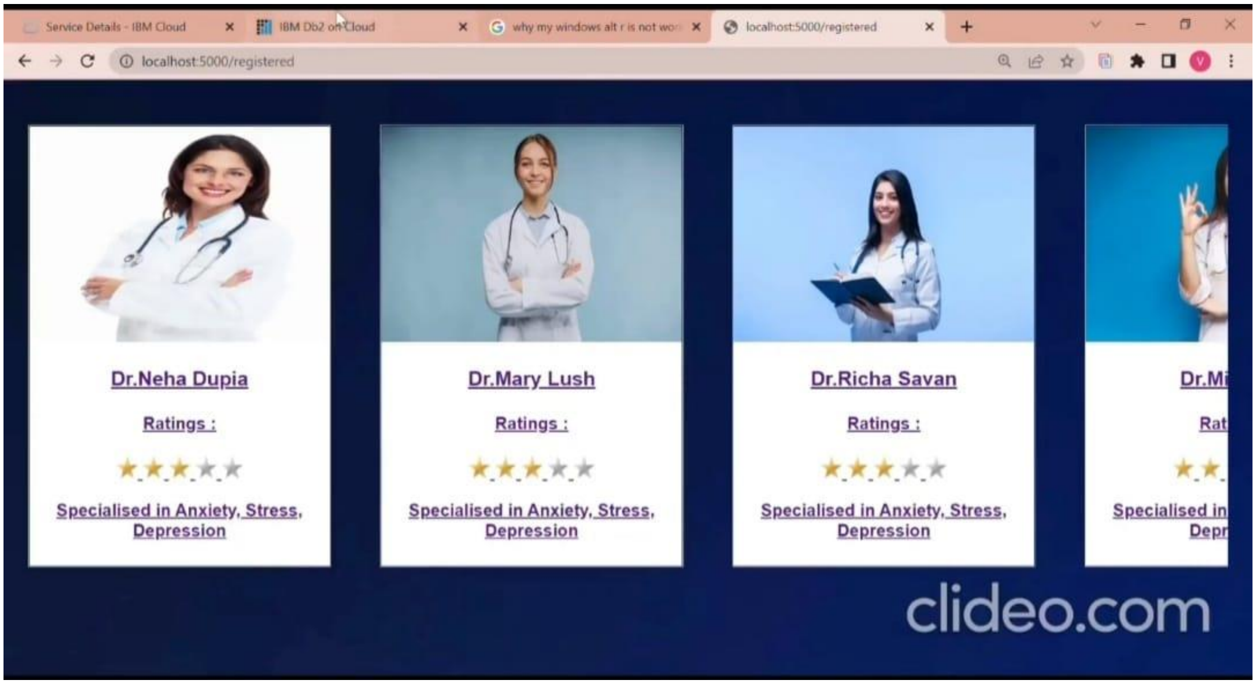
harsha@gamil.com

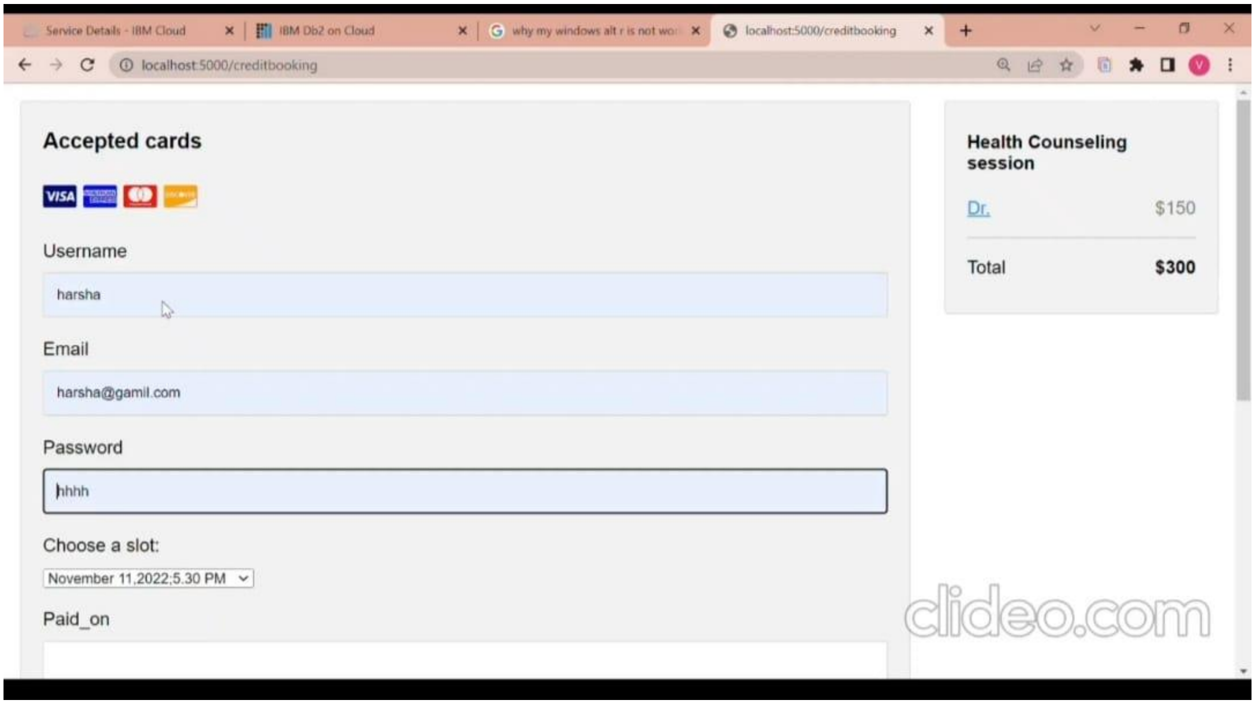
Password

hhhh

Paid_on

Name on Card


Service Details - IBM Cloud
IBM Db2 on Cloud
why my windows alt r is not wor
localhost:5000/registered
localhost:5000/registered
Dr.Neha Dupia
Ratings :
Specialised in Anxiety, Stress, Depression
Dr.Mary Lush
Ratings :
Specialised in Anxiety, Stress, Depression
Dr.Richa Savan
Ratings :
Specialised in Anxiety, Stress, Depression
Dr.Mi
Rat
Specialised in
Depr
clideo.com


Service Details - IBM Cloud
IBM Db2 on Cloud
why my windows alt r is not wor
localhost:5000/creditbooking
localhost:5000/creditbooking
clideo.com

Accepted cards

Username

harsha

Email

harsha@gamil.com

Password

hhhh

Choose a slot:

