IBM PROJECT REPORT

TEAM ID : PNT2022TMID49091

PROJECT TITLE : NUTRITION ASSISTANT APPLICATION

TEAM MEMBERS

BASKARABOOPATHY P (921819106002)
SURYA PRAKASH S (921819106018)
SENGAIPANDI N (921819106017)
THANGAPANDI R (921819106019)

ABSTRACT

The importance of food for human survival has been discussed in several medical conferences. Consumers now have more opportunities to learn about nutrition patterns, understand their daily eating habits, and maintain a balanced diet owing to modern dietary evaluation and nutrition analysis tools. Due to the ignorance of healthy food habits, obesity rates are increasing at an alarming speed, and this is reflective of the risks to people's health. People need to control their daily calorie intake by eating healthier foods, which is the most basic method to avoid obesity. However, although food packaging comes with nutrition (and calorie) labels, it's still not very convenient for people to refer to App-based nutrient dashboard systems which can analyze real-time images of a meal and analyze it for nutritional content which can be very handy and improves the dietary habits, and therefore, helps in maintaining a healthy lifestyle.This project aims at building application that automatically estimates food attributes such as ingredients and nutritional value by classifying the input image of food. This method employs deep learning model (CNN) for accurate food identification and Food API's to give the nutritional value of the identified food.

1. INTRODUCTION

1.1 PROJECT OVERVIEW

The style of life in the modern world is evolving daily, and with it, the requirements of the human body's composition, which includes a variety of consumables or prepared foods, are changing as well. Obesity, or being overweight, is one issue brought on by an abundance of food in our daily life. The body's excessive calorie consumption is the root cause of this issue. In today's world, obesity is becoming a major problem. Therefore, we need a system that can influence people's eating habits and give them guidance leading to healthy lifestyle maintenance. If a system alerts users to the nutritional information of a food item and categorises it as healthy or unhealthy as well as the nutrition content to the user, they can establish their daily intake of calories from their diet. First, we must determine the type of food, and then, after predicting the type of food (fruit or vegetable), our system must determine the type of that image (if the image is in the category of food or vegetable). A combination of deep learning techniques is used to recognise the image and determine the category based on it. Our approach incorporates a wide range of segmentation and picture features.

1.2 PURPOSE

The type of food must be determined first, and then our system must determine the type of that image after predicting the type of food (fruit or vegetable) (if the image is in the category of food or vegetable). The image is recognised, and a category is derived from it using a combination of deep learning algorithms. Our method includes a variety of segmentation and image properties.

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

2. LITERATURE REVIEW

2.1 EXISTING PROBLEM

Decreased appetite:

Lack of appetite, or decreased hunger, is one of the most troublesome nutrition problems you can experience. Although it's a common problem, its cause is often unknown. Appetite-stimulating medicines are available. Ask your provider if such medicines would help you.

Snack guidelines:

Wasting your energy eating foods that provide little or no nutritional value (such as potato chips, candy bars, colas and other snack foods).
A nutritional problem or deficiency refers to a condition when an individual's body experiences a shortage of essential nutrients or some specific nutrient. Such problems can give rise to several health issues such as anaemia

