**NALAIYA THIRAN - IBM PROJECT REPORT**

**(19EC406T - Professional Readiness for Innovation, Employability and Entrepreneurship)**

**ON**

# Nutrition Assistant Application

***Submitted by***


**TEAM ID: PNT2022TMID23530**

**YAKASIRI NITHEESH ( 113219041133 )**

**PARTHA SARATHY MJ ( 113219041079 )**

**HITESH KS ( 113219041040 )**

**KAMINENI YASWANTH BABU ( 113219041044 )**

**HARISH A ( 113219041039 )**


***in partial fulfillment for the award of the degree of***

## BACHELOR OF ENGINEERING

**IN**

## ELECTRONICS AND COMMUNICATION ENGINEERING

VELAMMAL
ENGINEERING COLLEGE
The Wheel of Knowledge rolls on!

## VELAMMAL ENGINEERING COLLEGE, CHENNAI-66.

(An Autonomous Institution, Affiliated to Anna University, Chennai)

**2022-2023**

# VELAMMAL ENGINEERING COLLEGE CHENNAI -66

(An Autonomous Institution, Affiliated to Anna University, Chennai)

## BONAFIDE CERTIFICATE

Certified that this NALAIYA THIRAN – IBM PROJECT REPORT "**Nutrition Assistant Application"** is the Bonafide work of "YAKASIRI NITHEESH (113219041133), KAMINENI YASWANTH BABU (113219041044),PARTHA SARATHY MJ (113219041079),HITESH KS (113219041040) and HARISH A (113219041039)" carried out in "PROFESSIONAL READINESS FOR INNOVATION, EMPLOYABILITY AND ENTREPRENEURSHIP (NALAIYA THIRAN - IBM PROJECT)**"** during the Academic Year 2022-2023.

**Mrs. DOLLY IRENE**
**ASSISTANT PROFESSOR I**
**FACULTY EVALUATOR**
Department of Electronics &
Communication Engineering
Velammal Engineering College
Ambattur - redhills road.
Chennai-66

**Dr.S. MARY JOANS**
**PROFESSOR &**
**HEAD OF THE DEPARTMENT**
Department of Electronics &
Communication Engineering
Velammal Engineering College
Ambattur - redhills road.
Chennai-66

# ABSTRACT

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 analyse it for the nutritional content which can be very handy and improves the dietary habits, and therefore, helps in maintaining a healthy lifestyle.

# CHAPTER 1
# INTRODUCTION

## 1.1 Project Overview

• Living a healthy lifestyle can help prevent chronic diseases and long-term illnesses. Feeling good about yourself and taking care of your health are important for your self-esteem and self-image. Maintain a healthy lifestyle by doing what is right for your body.

• Good nutrition promotes not only better physical health and reduced susceptibility to disease, but has also been demonstrated to contribute to cognitive development and academic success. Left to their own devices, children will not automatically select healthy foods.

• Nutrition is the process of consuming, absorbing, and using nutrients needed by the body for growth, development, and maintenance of life. To receive adequate, appropriate nutrition, people need to consume a healthy diet, which consists of a variety of nutrients the substances in foods that nourish the body.

• At last I want to conclude that food and health both are related to each other. Our health depends upon what food we eat and how much we eat. Therefore we should be careful while eating.

## 1.2 Purpose

The Purpose of our Project is :

The users continue to demand to know the nutritional value that is in their food. The users learn about the effect of different foods on human health. Evidently, the ultimate aim of this application is to provide the ways in which one can lead a healthy life by maintaining his/her diet. The user can access the nutritional information by taking a photo of the food, uploading a photo from the gallery, or by entering manually.

# CHAPTER 2

# LITERATURE SURVEY

1. **Djilani Kebaili, Eric Antoine Scuccimarra,Gaurav Singhal, Harris Heritier,Marcel Salathe, Sharada Prasanna Mohanty, Victor Boulanger (2016) "The Food Recognition Benchmark: Using Deep Learning to Recognize Food in Images", International Conference On Identification And Knowledge On IOT.**

Information and communication technologies (ICTs), such as computer, mobile phone, and internet, are emerged as an accelerator in the health sector development. They can play a critical role in pursuing outcome for most vulnerable groups, such as pregnant woman and children in developing countries. In this paper, we review the use of m-Health to improve mother and child health. We use the result of this review to propose a design for a set of m-Health applications focused on the improvement of mother and child nutrition in Indonesia.

**2. Alisha Lalani, Md.Riyazudin, Mousmi Ajay Chaurasia, Salva Fathima, Syed Ibrahim Ibaad (2022) "Estimation of Nutrition content".**

The health of women and children is the cornerstone of health of the whole people, and the foundation and premise of sustainable development of mankind. Sufficient nutrition is of great important to the health of pregnant women and infants, and an effective way to prevent birth defects. Pregnancy is a very important period for the formation and development of fetal tissues and organs. Macro-nutrients and micro-nutrients can affect DNA stability by affecting methylation, and they are also the direct regulators of DNA stability and phenotypic adaptability. Nutrients act as substrates, transcription factors and modifying factors of gene expression, and the changes of fetal nutrition may significantly affect offspring's structure and physiology.

**3. Bojia Qiu , Chenxi Huang , Kunhui Lin , Landu Jiang , Xue Liu (2022) "Deep Food: Food Image Analysis and Dietary Assessment via Deep Model", International Journal Of Scientific And Engineering Research**.

Some people tend to leave their food when eating caused by their changing lifestyle during the time. Leaving food means wasting its nutritional content acquired in people's bodies. By understanding the number of nutrient loss, the factors that influence of leftovers food is found, so that, it can prevent the number of food waste. In this paper, we present a method of leftovers nutrition estimation from food images in a single tray box employing an image processing approach. This feature is also embedded in our prototype named as Smart Nutrition Box (SNB). We apply the Automatic Food Leftover Estimation (AFLE) algorithm, which is suitable to predict the weight of food images placed in the tray box. The information on food weight is then utilized for calculating nutrition inside food leftovers. By using Root Mean Square Error (RMSE), the experimental result achieves 1.35 of error. It shows that the proposed method is able to project the leftover nutrition of food.

**4. Hui Deng , Jianbo Wu , Xianghui Zeng , Ying Wang (2021) "A Comprehensive Survey of Image-Based Food Recognition and Volume Estimation Methods for Dietary Assessment", International Conference on Journal Publication.**

The association between nutrition and health has been repeatedly established by the field of nutrition science and evidence-based practices. Nevertheless, inadequate nutrition is still prevalent among Filipino households. As a response to this public health issue, a nutrition system called Virtual Dietitian (VD) was conceived. Through a mixed-methods approach, VD was beta tested via a user study and System Usability Scale (SUS) by six information technology experts and six registered dietitians. Participants performed the standardized tasks with a mean of 85% completion rate and 106.2 seconds, and graded SUS with a mean score of 83.4 (excellent). Albeit the prototype successfully exhibited the potential of VD as a nutrition system, qualitative feedback from experts revealed some modifications that are needed to accomplish before going to the next phase of the study.

## 2.1 Existing Problem :

• In this pandemic situation, we need to lead a healthier life by means of taking healthier intake of foods .

• But in our fast moving world while we taking food we can’t find a chart and check whether the food is nutritional food or not .

• Thus to overcome that risky we created a application known as nutrition assistant application.

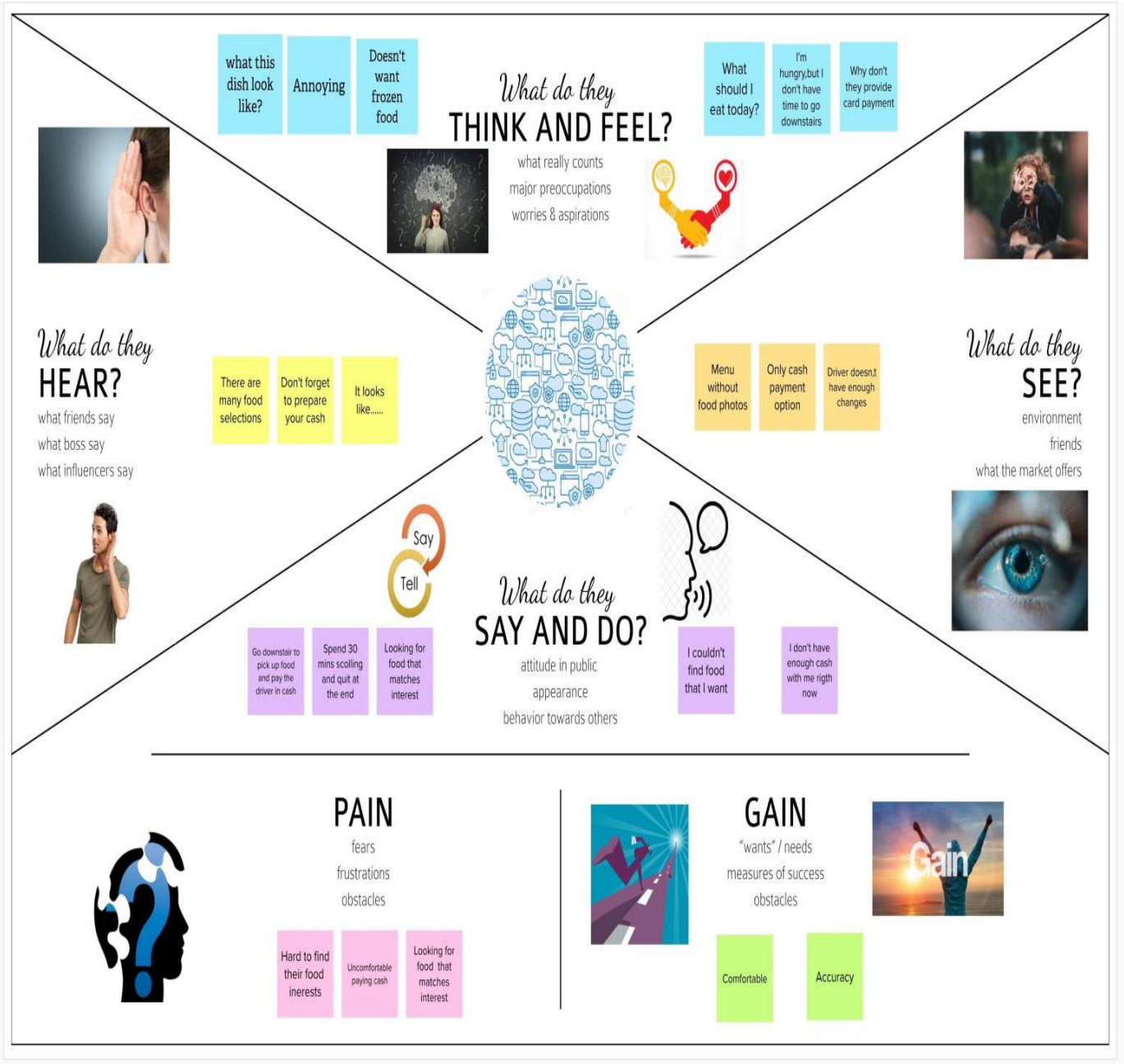
## 2.2 Problem Statement And Definition :

App-based nutrient dashboard systems which can analyse real-time images of a meal and analyse it for nutritional content which can be very handy and improves the dietary habits, and therefore, helps in maintaining a healthy lifestyle

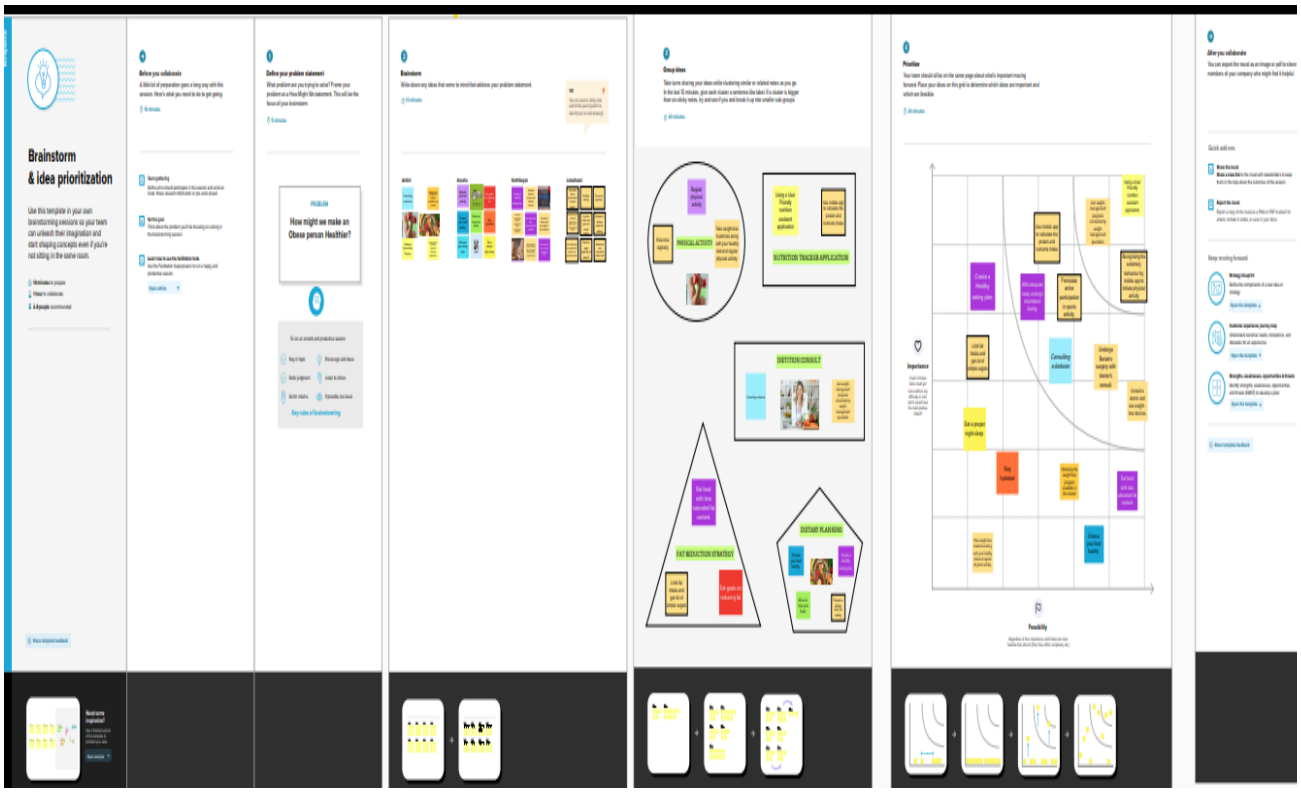
# CHAPTER 3

# IDEATION & PROPOSED SOLUTION

## 3.1 EMPATHY MAP CANVAS



## 3.2 IDEATION & BRAINSTORMING

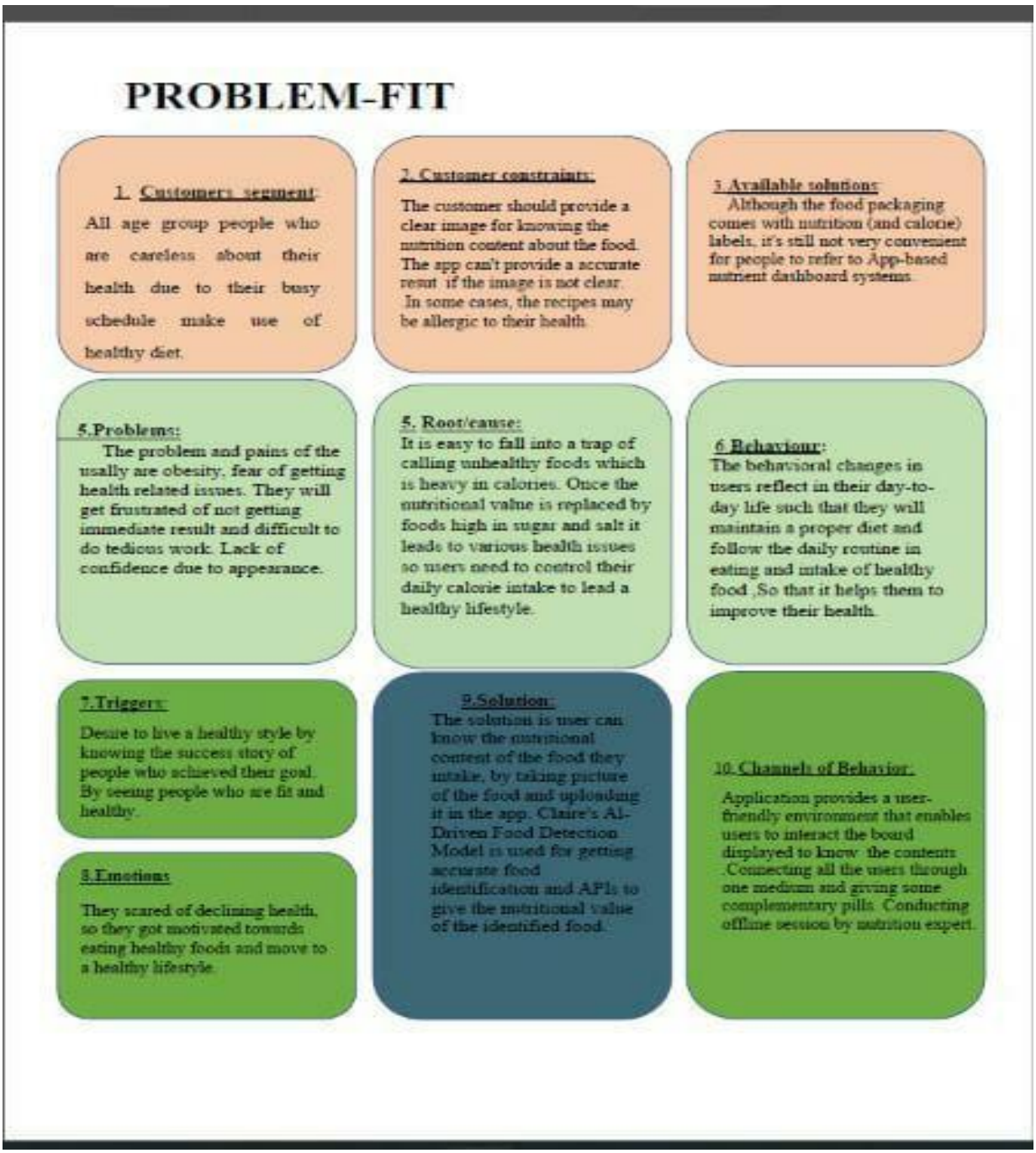



## 3.3 PROPOSED SOLUTION

| S.No. | Parameter | Description |
|---|---|---|
| 1. | Problem Statement (Problem to be solved) | 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. |
| 2. | Idea / Solution description | Nutrition Assistant Application,is the project on the above problem statement. The solution flows in a manner where user interacts with the web app to load an image. The image is passed to the server application. |

