NUTRITION ASSISTANT APPLICATION

A PROJECT REPORT

SUBMITTED BY

TEAM ID: PNT2022TMID00787

TEAM LEADER: KARTHIKEYAN B

TEAM MEMBER: KAVUSHICK P

TEAM MEMBER: KEETH ALOCIOUS K

TEAM MEMBER: MANIGANDAN L

In partial fulfillment for the award of the degree of

BACHELOR OF ENGINEERING

In

COMPUTER SCIENCE AND ENGINEERING

PANIMALAR ENGINEERING COLLEGE

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

NOVEMBER 2022

TABLE OF CONTENTS

1. INTRODUCTION

1.1 Project Overview

A key challenge in human nutrition is the assessment of usual food intake. This is of particular interest given recent proposals of eHealth personalized interventions. The adoption of mobile phones has created an opportunity for assessing and improving nutrient intake as they can be used for digitizing dietary assessments and providing feedback. It can be very helpful and improve eating habits to develop app-based nutrient dashboard systems that can evaluate real-time photographs of meals and assess them for nutritional content. The health tracking platform must, like any other nutrition app, have a specific capability set as well as a number of fundamental elements that assist users in bettering their physical condition and set it apart from other apps currently on the market. Diet services can provide more than just calorie counting, food intake monitoring, and physical activity tracking. In addition, it offers food diaries, diet tracking, a health activity tracker, and diet plans for pregnancy, bodybuilding, and veganism. Even if the main goal was to design an app for a diet plan with proper nutrition, the platform must be adaptable to future changes and the addition of new features. One of a person's most fundamental needs is food. Proper food consumption is crucial for human health and fitness and is frequently seen as much more than a means of survival. The four different food groups or the quantity we eat have a significant impact on our health. There are many fields, including. The study of healthy food consumption is done in the fields of sociology, psychology, nutrition science, and medicine. Lack of self-control, unhealthy habits, and a hectic lifestyle all have an adverse impact on food choices. However, excessively unhealthy lifestyles and poor eating practices, such as consuming more high-energy and high-fat foods, Overweight is a problem for more than 1.9 billion persons over the age of 18, according to the World Health Organization (WHO).

1.2 Purpose

They can help you to improve your diet so you can eat healthier. Using a nutrition app will inherently help you eat a better and more optimized diet. Not only do these apps typically have calorie counters in them, but they can show you how many macro and micronutrients you are getting into your diet.Without having an app that tracks your food intake, you're going to find it increasingly difficult to track this manually. These apps will generally make recommendations on what foods you should be looking to get into your diet.ne of the most important things that you can do when you are looking to improve your diet choices and your weight loss efforts is to track your progress. This is something that you will either need to do manually or in an automated fashion using an app like this.A nutrition app that can track and monitor your progress is worth using. Without progression tracking, you are going at things without a measuring stick. Having an app that tracks and monitors your progress is the only way to identify what you are doing right and what you are doing wrong. It also shows you what you may need to change to improve. Being able to check your progress at a glance can do wonders for your efforts.Not only will you have something providing you with additional motivation, but it can also put things into perspective when you aren't necessarily visualizing any results from your efforts.

2. LITERATURE SURVEY

2.1 Existing problem

A. Picture-to-Amount (PITA): Predicting Relative Ingredient Amounts from Food Images

PITA, a deep learning architecture for predicting the relative amount of each ingredient in a given food image, was proposed. From a domain-driven Wasserstein loss from image-to-recipe cross-modal retrieval system, they first learn an image embedding representation. Then, using an amount prediction network, they detect ingredients from ingredient detection networks and predict amounts based on the results of the ingredient detection networks. As part of the evaluation metric and loss function, ingredient substitution groups are created to facilitate functional ingredient substitutions.

B. Smart Log system that performs automated nutrition monitoring and meal prediction.

The smart sensor board consisting of Piezo Electric sensors is used for nutrition quantification. The nutrient data acquisition is done using Optical Character Recognition and by linking open source Application Program Interfaces (APIs) through barcodes. The meal prediction is done by collecting nutritional value of the leftover food along with the user's feedback on the type of food that is desired. The SR8 database available through the US Department of Agriculture website is also analyzed using their API which provides a food report of associated nutrient values for a particular food item and a nutrient report which gives an extensive list of food and their nutrient values for a selected amount of nutrients. The results have been analyzed by creating an AttributeRelation File Format which inputs the Waikato Environment for Knowledge Analysis (WEKA) tool which builds a better

prediction model and is observed that the Bayesian classifiers provided better results.

C. Using Deep Learning for Food and Beverage Image Recognition

NutriNet, a novel deep learning architecture, and a pixel-level classification solution for images of fake food were created by the researchers. NutriNet was the first to recognise beverage images after being trained on a larger food image dataset with more food classes than previous works. Their work on fake-food image recognition includes the development of the first automatic system for recognizing images of fake food, and the visual similarity between fake and real food makes it useful for both fake-food experiments and real food recognition.

2.2 References

1. Nutrition-Reference-Center-https://play.google.com/store/apps/details?id=com.ebsco.nurc&hl=en_IN&gl=US
2. Healtify- Me- app- https://www.healthifyme.com/us/
3. Alpha- coach -Evolve-https://www.alphacoach.app/evolve
4. Global Status Report on Noncommunicable Diseases 2010. Geneva, Switzerland: World Health Organization; 2011. URL: http://www.who.int/nmh/publications/ncd_report_full_en.pdf [accessed 2016-06-28] [WebCite Cache ID 6ibtFrqbc]
5. Simko MD, Cowell C, Gilbride JA. Nutrition Assessment: A Comprehensive Guide for Planning Intervention. 2nd edition. Gaithersburg, MD: Aspen Publishers; 1995.
6. Benedik E, Koroušić Seljak B, Simčič M, Rogelj I, Bratanič B, Ding EL, et al. Comparison of paper- and Web-based dietary records: A pilot study. Ann

Nutr Metab 2014;64(2):156-166. [doi: 10.1159/000363336] [Medline: 25116257]

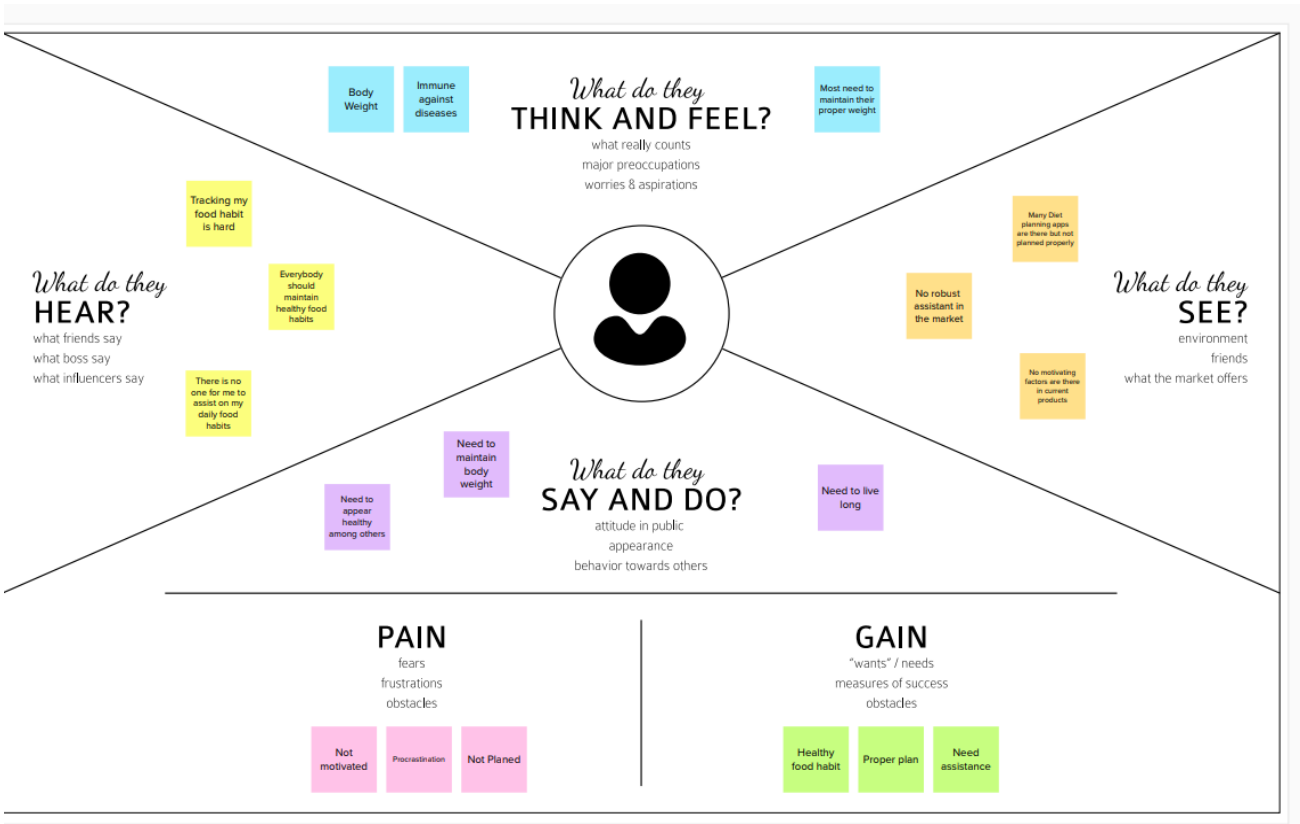
7. Thompson FE, Subar AF, Loria CM, Reedy JL, Baranowski T. Need for technological innovation in dietary assessment. JAm Diet Assoc 2010 Jan;110(1):48-51 [FREE Full text] [doi: 10.1016/j.jada.2009.10.008] [Medline: 20102826]
8. Klasnja P, Pratt W. Healthcare in the pocket: Mapping the space of mobile-phone health interventions. J Biomed Inform 2012 Feb;45(1):184-198 [FREE Full text] [doi: 10.1016/j.jbi.2011.08.017] [Medline: 21925288]
9. Probst Y, Nguyen DT, Tran MK, Li W. Dietary assessment on a mobile phone using image processing and pattern recognition techniques: Algorithm design and system prototyping. Nutrients 2015 Aug;7(8):6128-6138 [FREE Full text] [doi:10.3390/nu7085274] [Medline: 26225994]
10. Zhang W, Yu Q, Siddiquie B, Divakaran A, Sawhney H. "Snap-n-Eat": Food recognition and nutrition estimation on a smartphone. J Diabetes Sci Technol 2015 May;9(3):525-533 [FREE Full text] [doi: 10.1177/1932296815582222] [Medline:25901024]
11. West JH, Hall PC, Arredondo V, Berrett B, Guerra B, Farrell J. Health behavior theories in diet apps. J Consum HealthInternet 2013 Jan;17(1):10-24. [doi: 10.1080/15398285.2013.756343]
12. Weight Watchers UK. 2015. Weight loss plans URL: https://www.weightwatchers.com/uk/weight-loss-plans [accessed 2016-06-28] [WebCite Cache ID 6ibtx4G3x]
13. Fitbit. 2016. URL: http://www.fitbit.com/uk [accessed 2016-04-05] [WebCite Cache ID 6gXe73WuA]