November 11,2022;5.30 PM

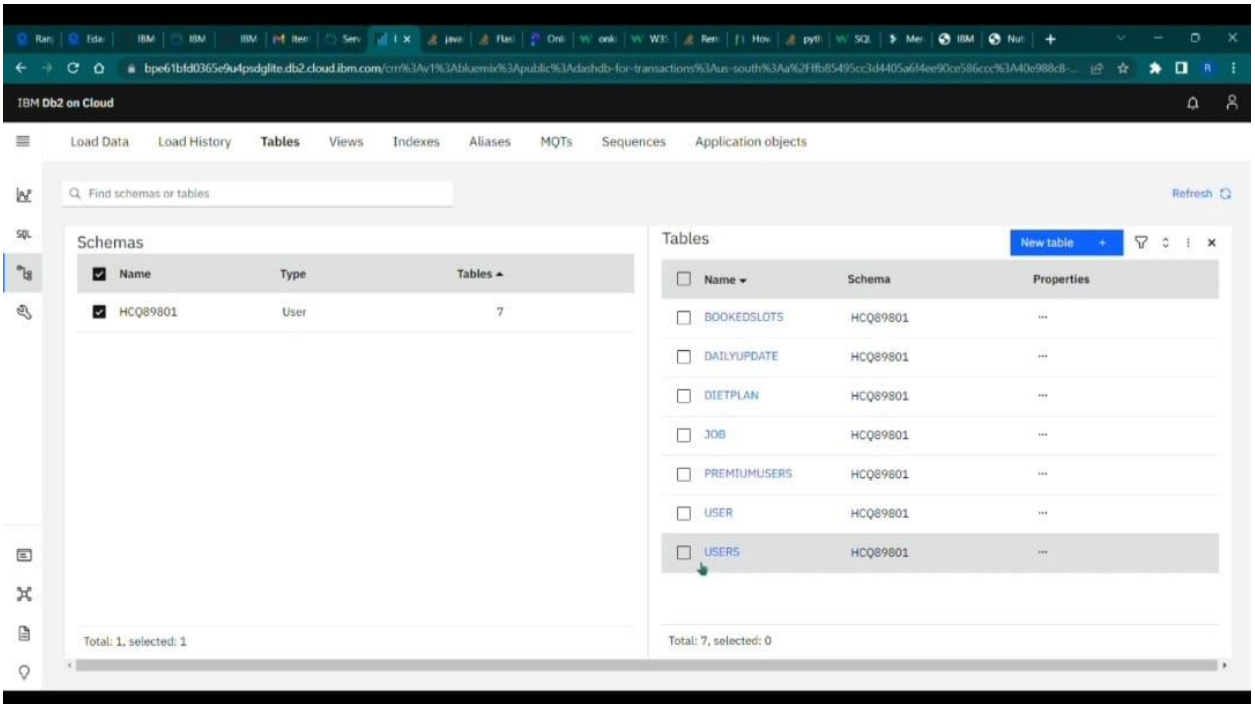
Paid_on

Health Counseling session

Dr. $150

Total $300

7.3 Database Schema



Schemas

☑ Name	Type	Tables ▲
☑ HCQ89801	User	7

Total: 1, selected: 1

Tables

☐ Name ▾	Schema	Properties
☐ BOOKEDSLOTS	HCQ89801	···
☐ DAILYUPDATE	HCQ89801	···
☐ DIETPLAN	HCQ89801	···
☐ JOB	HCQ89801	···
☐ PREMIUMUSERS	HCQ89801	···
☐ USER	HCQ89801	···
☐ USERS	HCQ89801	···

Total: 7, selected: 0

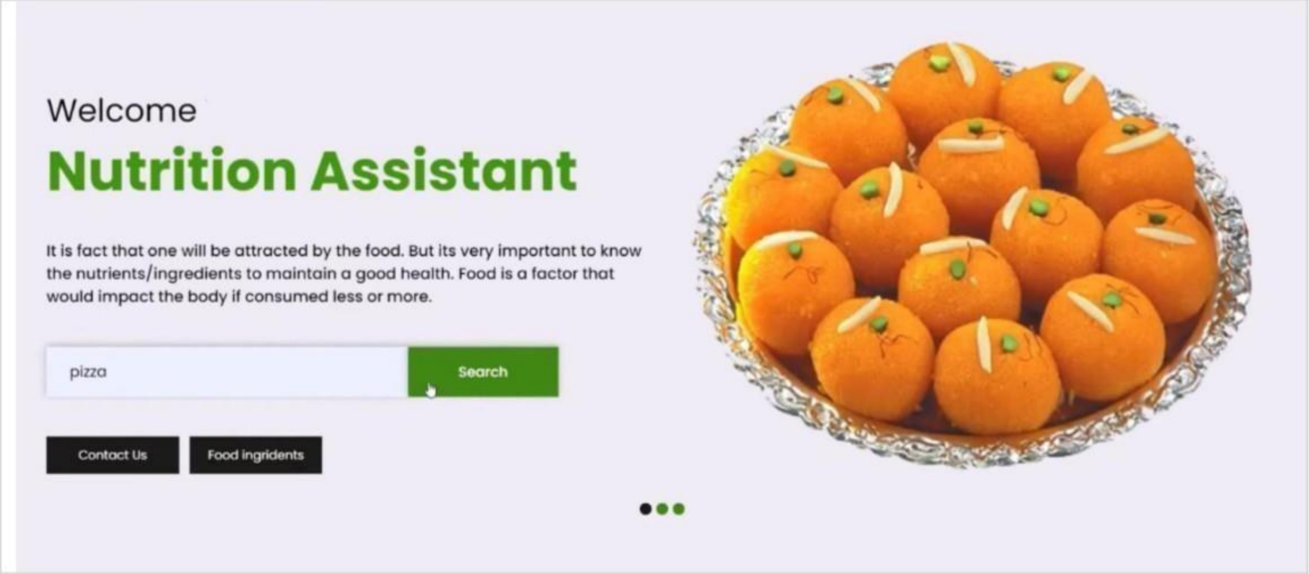
TESTING

1) Test Cases

i. Our code was tested on various food to check whether it gives the correctoutput
ii. To satisfy the customer’s expectations we tested it fully.

2) User Acceptance Testing

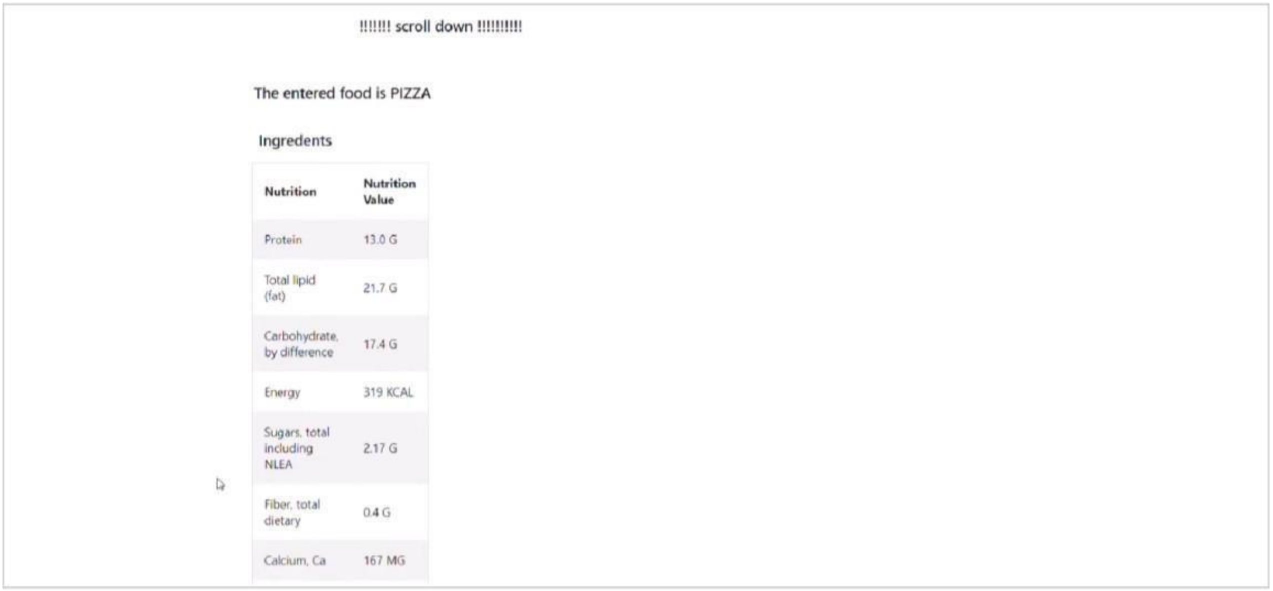
Our project was tested by an end user to verify that it’s working correctly.