| | | |
|---|---|---|
| 3. | Novelty / Uniqueness | The innovative approach has<br>(i) Artificial intelligence based image recognization<br>(ii)Developing an web application (iii)Using the cloud for storing nutrition details (iv)Using API<br>(v)Spatial analysis |
| 4. | Social Impact / Customer Satisfaction | (i) User interacts with the Web App to Load an image.<br>(ii) The application model will be helpful for speed recognization of nutrition in food by image processing. |
| 5. | Business Model (Revenue Model) | **maintenance-as-a-service:** This is an application where users can maintain their health according to the artificial intelligence guidance through IBM Watson.The application name is Nutrition Assistant Application.The name itself reflects , it will assist the nutrition levels from capturing images of food. |
| 6. | Scalability of the Solution | Since most Nutrition assistant data are small files that lead to many small files, IBM cloud cannot be effective without a distributed system equipped with a high-performance computing system. To address this problem, IBM DB2 and SendGrid has been designed to process large (and small size) datasets. Using cloud computing technology in a Nutrition Assistant Apllication platform is another solution that can address scalability challenges related to capacity due to flexible and robust |

## 3.4 PROBLEM SOLUTION FIT



# CHAPTER 4
# REQUIREMENT ANALYSIS

## 4.1 FUNCTIONAL REQUIREMENT

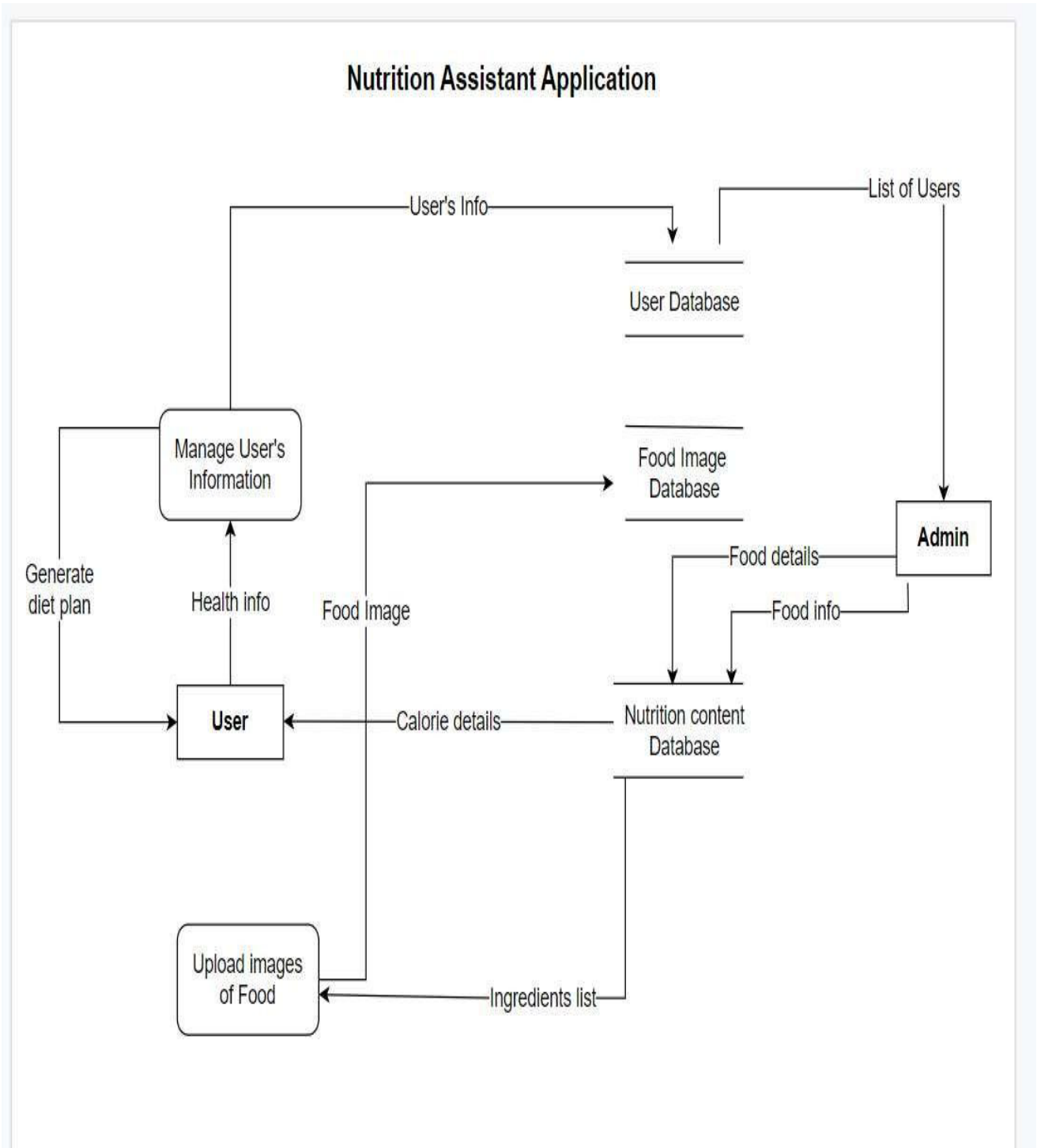
| FR No. | Functional Requirement (Epic) | Sub Requirement (Story / Sub-Task) |
|---|---|---|
| FR-1 | User Registration | Registration through Gmail<br>Registration through Form<br>Registration through LinkedIN |
| FR-2 | User Confirmation | Confirmation via Email<br>Confirmation via OTP |
| FR-3 | Diet Plan | can customize based on user needs |
| FR-4 | My Nutrition | Can add or update the existing food plan based on user needs |
| FR-5 | Workout | walking, gym |
| FR-6 | Nutrients Display | Display nutrients through IBM Cloud |
| FR-7 | Consultation | User can get consultation with doctor whenever the user want |

## 4.2 NON FUNCTIONAL REQUIREMENT

| No. | Non-Functional Requirement | Description |
|---|---|---|
| NFR-1 | Usability | Using Android or IOS or windows application. |
| NFR-2 | Security | The user data is stored securely in IBM Cloud. |
| NFR-3 | Reliability | The Quality Of the Service are trusted. |
| NFR-4 | Performance | It provide smooth user experience. |
| NFR-5 | Availability | The Service are available for 24 /7. |
| NFR-6 | Scalability | It is easy to scalable size for users. |