2.3 Problem Statement Definition

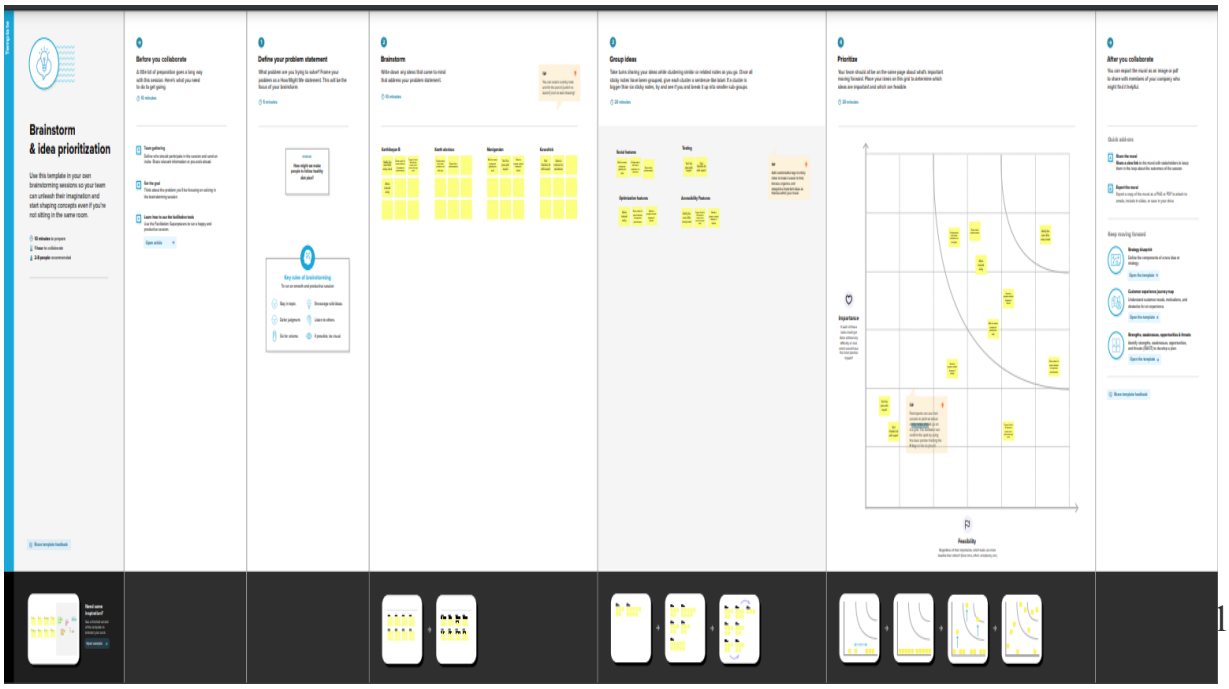
I am an unhealthy person, I'm trying to maintain healthy food habits but It is hard to form or maintain any healthy diet habits because I don't have any proper diet plan and have no one to assist which makes me feel sad.Obesity rates are rising rapidly due to a lack of knowledge about healthy eating habits, and this reflects the risks to people's health. The most basic method for avoiding obesity is for people to control their daily calorie intake by eating healthier foods. However, even though food packaging includes nutrition (and calorie) labels, it is still inconvenient for people to refer to App-based nutrient dashboard systems that can analyze real-time images of a meal and analyze it for nutritional content, which can be very useful and improves dietary habits, and thus helps in maintaining a healthy lifestyle. The primary goal of this project is to create a web app that estimates food attributes such as ingredients and nutritional value automatically.

3. IDEATION & PROPOSED SOLUTION

3.1 Empathy Map Canvas



3.2 Ideation & Brainstorming

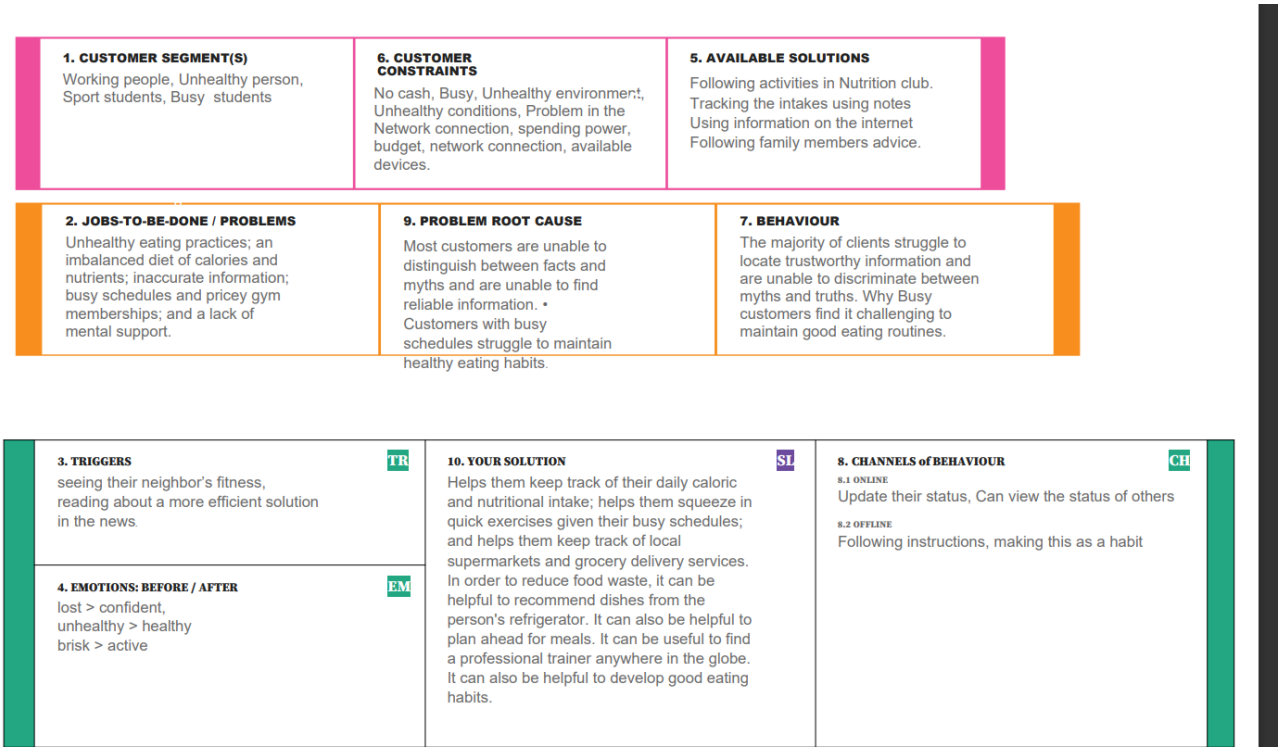


3.3 Proposed Solution

S. No.	Parameter	Description
1.	Problem Statement	Obesity rates are rising alarmingly quickly as a result of people's lack of knowledge about appropriate eating practises, which reflects the hazards to their health. The simplest way to prevent obesity is for people to limit their daily calorie consumption by eating healthier meals
2.	Idea/Solution description	This project seeks to create a web application that, using the classification of the supplied food image, automatically predicts food features like ingredients and nutritional value.
3.	Novelty/Uniqueness	The suggested approach employs a cutting-edge detection model to accurately and instantly identify a food.
4.	Social Impact / Customer Satisfaction	This ensures the safety of all humans and promotes a safe and healthy food habits