Welcome

Nutrition Assistant

It is fact that one will be attracted by the food. But its very important to know the nutrients/ingredients to maintain a good health. Food is a factor that would impact the body if consumed less or more.



!!!!!! scroll down !!!!!!!!!

The entered food is PIZZA

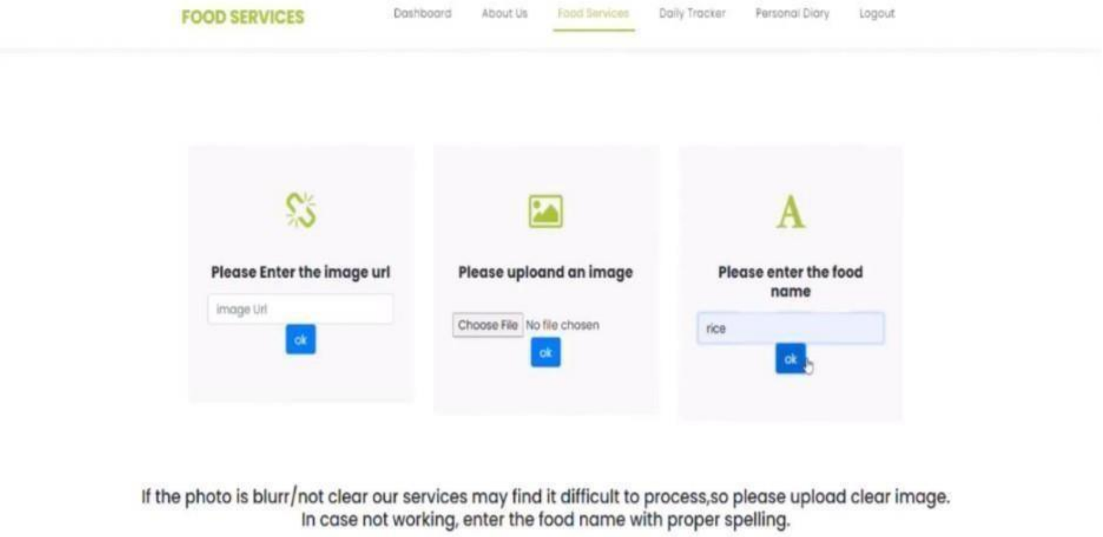
Ingredents

Nutrition	Nutrition Value
Protein	13.0 G
Total lipid (fat)	21.7 G
Carbohydrate, by difference	17.4 G
Energy	319 KCAL
Sugars, total including NLEA	2.17 G
Fiber, total dietary	0.4 G
Calcium, Ca	167 MG

<u>RESULTS</u>

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** in shown on the screen



Please Enter the image url

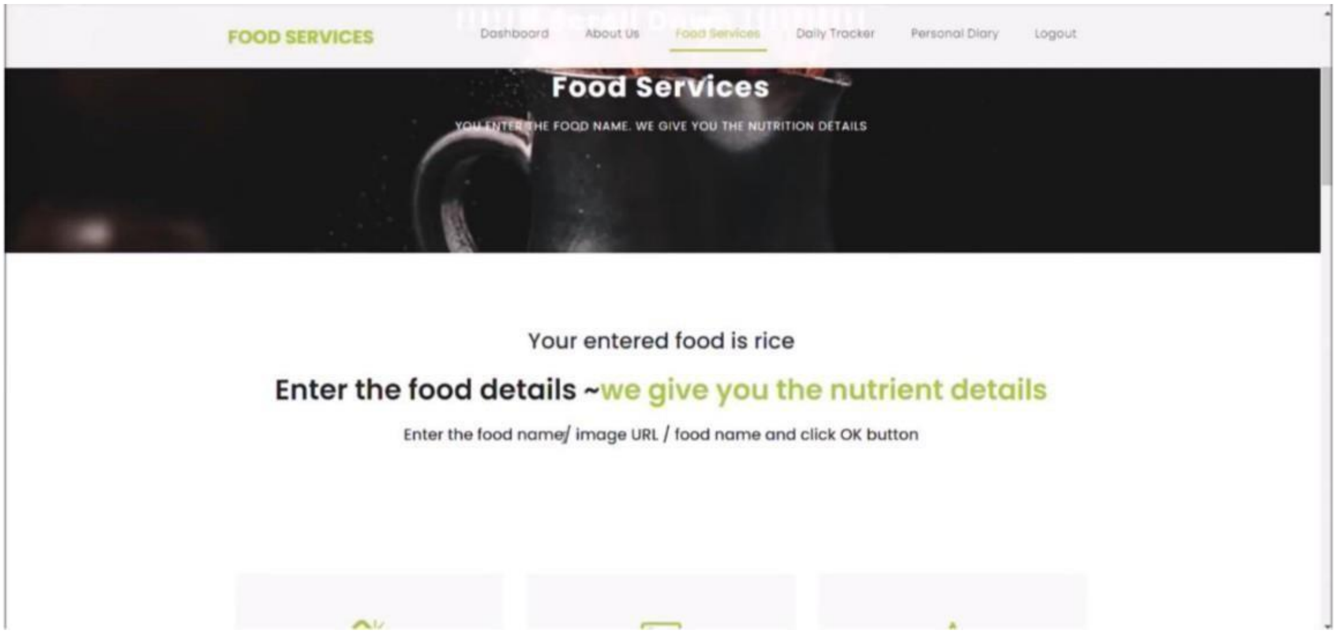
image Url

Please uploand an image

Please enter the food name

rice

If the photo is blurr/not clear our services may find it difficult to process,so please upload clear image.
In case not working, enter the food name with proper spelling.