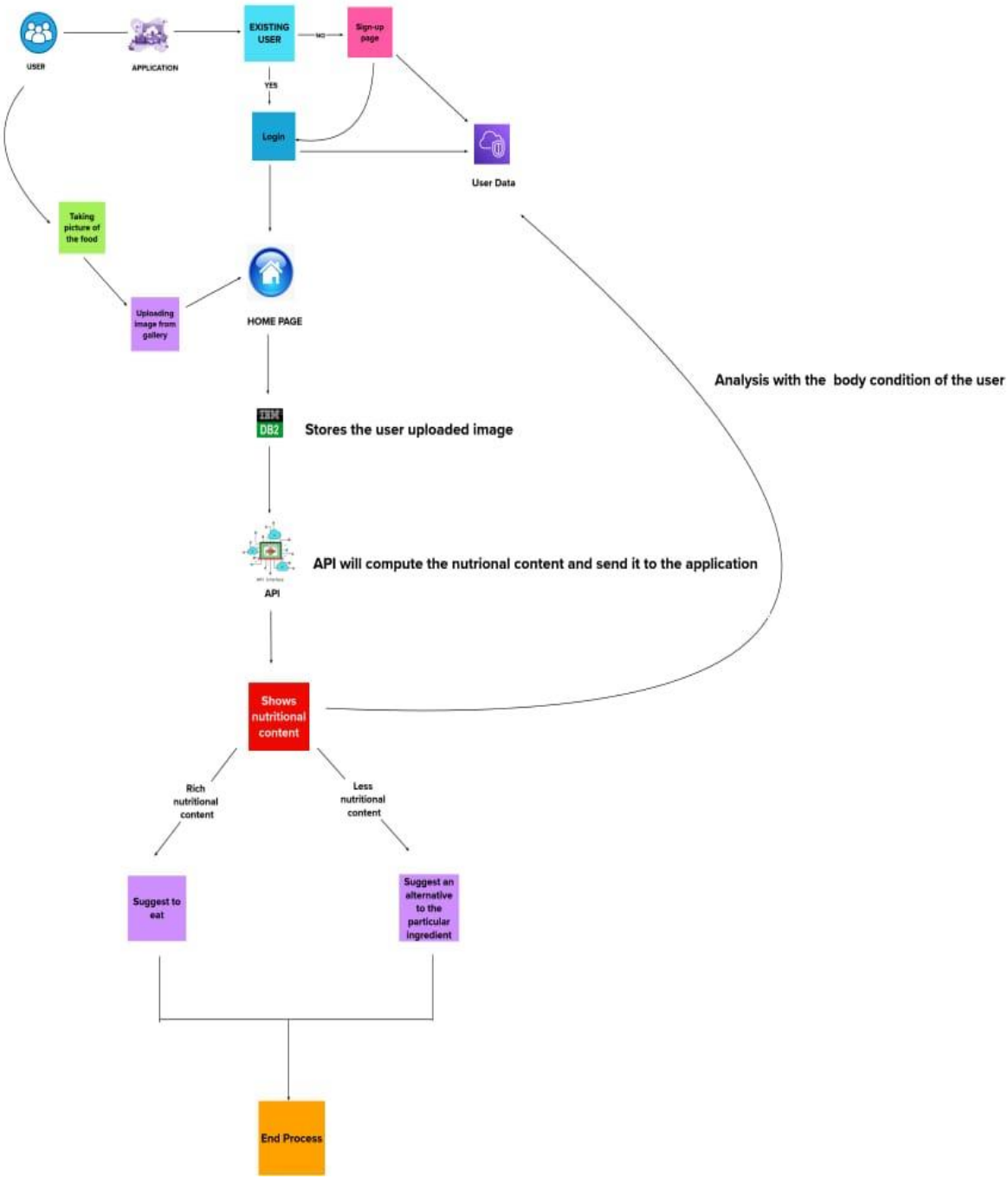
# CHAPTER 5
# PROJECT DESIGN

## 5.1 DATA FLOW DIAGRAM

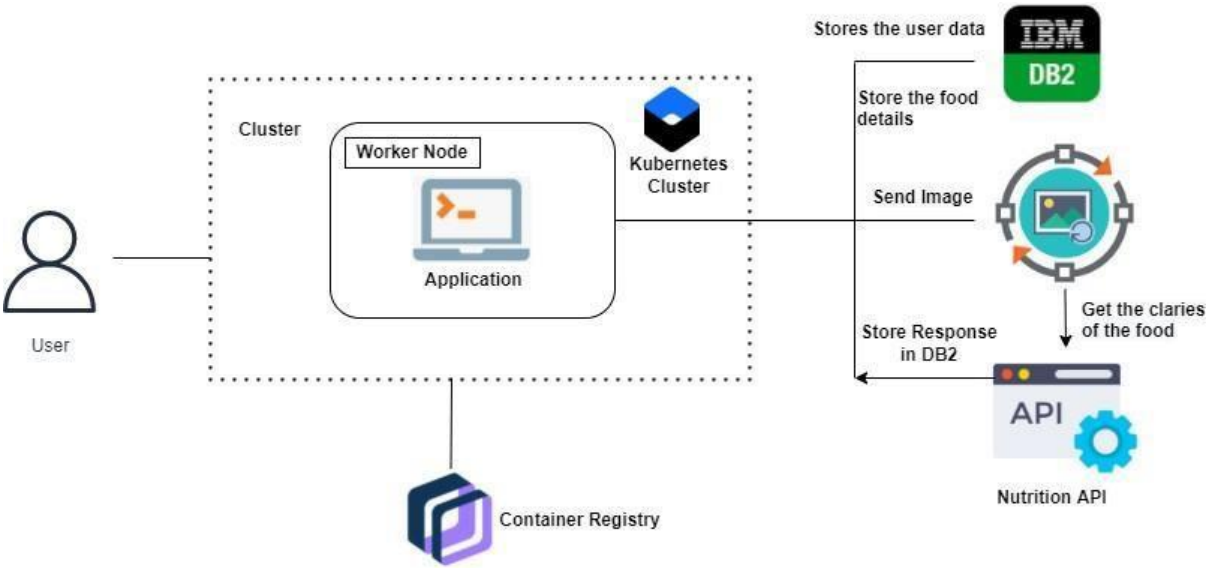


## 5.2 SOLUTION & TECHNICAL ARCHITECTURE

SOLUTION ARCHITECTURE FOR NUTRITIONAL ASSISTANT APPLICATION



**Technical Architecture:**




## 5.3 USER STORIES

1. As a user, I can register for the application by entering my email, password, and Confirm my password

2. As a user, I will receive confirmation email once I have registered for the application

3. As a user, I can log into the application by entering email & password

4. As a user, I can fill the details.

5. As a user, I can register for the application by entering my email, password, and Confirm my password

6. As a user, I will receive confirmation email once I have registered for the application

7. As a user, I can log into the application by entering email & password

8. As a user, I can fill the details.

9. As a user,I will search the food items.

10. As a user, I can scan the food an get the nutrition details and recipe for related scanned food.

# CHAPTER 6
# PROJECT PLANNING & SCHEDULING

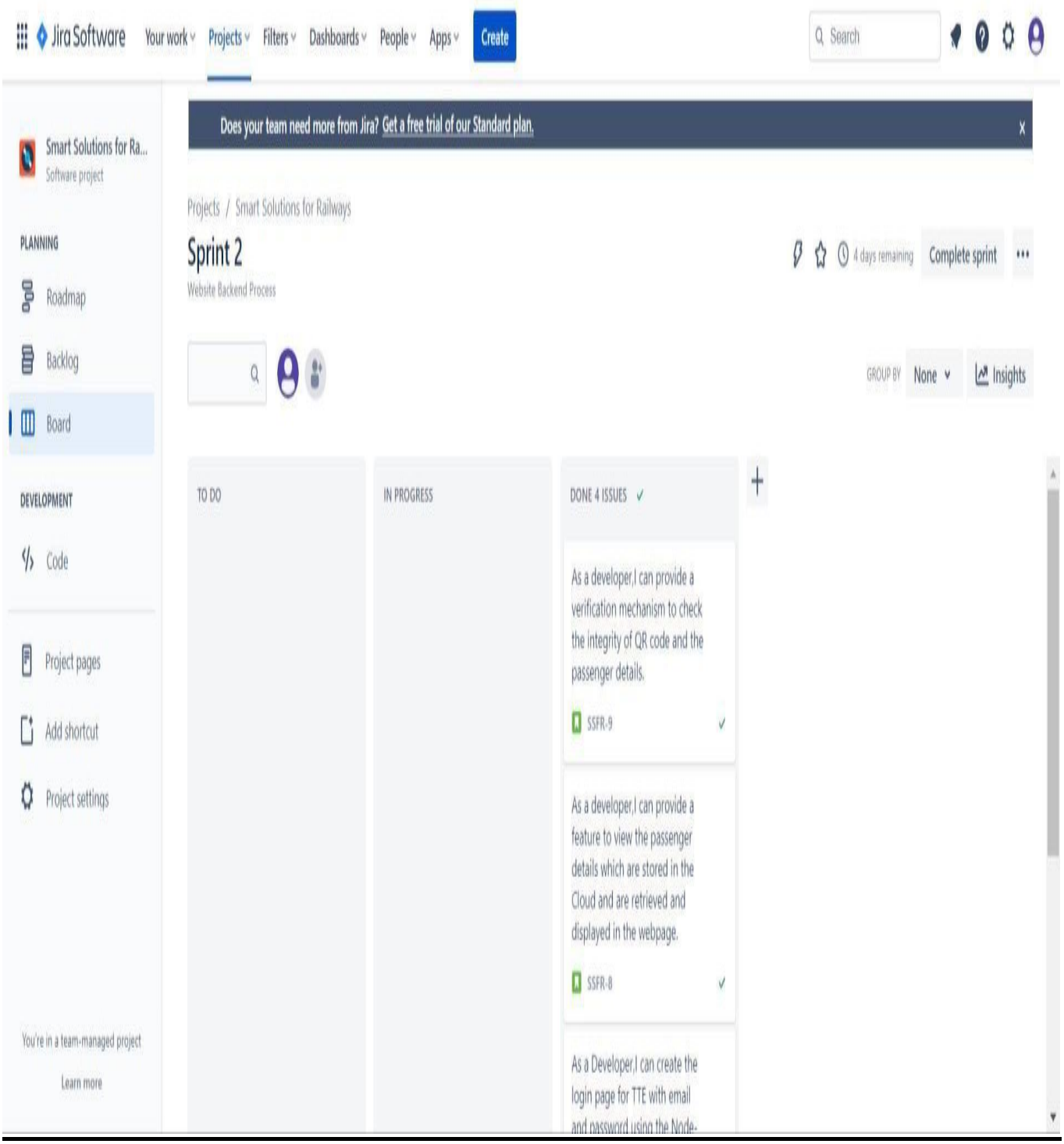
## 6.1 SPRINT PLANNING & ESTIMATION

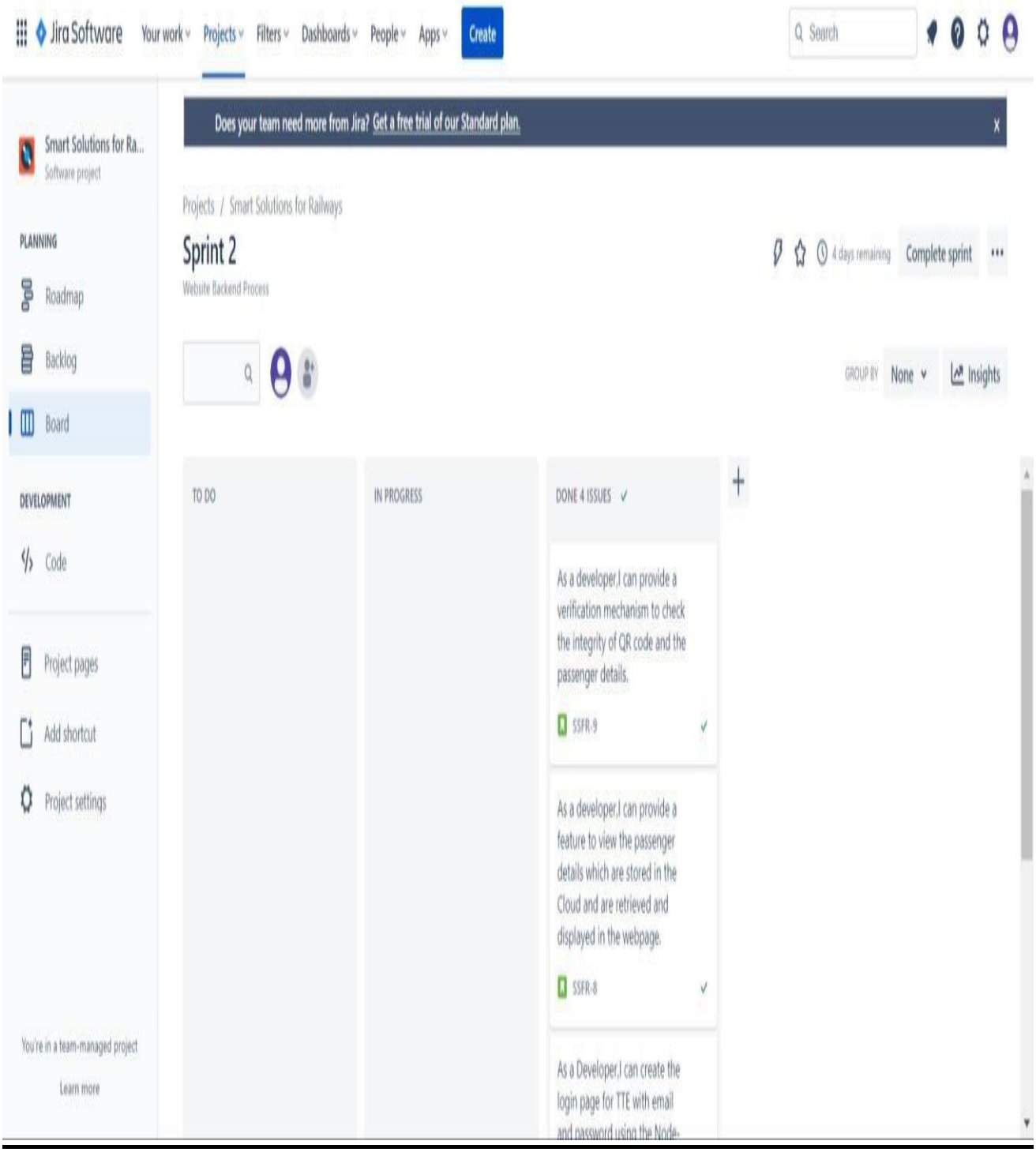
| Sprint | Functional Requirement(epic) | User Story number | Uer story and task | Story points | priority | Team Members |
|---|---|---|---|---|---|---|
| Sprint 1 | Registration | USN 1 | As a user, I can register for the application entering my email, password, and confirming my password | 2 | high | NITEESH REDDY Y<br>PARTHASARATHY M J<br>YASWANTH BABU K<br>HITESH K S<br>HARISH A |
| Sprint 1 | | USN 2 | As a user, I will receive confirmation email once I have registered for the application | 1 | high | NITEESH REDDY Y<br>PARTHASARATHY M J<br>YASWANTH BABU K<br>HITESH K S<br>HARISH A |
| Sprint 1 | Login | USN 3 | As a user, I can log into the application entering email & password | 1 | high | NITEESH REDDY Y<br>PARTHASARATHY M J<br>YASWANTH BABU K<br>HITESH K S<br>HARISH A |
| Sprint 2 | User details | USN 4 | As a user , I can fill the Details. | 2 | medium | NITEESH REDDY Y<br>PARTHASARATHY M J<br>YASWANTH BABU K<br>HITESH K S<br>HARISH A |
| Sprint 3 | Push notification | USN 5 | As a user, I will search the food items | 2 | high | NITEESH REDDY Y<br>PARTHASARATHY M J<br>YASWANTH BABU K<br>HITESH K S<br>HARISH A |
| Sprint 4 | Shown the nutrition details and Recipe for scanned food | USN 6 | As a user, I can scan the food an get the nutrition details and recipe for related scanned | 1 | high | NITEESH REDDY Y<br>PARTHASARATHY M J<br>YASWANTH BABU K<br>HITESH K S<br>HARISH A |

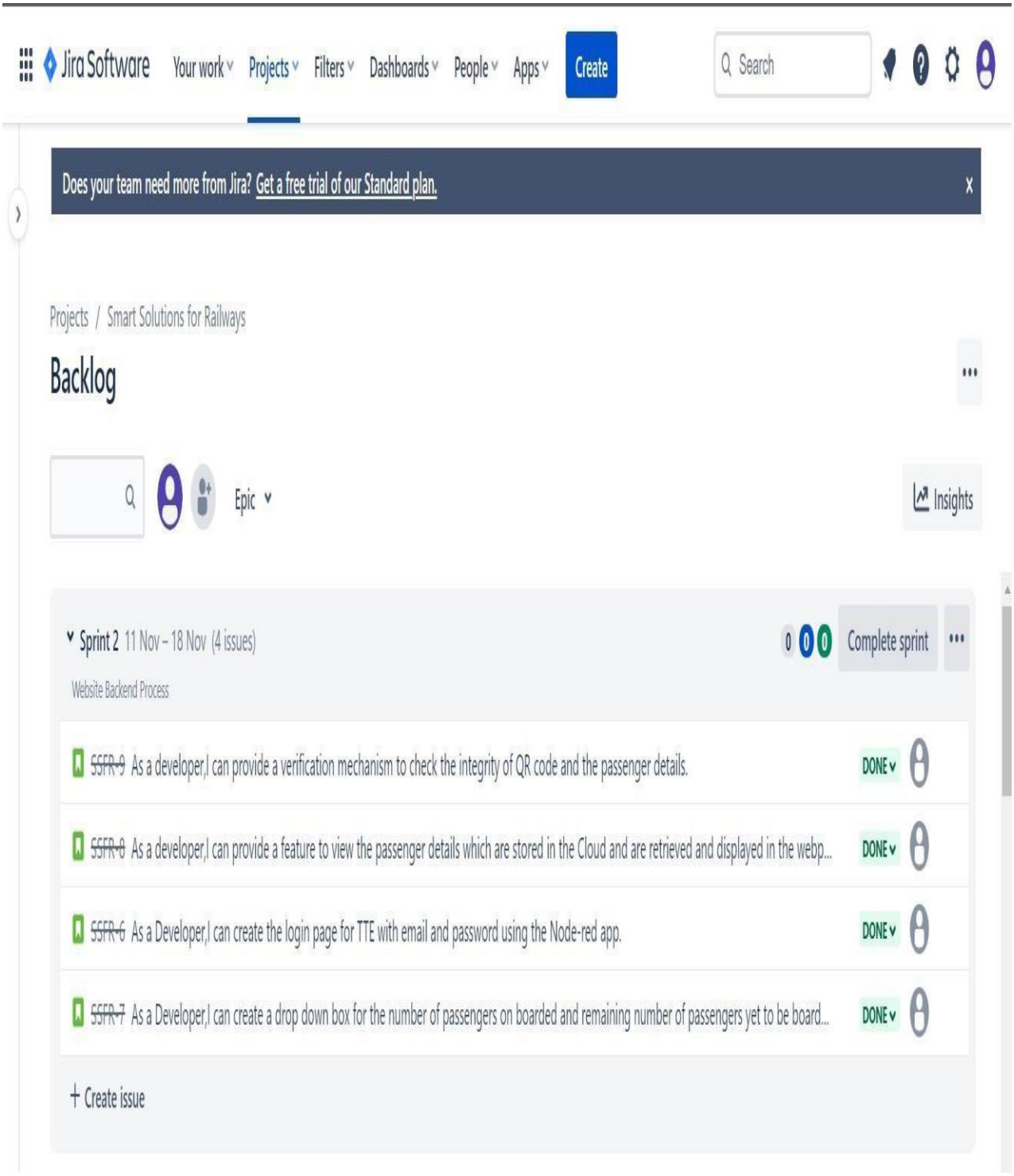
## 6.2 SPRINT DELIVERY SCHEDULE

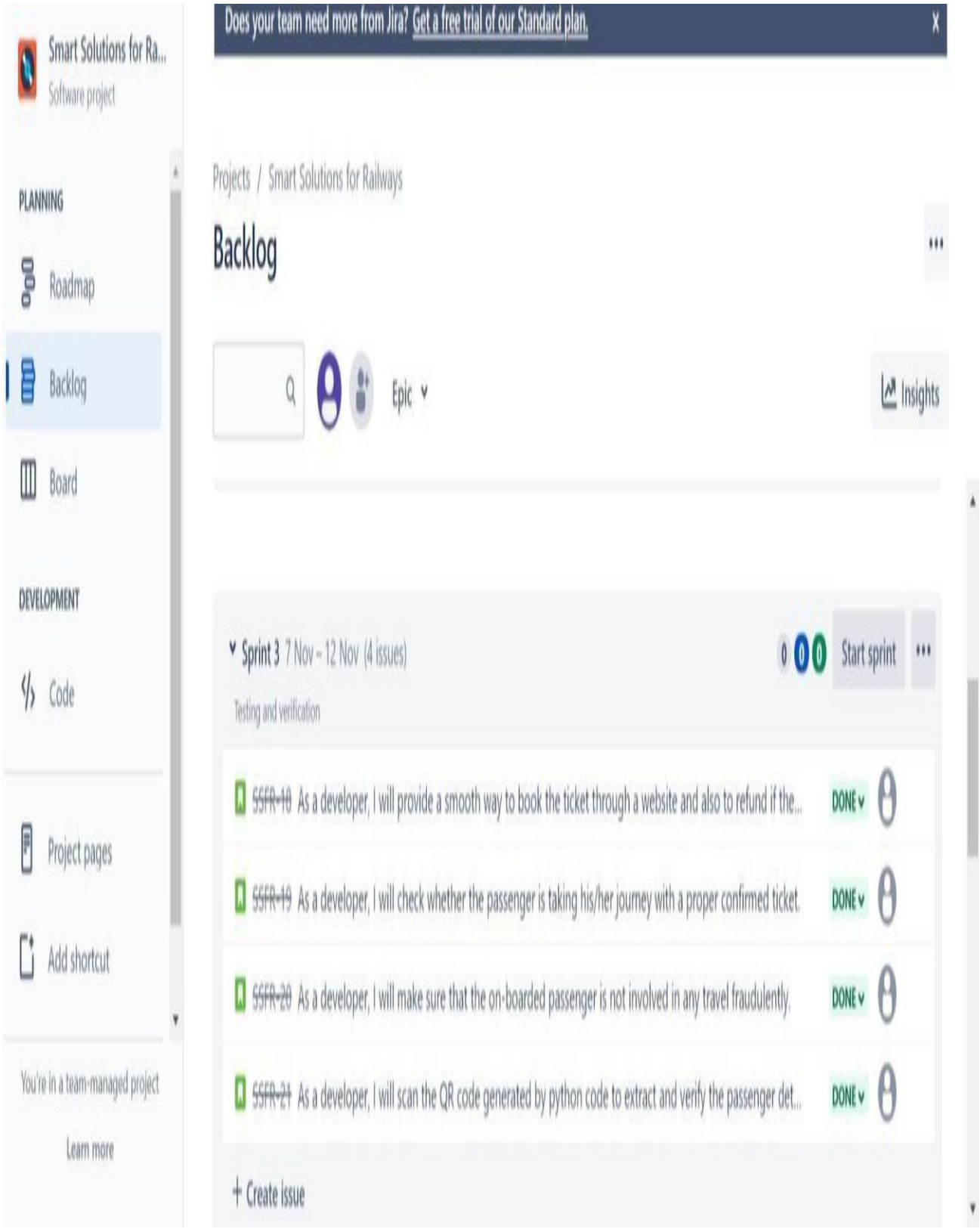
| | | | | | | |
|---|---|---|---|---|---|---|
| Sprint-1 | 20 | 6 Days | 24 Oct 2022 | 29 Oct 2022 | 20 | 29 Oct 2022 |
| Sprint-2 | 20 | 6 Days | 31 Oct 2022 | 05 Nov 2022 | 20 | 07 Oct 2022 |
| Sprint-3 | 20 | 6 Days | 07 Nov 2022 | 12 Nov 2022 | 20 | 10 Oct 2022 |
| Sprint-4 | 20 | 6 Days | 14 Nov 2022 | 19 Nov 2022 | 20 | 15 Oct 2022 |