2.2 REFERENCES

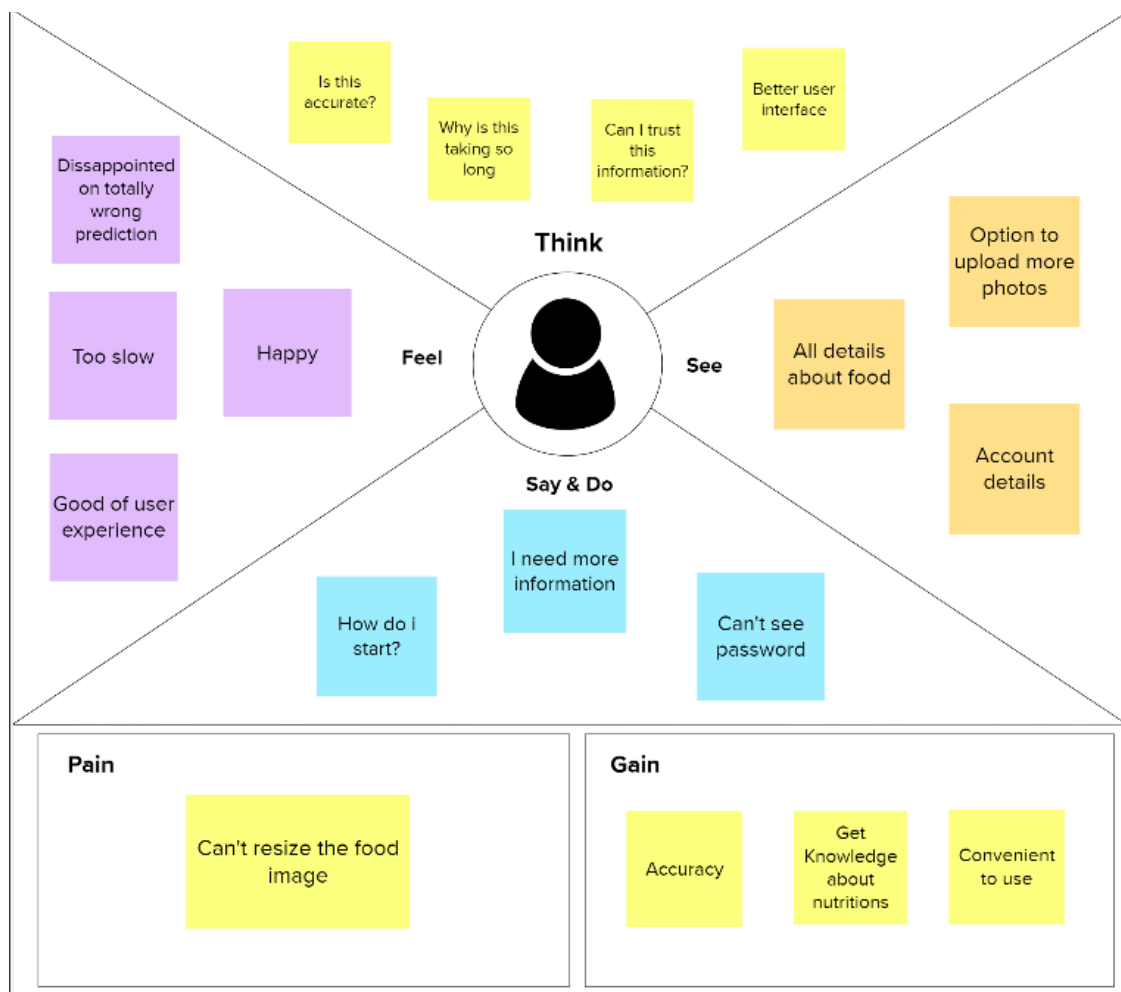| Title	Author(s)	Year	Findings/Pros/Cons
Application of Artifical Intelligence in Food industry-a guideline	Nidhi Mavani, Jarinah Mohd Ali, Suhaili Othman Mariya Hussian	2021	Helps to find the integration of AI technology in food industry
AI for Healthy meal preparation in smar cities	Bhuvana Namasivayam	2022	This aims to help the smart cities to process of preparation of healthy and nutritional foods with the help of AI technology
Agile Software development methodologies in cloud computing	Justin Ogala	2022	It give the information about the agile methodologies of cloud development
Development of a cloud based solution for effective nutrition intervention in the management of lifestyle diseases	Manju P George, C.A. Kalpana	2018	Nutritional cloud application development studies

NutriTrack: Android-based food recognition app for nutrition awareness	Arnel B. Ocay, Jane M. Fernandez, Thelma D. Palaoag	2017	This paper presents an analysis of the impact of the food recognition app to change people's concept of food nutrition.
A Performance evaluation of containers running on Kubernetes services	Arnaldo Pereira Ferreria, Richard Sinnott	2019	This article helps to get a knowledge about the deployment of application containers on kubernetes
Efficient way of Web development using Python and Flask	Fankar Armash Aslam, Hawa Nabeel mohammed	2015	This journal presumed the web application services of python and its flask framework.
A Study on Cloud Database	Deka Ganesh Chandra, Ravi Prakash, Swati Lamdharia	2012	This paper about the studies of varies cloud databases and its activities
Cloud database Database as a service	Waleed AI Shehri	2013	This paper includes all the basic information aboutthe database as a service. This paper provide the advantages and disadvantages of cloud database as a service
Effects and challenges of using a nutrition assistance system: Results of s long-term mixed-method study	Hanna Hauptmann, Nadja Leipold, Mira Madenach, Monika Wintergerst, Martin Lurz, Georg Groh, Markus Bohm, Kurt Gedrich & Helmut Krcmar	2021	This article survey helps to know the use of nutrition assistance systems is promising since previous studies have shown that persuasive technologies can help people to eat healthier