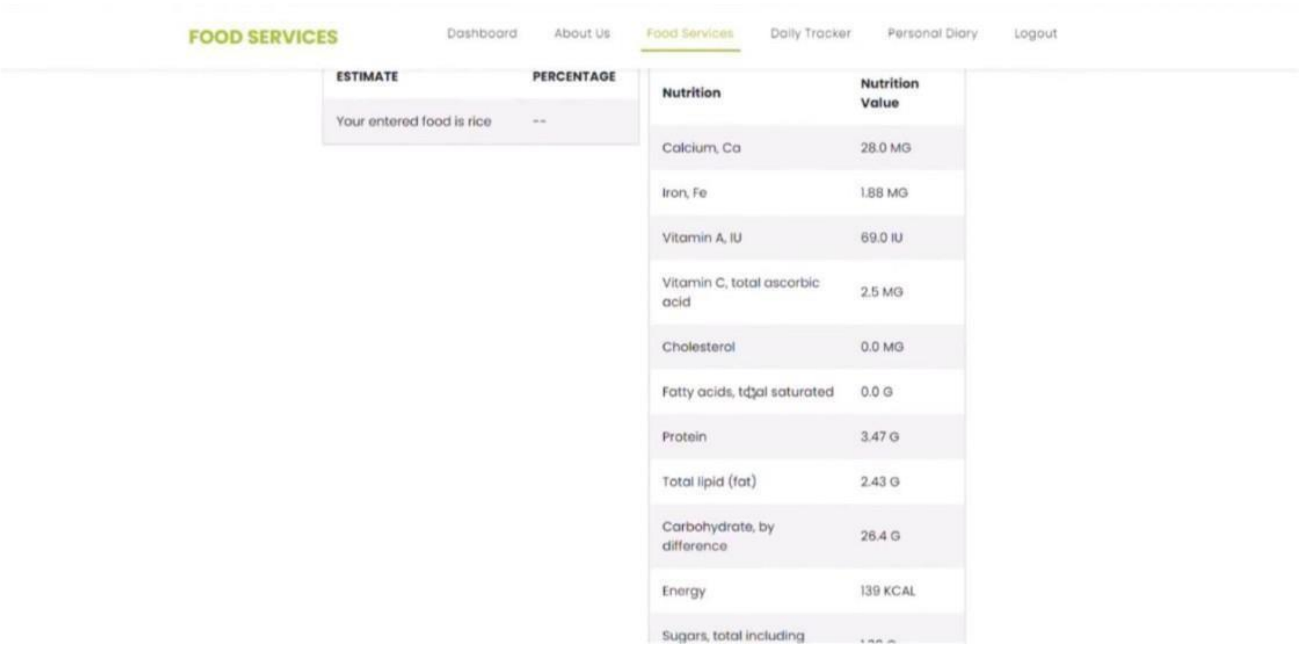
Food Services

YOU ENTER THE FOOD NAME. WE GIVE YOU THE NUTRITION DETAILS

Your entered food is rice

Enter the food details ~we give you the nutrient details

Enter the food name/ image URL / food name and click OK button



ESTIMATE	PERCENTAGE
Your entered food is rice	--

Nutrition	Nutrition Value
Calcium, Ca	28.0 MG
Iron, Fe	1.88 MG
Vitamin A, IU	69.0 IU
Vitamin C, total ascorbic acid	2.5 MG
Cholesterol	0.0 MG
Fatty acids, total saturated	0.0 G
Protein	3.47 G
Total lipid (fat)	2.43 G
Carbohydrate, by difference	26.4 G
Energy	139 KCAL
Sugars, total including	[illegible]

ADVANTAGES

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

CONCLUSION

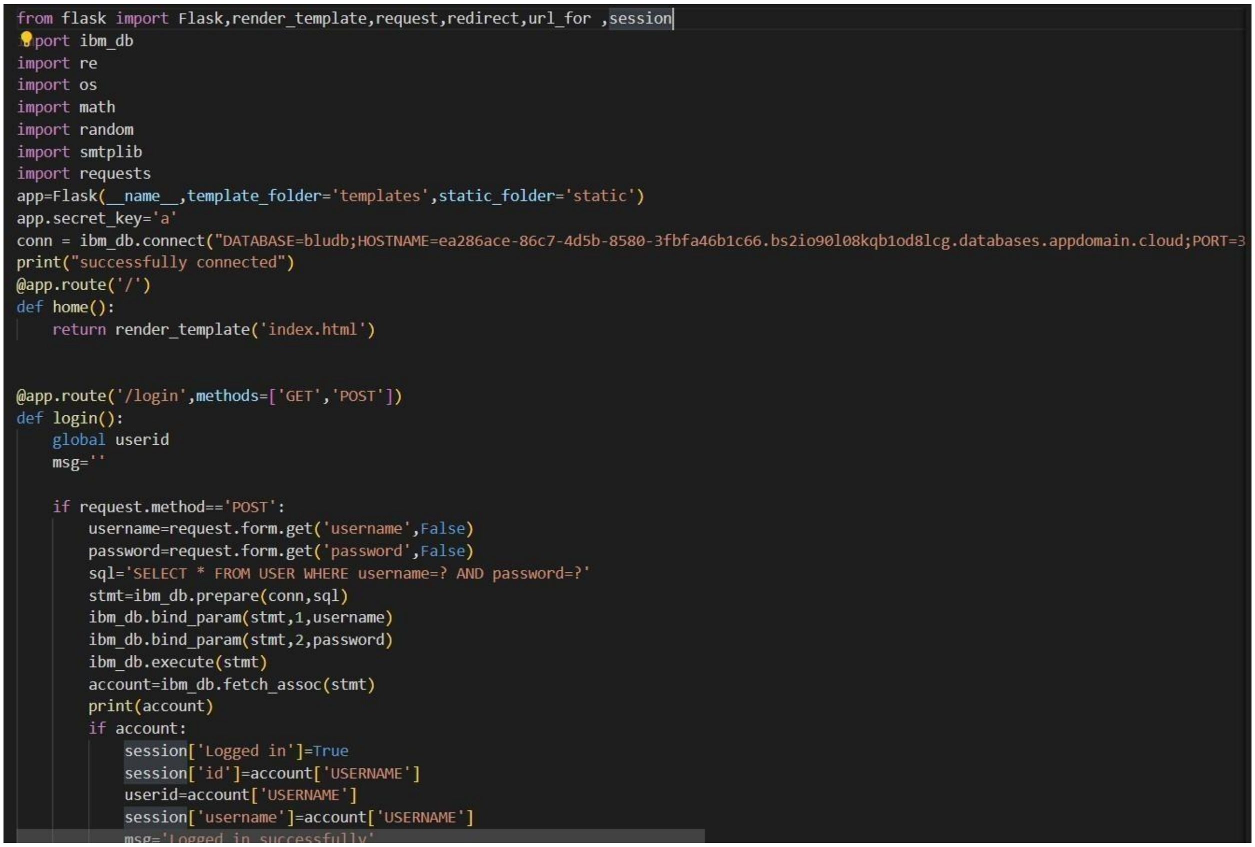
In conclusion, many people have become aware of their health. Moreover, they arealso informed how to live a healthy lifestyle. Most of the research related to these themes aims to identify changes in healthy lifestyle behavior with web applications that are considered effective in dietary self-monitoring.