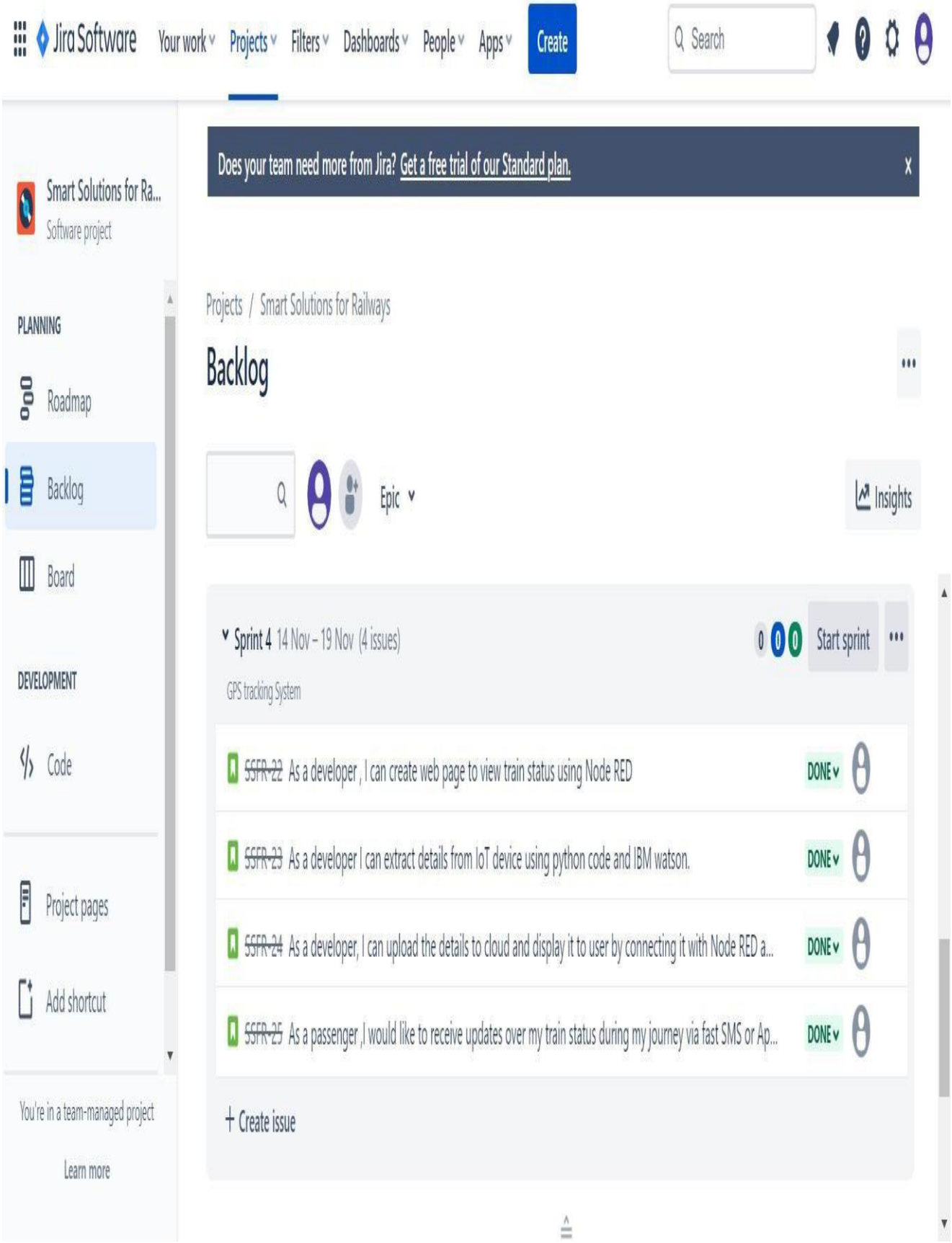
## 6.3 JIRA REPORT:

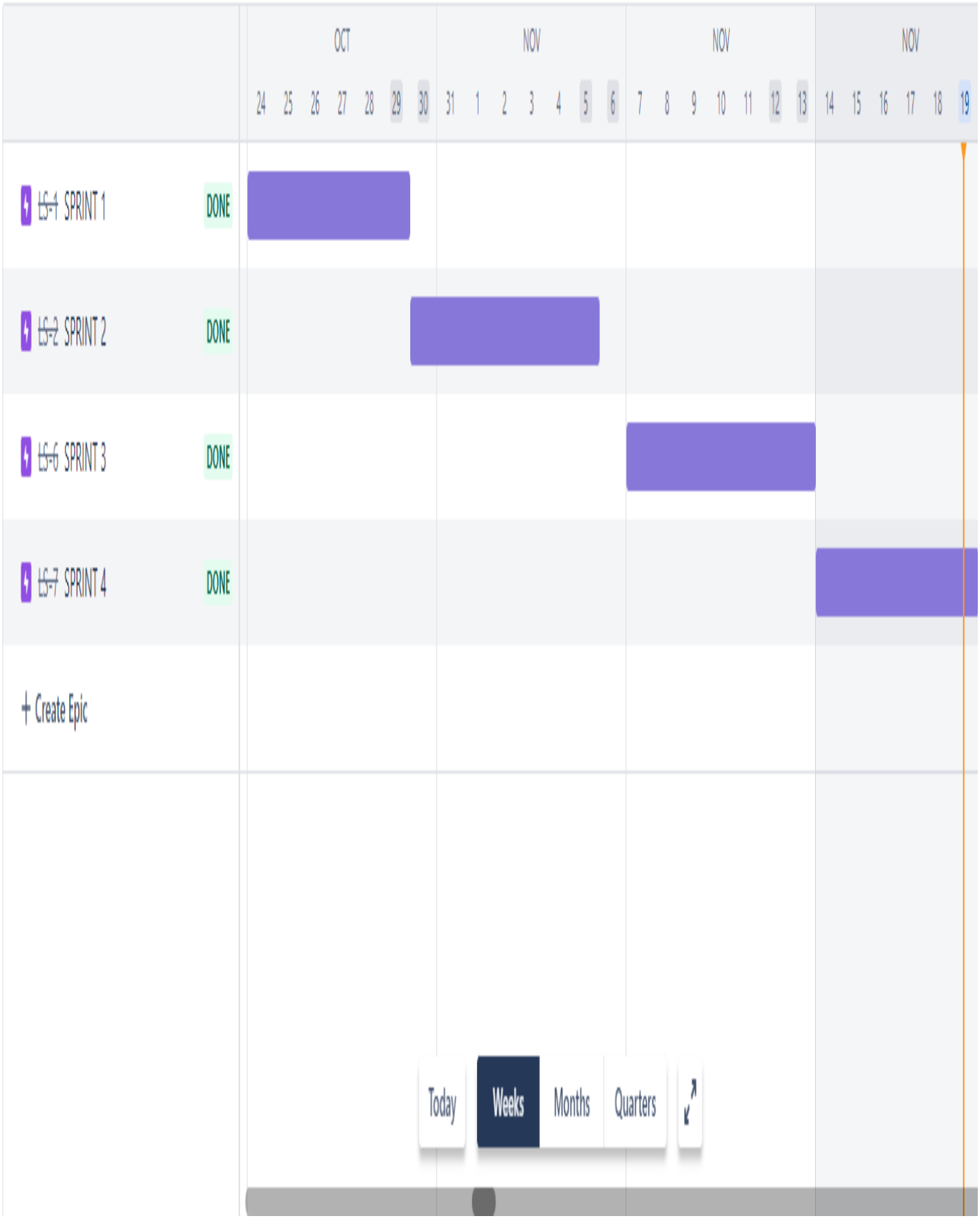



Jira Software
Your work
Projects
Filters
Dashboards
People
Apps
Create
Search
Smart Solutions for Ra...
Software project
PLANNING
Roadmap
Backlog
Board
DEVELOPMENT
Code
Project pages
Add shortcut
Project settings
You're in a team-managed project
Learn more
Does your team need more from Jira? Get a free trial of our Standard plan.
Projects / Smart Solutions for Railways
Sprint 2
Website Backend Process
4 days remaining
Complete sprint
GROUP BY
None
Insights
TO DO
IN PROGRESS
DONE 4 ISSUES
As a developer,I can provide a verification mechanism to check the integrity of QR code and the passenger details.
SSFR-9
As a developer,I can provide a feature to view the passenger details which are stored in the Cloud and are retrieved and displayed in the webpage.
SSFR-8
As a Developer,I can create the login page for TTE with email and password using the Node-


Jira Software
Your work
Projects
Filters
Dashboards
People
Apps
Create
Search
Does your team need more from Jira? Get a free trial of our Standard plan.
Projects / Smart Solutions for Railways
Backlog
Epic
Insights
Sprint 2 11 Nov – 18 Nov (4 issues)
0 0 0
Complete sprint
Website Backend Process
SSFR-9 As a developer,I can provide a verification mechanism to check the integrity of QR code and the passenger details.
DONE
SSFR-8 As a developer,I can provide a feature to view the passenger details which are stored in the Cloud and are retrieved and displayed in the webp...
DONE
SSFR-6 As a Developer,I can create the login page for TTE with email and password using the Node-red app.
DONE
SSFR-7 As a Developer,I can create a drop down box for the number of passengers on boarded and remaining number of passengers yet to be board...
DONE
Create issue

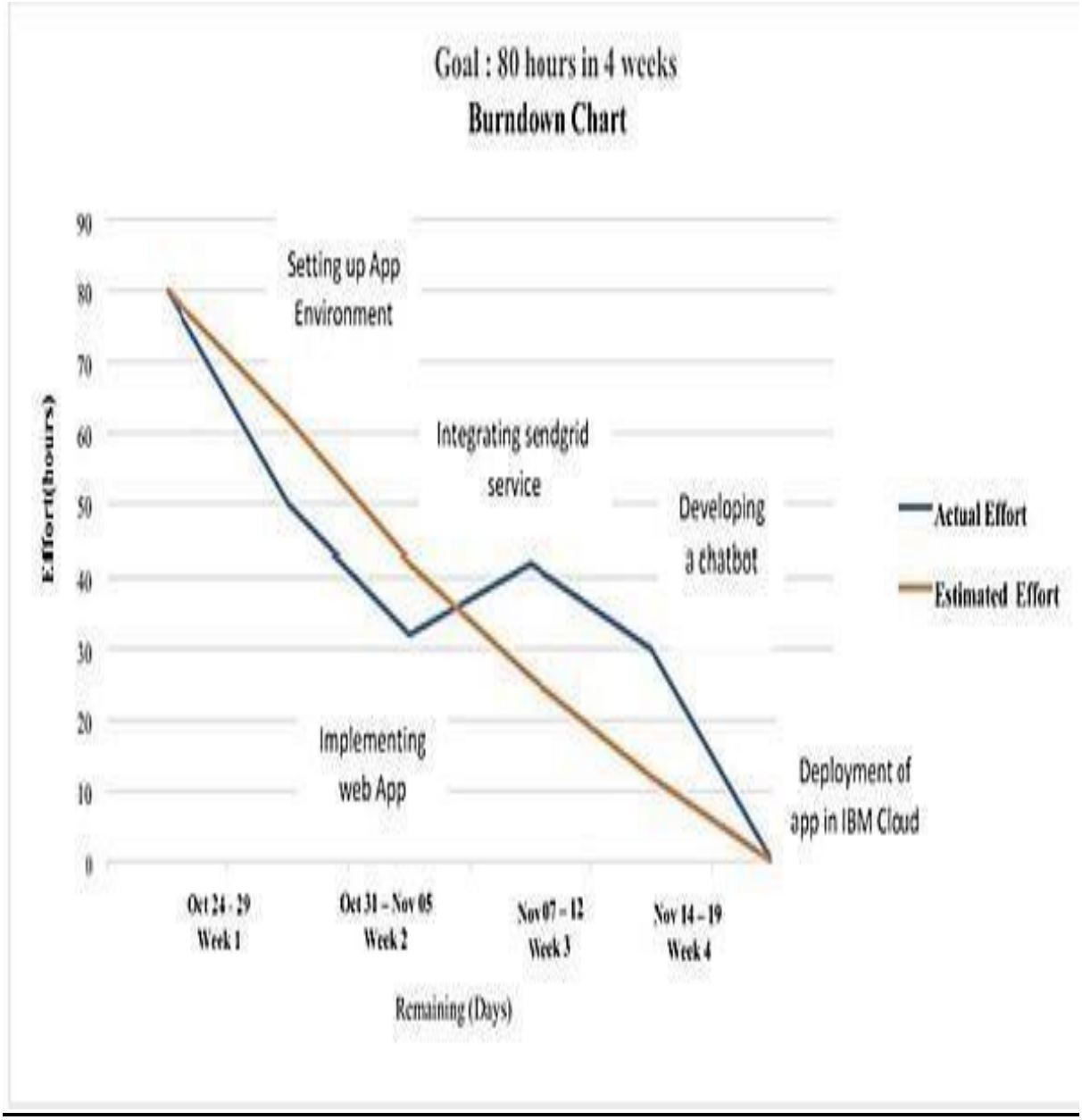

Smart Solutions for Ra...
Software project
PLANNING
Roadmap
Backlog
Board
DEVELOPMENT
Code
Project pages
Add shortcut
You're in a team-managed project
Learn more
Does your team need more from Jira? Get a free trial of our Standard plan.
Projects / Smart Solutions for Railways
Backlog
Epic
Insights
Sprint 3 7 Nov – 12 Nov (4 issues)
Start sprint
Testing and verification
SSFR-18 As a developer, I will provide a smooth way to book the ticket through a website and also to refund if the...
DONE
SSFR-19 As a developer, I will check whether the passenger is taking his/her journey with a proper confirmed ticket.
DONE
SSFR-20 As a developer, I will make sure that the on-boarded passenger is not involved in any travel fraudulently.
DONE
SSFR-21 As a developer, I will scan the QR code generated by python code to extract and verify the passenger det...
DONE
Create issue


Jira Software
Your work
Projects
Filters
Dashboards
People
Apps
Create
Search
Does your team need more from Jira? Get a free trial of our Standard plan.
Smart Solutions for Ra...
Software project
PLANNING
Roadmap
Backlog
Board
DEVELOPMENT
Code
Project pages
Add shortcut
You're in a team-managed project
Learn more
Projects / Smart Solutions for Railways
Backlog
Epic
Insights
Sprint 4 14 Nov – 19 Nov (4 issues)
Start sprint
GPS tracking System
SSFR-22 As a developer , I can create web page to view train status using Node RED
DONE
SSFR-23 As a developer I can extract details from IoT device using python code and IBM watson.
DONE
SSFR-24 As a developer, I can upload the details to cloud and display it to user by connecting it with Node RED a...
DONE
SSFR-25 As a passenger ,I would like to receive updates over my train status during my journey via fast SMS or Ap...
DONE
Create issue


OCT
NOV
NOV
NOV
24 25 26 27 28 29 30 31 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19
LG-1 SPRINT 1
DONE
LG-2 SPRINT 2
DONE
LG-6 SPRINT 3
DONE
LG-7 SPRINT 4
DONE
Create Epic
Today
Weeks
Months
Quarters

## Burn down Chart:

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



# CHAPTER 7
# CODING & SOLUTIONING

## 7.1 FEATURE 1

### Python Flask

Python Flask is used to develop chatbot applications using python. Flask is mainly used to render and integrate the nutrition assistant application in the browser by providing API. By running the python application, the suitable server domain link is obtained and run in the browser.

### HTML

The HTML and CSS is used to design the overall nutrition assistant application's UI**.** HTML is used to add UI components and CSS is used to add style to those components.