5.	Business Model (Revenue Model)	Subscription Based - A consumer who wants access to a good or service must pay a recurring fee at regular intervals, according to the subscription business model.
6.	Scalability of the Solution	Since it is web based application, it can be accessed from anywhere from any user devices

3.4 Problem Solution fit



4. REQUIREMENT ANALYSIS

4.1 Functional requirement

FR No.	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)
FR-1	User Registration	Registration through Form
FR-2	User Confirmation	Confirmation via Email Confirmation via OTP
FR-3	Interoperability	Implemented using web technology so it can be accessed from any platform
FR-4	NutritionPrediction Accuracy	Each prediction should need to be accurate
FR-5	Diet Plan	Each diet plan should be clear and easy to follow

4.2 Non - Functional requirement

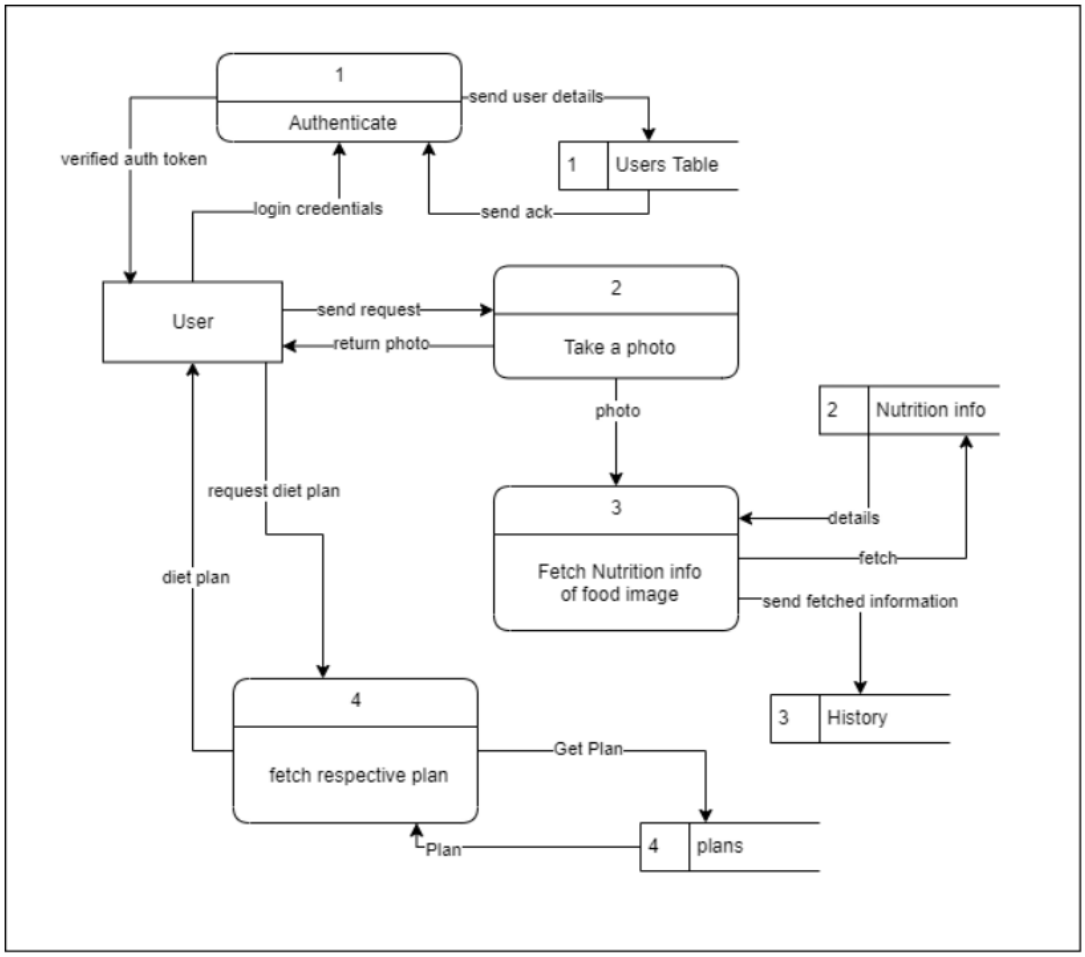
FR No.	Non-Functional Requirement	Description
NFR-1	Usability	This web page (UI) are designed to offer a comprehensive overview of diet plan and allow easy upload of photos.
NFR-2	Security	The basic authentication system can prevent unauthorized access.
NFR-3	Reliability	This webpage will be consistent and reliable to the users and helps the user to use in effective, efficient and reliable manner..
NFR-4	Performance	The image analysis model is trained well and it will also performed well in production environment.
NFR-5	Availability	The app can meet user's requirements and it is also helps to provide necessary information to the user's request
NFR-6	Scalability	The app is hosted as micro service so it can scale easily to million users

5. PROJECT DESIGN

5.1 Data Flow Diagrams

A Data Flow Diagram (DFD) is a traditional visual representation of the information flows within a system. A neat and clear DFD can depict the right amount of the system requirement graphically. It shows how data enters and leaves the system, what changes the information, and where data is stored**.**

Data Flow Diagrams

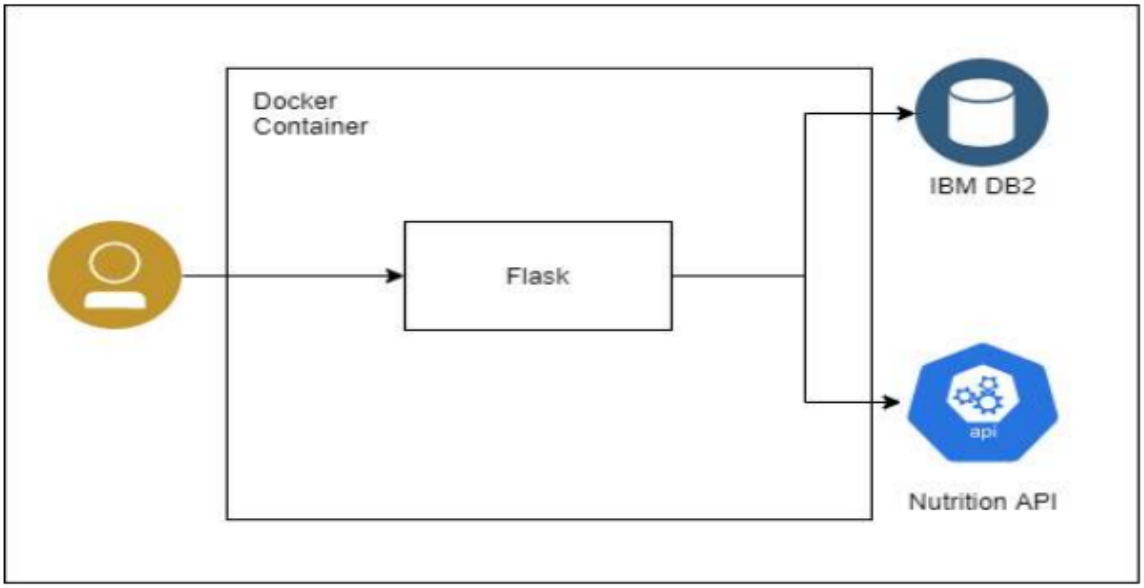


5.2 Solution & Technical Architecture

Solution architecture is a complex process – with many sub-processes – that bridges the gap between business problems and technology solutions. Its goals are to:

- When we used models pretrained on unrelated Image Net dataset for the construction of the ensemble architectures
- It significantly enhanced the performance on detecting PD compared to untrained models.
- Our finding suggests a promising direction, where unrelated training data can be considered when insufficient or no training data is available for a particular application.