FUTURE SCOPE

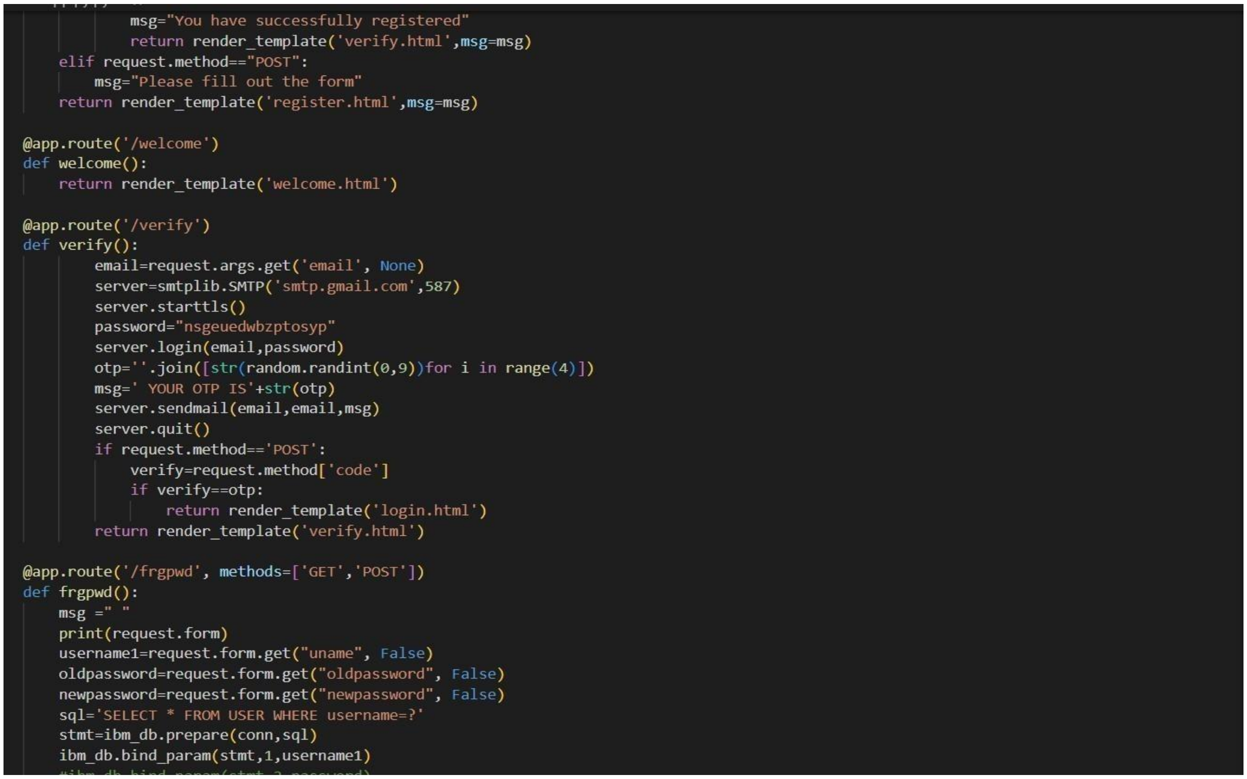
Nutrition assistants help dietitians with providing proper nutrition at healthcare facilities. They determine patients' nutritional needs, assess risk factors, and plan meals and menus.They also ensure proper sterilization of plates and utensils.