### Build PYTHON FLASK Code*:*

### APP.PY

```
from flask import Flask, render_template, redirect, url_for, request, flash, escape, session
from flask_wtf import FlaskForm
from wtforms import StringField, PasswordField, EmailField
from wtforms.validators import InputRequired, Length, Email, EqualTo
import ibm_db
import time
import openapi_client
from com.spoonacular import misc_api
from openapi_client.model.image_analysis_by_url200_response import
ImageAnalysisByURL200Response
from pprint import pprint
from flask import Flask,redirect,url_for,render_template,request
import ibm_boto3
from ibm_botocore.client import Config, ClientError
import json
import os
from dotenv import load_dotenv
load_dotenv()
```

```
COS_ENDPOINT="https://s3.jp-tok.cloud-object-storage.appdomain.cloud"
COS_API_KEY_ID= os.getenv("COSAPI")
COS_INSTANCE_CRN= os.getenv("COSINSTANCE")
# Create resource https://s3.ap.cloud-object-storage.appdomain.cloud
cos = ibm_boto3.resource("s3",
ibm_api_key_id=COS_API_KEY_ID,
ibm_service_instance_id=COS_INSTANCE_CRN,
config=Config(signature_version="oauth"),
endpoint_url=COS_ENDPOINT
)
DB_HOSTNAME = os.getenv("DB_HOSTNAME")
DB_PORT = os.getenv("DB_PORT")
DB_USERNAME = os.getenv("DB_USERNAME")
DB_PASS = os.getenv("DB_PASS")
#conn =
ibm_db.connect(f"DATABASE=bludb;HOSTNAME={DB_HOSTNAME};PORT={DB_PORT};SECU
RITY=SSL;SSLServerC
ertificate=DigiCertGlobalRootCA.crt;UID={DB_USERNAME};PWD={DB_PASS}",",")
conn =
ibm_db.connect("DATABASE=bludb;HOSTNAME=3883e7e4-18f5-4afe-be8c-
fa31c41761d2.bs2io90l08kqb1od
8lcg.databases.appdomain.cloud;PORT=
31498;SECURITY=SSL;SSLServerCertificate=DigiCertGlobalRootCA.crt;UID=sxq13787;PWD=h4bJ
WERMtK7
2WIkp",", ")
print(conn)
app = Flask(__name__)
app.config['SECRET_KEY'] = os.getenv("SECERT_KEY")
class LoginForm(FlaskForm):
email = EmailField("email", validators=[InputRequired("Email is required"), Email()])
password = PasswordField("password", validators=[InputRequired("Password is required")])
class RegisterForm(FlaskForm):
username = StringField("username", validators=[InputRequired("Username is required")])
email = EmailField("email", validators=[InputRequired("Email is required"), Email()])
```

```
pass1 = PasswordField("pass1", validators=[InputRequired("Password is required"),
EqualTo('pass2', message="Passwords must match"), Length(min=4, max=30, message="Length must
be between 4 and 30")])
pass2 = PasswordField("pass2")
class ForgetPassword(FlaskForm):
email = EmailField("email", validators=[InputRequired("Email is required"), Email()])
@app.route("/")
def home():
username = request.cookies.get('username')
return render_template("home.html", username=username)
@app.route('/login', methods=['GET', 'POST'])
def login():
form = LoginForm()
if request.method=='POST' and form.validate_on_submit():
email = request.form['email']
password = request.form['password']
sql = f"SELECT * FROM user WHERE EMAIL='{escape(email)}'"
stmt = ibm_db.exec_immediate(conn, sql)
dic = ibm_db.fetch_both(stmt)
if not dic or password != dic['PASSWORD']:
flash("Incorrect email or password", "error")
return redirect(url_for('login'))
session['username'] = dic['USERNAME']
return redirect(url_for('home'))
Else:
return render_template("login.html", form=form)
@app.route('/signup', methods=['GET', 'POST'])
def signup():
form = RegisterForm()
if request.method=='POST' and form.validate_on_submit():
username = request.form['username']
email = request.form['email']
pass1 = request.form['pass1']
sql = f"SELECT * FROM user WHERE EMAIL='{escape(email)}'"
```

```
stmt = ibm_db.exec_immediate(conn, sql)
dic = ibm_db.fetch_both(stmt)
if dic:
flash("User with the email already exist", "error")
return redirect(url_for('login'))
sql = "INSERT INTO user(username,email,password) VALUES (?, ?, ?)"
prep_stmt = ibm_db.prepare(conn, sql)
ibm_db.bind_param(prep_stmt, 1, username)
ibm_db.bind_param(prep_stmt, 2, email)
ibm_db.bind_param(prep_stmt, 3, pass1)
ibm_db.execute(prep_stmt)
flash("Registration Successful", "success")
response = redirect('/login',307)
return response
else:
return render_template("register.html", form=form)
@app.route('/forgot', methods=['GET', 'POST'])
def forgot_password():
form = ForgetPassword()
if request.method=='POST' and form.validate_on_submit():
email = request.form['email']
sql = f"SELECT * FROM user WHERE EMAIL='{escape(email)}'"
stmt = ibm_db.exec_immediate(conn, sql)
dic = ibm_db.fetch_both(stmt)
if dic:
flash("Email has been sent if user exist", "success")
return redirect(url_for('forgot_password'))
return render_template("forgot_password.html", form=form)
return render_template("forgot_password.html", form=form)
@app.route('/logout', methods=['GET', 'POST'])
def logout():
session.pop('username')
return redirect(url_for('home'))
@app.route('/pictures')
```

```
def index():
files = get_bucket_contents('flask-app-2k22')
return render_template('index.html', files = files)
@app.route('/uploader', methods = ['GET', 'POST'])
def upload():
if request.method == 'POST':
bucket=request.form['bucket']
name_file=request.form['filename']
f = request.files['file']
multi_part_upload(bucket,name_file,f.filename)
sql = f"INSERT INTO imagedetails(img_link) VALUES(?)"
imagelink="https://flask-app-2k22.s3.jp-tok.cloud-object-storage.appdomain.cloud/"+name_file
print(imagelink)
prep_stmt = ibm_db.prepare(conn, sql)
ibm_db.bind_param(prep_stmt, 1, imagelink)
ibm_db.execute(prep_stmt)
sql = f"SELECT ID FROM imagedetails WHERE img_link='{escape(imagelink)}'"
stmt = ibm_db.exec_immediate(conn, sql)
image_id = ibm_db.fetch_both(stmt)
nutitionapi(imagelink,image_id)
return redirect("/foodinfo", code=307)
if request.method == 'GET':
return render_template('upload.html')
def get_item(bucket_name, item_name):
print("Retrieving item from bucket: {0}, key: {1}".format(bucket_name, item_name))
try:
file = cos.Object(bucket_name, item_name).get()
print("File Contents: {0}".format(file["Body"].read()))
except ClientError as be:
print("CLIENT ERROR: {0}\n".format(be))
except Exception as e:
print("Unable to retrieve file contents: {0}".format(e))
def get_bucket_contents(bucket_name):
print("Retrieving bucket contents from: {0}".format(bucket_name))
```

```
try:
files = cos.Bucket(bucket_name).objects.all()
files_names = []
for file in files:
files_names.append(file.key)
print("Item: {0} ({1} bytes).".format(file.key, file.size))
return files_names
except ClientError as be:
print("CLIENT ERROR: {0}\n".format(be))
except Exception as e:
print("Unable to retrieve bucket contents: {0}".format(e))
def multi_part_upload(bucket_name, item_name, file_path):
try:
print("Starting file transfer for {0} to bucket: {1}\n".format(item_name,
bucket_name))
# set 5 MB chunks
part_size = 1024 * 1024 * 5
# set threadhold to 15 MB
file_threshold = 1024 * 1024 * 15
# set the transfer threshold and chunk size
transfer_config = ibm_boto3.s3.transfer.TransferConfig(
multipart_threshold=file_threshold,
multipart_chunksize=part_size
)
# the upload_fileobj method will automatically execute a multi-part upload
# in 5 MB chunks for all files over 15 MB
with open(file_path, "rb") as file_data:
cos.Object(bucket_name, item_name).upload_fileobj(
Fileobj=file_data,
Config=transfer_config
print("Transfer for {0} Complete!\n".format(item_name))
except ClientError as be:
print("CLIENT ERROR: {0}\n".format(be))
except Exception as e:
```

```
print("Unable to complete multi-part upload: {0}".format(e))
def nutitionapi(imagelink,image_id):
configuration = openapi_client.Configuration(
host = "https://api.spoonacular.com"
)
configuration.api_key['apiKeyScheme'] = os.getenv("NUTRITIONAPI")
with openapi_client.ApiClient(configuration) as api_client:
api_instance = misc_api.MiscApi(api_client)
image_url =imagelink
try:
api_response = api_instance.image_analysis_by_url(image_url)
pprint(api_response)
y =api_response
cal= y["nutrition"]["calories"]["value"]
print(cal)
Carb= y["nutrition"]["carbs"]["value"]
fat= y["nutrition"]["fat"]["value"]
protein= y["nutrition"]["protein"]["value"]
name=y["category"]["name"]
image=image_id["ID"]
sql = f"INSERT INTO nutritiondetails1(calories,carbs,fat,protein,ref_id,name)
VALUES('{escape(cal)}','{escape(Carb)}','{escape(fat)}','{escape(protein)}','{escape(image)}'
,'{escape(name)}')"
prep_stmt = ibm_db.prepare(conn, sql)
ibm_db.execute(prep_stmt)
flash("Successful db operation", "success"
except openapi_client.ApiException as e:
print("Exception when calling MiscApi->image_analysis_by_url: %s\n" % e)
@app.route('/foodinfo', methods = ['GET', 'POST'])
def test():
sql = f"SELECT * FROM imagedetails "
stmt = ibm_db.exec_immediate(conn, sql)
pic = ibm_db.fetch_both(stmt)
pics=[]
```

```
while pic != False:
x=[pic["IMG_LINK"],pic["ID"]]
pics.append(x)
print(pic)
pic = ibm_db.fetch_both(stmt)
print(pics)
return render_template('index.html', files = pics)
@app.route('/nutritioninfo/<id>', methods = ['GET', 'POST'])
def test1(id):
print(id)
sql = f"SELECT * FROM nutritiondetails1,imagedetails where
nutritiondetails1.ref_id=imagedetails.id and ref_id='{escape(id)}'"
stmt = ibm_db.exec_immediate(conn, sql)
pic = ibm_db.fetch_both(stmt)
print(pic)
return render_template('foodinfo.html', files = pic)
if __name__ == '__main__':
app.run(debug=True)
```

## REGISTER.HTML:

```
{% extends "base.html" %}
{% from "_render_field.html" import render_error_field %}
{% block head %}
<title>Registration</title>
<link rel="stylesheet" href="{{url_for('static', filename='css/login_style.css')}}">
{% endblock %}
{% block body %}
<div class="container h-100">
<div class="wrapper">
<div class="card">
<form class="d-flex flex-column" action="{{ url_for('signup') }}" method="POST"
novalidate>
{{ form.csrf_token }}
```

```
<div class="h3 text-center text-white">Sign Up</div>
<div class="d-flex align-items-center input-field my-3 mb-4">
<span class="far fa-user p-2"></span>
{{ form.username(class="form-control", placeholder="Username") }}
</div>
{{ render_error_field(form.username) }}
<div class="d-flex align-items-center input-field mb-4">
<span class="far fa-envelope p-2"></span>
{{ form.email(class="form-control", placeholder="Email") }}
</div>
{{ render_error_field(form.email) }}
<div class="d-flex align-items-center input-field mb-4">
<span class="fas fa-lock p-2"></span>
{{ form.pass1(class="form-control", placeholder="New Password", id='pwd')
}}
<button type="button" class="btn" onclick="showPassword()">
<span class="fas fa-eye-slash"></span>
</button>
</div>
{{ render_error_field(form.pass1) }}
<div class="d-flex align-items-center input-field mb-4">
<span class="fas fa-lock p-2"></span>
{{ form.pass2(class="form-control", placeholder="New Password",
id='pwd1') }}
<button type="button" class="btn" onclick="Password()">
<span class="fas fa-eye-slash"></span>
</button>
</div>
{{ render_error_field(form.pass2) }}
<div class="my-3">
<input type="submit" value="Register" class="btn btn-primary">
</div>
<div class="mb-3">
<span class="text-light-white">Already have an account?</span>
```

```
<a href="/login">Login</a>
</div>
</form>
</div>
</div>
</div>
{% endblock %}
```

## HOME.HTML:

```
{% extends "base.html" %}
{% block head %}
<title>Nutrify</title>
<link rel="stylesheet" type="text/css" href="{{url_for('static',
filename='css/indexstyle.css')}}">
{% endblock %}
{% block body %}
<header>
<nav class="navbar navbar-expand-lg navigation-wrap">
<div class="container">
<a class="navbar-brand" href="/">Nutrition Assistant Application</a>
<button class="navbar-toggler" type="button" data-bs-toggle="collapse"
data-bs-target="#navbarNav"
aria-controls="navbarNav" aria-expanded="false" aria-label="Toggle
navigation">
<span class="navbar-toggler-icon"></span>
</button>
<div class="collapse navbar-collapse" id="navbarText">
<ul class="navbar-nav ms-auto mb-2 mb-lg-0">
<li class="nav-item">
<a class="nav-link" aria-current="page" href="/">Home</a>
</li>
<li class="nav-item">
<a class="nav-link" aria-current="page" href="/foodinfo">Food
```

```
Info</a>
</li>
{% if session['username'] %}
<li class="nav-item">
<a class="nav-link" aria-current="page" href="/uploader">Upload
Images</a>
</li>
{% endif %}
<li class="nav-item">
<a class="nav-link" aria-current="page" href="/#aboutus">About Us</a>
</li>
{% if session['username'] %}
<li>
<a class="btn btn-outline-danger ms-2" href="{{ url_for('logout')
}}">Logout</a>
</li>
{% else %}
<li>
<a class="btn btn-outline-danger ms-2" href="/login"
role="button">Login</a>
</li>
<li>
<a class="btn btn-primary ms-4" href="/signup"
role="button">Register</a>
</li>
{% endif %}
</ul>
</div>
</div>
</nav>
</header>
<section id="home">
<div class="test">
<div class="main">
```

```
</div>
<div class="textbox">
{% if session['username'] %}
<h1>Welcome <span style="color:#91eb48;">{{ session['username'] | title
}}</span></h1>
<!-- <a href="{{ url_for('logout') }}">Logout</a> -->
{% else %}
<h1>Welcome to <span style="color:#40a008;">Nutrify</span></h1>
{% endif %}
<h4>All groundwork of happiness starts from health.</h4>
</div>
</div
</section>
<section class="feature section-padding" id="feature">
<div class="container-fluid px-0 top-banner1">
<div class="container">
<div class="feature-heading" style="visibility: visible; animation-name: zoom">
<h2><span style="color:#40a008;">Nutrify</span>'s Main Features</h2>
</div>
<div class="serv-field row mt-4">
<div class="col-12 col-md-6 col-lg-4 text-center">
<div class="serv-box">
<div class="icon">
<i class="bi bi-stopwatch-fill"></i>
</div>
<h4>60 seconds</h4>
<p>Professional grade weekly meal plan creation in under 60
seconds.</p>
</div>
</div>
<div class="col-12 col-md-6 col-lg-4 text-center">
<div class="serv-box">
<div class="icon">
<i class="bi bi-menu-up"></i>
```

```
</div>
<h4>3,000+ variations</h4>
<p>Such as smoothies, soups, main dishes and so much more.</p>
</div>
</div>
<div class="col-12 col-md-6 col-lg-4 text-center">
<div class="serv-box">
<div class="icon">
<i class="bi bi-cloud-arrow-down-fill"></i>
</div>
<h4>1-Click Download</h4>
<p>Easily access from any device &amp;amp; share meal plans from any
device.</p>
</div>
</div>
</div>
</div>
</div>
</section>
<script type="text/javascript" >
let nav=document.querySelector(".navigation-wrap");
window.onscroll = function(){
if(document.documentElement.scrollTop > 20){
nav.classList.add("scroll-on");
}else{
nav.classList.remove("scroll-on");
}
}
</script>
<section class="feature section-padding" id="trackfood">
<div class="container-fluid px-0 top-banner1">
<div class="container">
<div class="feature-heading" style="visibility: visible; animation-name: zoom">
<h2>KNOW THE NUTRIENTS PRESENT IN YOUR FOOD</h2>
```

```
</div>
<div class="serv-field row mt-4">
<div class="col-12 col-md-6 col-lg-4 text-center">
<div class="serv-box">
<div class="icon">
<i class="bi bi-camera-fill"></i>
</div>
<h1 style="font-size: xxx-large;
color: red;
font-family: 'Gill Sans', 'Gill Sans MT', Calibri, 'Trebuchet MS',
sans-serif;">01</h1>
<p>Click a Picture of Your Food</p>
</div>
</div>
<div class="col-12 col-md-6 col-lg-4 text-center">
<div class="serv-box">
<div class="icon">
<i class="bi bi-cloud-arrow-up-fill"></i>
</div>
<h1 style="font-size: xxx-large;
color: red;
font-family: 'Gill Sans', 'Gill Sans MT', Calibri, 'Trebuchet MS',
sans-serif;">02</h1>
<p>Upload the Picture</p>
</div>
</div>
<div class="col-12 col-md-6 col-lg-4 text-center">
<div class="serv-box">
<div class="icon">
<i class="bi bi-card-list"></i>
</div>
<h1 style="font-size: xxx-large;
color: red;
font-family: 'Gill Sans', 'Gill Sans MT', Calibri, 'Trebuchet MS',
```

```
sans-serif;">03</h1>
<p>Know the Nutrients Present</p>
</div>
</div>
</div>
<br>
<br>
<center>
{% if session['username'] %}
<a href="/uploader" class="btn btn-outline-primary" role="button" aria-pressed="true">
TRY NOW </a>
{% endif %}
</center>
</div>
</div>
</section>
<footer id="aboutus">
<div class="container">
<div class="row pb-4">
<div class="foot-info col-12 col-md-6 col-lg-9">
<a class="foot-logo" href="#home">
Nutri<span style="color:#40a008;" >fy</span>
</a>
<div class="mail">
<i class="fas fa-envelope"></i>
<a
href="mailto:nandita.sajeev.2019.cse@rajalakshmi.edu.in">nandita.sajeev.2019.cse@rajalakshmi.
edu.in</a>
</div>
<div class="mail">
<p>Contact Us: +91 12345 67890</p>
</div>
</div>
</div>
```

</footer>{% endblock %

## 7.2 Feature 2:

## Upload Image

This page allows users to upload images of food in order to get the nutrition information of it.