Solution Architecture Diagram:



5.3 User Stories

User Type	Epic	UserStory Number	User Story / Task	Acceptance criteria	Priority	Release
Customer (Web user)	Registration	USN-1	As a user, I can register for the application by entering my email, password, and confirming my password.	can access my account / dashboard	High	Sprint-1
Customer (Web user)	Registration	USN-2	As a user, I will receive confirmation email once I have registered for the application	I can receive confirmation email & click confirm	Medium	Sprint-3

Customer (Web user)	Login	USN-1	As a user, I can login for the application by entering my email and password	I can access my account / dashboard	High	Sprint-1
Customer (Web user)	Main	USN-1	As a user, I want to take photo by clicking take photo button	I can access device camera and take photo	High	Sprint-2
Customer (Web user)	Main	USN-2	As a user, I want to get nutrition info by uploading photo	I can view the information	High	Sprint-2
Customer (Web user)	Main	USN-3	As a user, I want to get diet plan by send criteria	I can view the plan	Medium	Sprint-3

6. PROJECT PLANNING & SCHEDULING

6.1 Sprint Planning & Estimation

Sprint	Epic	UserStory Number	User Story / Task	Story Points	Priority	Team Members
Sprint-1	Authentication	USN-1	As a user, I can register for the application by entering my email, password, and confirming my password	2	Medium	KARTHIKEYAN
Sprint-1	Authentication	USN-2	As a user, I will receive confirmation email once I have registered for the application	1	Medium	MANIGANDAN
Sprint-1	Authentication	USN-3	As a user, I can log into the application by entering email & password	2	Low	KAVUSHICK
Sprint-1	Application	USN-1	As a developer, I want to install requirements, create IBM account, API access & send grid account.	1	Medium	KEETHALOCIOUS
Sprint-2	Application	USN-1	As a user, I can see the homepage by log into the	1	Low	Kavushick

			application			
Sprint-2	Applicaton	USN-2	As a user, I can see the upload page by navigating using navigation bar	1	Low	Manigandan
Sprint-2	Application	USN-3	As a user, I can upload my food image by using camera or pick from gallery	5	High	KARTHIKEYAN
Sprint-2	Application	USN-4	As a user, I can get the ingredients of the food and nutritional details of food by using AI-Driven Food Detection Model	5	High	KARTHIKEYAN
Sprint-3	Application	USN-5	As a user, I can see my diet history by navigating using navigation bar	2	Low	Keeth Aloclous
Sprint-3	Application	USN-6	As a user, I can get some diet plans by queries nutrition API	5	High	Manigandan
Sprint-3	Chat Bot	USN-1	As a user, I can get some diet based queries by using the chat bot	3	Medium	Kavushick

Sprint-4	Final	USN-1	As a developer, I can containerize the application by using Docker	5	High	KARTHIKEYAN
Sprint-4	Final	USN-2	As a developer, I can deploy the application image to the IBM Cloud Kubernetes Service	5	High	KARTHIKEYAN

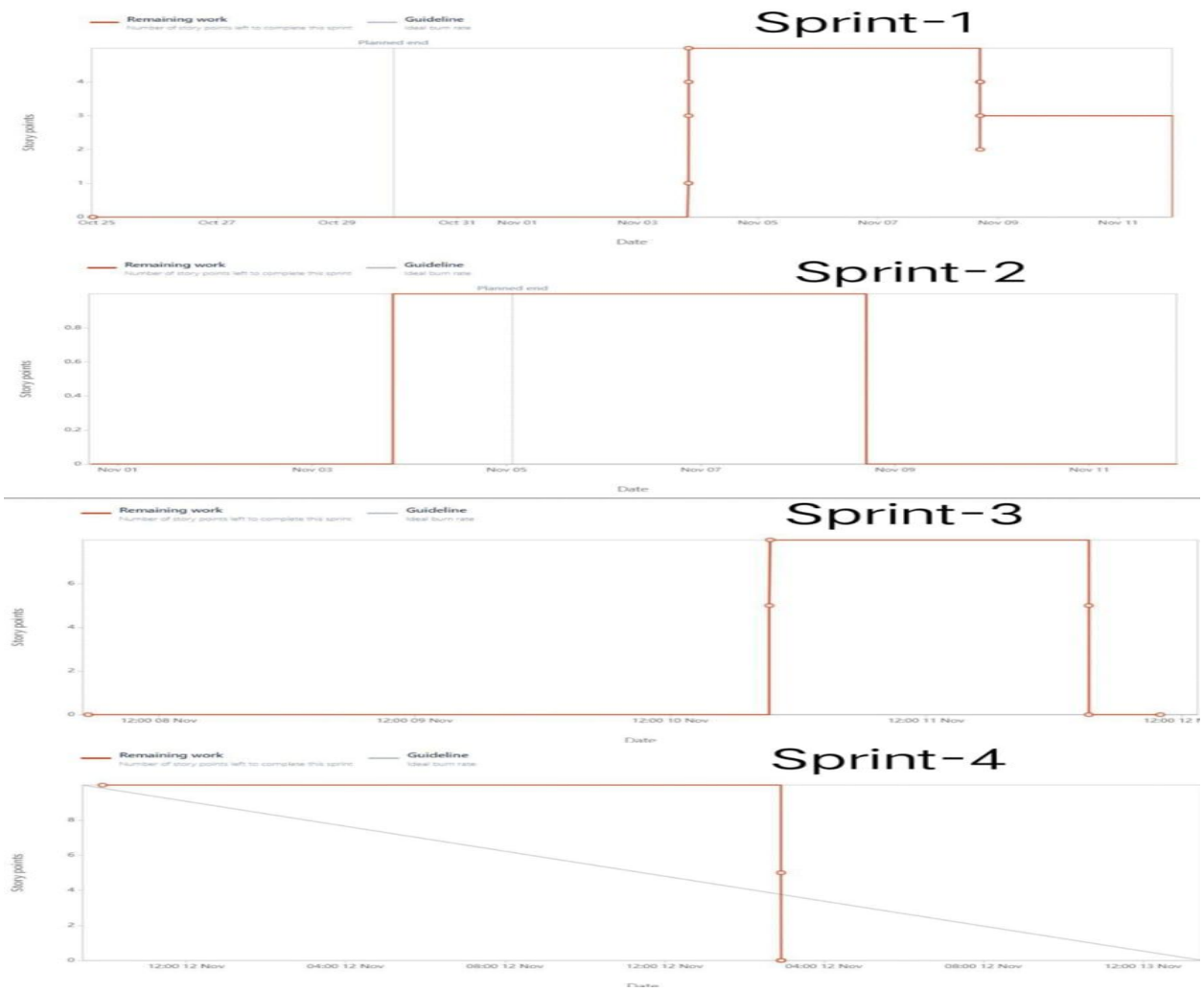
6.2 Sprint Delivery Schedule

Sprint	**Total Story Points**	**Duration**	**Sprint Start Date**	**Sprint End Date (Planned)**	**Story Points Completed (as on Planned EndDate)**	**Sprint Release Date (Actual)**
Sprint-1	6	6 Days	24 Oct 2022	29 Oct 2022	4	29 Oct 2022
Sprint -2	12	6 Days	31 Oct 2022	05 Nov 2022	12	05 Nov 2022

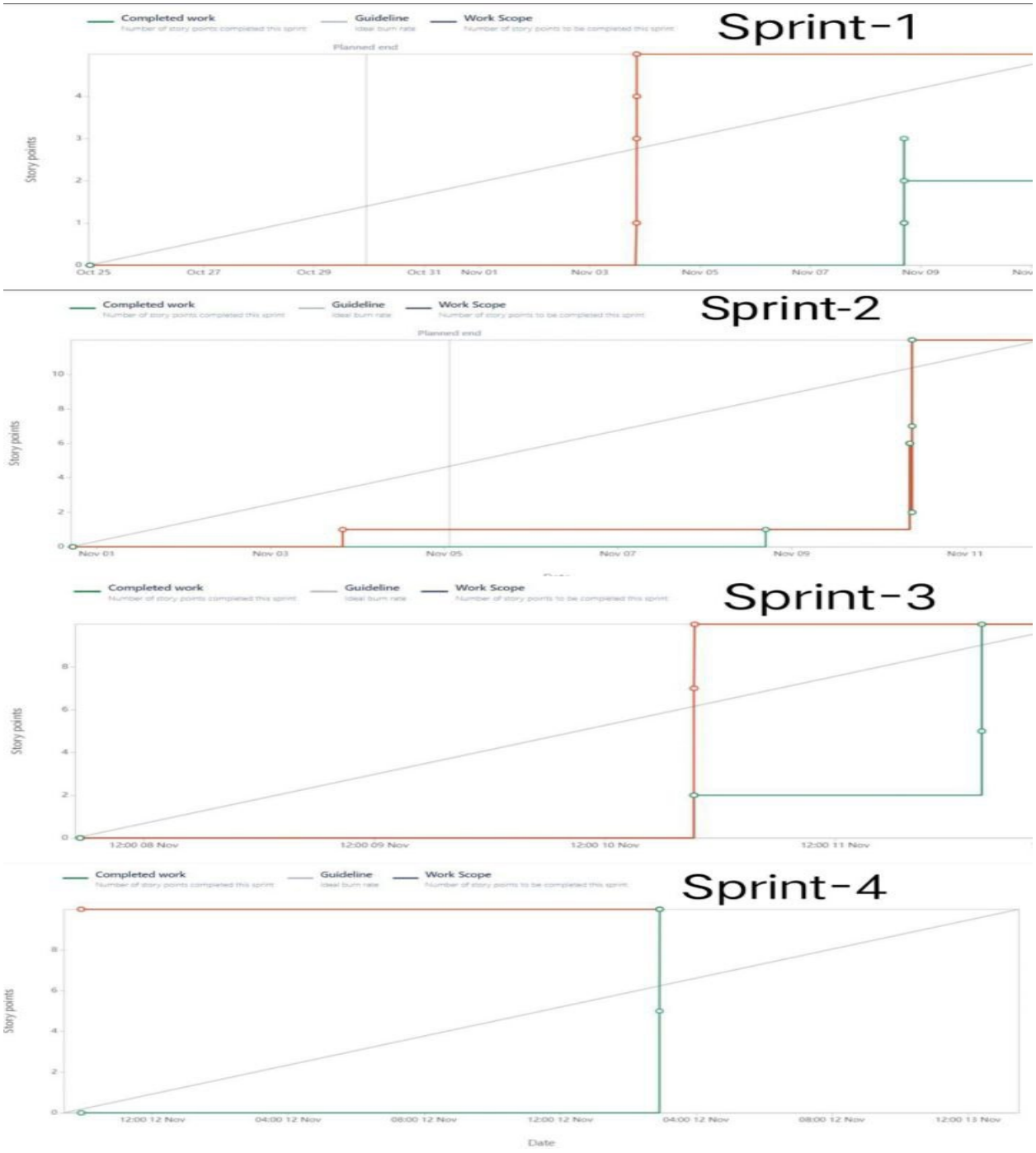
Sprint -3	10	6 Days	07 Nov 2022	12 Nov 2022	10	12 Nov 2022
Sprint -4	10	6 Days	14 Nov 2022	19 Nov 2022	10	13 Nov 2022