<u>APPENDIX</u>

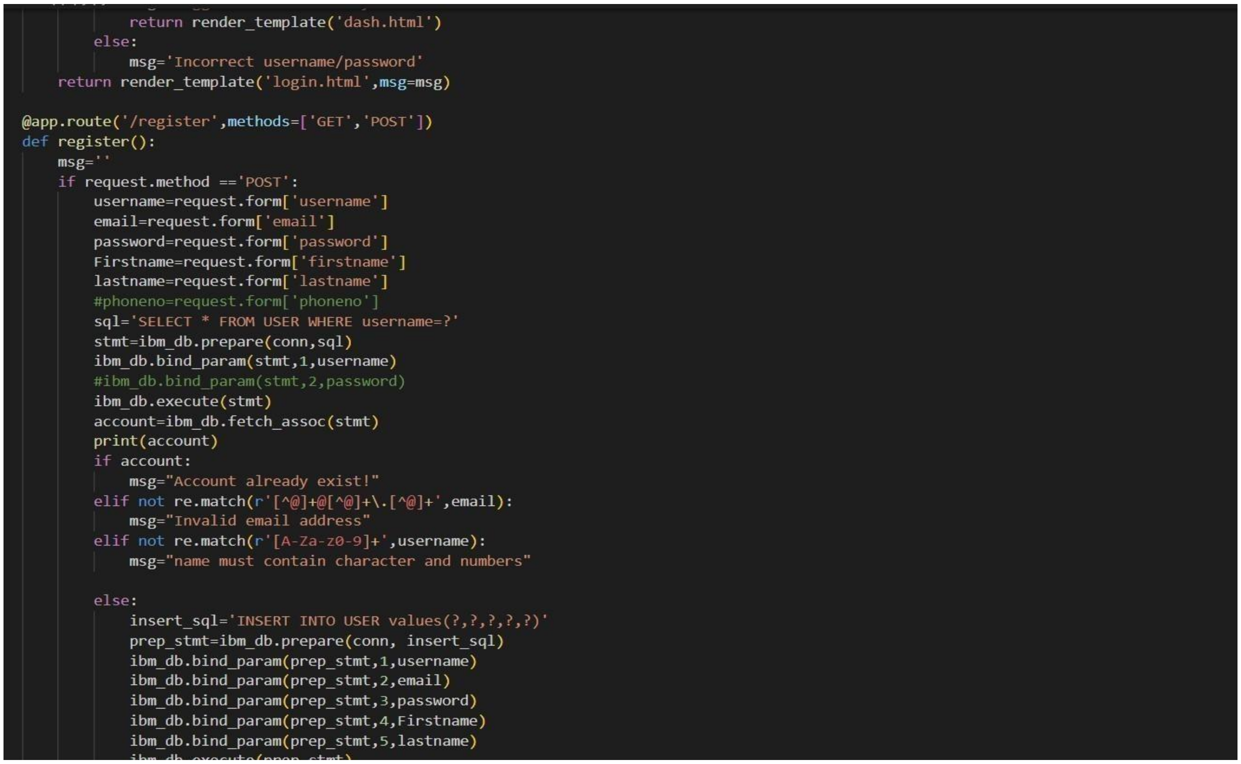
1) Source Code



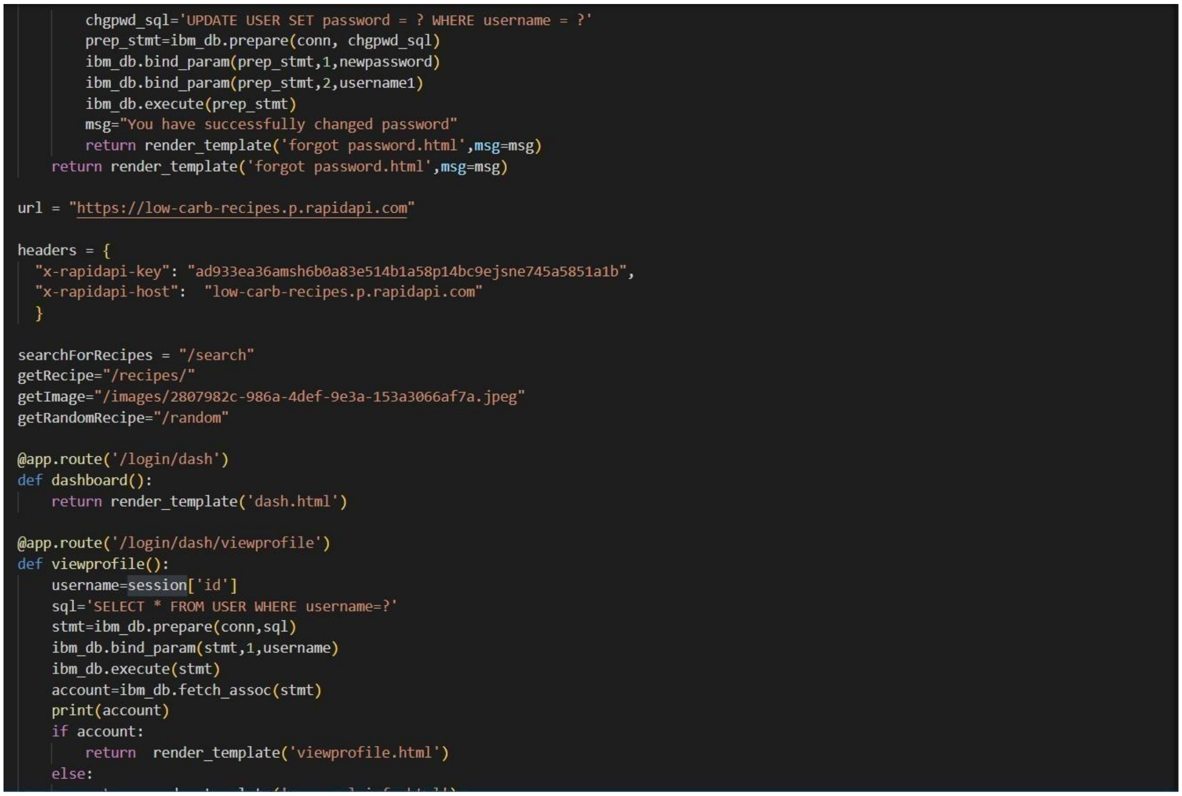
```
from flask import Flask,render_template,request,redirect,url_for ,session
import ibm_db
import re
import os
import math
import random
import smtplib
import requests
app=Flask(__name__,template_folder='templates',static_folder='static')
app.secret_key='a'
conn = ibm_db.connect("DATABASE=bludb;HOSTNAME=ea286ace-86c7-4d5b-8580-3fbfa46b1c66.bs2io90l08kqb1od8lcg.databases.appdomain.cloud;PORT=3
print("successfully connected")
@app.route('/')
def home():
    return render_template('index.html')


@app.route('/login',methods=['GET','POST'])
def login():
    global userid
    msg=''

    if request.method=='POST':
        username=request.form.get('username',False)
        password=request.form.get('password',False)
        sql='SELECT * FROM USER WHERE username=? AND password=?'
        stmt=ibm_db.prepare(conn,sql)
        ibm_db.bind_param(stmt,1,username)
        ibm_db.bind_param(stmt,2,password)
        ibm_db.execute(stmt)
        account=ibm_db.fetch_assoc(stmt)
        print(account)
        if account:
            session['Logged in']=True
            session['id']=account['USERNAME']
            userid=account['USERNAME']
            session['username']=account['USERNAME']
            msg='Logged in successfully'
```



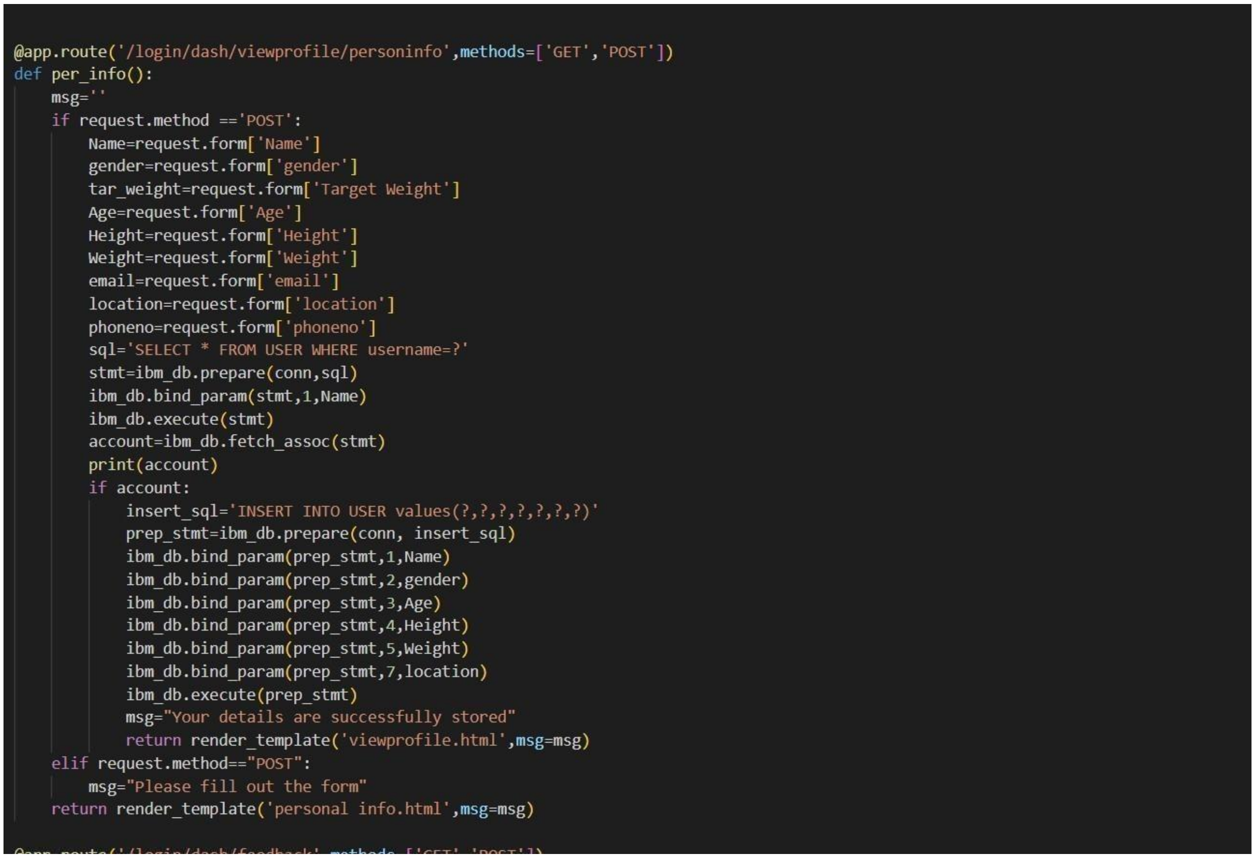
```
            msg="You have successfully registered"
            return render_template('verify.html',msg=msg)
    elif request.method=="POST":
        msg="Please fill out the form"
    return render_template('register.html',msg=msg)

@app.route('/welcome')
def welcome():
    return render_template('welcome.html')

@app.route('/verify')
def verify():
        email=request.args.get('email', None)
        server=smtplib.SMTP('smtp.gmail.com',587)
        server.starttls()
        password="nsgeuedwbzptosyp"
        server.login(email,password)
        otp=''.join([str(random.randint(0,9))for i in range(4)])
        msg=' YOUR OTP IS'+str(otp)
        server.sendmail(email,email,msg)
        server.quit()
        if request.method=='POST':
            verify=request.method['code']
            if verify==otp:
                return render_template('login.html')
        return render_template('verify.html')

@app.route('/frgpwd', methods=['GET','POST'])
def frgpwd():
    msg =" "
    print(request.form)
    username1=request.form.get("uname", False)
    oldpassword=request.form.get("oldpassword", False)
    newpassword=request.form.get("newpassword", False)
    sql='SELECT * FROM USER WHERE username=?'
    stmt=ibm_db.prepare(conn,sql)
    ibm_db.bind_param(stmt,1,username1)
    #ibm_db.bind_param(stmt,2,password)
```