**UPLOADIMG.HTML**

```
{% extends "base.html" %}
{% block head %}
<title>Nutrify</title>
<link rel="stylesheet" type="text/css" href="{{url_for('static',
filename='css/indexstyle.css')}}">
<link href="https://cdn.jsdelivr.net/npm/bootstrap@5.2.2/dist/css/bootstrap.min.css"
rel="stylesheet"
integrity="sha384-Zenh87qX5JnK2Jl0vWa8Ck2rdkQ2Bzep5IDxbcnCeuOxjzrPF/et3URy9Bv1WTRi"
crossorigin="anonymous">
<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>
{% endblock %}
{% block body %}
<div class="row">
<div class="col-lg-6">
<div class="upload card">
<h1>Upload Food Image</h1>
<form class="uploaderform" action = "/uploader" method = "POST"
enctype = "multipart/form-data">
<div class="mb-3">
<label for="exampleInputname" class="form-label">Name</label>
<input placeholder="Enter bucket name" type="text" name="bucket"
class="form-control" id="exampleInputname" aria-describedby="emailHelp">
```

```
</div>
<div class="mb-3">
<label for="exampleInputfilename" class="form-label">Filename</label>
<input placeholder="Enter file name" name="filename" type="text"
class="form-control" id="exampleInputfilename">
</div>
<div class="mb-3">
<input class="form-control" type = "file" name = "file" />
</div>
<button type="submit" class="btn btn-primary">Submit</button>
</form>
</div>
</div>
<div class="col-lg-6">
<img src="/static/images/logo.png" alt="">
</div>
</div>
{% endblock %}
```

## 7.3 Feature 3:

**View History of Items**

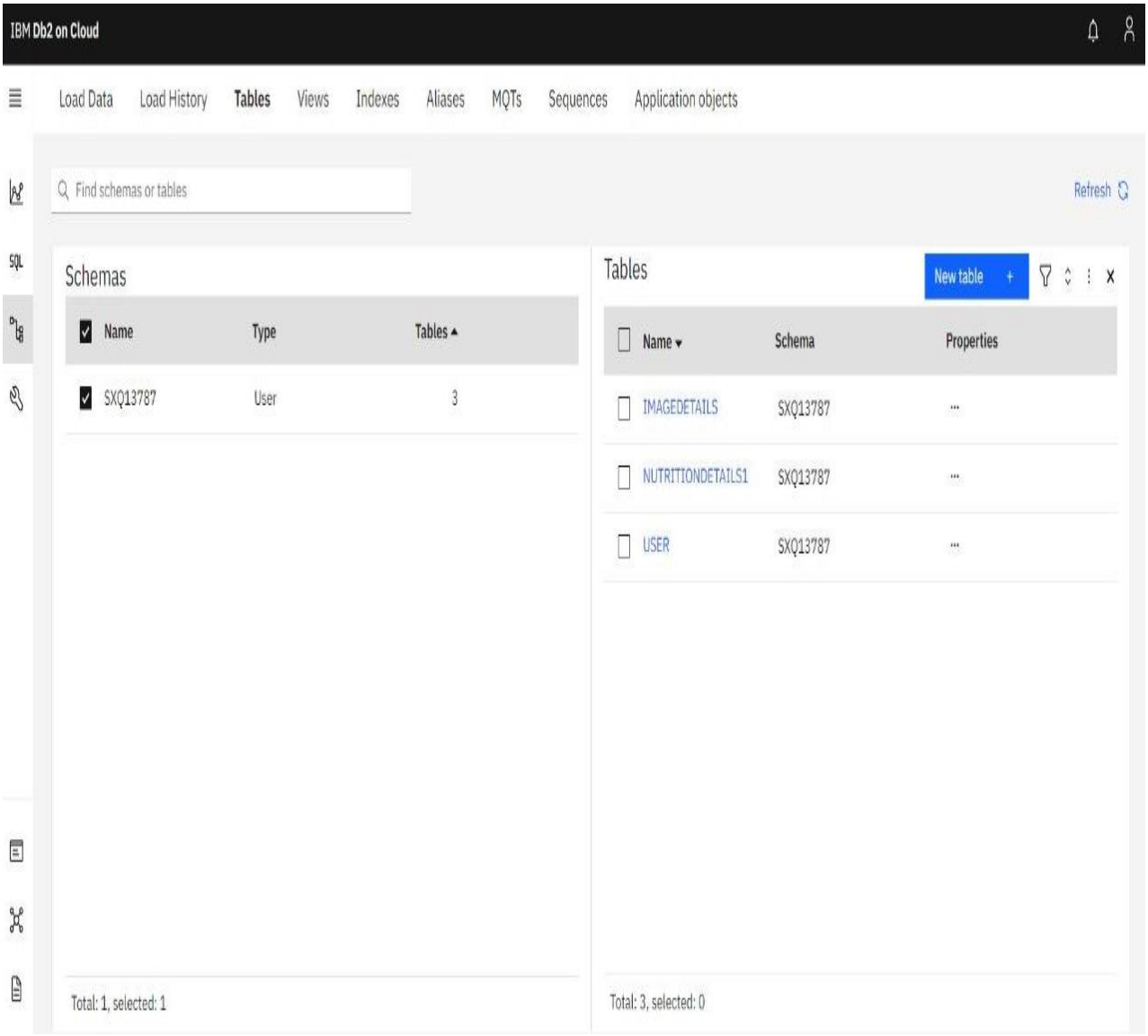
The users can view the nutritional information of all the past foods they have uploaded pictures of

**FOODINFO.HTML**

```
{% extends "base.html" %}
{% block head %}
<title>Nutrify</title>
<link rel="stylesheet" type="text/css" href="{{url_for('static',
filename='css/indexstyle.css')}}">
{% endblock %}
```

```
{% block body %}
<br>
<br>
<br>
<div class="foodinfomain">
<div class="foodinfo card">
<div class="row">
<div class="col-lg-6">
<img class="fo" src="{{files[8]}}" alt="">
</div>
<div class="col-lg-6">
<p>Food item: {{files[1]}}</p>
<hr>
<p>Calories: {{files[2]}} Kcal</p>
<hr>
<p>Carbs: {{files[3]}} Gm</p>
<hr>
<p>Fat: {{files[4]}} Gm</p>
<hr>
<p>Protein: {{files[5]}} Gm</p>
</div>
</div>
</div>
</div>
{% endblock %}
```

## 7.4 Database Schemas:



# CHAPTER 8
# TESTING

## 8.1 TEST CASES

| | **Test Scenarios** |
|---|---|
| **1** | Verify if the user is able to open and view the homepage |
| 2 | Verify if the user is able to interact with the elements in the homepage |
| 3 | Verify if the user is able to navigate to the other pages of the application from the homepage. |
| | **Upload Image Page Actions** |
| 1 | User is able to upload image |
| 2 | User is able to submit the image and obtain results |
| | **View History of Items Related Actions** |
| 1 | User is able to view all past uploaded images |
| 2 | User is able to see the nutritional breakdown of the previously uploaded images |
| | **User is able to log in and sign up** |
| 1 | User is able to create an account and log in |

## 8.2 User Acceptance Testing:

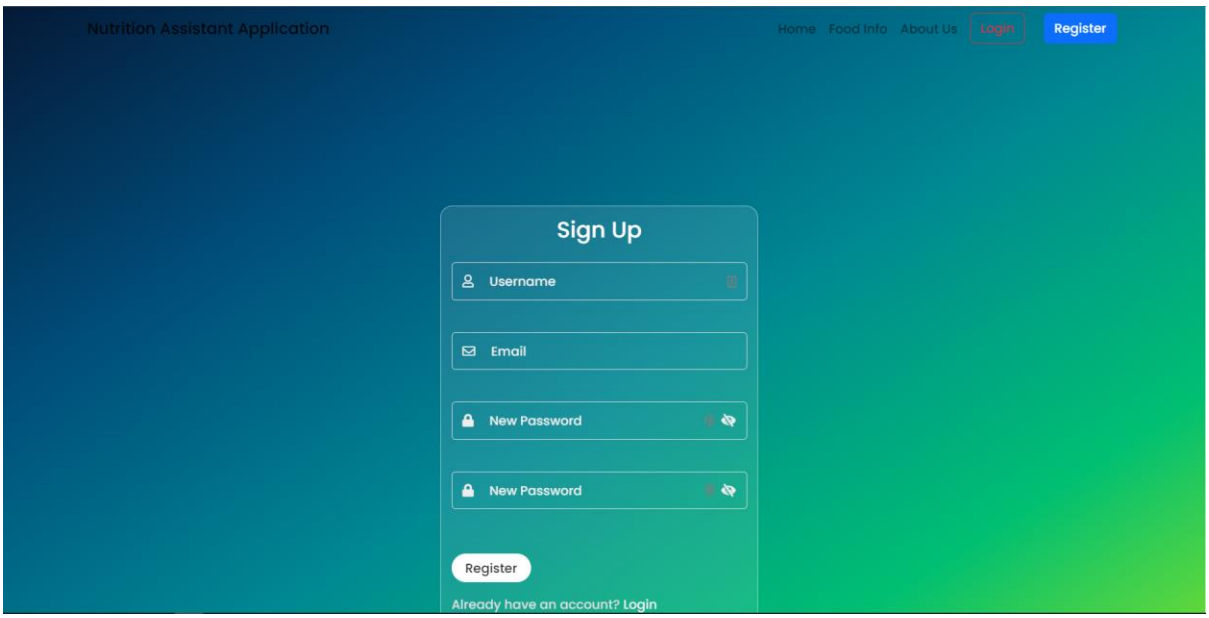
| Test Case ID | Feature Type | Component | Test Scenario | Pre-requisite | Steps to Execute | Test Data | Expected Result | Actual Result | Status | Comments | TC for automation | Bug ID | Executed By |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Homepage TC 01 | UI | Homepage | Verify if the user is able to open and view the homepage | None | 1. Click on URL and go to the homepage | URL Link | Homepage is viewable | Working as expected | Pass | - | N | - | Nandita S |
| Homepage TC 02 | Functional | Homepage | Verify if the user is able to interact with the elements in the homepage | Homepage is accesible | 1. Click on the various elements of the page and see if its working | Homepage | Elements Work | Working as expected | Pass | - | N | - | Nithish Kumar N |
| Homepage TC 03 | Functional | Homepage | Verify if the user is able to navigate to other pages from the homepage | Homepage is accesible | 1. Click on the various links of the page and see if its working | Homepage | We can navigate | Working as expected | Pass | - | N | - | Abuthahir |
| Upload Image TC 01 | UI | Upload Image Page | User is able to upload images | Page is accessible | 1. Click on upload image button and see if its working | Upload Image Page | We can upload | Working as expected | Pass | - | N | - | Parthiban |
| Upload Image TC 02 | Functional | Upload Image Page | User is able to submit images and get results | Page is accessible | 1. See if the uploaded images yield results | Upload Image Page | We can see results | Working as expected | Pass | - | N | - | Nandita S Abuthahir |
| View History TC 01 | UI | View History Page | User is able to view past uploaded images | Page is accessible | 1. See if the uploaded images are there | View History Page | We can see history | Working as expected | Pass | - | N | - | Abuthahir Nithish Kumar N |
| View History TC 02 | Functional | View History Page | User is able to the nutritional breakdown of the previously uploaded images | Page is accessible | 1. See if the nutritional breakdown is visible | View History Page | We can see the nutrients | Working as expected | Pass | - | N | - | Abuthahir Nithish Kumar N Nandita Sajeev |

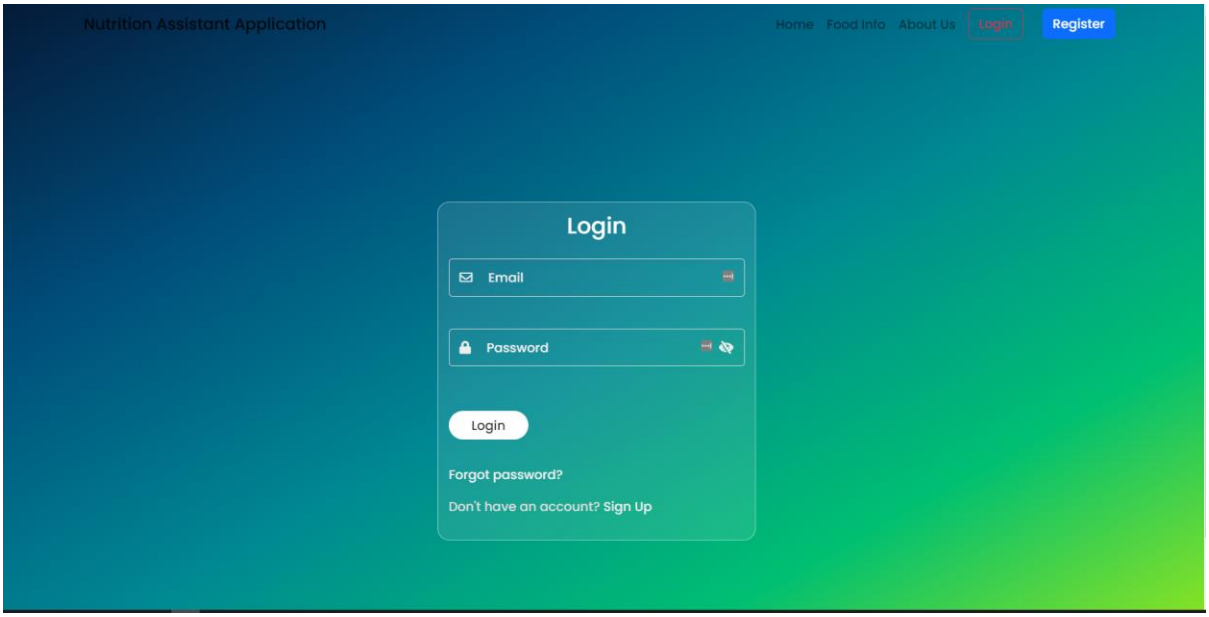
# CHAPTER 9
# RESULTS

## 9.1 Performance Metrics

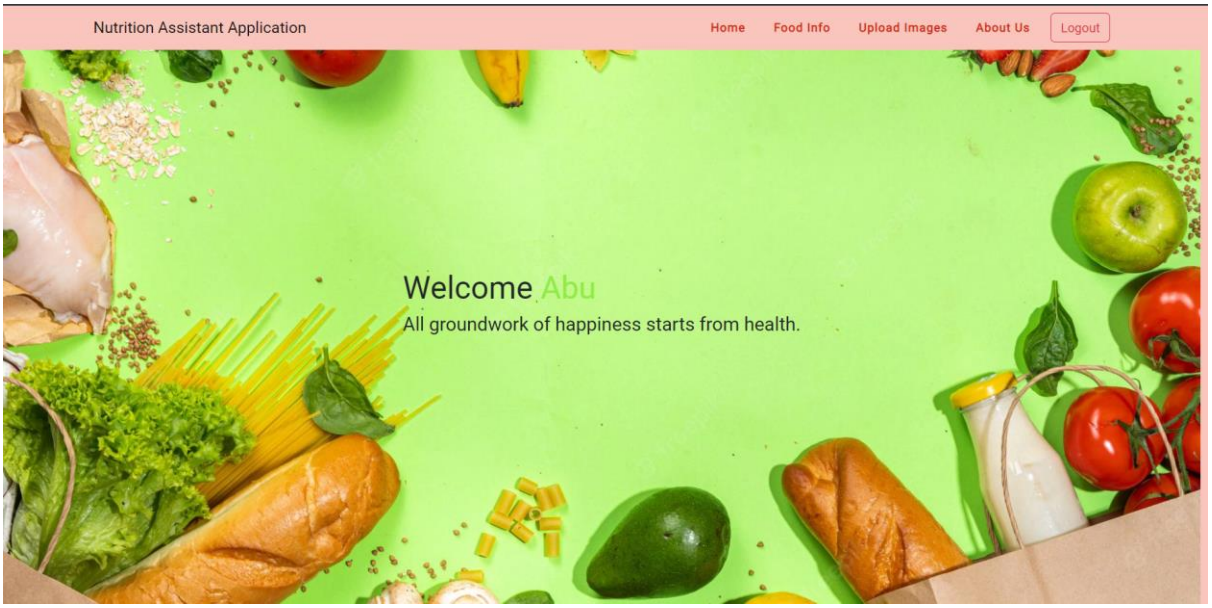
| S.No | Parameters | Values | Screenshots |
|---|---|---|---|
| 1 | Homepage | This page allows the user to get a glimpse of the app and allows them to navigate the |  |
| 2 | Upload Image Page | This page allows users to upload food images and get results |  |
| 3 | View History Page | User is able to view the past uploaded items |  |
| 4 | Login/SignUp | User can log in and sign up |  |