6.3 Reports from Jira

6.3.1 Burndown chart



6.3.2 Burnup chart



7. CODING & SOLUTIONING

7.1 Authentication

7.1.1 Registration

```
@app.route('/register', methods = ['POST', 'GET'])
def register():
    if request.method == 'POST':
        name = request.form["name"]
        email = request.form["email"]
        password = request.form["password"]
        user_controller.create_user(name, email, password)
        return redirect('/main/')

    return render_template('register.html')
```

When a user clicks the "Register" button on the registration form, their email address, name and password are collected from them. When a POST request with valid data is sent to the server, a new user is successfully created. If the POST request fails, an error message is displayed to inform the user that the data was not sent successfully. If they click "Register" again, they will be able to try again.

7.1.2 Login

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

```
    if request.method == 'POST':
        email = request.form["email"]
        password = request.form["password"]
        user = user_controller.fetch_user_by_email(email)

        if user.password == password:
            return redirect('/main')

    return render_template('login.html')
```

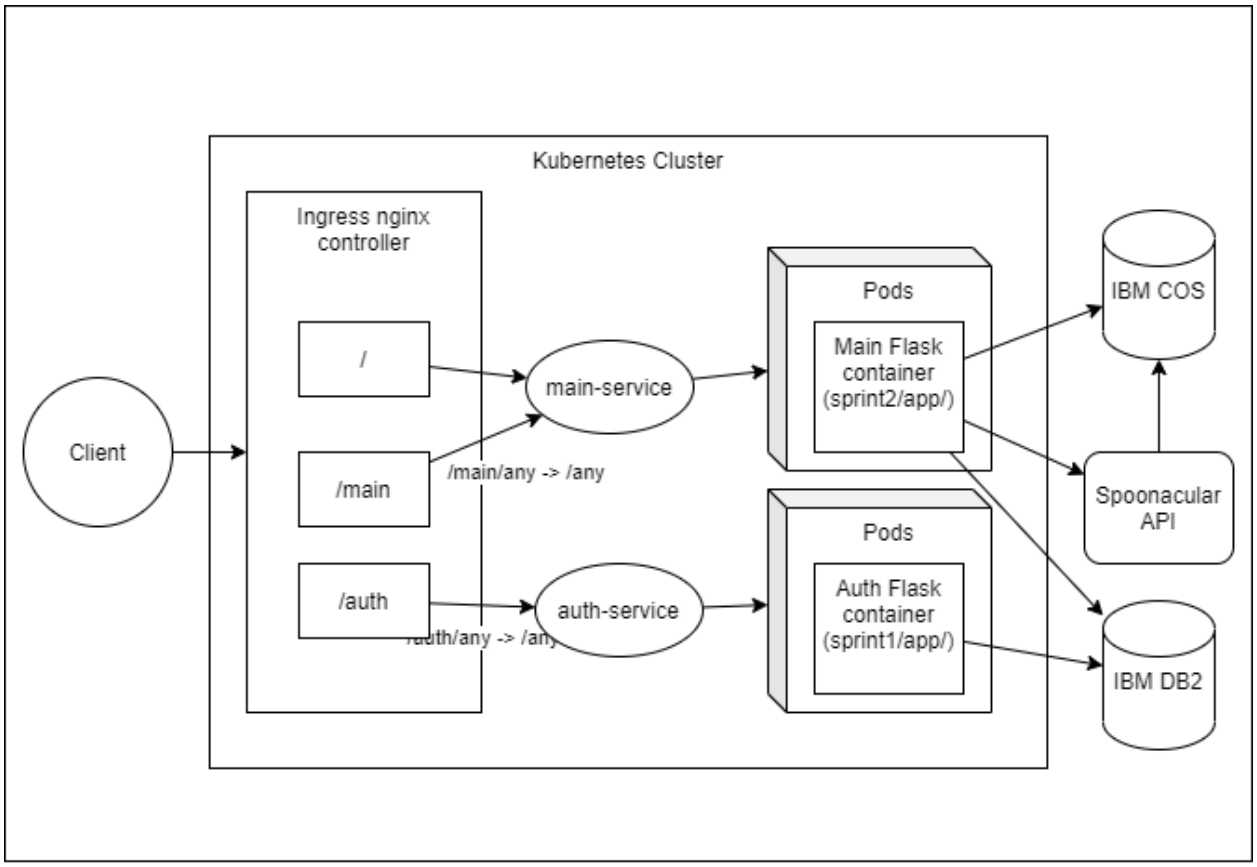
7.2 Image Analysis

```
def analyze_image(image_file_from_client):
    #init
    img_filename = random_name()
    local_path = os.path.join('upload', img_filename)
    # save
    image_file_from_client.save(local_path)
    image_ibm_cos_url = ibm_cos.upload_file(local_path, img_filename)
    # get details
    nutrition_details = fetch_nutrition_details(image_ibm_cos_url)
    # delete the image file
    os.remove(local_path)
    ibm_cos.delete_item(rand_img_filename)

    return nutrition_details
```

- **#init -** generates random name & local path to save the image.
- **#save** - Image is first saved locally and then upload it to the ibm cos storage
- **#get details -** image is then analyzed using nutrition api by passing the image's public url (ibm cos url)
- **#delete -** Finally, All the images are cleared and deleted both in server and ibm cos

7.3 Kubernetes cluster deployment



7.3.1 Ingress deployment script

```
apiVersion: networking.k8s.io/v1
    kind: Ingress
    metadata:
        annotations:
            nginx.ingress.kubernetes.io/rewrite-target: /$2
        name: naa-ingress
    spec:
        ingressClassName: nginx
        rules:
            - http:
                paths:
                    - path: /auth(/|$)(.*)
                      pathType: Prefix
                      backend:
                          service:
                              name: auth-service
                          port:
                              number: 5000
                paths:
                    - path: /main(/|$)(.*)
                      pathType: Prefix
                      backend:
                          service:
                              name: main-service
```

port: number: 5000

8. TESTING

8.1 Test Cases

- Verify user is able to see login page
- Verify user is able to login to application or not
- Verify user is able to navigate to registration page
- Verify user is able to receive verification email
- Verify user is able to see their food consumption history
- Verify user is able to view the stats of their history
- Verify user upload image
- Verify user is get correct predictions
- Verify user is able to add details manually
- Verify user is able to chat with Watson chat bot

8.2 User Acceptance Testing

Test case ID	Test Scenario	Steps To Execute	Status	Comments
Auth_TC_OO1	Verify user is able to see login/registration page	1.Enter URL and click go 2.Verify login/registration page displayed or not	Pass	
Auth_TC_OO2	Verify user is able to login/register to application or not	1.Enter URL and click go 2.Verify login/registration page displayed or not	Pass	