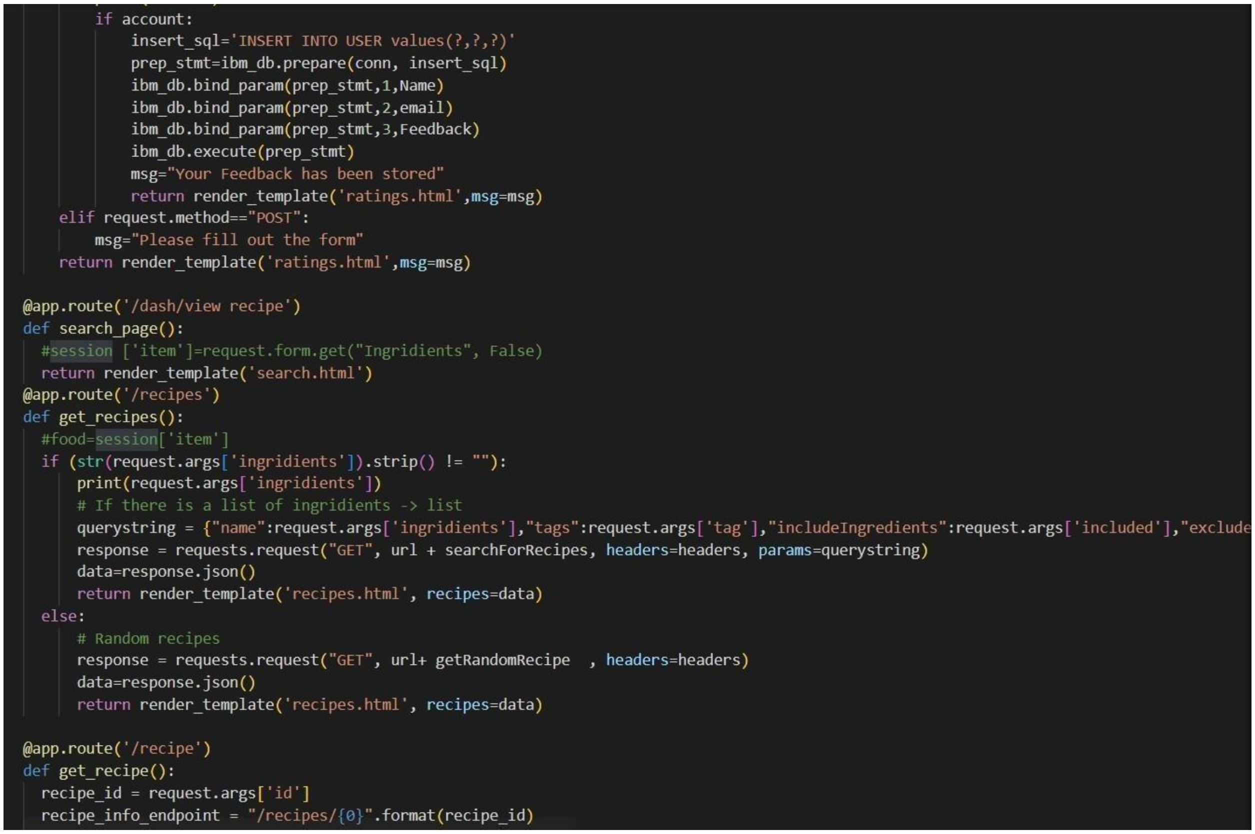
```
            return render_template('dash.html')
        else:
            msg='Incorrect username/password'
    return render_template('login.html',msg=msg)

@app.route('/register',methods=['GET','POST'])
def register():
    msg=''
    if request.method =='POST':
        username=request.form['username']
        email=request.form['email']
        password=request.form['password']
        Firstname=request.form['firstname']
        lastname=request.form['lastname']
        #phoneno=request.form['phoneno']
        sql='SELECT * FROM USER WHERE username=?'
        stmt=ibm_db.prepare(conn,sql)
        ibm_db.bind_param(stmt,1,username)
        #ibm_db.bind_param(stmt,2,password)
        ibm_db.execute(stmt)
        account=ibm_db.fetch_assoc(stmt)
        print(account)
        if account:
            msg="Account already exist!"
        elif not re.match(r'[^@]+@[^@]+\.[^@]+',email):
            msg="Invalid email address"
        elif not re.match(r'[A-Za-z0-9]+',username):
            msg="name must contain character and numbers"

        else:
            insert_sql='INSERT INTO USER values(?,?,?,?,?)'
            prep_stmt=ibm_db.prepare(conn, insert_sql)
            ibm_db.bind_param(prep_stmt,1,username)
            ibm_db.bind_param(prep_stmt,2,email)
            ibm_db.bind_param(prep_stmt,3,password)
            ibm_db.bind_param(prep_stmt,4,Firstname)
            ibm_db.bind_param(prep_stmt,5,lastname)
            ibm_db.execute(prep_stmt)
```