## Screenshots of UI:

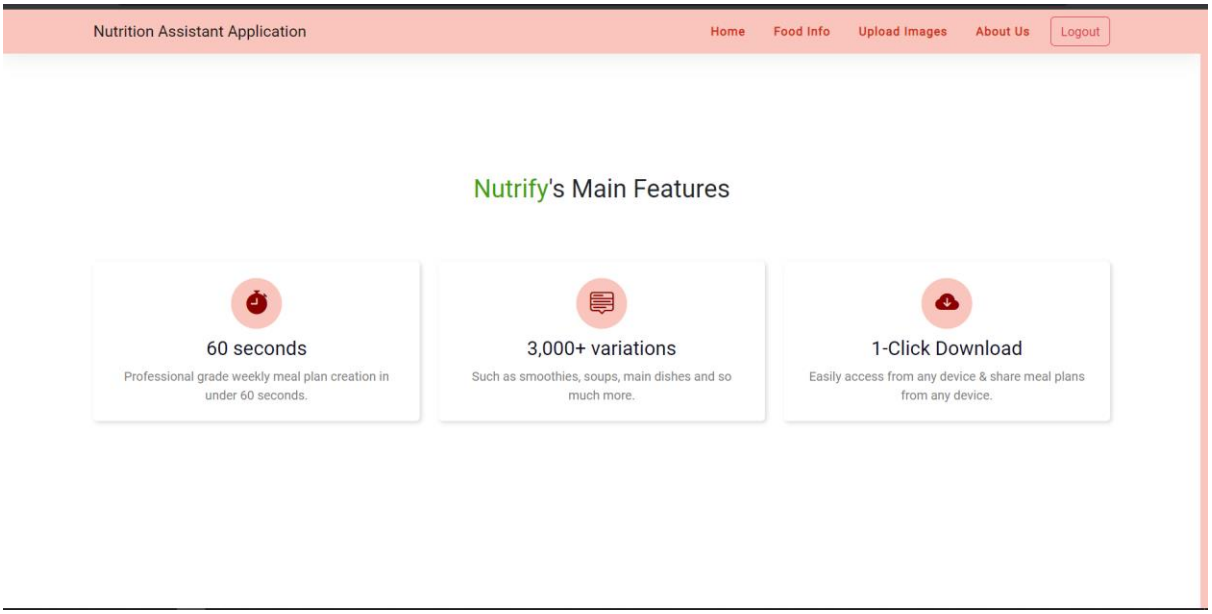


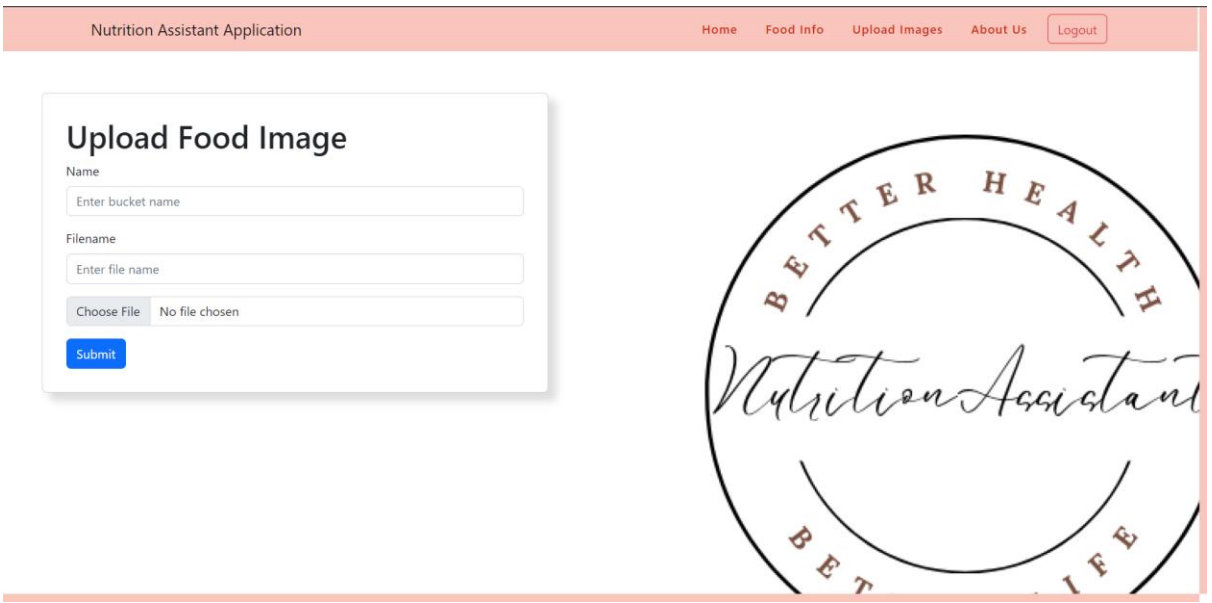


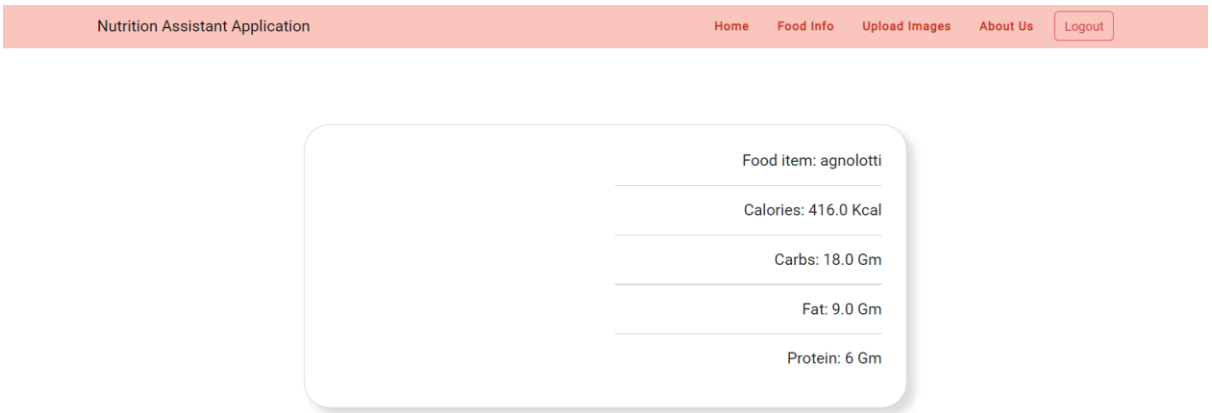

## Interface after registration:



Home page:


Nutrition Assistant Application
Home
Food Info
Upload Images
About Us
Logout
Nutrify's Main Features
60 seconds
Professional grade weekly meal plan creation in under 60 seconds.
3,000+ variations
Such as smoothies, soups, main dishes and so much more.
1-Click Download
Easily access from any device & share meal plans from any device.


Nutrition Assistant Application
Home
Food Info
Upload Images
About Us
Logout
Upload Food Image
Name
Enter bucket name
Filename
Enter file name
Choose File
No file chosen
Submit
BETTER HEALTH
Nutrition Assistant


Nutrition Assistant Application
Home
Food Info
Upload Images
About Us
Logout
Food item: agnolotti
Calories: 416.0 Kcal
Carbs: 18.0 Gm
Fat: 9.0 Gm
Protein: 6 Gm

## After uploading the image:

# CHAPTER-11

# ADVANTAGES AND DISADVANTAGES

## Advantages:

1. The user is now able to track his daily calorie intake.
2. He/she can now take effective measures to maintain a healthy body weight.
3. It delivers information on the nutritional value of food and how it should be maintained on a daily basis.

## Disadvantages:

1. It cannot be used without an Internet Connection.
2. Usage of 3rd party API may cause a time delay.

# CONCLUSION & FUTURE SCOPE

The nutrition assistant application using cloud computing is able to get images from the users and analyse them and show the nutritional breakdown of the food item. It is able to do this in an efficient and cost-effective way. This application allows people to get to know the nutrients of foods at any time which makes it more convenient for the users.This can be scaled to include APIs that have a larger variety of foods to have it cater to larger audiences of different backgrounds and ethnicities.

## FUTURE SCOPE

The application can be improved to cater to more people

### 1. ADDING GRAPHICAL DATA ON THE FOODS CONSUMES

Adding a pie chart or a breakdown of what nutritional components are being consumed can give more insight into the food habits of a user. This can help the user make changes and increase or decrease their consumption of a particular nutrient or food.

### 2. CREATING A PERSONALISED FOOD RECOMMENDATION SYSTEM

Based on the previously uploaded images we can provide recommendations for the kinds of foods to eat to have a balanced diet.

## REFERENCES

1. Djilani Kebaili, Eric Antoine Scuccimarra,Gaurav Singhal, Harris Heritier,Marcel Salathe, Sharada Prasanna Mohanty, Victor Boulanger (2016) “The Food Recognition Benchmark: Using Deep Learning to Recognize Food in Images”, International Conference On Identification And Knowledge On IOT.

2. Alisha Lalani, Md.Riyazudin, Mousmi Ajay Chaurasia, Salva Fathima, Syed Ibrahim Ibaad (2022) “Estimation of Nutrition content”.

3. “Quantity and Nutritional Information Using Image Processing”, International Journal Of Scientific And Engineering Research.

4. Bojia Qiu , Chenxi Huang , Kunhui Lin , Landu Jiang , Xue Liu (2022) “Deep Food: Food Image Analysis and Dietary Assessment via Deep Model”, International Journal Of Scientific And Engineering Research.

5. Hui Deng , Jianbo Wu , Xianghui Zeng , Ying Wang (2021) “A Comprehensive Survey of Image-Based Food Recognition and Volume Estimation Methods for Dietary Assessment”, International Conference on Journal Publication.

6. Lily Indriani Octovia; Nurul Ratna Mutu Manikam; Fiastuti Witjaksono; Krisadelfa Sutanto;Allya P. Koesoema; Soegijardjo Soegijoko;Yoke S. Irawan (2017) , “mHealth for Mother and Child Health Nutrition - A Review and Proposed Design for Indonesia Focus on Early Life Nutrition in Indonesia” .

7. Jitao Yang (2021), “Mother Care Precision Nutrition Service Based on Nutrigenetics and Nutrition Blood Test”.

8. Yuita Arum Sari; Luthfi Maulana; Yusuf Gladiesnyah Bihanda; Jaya Mahar Maligan; Nabila Nur’aini; Dhea Rahma Widyadhana (2020), “Leftovers Nutrition Prediction for Augmenting Smart Nutrition Box Prototype Feature Using Image Processing Approach and AFLE Algorithm”.

9. Manuel B. Garcia; Joel B. Mangaba; Celeste C. Tanchoco (2021), “Virtual Dietitian: A Nutrition Knowledge-Based System Using Forward Chaining Algorithm”.
10. Fang Sun (2019), “Computer Optimization of Food Nutrition Formula Based on the Consideration of Adaptive Genetic Algorithm”.
11. Maryam Hazman; Amira M. Idrees (2015),” A healthy nutrition expert system for children”.
12. İbrahim Berkan Aydilek (2017), “Approximate estimation of the nutritions of consumed food by deep learning”.
13. Asmabee Khan; Sachin Deshpande; Amiya K. Tripathy (2019), “Optimizing Nutrition using Machine Learning Algorithms-a Comparative Analysis”.
14. Niloofar Hezarjaribi; Sepideh Mazrouee; Hassan Ghasemzadeh (2018), “Speech2Health: A Mobile Framework for Monitoring Dietary Composition From Spoken Data”.
15. Poltak Sihombing; Muhammad Zarlis; Herriyance (2019), “Automatic Nutrition Detection System (ANDES) for Hydroponic Monitoring by using Micro controller and Smartphone Android”.

***https://ieeexplore.ieee.org/document/4782671***
***https://ieeexplore.ieee.org/document/8118575***

# APPENDIX

## Source Code

### home.html:

```
{% extends "base.html" %}
{% block head %}
<title>Nutrify</title>
<link rel="stylesheet" type="text/css" href="{{url_for('static',
filename='css/indexstyle.css')}}">
{% endblock %}
{% block body %}
<header>
<nav class="navbar navbar-expand-lg navigation-wrap">
<div class="container">
<a class="navbar-brand" href="/">Nutrition Assistant Application</a>
<button class="navbar-toggler" type="button" data-bs-toggle="collapse"
data-bs-target="#navbarNav"
aria-controls="navbarNav" aria-expanded="false" aria-label="Toggle
navigation">
<span class="navbar-toggler-icon"></span>
</button>
<div class="collapse navbar-collapse" id="navbarText">
<ul class="navbar-nav ms-auto mb-2 mb-lg-0">
<li class="nav-item">
<a class="nav-link" aria-current="page" href="/">Home</a>
</li>
<li class="nav-item">
<a class="nav-link" aria-current="page" href="/foodinfo">Food
Info</a>
</li>
```

```
{% if session['username'] %}
<li class="nav-item">
<a class="nav-link" aria-current="page" href="/uploader">Upload
Images</a>
</li>
{% endif %}
<li class="nav-item">
<a class="nav-link" aria-current="page" href="/#aboutus">About Us</a>
</li>
{% if session['username'] %}
<li>
<a class="btn btn-outline-danger ms-2" href="{{ url_for('logout')
}}">Logout</a>
</li>
{% else %}
<li>
<a class="btn btn-outline-danger ms-2" href="/login"
role="button">Login</a>
</li>
<li>
<a class="btn btn-primary ms-4" href="/signup"
role="button">Register</a>
</li>
{% endif %}
</ul>
</div>
</div>
</nav>
</header>
```

```
<section id="home">
<div class="test">
<div class="main">
</div>
<div class="textbox">
{% if session['username'] %}
<h1>Welcome <span style="color:#91eb48;">{{ session['username'] | title
}}</span></h1>
<!-- <a href="{{ url_for('logout') }}">Logout</a> -->
{% else %}
<h1>Welcome to <span style="color:#40a008;">Nutrify</span></h1>
{% endif %}
<h4>All groundwork of happiness starts from health.</h4>
</div>
</div>
</section>
<section class="feature section-padding" id="feature">
<div class="container-fluid px-0 top-banner1">
<div class="container">
<div class="feature-heading" style="visibility: visible; animation-name: zoom">
<h2><span style="color:#40a008;">Nutrify</span>'s Main Features</h2>
</div>
<div class="serv-field row mt-4">
<div class="col-12 col-md-6 col-lg-4 text-center">
<div class="serv-box">
<div class="icon">
<i class="bi bi-stopwatch-fill"></i>
</div>
<h4>60 seconds</h4>
```

```
<p>Professional grade weekly meal plan creation in under 60
seconds.</p>
</div>
</div>
<div class="col-12 col-md-6 col-lg-4 text-center">
<div class="serv-box">
<div class="icon">
<i class="bi bi-menu-up"></i>
</div>
<h4>3,000+ variations</h4>
<p>Such as smoothies, soups, main dishes and so much more.</p>
</div>
</div>
<div class="col-12 col-md-6 col-lg-4 text-center">
<div class="serv-box">
<div class="icon">
<i class="bi bi-cloud-arrow-down-fill"></i>
</div>
<h4>1-Click Download</h4>
<p>Easily access from any device &amp;amp; share meal plans from any
device.</p>
</div>
</div>
</div>
</div>
</div>
</section>
<script type="text/javascript" >
let nav=document.querySelector(".navigation-wrap");
```

```
window.onscroll = function(){
if(document.documentElement.scrollTop > 20){
nav.classList.add("scroll-on");
}else{
nav.classList.remove("scroll-on");
}
}
</script>
<section class="feature section-padding" id="trackfood">
<div class="container-fluid px-0 top-banner1">
<div class="container">
<div class="feature-heading" style="visibility: visible; animation-name: zoom">
<h2>KNOW THE NUTRIENTS PRESENT IN YOUR FOOD</h2>
</div>
<div class="serv-field row mt-4">
<div class="col-12 col-md-6 col-lg-4 text-center">
<div class="serv-box">
<div class="icon">
<i class="bi bi-camera-fill"></i>
</div>
<h1 style="font-size: xxx-large;
color: red;
font-family: 'Gill Sans', 'Gill Sans MT', Calibri, 'Trebuchet MS',
sans-serif;">01</h1>
<p>Click a Picture of Your Food</p>
</div>
</div>
<div class="col-12 col-md-6 col-lg-4 text-center">
<div class="serv-box">
```

```
<div class="icon">
<i class="bi bi-cloud-arrow-up-fill"></i>
</div>
<h1 style="font-size: xxx-large;
color: red;
font-family: 'Gill Sans', 'Gill Sans MT', Calibri, 'Trebuchet MS',
sans-serif;">02</h1>
<p>Upload the Picture</p>
</div>
</div>
<div class="col-12 col-md-6 col-lg-4 text-center">
<div class="serv-box">
<div class="icon">
<i class="bi bi-card-list"></i>
</div>
<h1 style="font-size: xxx-large;
color: red;
font-family: 'Gill Sans', 'Gill Sans MT', Calibri, 'Trebuchet MS',
sans-serif;">03</h1>
<p>Know the Nutrients Present</p>
</div>
</div>
</div>
<br>
<br>
<center>
{% if session['username'] %}
<a href="/uploader" class="btn btn-outline-primary" role="button" aria-pressed="true">
TRY NOW </a>
```

```
{% endif %}
</center>
</div>
</div>
</section>
<footer id="aboutus">
<div class="container">
<div class="row pb-4">
<div class="foot-info col-12 col-md-6 col-lg-9">
<a class="foot-logo" href="#home">
Nutri<span style="color:#40a008;" >fy</span>
</a>
<div class="mail">
<i class="fas fa-envelope"></i>
<a
href="mailto:nandita.sajeev.2019.cse@rajalakshmi.edu.in">nandita.sajeev.2019.cse@raja
lakshmi.edu.in</a
>
</div>
<div class="mail">
<p>Contact Us: +91 12345 67890</p>
</div>
</div>
</div>
</footer>
{% endblock %}
```