		3. enter info 4. click continue button		
Auth_TC_OO3	Verify user is able to receive verification email	1.Enter URL and register new account 2. click continue button	Pass	
Main_TC_OO4	Verify user is able to see their food consumption history?	1. Login into app 2. Go to home/main page 3. Verify previous history is visible	Pass	
Main_TC_OO5	Verify user is able to view the stats of their history	1. Login into app 2. Go to stats page 3. Verify stats are visible	Pass	
Main_TC_OO6	Verify user upload image	2. Login into app 2. Go to add food page 3. Verify user able to upload image	Pass	
Main_TC_OO7	Verify user is get correct predictions	1. Login into app 2. Go to upload page 3. click continue	Pass	
Main_TC_OO8	Verify user is able to add details manually	4. Login into app 2. Go to stats page 3. Verify stats are visible	Pass	Not predicating the Indian foods correctly
Bot_TC_OO9	Verify user is able to chat with Watson chat bot	1. login into app 2. click bot floating button on bottom right	Fail	responses are slow

9. RESULTS

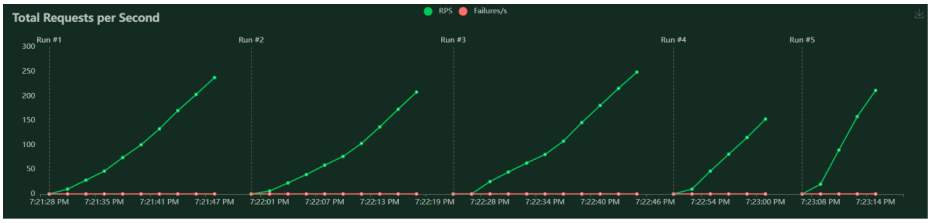
9.1 Performance Metrics

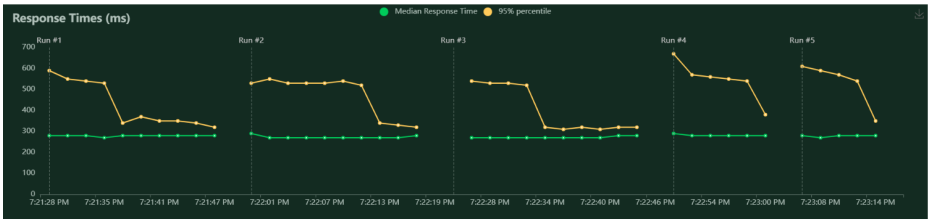
9.1.1 End Result

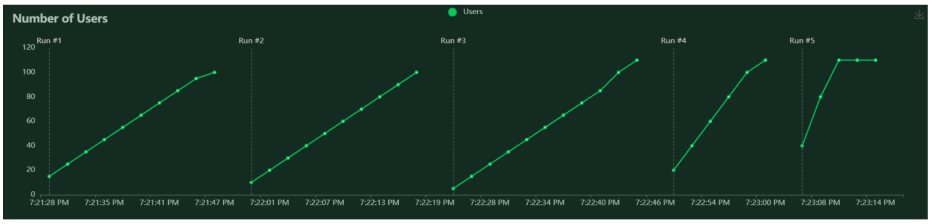
S.No	NFT Test Approach	Risk	Result
1	LOAD	Page gets slow down, when high volume of requests are coming to the server	PASS

9.1.2 Load Test Report

Request Count	Average Response Time	Average Content Size	Failures/s
3062	293.54906260641	1380	0