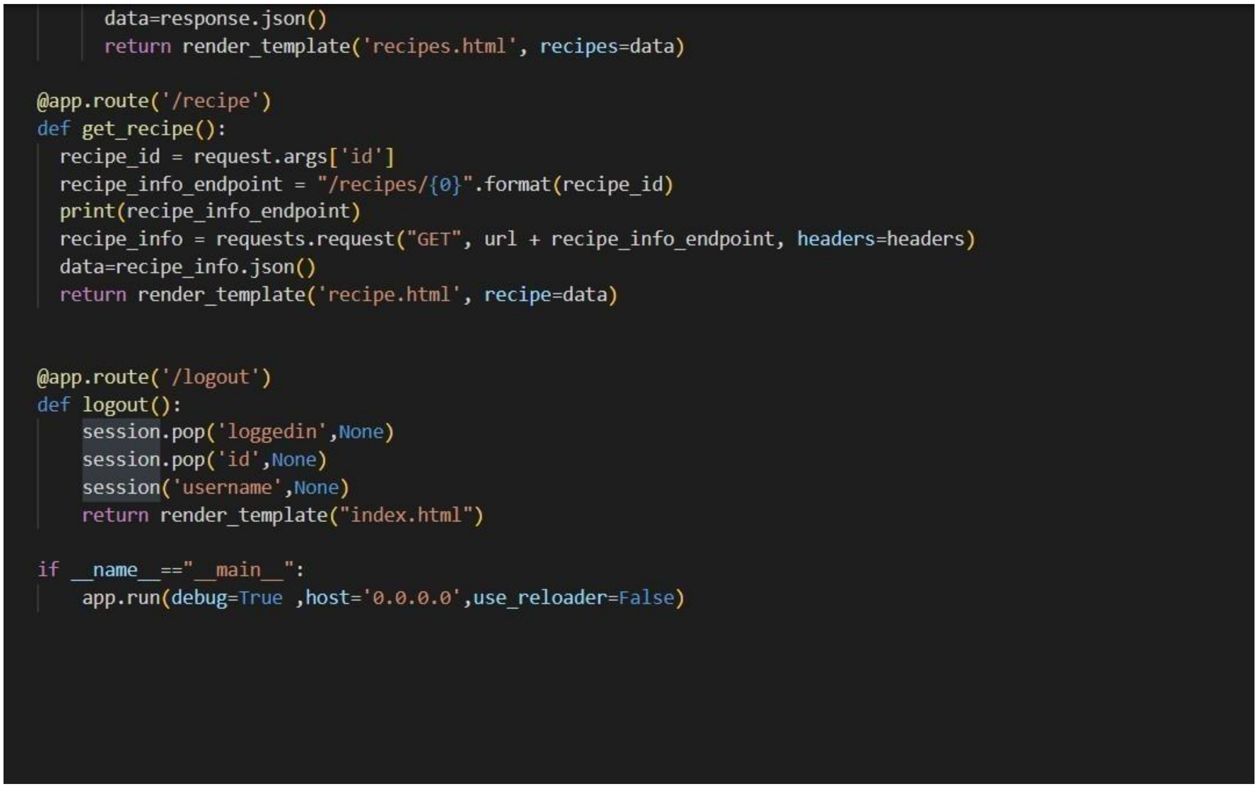
```
        chgpwd_sql='UPDATE USER SET password = ? WHERE username = ?'
        prep_stmt=ibm_db.prepare(conn, chgpwd_sql)
        ibm_db.bind_param(prep_stmt,1,newpassword)
        ibm_db.bind_param(prep_stmt,2,username1)
        ibm_db.execute(prep_stmt)
        msg="You have successfully changed password"
        return render_template('forgot password.html',msg=msg)
    return render_template('forgot password.html',msg=msg)

url = "https://low-carb-recipes.p.rapidapi.com"

headers = {
  "x-rapidapi-key": "ad933ea36amsh6b0a83e514b1a58p14bc9ejsne745a5851a1b",
  "x-rapidapi-host":  "low-carb-recipes.p.rapidapi.com"
  }

searchForRecipes = "/search"
getRecipe="/recipes/"
getImage="/images/2807982c-986a-4def-9e3a-153a3066af7a.jpeg"
getRandomRecipe="/random"

@app.route('/login/dash')
def dashboard():
    return render_template('dash.html')

@app.route('/login/dash/viewprofile')
def viewprofile():
    username=session['id']
    sql='SELECT * FROM USER WHERE username=?'
    stmt=ibm_db.prepare(conn,sql)
    ibm_db.bind_param(stmt,1,username)
    ibm_db.execute(stmt)
    account=ibm_db.fetch_assoc(stmt)
    print(account)
    if account:
        return  render_template('viewprofile.html')
    else:
```



```
@app.route('/login/dash/viewprofile/personinfo',methods=['GET','POST'])
def per_info():
    msg=''
    if request.method =='POST':
        Name=request.form['Name']
        gender=request.form['gender']
        tar_weight=request.form['Target Weight']
        Age=request.form['Age']
        Height=request.form['Height']
        Weight=request.form['Weight']
        email=request.form['email']
        location=request.form['location']
        phoneno=request.form['phoneno']
        sql='SELECT * FROM USER WHERE username=?'
        stmt=ibm_db.prepare(conn,sql)
        ibm_db.bind_param(stmt,1,Name)
        ibm_db.execute(stmt)
        account=ibm_db.fetch_assoc(stmt)
        print(account)
        if account:
            insert_sql='INSERT INTO USER values(?,?,?,?,?,?,?)'
            prep_stmt=ibm_db.prepare(conn, insert_sql)
            ibm_db.bind_param(prep_stmt,1,Name)
            ibm_db.bind_param(prep_stmt,2,gender)
            ibm_db.bind_param(prep_stmt,3,Age)
            ibm_db.bind_param(prep_stmt,4,Height)
            ibm_db.bind_param(prep_stmt,5,Weight)
            ibm_db.bind_param(prep_stmt,7,location)
            ibm_db.execute(prep_stmt)
            msg="Your details are successfully stored"
            return render_template('viewprofile.html',msg=msg)
    elif request.method=="POST":
        msg="Please fill out the form"
    return render_template('personal info.html',msg=msg)

@app.route('/login/dash/feedback',methods=['GET','POST'])
```



```
        if account:
            insert_sql='INSERT INTO USER values(?,?,?)'
            prep_stmt=ibm_db.prepare(conn, insert_sql)
            ibm_db.bind_param(prep_stmt,1,Name)
            ibm_db.bind_param(prep_stmt,2,email)
            ibm_db.bind_param(prep_stmt,3,Feedback)
            ibm_db.execute(prep_stmt)
            msg="Your Feedback has been stored"
            return render_template('ratings.html',msg=msg)
    elif request.method=="POST":
        msg="Please fill out the form"
    return render_template('ratings.html',msg=msg)

@app.route('/dash/view recipe')
def search_page():
  #session ['item']=request.form.get("Ingridients", False)
  return render_template('search.html')
@app.route('/recipes')
def get_recipes():
  #food=session['item']
  if (str(request.args['ingridients']).strip() != ""):
      print(request.args['ingridients'])
      # If there is a list of ingridients -> list
      querystring = {"name":request.args['ingridients'],"tags":request.args['tag'],"includeIngredients":request.args['included'],"exclude
      response = requests.request("GET", url + searchForRecipes, headers=headers, params=querystring)
      data=response.json()
      return render_template('recipes.html', recipes=data)
  else:
      # Random recipes
      response = requests.request("GET", url+ getRandomRecipe  , headers=headers)
      data=response.json()
      return render_template('recipes.html', recipes=data)

@app.route('/recipe')
def get_recipe():
  recipe_id = request.args['id']
  recipe_info_endpoint = "/recipes/{0}".format(recipe_id)
```



```
      data=response.json()
      return render_template('recipes.html', recipes=data)

@app.route('/recipe')
def get_recipe():
  recipe_id = request.args['id']
  recipe_info_endpoint = "/recipes/{0}".format(recipe_id)
  print(recipe_info_endpoint)
  recipe_info = requests.request("GET", url + recipe_info_endpoint, headers=headers)
  data=recipe_info.json()
  return render_template('recipe.html', recipe=data)


@app.route('/logout')
def logout():
    session.pop('loggedin',None)
    session.pop('id',None)
    session('username',None)
    return render_template("index.html")

if __name__=="__main__":
    app.run(debug=True ,host='0.0.0.0',use_reloader=False)
```

2) GitHub

https://github.com/IBM-EPBL/IBM-Project-17008-1659626698

Drive link

https://drive.google.com/file/d/1ZwOj-8ZHMsRUkWnZXIY1VMZMmGeq8hqe/view?usp=drivesdk