**index.html:**

```
{% extends "base.html" %}
```

```
{% block head %}
<title>Nutrify</title>
<link rel="stylesheet" type="text/css" href="{{url_for('static',
filename='css/indexstyle.css')}}">
<link href="https://cdn.jsdelivr.net/npm/bootstrap@5.2.2/dist/css/bootstrap.min.css"
rel="stylesheet" integrity="sha384-
Zenh87qX5JnK2Jl0vWa8Ck2rdkQ2Bzep5IDxbcnCeuOxjzrPF/et3URy9Bv1WTRi"
crossorigin="anonymous">
<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>
{% endblock %}
{% block body %}
{% for row in files %}
<div class="piccard card">
<img class="pic" src="{{row[0]}}">
<a href="/nutritioninfo/{{row[1]}}"><button style="width: 30%;" type="submit"
class="btn btn-outline-primary">Learn More..</button>
</a>
</div>
{% endfor %}
{% endblock %}
```

**upload.html:**

```
{% extends "base.html" %}
{% block head %}
<title>Nutrify</title>
```

```
<link rel="stylesheet" type="text/css" href="{{url_for('static',
filename='css/indexstyle.css')}}">
<link href="https://cdn.jsdelivr.net/npm/bootstrap@5.2.2/dist/css/bootstrap.min.css"
rel="stylesheet" integrity="sha384-
Zenh87qX5JnK2Jl0vWa8Ck2rdkQ2Bzep5IDxbcnCeuOxjzrPF/et3URy9Bv1WTRi"
crossorigin="anonymous">
<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>
{% endblock %}
{% block body %}
<div class="row">
<div class="col-lg-6">
<div class="upload card">
<h1>Upload Food Image</h1>
<form class="uploaderform" action = "/uploader" method = "POST"
enctype = "multipart/form-data">
<div class="mb-3">
<label for="exampleInputname" class="form-label">Name</label>
<input placeholder="Enter bucket name" type="text" name="bucket"
class="form-control" id="exampleInputname" aria-describedby="emailHelp">
</div>
<div class="mb-3">
<label for="exampleInputfilename" class="form-label">Filename</label>
<input placeholder="Enter file name" name="filename" type="text"
class="form-control" id="exampleInputfilename">
</div>
```

```
<ul class=flashes>
{% for message in messages %}
<li>{{ message }}</li>
{% endfor %}
</ul>
</div>
{% endif %}
{% endwith %}
<header>
<nav class="navbar navbar-expand-lg navigation-wrap">
<div class="container">
<a class="navbar-brand" href="/">Nutrition Assistant Application</a>
<button class="navbar-toggler" type="button" data-bs-toggle="collapse"
data-bs-target="#navbarNav"
aria-controls="navbarNav" aria-expanded="false" aria-label="Toggle
navigation">
<span class="navbar-toggler-icon"></span>
</button>
<div class="collapse navbar-collapse" id="navbarText">
<ul class="navbar-nav ms-auto mb-2 mb-lg-0">
<li class="nav-item">
<a class="nav-link" aria-current="page" href="/">Home</a>
</li>
<li class="nav-item">
<a class="nav-link" aria-current="page" href="/foodinfo">Food
Info</a>
</li>
{% if session['username'] %}
<li class="nav-item">
```

```
<a class="nav-link" aria-current="page" href="/uploader">Upload
Images</a>
</li>
{% endif %}
<li class="nav-item">
<a class="nav-link" aria-current="page" href="/#aboutus">About Us</a>
</li>
{% if session['username'] %}
<li>
<a class="btn btn-outline-danger ms-2" href="{{ url_for('logout')
}}">Logout</a>
</li>
{% else %}
<li>
<a class="btn btn-outline-danger ms-2" href="/login"
role="button">Login</a>
</li>
<li>
<a class="btn btn-primary ms-4" href="/signup"
role="button">Register</a>
</li>
{% endif %}
</ul>
</div>
</div>
</nav>
</header>
{% block body %} {% endblock %}
</body>
```

```
</html>
```

**delete.html:**

```
<html>
<body>
<a href="/">HOME</a>
<a href="/uploader">Upload </a>
<a href="/deletefile">Delete </a>
<br><hr>
<h1>IBM Object Storage</h1>
<div>
<form action = "/deletefile" method = "POST" >
<input type = "text" placeholder="Enter bucket name" name = "bucket" />
<br>
<br>
<input type = "text" placeholder="Enter file name" name = "filename" />
<br>
<br>
<input type = "submit"/>
</form>
</div>
</body>
</html>
```

**_render_field:**

```
{% macro render_error_field(field) %}
<ul>
{% for error in field.errors %}
<li class="field_error">{{ error }}</li>
```

```
{% endfor %}
</ul>
{% endmacro %}
```

**indexstyle.css:**

```
body, html {
height: 100%;
}
a,
a:hover{
text-decoration: none;
}
a:hover{
color: #FF0000;
}
html{
scroll-behavior: smooth;
}
body{
font-family: 'Roboto', sans-serif;
font-size: 100%;
font-weight: 400;
}
::-webkit-scrollbar{
width: 0.625rem;
}
::-webkit-scrollbar-track{
background: #f9c5bd;
}
```

```
::-webkit-scrollbar-thumb{
background: #f9c5bd;
}
.navigation-wrap{
background-color:#f9c5bd;
position: fixed;
width: 100%;
left:0;
z-index: 1000;
-webkit-transition:all 0.3s ease-out;
transition: all 0.3s ease-out;
}
.navigation-wrap .nav-item{
padding: 0 0.625rem;
transition: all 200ms linear;
}
.navbar-toggler:focus{
outline: unset;
border: unset;
box-shadow: none;
}
.nav-item .nav-link{
font-size: 0.9375rem;
font-weight: 600;
text-transform: capitalize;
color: #D12B10;
letter-spacing: 1px;
} .
nav-item .
```

```
nav-link a:hover{
color: #FF0000;
}
.navigation-wrap.scroll-on{
position: fixed;
top:0;
left: 0;
width: 100%;
background: #f9c5bd;
box-shadow: 0 0.125rem 1.75rem 0 rgb(0,0,0,0.09);
transition: all .15s ease-in-0ut 0s;
}
#home{
height: 100%;
}
.test{
height: 100%;
position: relative;
}
.top-banner{
width: 80%;
padding: 10rem 0 7rem;
}
.main{
background-image: url('/static/images/home.webp');
height: 100%;
width: 100%;
font-size: 70px;
background-position: center;
```

```
background-repeat: no-repeat;
background-size: cover;
}
.textbox{
position: absolute;
top: 50%;
left: 50%;
transform: translate(-50%, -50%);
}
.upload{
padding: 4%;
margin:7%;
box-shadow: 10px 10px 10px #ddd;
}
.piccard{
width: 30%;
padding: 2%;
border-radius: 7rem;
}
.pic{ margin-bottom: 7%;
}
.foodinfomain{
padding: 3%;
}
.foodinfo{
text-align: right;
position: absolute;
top: 20%;
left: 25%;
```

```
padding: 2%;
width:50%;
border-radius: 2rem;
box-shadow: 10px 10px 10px #ddd;
font-size: 20px;
} .
fo{
width: 130%;
}
.top-banner h1{
font-size: 48px;
}
.top-banner1{
width: 100%;
padding: 9.875rem 0 7.375rem;
}
.feature-heading{
text-align: center;
}
.serv-field {
padding-top: 50px;
} .
serv-field .
serv-box {
padding: 20px;
margin-bottom: 20px;
border-radius: 5px;
border: 1px solid transparent;
box-shadow: 2px 2px 5px rgba(136, 136, 136, 0.3);
```

```
transition: 0.4s ease
} .
serv-field .
serv-box:hover {
box-shadow: rgba(0, 0, 0, 0.4) 0px 30px 90px;
border-color:darkred;
} .
serv-field .
serv-box .
icon {
position: relative;
width: 65px;
height: 65px;
display: flex;
align-items: center;
justify-content: center;
margin: 0 auto 10px;
color:darkred;
border-radius: 50%;
} .
serv-field .serv-box .
icon:before {
content: '';
position: absolute;
top: 50%;
left: 50%;
transform: translate(-50%,-50%) scale(1);
width: 100%;
height: 100%;
```

```
border-radius: 50%;
background-color: #f9c5bd;
z-index: -1;
} .
serv-field .
serv-box .
icon i
{
font-size: 30px;
} .
serv-field .
serv-box h4 {
font-size: 25px;
color: #090719;
} .
serv-field .
serv-box p
{
color: #787878;
margin-bottom: 0px;
}
footer {
background-color: #222;
color: #fff;
padding-top: 50px;
} footer a
{
color: #fff;
transition: color 0.4s ease;
```

```
text-decoration: none;
} footer a:hover{
color: #D12B10;
} footer ul {
padding-left: 0;
} footer .
foot-logo {
display: inline-block;
font-size: 35px;
font-weight: 700;
margin-bottom: 15px;
} footer .
foot-info >
div { display: flex;
margin-bottom: 15px;
} footer .
foot-info > div i
{
width: 35px;
font-size: 18px;
}
```

**login_style.css:**

```
* {
margin: 0;
padding: 0;
box-sizing: border-box;
font-family: 'Poppins', sans-serif;
}
```

```
body {
background-image: linear-gradient(to right bottom, #051937, #004d7a, #008793, #00bf72,
#a8eb12);
background-repeat: no-repeat;
height: auto;
}
.wrapper {
max-width: 500px;
margin-top: 15%;
margin-left: 32%;
margin-bottom: 13.5%;
}
.wrapper .card {
max-width: 400px;
min-height: 380px;
margin: 30px;
background: rgba(255, 255, 255, 0.1);
overflow: hidden;
backdrop-filter: blur(10px);
border: 1px solid rgba(255, 255, 255, 0.5);
border-radius: 15px;
cursor: pointer;
padding: 0.8rem;
}
.wrapper .card a {
text-decoration: none;
color: #eee;
}
.wrapper .card a:hover {
```

```
color: #fff;
}
.wrapper .card .input-field {
border: 1px solid #ddd;
border-radius: 5px;
color: #eee;
padding: 0.3rem;
}
.wrapper .card .input-field input {
background-color: inherit;
}
.wrapper .card .input-field input.form-control,
.wrapper .card .input-field input.form-control:focus {
border: none;
outline: none;
box-shadow: none;
color: #eee;
}
.wrapper .card .input-field button.btn {
color: #eee;
padding: 0rem;
padding-right: 0.5rem;
}
.wrapper .card .input-field button.btn:hover {
color: #fff;
}
.wrapper .card .input-field button.btn:focus {
border: none;
outline: none;
```

```
box-shadow: none;
}
.wrapper .card .input-field input::placeholder {
color: #eee;
}
.wrapper .card .option {
display: block;
position: relative;
padding-left: 25px;
cursor: pointer;
user-select: none
}
.wrapper .card .option span.text-light-white:hover {
color: #fff;
}
.wrapper .card .option input {
position: absolute;
opacity: 0;
cursor: pointer;
height: 0;
width: 0
}
.checkmark {
position: absolute;
top: 3px;
left: 0;
height: 18px;
width: 18px;
background-color: #fff;
```

```
border-radius: 2px
}
.wrapper .card .btn.btn-primary {
border-radius: 20px;
width: 100px;
background-color: #fff;
color: #333;
border: none;
}
.wrapper .card .btn.btn-primary:hover {
color: #fff;
background: #333;
}
.wrapper .card .btn.btn-primary:focus {
border: none;
box-shadow: none;
}
.wrapper .card .text-light-white {
color: #ddd;
}
.wrapper .card .line span.connect {
position: absolute;
top: -12px;
left: 33%;
color: #000;
padding: 0 0.3rem;
z-index: 100;
border-radius: 2px;
background-color: #fff;
```

```
}
.wrapper .card .connections a img {
width: 40px;
height: 40px;
border-radius: 50%;
object-fit: cover;
}
```

**navbar.css:**

```
a,
a:hover{
text-decoration: none;
}
a:hover{
color: #FF0000;
}
html{
scroll-behavior: smooth;
}
body{
font-family: 'Roboto', sans-serif;
font-size: 100%;
font-weight: 400;
}
```

**reset_style.css:**

```
* {
margin: 0;
padding: 0;
box-sizing: border-box;
font-family: 'Poppins', sans-serif;
```

```
}
body {
background-image: linear-gradient(to right bottom, #051937, #004d7a, #008793, #00bf72,
#a8eb12);
background-repeat: no-repeat;
height: auto;
}
.wrapper {
max-width: 500px;
margin-top: 17%;
margin-left: 32%;
margin-bottom: 24%;
}
.wrapper .card {
max-width: 400px;
min-height: 200px;
margin: 30px;
background: rgba(255, 255, 255, 0.1);
overflow: hidden;
backdrop-filter: blur(10px);
border: 1px solid rgba(255, 255, 255, 0.5);
border-radius: 15px;
cursor: pointer;
padding: 0.8rem;
}
.wrapper .card a {
text-decoration: none;
color: #eee;
}
```

```
.wrapper .card a:hover {
color: #fff;
}
.wrapper .card .input-field {
border: 1px solid #ddd;
border-radius: 5px;
color: #eee;
padding: 0.3rem;
}
.wrapper .card .input-field input {
background-color: inherit;
}
.wrapper .card .input-field input.form-control,
.wrapper .card .input-field input.form-control:focus {
border: none;
outline: none;
box-shadow: none;
color: #eee;
}
.wrapper .card .input-field button.btn {
color: #eee;
padding: 0rem;
padding-right: 0.5rem;
}
.wrapper .card .input-field button.btn:hover {
color: #fff;
}
.wrapper .card .input-field button.btn:focus {
border: none;
```

```
outline: none;
box-shadow: none;
}
.wrapper .card .input-field input::placeholder {
color: #eee;
}
.wrapper .card .option {
display: block;
position: relative;
padding-left: 25px;
cursor: pointer;
user-select: none
}
.wrapper .card .option span.text-light-white:hover {
color: #fff;
}
.wrapper .card .option input {
position: absolute;
opacity: 0;
cursor: pointer;
height: 0;
width: 0
}
.checkmark {
position: absolute;
top: 3px;
left: 0;
height: 18px;
width: 18px;
```

```
background-color: #fff;
border-radius: 2px
}
.wrapper .card .btn.btn-primary {
border-radius: 20px;
width: 100px;
background-color: #fff;
color: #333;
border: none;
}
.wrapper .card .btn.btn-primary:hover {
color: #fff;
background: #333;
}
.wrapper .card .btn.btn-primary:focus {
border: none;
box-shadow: none;
}
.wrapper .card .text-light-white {
color: #ddd;
}
.wrapper .card .line span.connect {
position: absolute;
top: -12px;
left: 33%;
color: #000;
padding: 0 0.3rem;
z-index: 100;
border-radius: 2px;
```

```
background-color: #fff;
}
.wrapper .card .connections a img {
width: 40px;
height: 40px;
border-radius: 50%;
object-fit: cover;
}
.nav-link{
font-family: 'Bungee Spice', Georgia;
font-size: 20px;
padding-top: 29px;
color: rgb(127,255,0);
}
.nav-link:hover{
color: #00FF00!important;
}
```

**GITHUB** - **IBM-Project-22953-1659862147**

**PROJECT DEMO LINK** :

**https://drive.google.com/file/d/1XoV6zSsK1NvhXQNb9wirV5trCYLBySKp/view?usp=drivesdk**