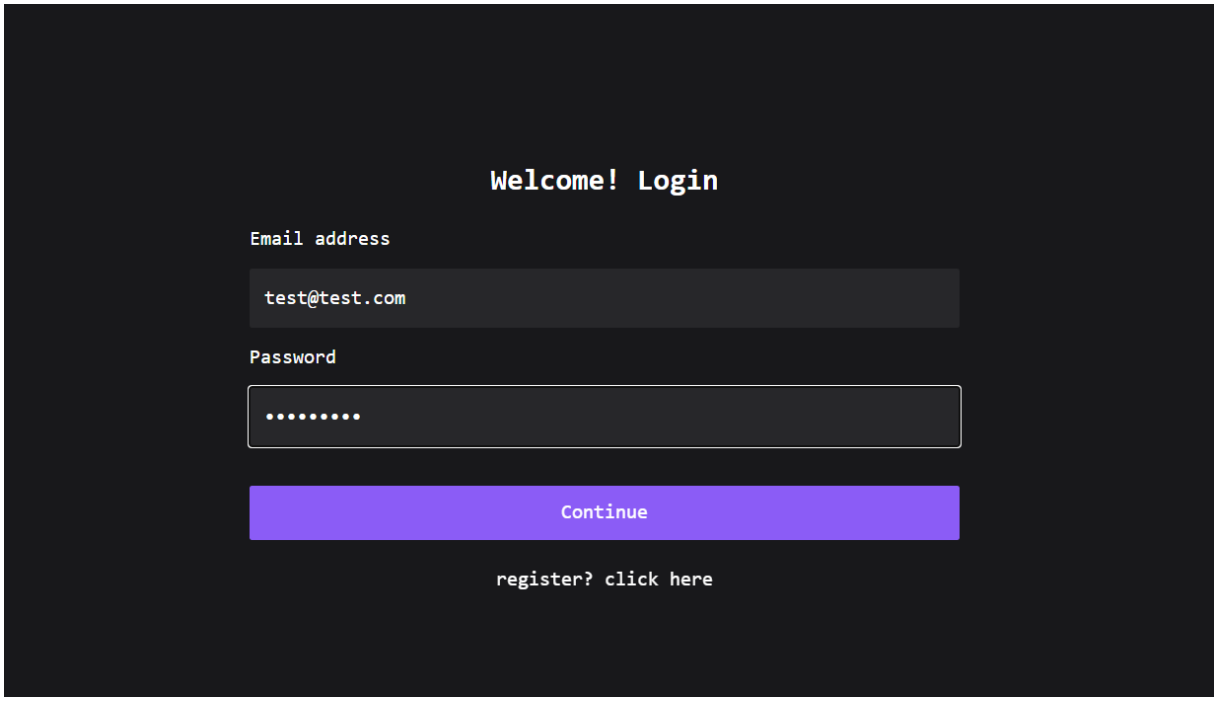


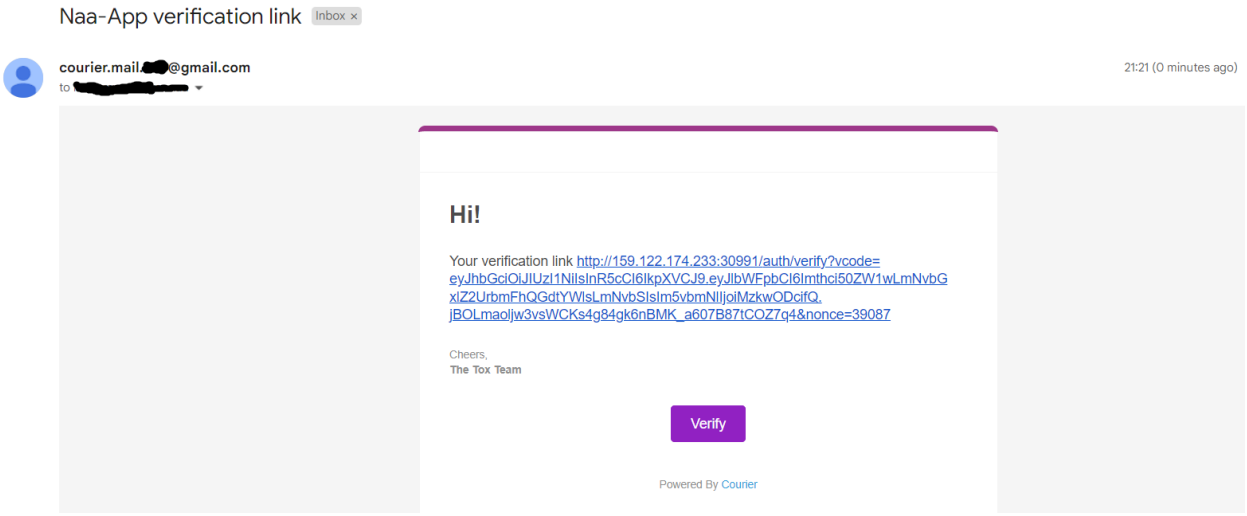




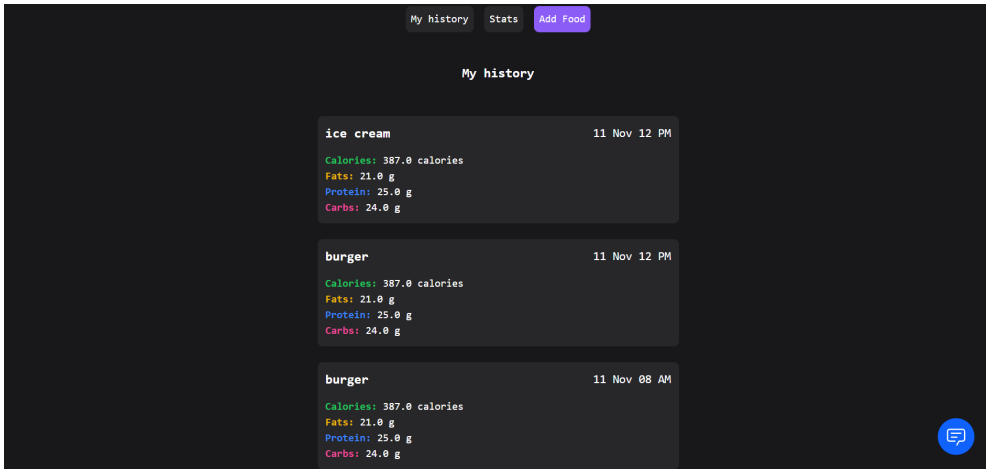
9.1.3 Screenshots

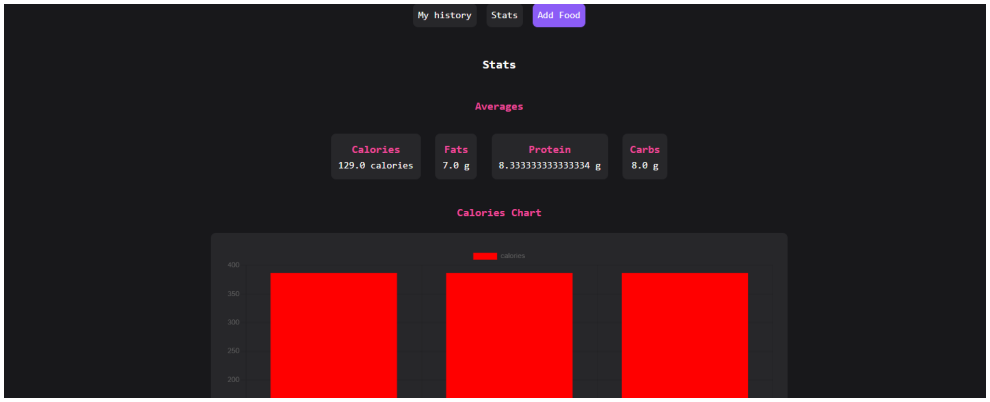
9.1.3.1 Authentication System

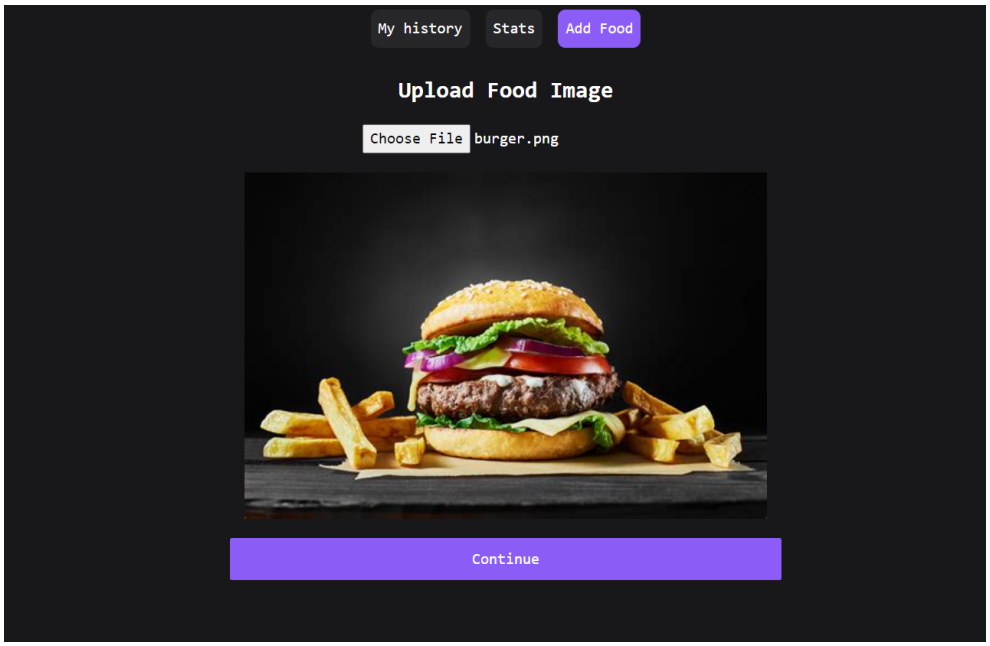




9.1.3.2 Main System







10. ADVANTAGES & DISADVANTAGES

10.1 Advantages:

10.1.1 They Can Help You Eat Healthier : Perhaps most importantly, they can help you improve your diet so you are eating healthier. Using a nutrition app will inherently help you eat a better and more optimized diet. Not only do these apps typically have calorie counters in them, but they can show you how many macro and micronutrients you are getting into your diet. Without having an app that tracks your food intake, you're going to find it increasingly difficult to track this

manually. These apps will generally make recommendations on what foods you should be looking to get into your diet. You cannot fool yourself when you are using a nutrition app. You will see the foods you are eating and the app will generally tell you if you are consuming too much of any type of food.

10.1.2.Track Your Progress : One of the most important things that you can do when you are looking to improve your diet choices and your weight loss efforts is to track your progress. This is something that you will either need to do manually or in an automated fashion using an app like this. A nutrition app that can track and monitor your progress is worth using. Without progression tracking, you are going at things without a measuring stick. Having an app that tracks and monitors your progress is the only way to identify what you are doing right and what you are doing wrong. It also shows you what you may need to change to improve. Being able to check your progress at a glance can do wonders for your efforts. Not only will you have something providing you with additional motivation, but it can also put things into perspective when you aren't necessarily visualizing any results from your efforts

10.1.3.Recipes : A nutrition app can be a great benefit to anyone who is looking to improve their diet for various reasons. One of the best reasons has to do with the recipes you will find. One of the biggest pain points of anyone who is looking to lose weight is not knowing what to eat. A lot of diets fail because they pigeonhole you into eating a specific type of meal continuously.

10.2 Disadvantages:

10.2.1. Inaccuracy

- One of the downsides to using an app for calorie counting is the fact that it's not always going to be accurate. Whether you mistype something or it tracks incorrectly, it can completely throw off your true macronutrient count.

- If you aren't logging things accurately, you could be hurting yourself more than helping.

- However, these same pitfalls are true if you are manually counting them. You have just as much room for user error. However, it needs to be noted.

10.2.2. Time-Consuming

- Another con associated with using and relying on these apps has to do with the time-consuming nature of logging all of your data. Many people won't want to spend more than a couple of minutes logging their information.

- These apps generally require a good amount of input to track everything. If you feel like it's too much work, you are likely to quit doing it. limited time and you think it will stop you from wanting to progress, it may be best to leave these apps alone

11. CONCLUSION

Good nutritional habits and a balanced diet aren't developed in one day, nor are they destroyed in one unbalanced meal. Healthful eating means a lifestyle of making choices and decisions, planning, and knowing how to make quick and wise choices when you haven't planned. What you learn about eating in these first years on your own will help establish good dietary patterns for the rest of your life. Making a break from home cooking and becoming responsible for choosing

the foods you eat is part of the challenge of becoming a mature and independent adult. It is a challenge that should not be taken lightly. The nutritional habits you develop now will be difficult to change in the coming years when your body stops growing and your lifestyle may become more sedentary. Learning to make sensible choices from a confusing array of options is not easy, but the rewards are great. Eating nutritious and healthful food while maintaining your proper body weight will contribute to a better performance in the classroom, in the gym, and on the dance floor. You will feel and look your best. In contrast, a poor diet can lead to insidious health problems that can interfere with success in academic and social performance and may eventually mean confronting a serious long-term illness, such as heart disease or diabetes. Knowing how much and what to eat is important knowledge. Good diet not only boosts physical health and disease susceptibility, but it has also been shown to contribute to cognitive growth and academic achievement. Good diet not only boosts physical health and disease susceptibility, but it has also been shown to contribute to cognitive growth and academic achievement. Finally, a healthy diet is essential for good mental health. Studies have shown that people who eat a lot of sugar and processed foods are more likely to suffer from anxiety and depression. When you eat well, you have more energy and feel happier by using the Nutrition assistant app.

12. FUTURE SCOPE

- The tool will also help you estimate the quantity and intensity of the food's flavor. In order to better serve user needs, future objectives include improving the precision of our machine learning model and extending the range of food categories.

- In order to improve the effectiveness of our output, we are also expanding our dataset of picture and nutrition categories. Our team members set the threshold for our project so that we also understand the components and the number of nutrients in a particular cuisine, but our research mainly just identifies the nutrients. In the late 18th century, during the chemical revolution, scientific examination of food and nutrition started.

- In order to establish nutritional theories, chemists in the 18th and 19th centuries experimented with various elements and food sources. The discovery of certain micronutrients in the 1910s marked the beginning of modern nutrition science. Thiamine was the first vitamin to be chemically identified, and in the decades that followed, the role of vitamins in nutrition was researched. During the Great Depression and World War II, the first nutritional guidelines for humans were created. The study of nutrition has placed a strong emphasis on human nutrition and agriculture, while placing ecology as a secondary issue because of its significance to human health. In order to live a healthy life, proper nutrition is essential.

13. APPENDIX

13.1 Source Code

13.1.1 Auth Server

```
from flask import Flask, jsonify, make_response, redirect, request,
render_template

from keys import SERVER_SECRET, MAIN_SERVER_REDIRECT_URL
```

```
import controllers.auth_controller as auth_controller
import controllers.user_controller as user_controller

from models.user import User


# init
app = Flask(__name__)
app.secret_key = SERVER_SECRET

User.create_users_table()

# routes

@app.route('/main', methods = ['POST', 'GET'])
# Dummy route
def main():
  try:
    token = request.cookies.get('access-token')
    user = auth_controller.decode_token(token)
    return f"<p>{token}</p><p>{user}</p>"
  except Exception as error:
    return str(error)



@app.route('/register', methods = ['POST', 'GET'])
def register():
  if request.method == 'GET':
    return render_template('register.html')

  elif request.method == 'POST':
    name = request.form["name"]
    email = request.form["email"]
    password = request.form["password"]
    retyped_password = request.form["retyped_password"]

    if password != retyped_password:
```

```
    return render_template('register.html', error="password is not matching")

  user = user_controller.fetch_user_by_email(email)

  if not user:
    user = user_controller.create_user(User(userid=None, name=name,
email=email, password=password))
    auth_controller.send_verification_mail(user)

  user = user_controller.fetch_user_by_email(email)

  if user.password == password:
    token = auth_controller.gen_access_token(user)

    res = make_response(redirect(MAIN_SERVER_REDIRECT_URL +
f'?access-token={token}'))
    res.set_cookie('access-token', token)
    return res
  else:
    return render_template('register.html', error="user already exists")

@app.route('/', methods = ['POST', 'GET'])
@app.route('/login', methods = ['POST', 'GET'])
def login():
 if request.method == 'POST':
  email = request.form["email"]
  password = request.form["password"]

  user = user_controller.fetch_user_by_email(email)

  if user:
    if user.password == password:
      token = auth_controller.gen_access_token(user)
      print("test============", MAIN_SERVER_REDIRECT_URL)
      res = make_response(redirect(MAIN_SERVER_REDIRECT_URL +
f'?access-token={token}'))
      res.set_cookie('access-token', token)
      return res
    else:
```

```
      return render_template('login.html', error="password is wrong")
    else:
      return render_template('login.html', error="user is not exists")

  return render_template('login.html')


@app.route('/verify')
def verify():
  vcode = request.args.get('vcode') or ''
  nonce = request.args.get('nonce') or ''

  try:
    decoded_vcode = auth_controller.decode_token(vcode)

    if nonce == decoded_vcode['nonce']:
      return render_template('verify.html', msg=decoded_vcode['email'] + " is
verified, Thank you!")
    else:
      return render_template('verify.html', msg="error occured! Not verified")
  except Exception as error:
      return render_template('verify.html', msg=error)


@app.route('/isvalid')
def isvalid():
  try:
    token = request.args.get('token') or ''
    ss = request.args.get('ss') or ''

    if ss != SERVER_SECRET:
      return jsonify({
        'status': True,
        'error': 'Not Authorized'
      })

    user = auth_controller.decode_token(token)
    return jsonify({
      'status': True,
```

```
      'user': user
    })
  except Exception as error:
    return jsonify({
      'status': False,
      'error': str(error)
    })

if __name__ == '__main__':
    app.run(host='0.0.0.0')
```

13.2 Main Server

```
from flask import Flask, make_response, redirect, render_template, request,
session
from datetime import datetime

try:
    import setkeys
    setkeys.load_keys()
except ImportError:
    print("Environments variables are not set")

from keys import SERVER_SECRET, AUTH_SERVER_REDIRECT_URL

from utils.jwt_token import validate_token

from models.food import Food

from controllers import main_controller


from models.food import Food

app = Flask(__name__)
app.secret_key = SERVER_SECRET
```

```
Food.create_foods_table()


@app.route("/")
def set_token():
    token = request.args.get('access-token') or request.cookies.get('access-token')
or ''

    if token != '':
        res = make_response(redirect('my_history'))
        res.set_cookie('access-token', token)
        return res
    else:
        res = make_response(redirect(AUTH_SERVER_REDIRECT_URL))
        return res

@app.route('/my_history', methods = ['POST', 'GET'])
@validate_token
def main():
    try:
        nds = main_controller.fetch_food(session['user']['userid'])
        for nd in nds:
            nd['time'] = nd['time'].strftime('%d %b %I %p')

        print(nds)

        return render_template('my_history.html', nd_len=len(nds),
nutrition_details=nds)
    except Exception as e:
        return str(e)


@app.route('/stats')
@validate_token
def overview():
    try:
        nutrition_details = main_controller.fetch_food(session['user']['userid'])

        time_arr = []
```

```
        calories_arr = []
        fat_arr = []
        protein_arr = []
        carbs_arr = []

        calories=0
        fat=0
        protein=0
        carbs=0

        for nd in nutrition_details:
            time_arr.append(nd['time'].strftime("%d/%m/%Y %H:%M:%S"))
            calories_arr.append(nd['calories'])
            fat_arr.append(nd['fat'])
            protein_arr.append(nd['protein'])
            carbs_arr.append(nd['carbs'])

            calories = nd['calories']
            fat = nd['fat']
            protein = nd['protein']
            carbs = nd['carbs']

        nd_len = len(nutrition_details)
        if nd_len <= 0:
            nd_len = 1
        calories=calories/nd_len
        fat=fat/nd_len
        protein=protein/nd_len
        carbs=carbs/nd_len

        return render_template('stats.html', calories=calories, fat=fat,
protein=protein, carbs=carbs, time_arr=time_arr, calories_arr=calories_arr,
fat_arr=fat_arr, protein_arr=protein_arr, carbs_arr=carbs_arr)
    except Exception as e:
        print(e)
        return "null"


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

```
@validate_token
def add_food():
    return render_template('add_food/actions.html')


@app.route('/upload_image', methods=['GET', 'POST'])
@validate_token
def upload_image():
    if request.method == 'POST':
        if 'food_image' not in request.files:
            return render_template("add_food/upload_image.html", error="food
image is not given")

        image_file_from_client = request.files['food_image']

        nutrition_details = main_controller.analyze_image(image_file_from_client)

        query_str = '?'
        query_str += 'name=' + nutrition_details['name'] + '&'
        query_str += 'accuracy=' + str(nutrition_details['accuracy']) + '&'
        query_str += 'calories=' + str(nutrition_details['calories']) + '&'
        query_str += 'fat=' + str(nutrition_details['fat']) + '&'
        query_str += 'protein=' + str(nutrition_details['protein']) + '&'
        query_str += 'carbs=' + str(nutrition_details['carbs'])

        return redirect('add_details' + query_str)

    return render_template('add_food/upload_image.html')



@app.route('/add_details', methods=['GET', 'POST'])
@validate_token
def add_details():
    name = ''
    accuracy = ''
    calories = ''
    fat = ''
    protein = ''
```

```
    carbs = ''

    if request.method == 'POST':
        try:
            # collect details
            name = request.form['food_name']
            calories = request.form['calories']
            fat = request.form['fat']
            protein = request.form['protein']
            carbs = request.form['carbs']

            # store in db
            user = session['user']
            food = Food(name, datetime.today().timestamp(), calories, fat, protein,
carbs)
            main_controller.save_food(user['userid'], food)

            # return to home page
            return redirect('my_history')

        except Exception as e:
            return render_template('add_food/add_details.html', name=name,
accuracy=accuracy, calories=calories, fat=fat, protein=protein, carbs=carbs,
error=str(e))




    try:
        name = request.args['name'] or ''
        accuracy = float(request.args['accuracy'] or '0')
        calories = float(request.args['calories'] or '0')
        fat = float(request.args['fat'] or '0')
        protein = float(request.args['protein'] or '0')
        carbs = float(request.args['carbs'] or '0')
    except:
        pass
```

```
    return render_template('add_food/add_details.html', name=name,
accuracy=accuracy, calories=calories, fat=fat, protein=protein, carbs=carbs)


if __name__ == '__main__':
    app.run(host='0.0.0.0')
```

13.2 GitHub & Project Demo Link

13.1 Video Link

https://youtu.be/JFRLPu-cDVw

13.2.1 Github Link

https://github.com/IBM-EPBL/IBM-Project-3376-1658557435

13.2.1 Demo Link

13.2.1.1 k8s cluster 1

http://159.122.174.233:30991/main/

Expires after 30-11-22, because it is hosted on free cluster

13.2.1.2 k8s cluster 2

http://169.51.204.20:30634/main/

Expires after 20-12-22, because it is hosted on free cluster