2.3 PROBLEM STATEMENT DEFINITION

In existing system, the way food data is analysed is evolving as a result of recent developments in computer vision and machine learning. Food-related photos, however, are frequently challenging to recognise and slow to detect due to the high quantity of food items and the ineffectiveness of the detection method. A diet plan is also fully manual to obtain. People may occasionally have to wait for many hours before learning about the suggested diet plan after contacting their dietician. For users, this makes getting their diet plan challenging. From the perspective of the customer, this is ineffective.

3. IDEATION & PROPOSED SOLUTION

3.1 EMPATHY MAP CANVAS

3.2 IDEATION & BRAINSTORMING

3.3 PROPOSED SOLUTION

Sufficient nutrition and energy consumption, which can only be obtained through proper eating habits, are essential for maintaining a healthy existence. In addition to maintaining a good standard of living, a healthy diet helps people prevent chronic conditions including diabetes, high blood pressure, mental illness, asthma, and others. Obesity is one of the conditions brought on by over eating that is most common. When a person is obese, excess body fat accumulates to the point where it endangers their health. For the suggested system to recognise and locate food items from the input images, develop an automated nutrition analysis system. By locating possible locations and classifying them with deep neural networks, develop a three-step procedure specifically for detecting various foods in photographs. In the first stage, the automated system generates numerous regions of proposals using the provided photos. Then, it collects each region of thoughts by locating them in the original pictures, putting them on feature maps, and classifying them according to different food groups. Finally, by analysing the nutritional elements in the images, determine the food's calories, fat, carbs, and protein contents as well as its ingredients to generate a dietary evaluation report. The system's effectiveness and accuracy will also be increased by expanding the dataset to cover a larger variety of food types.

S.No.	Parameter	Description
1.	Problem Statement (Problem to be solved)	Users not knowing which food contains which ingredients or nutrition
2.	Idea	Using AI model for predicting the food name, ingredients and nutrition values
3.	Uniqueness	Usage of AI APIs
4.	Customer Satisfaction	Accurate nutrition information
5.	Business Model (Revenue Model)	Open source
6.	Scalability of the Solution	Using kubernetes for scalability

3.4 PROBLEM SOLUTION FIT

There is no organised way to quickly collect dietary data. Waiting times for diet appointments might be hours long. Calculate the calories, fat, carbohydrates, and protein content after examining the nutritional components in the images to produce a report on the dietary quality. The system's effectiveness and accuracy will be increased by broadening the information to include a wider variety of food types.

Problem Solution Fit

4. REQUIREMENT ANALYSIS

4.1 FUNCTIONALREQUIREMENT

Upload Image

In this module, the nutrition datasets should be uploaded as CSV files. Additionally, the information is kept in a database for later use. The dataset includes information on the calories, protein, fat, and food ingredient values of fruits and vegetables. These numbers are preserved as integer values and acquired from the Kaggle website.

Filtering Noise

Ability to determine the nutritional content of diverse fruits and vegetables is made possible by the use of filter techniques that reduce noise in photos. The filter's goal is to get rid of noise, which makes photographs look bad. Statistics back up this statement. A filter's typical frequency response is constructed. Filtering is a nonlinear technique widely used in image processing to get rid of "salt and pepper" noise. When edge preservation and noise reduction are top concerns, a median filter is preferable to convolution. Practice photo binarization in a similar way. In the pre-processing stage of document analysis, document picture binarization is used to separate the text in the foreground from the backdrop of the document. The succeeding document image processing tasks require a quick and precise binarization approach.

Classification

Classification is the process of dividing data into various categories. The method starts by determining the class of the given data points. Classification is achievable for both structured and unstructured data. The terms target, label, and classes are occasionally used to describe the classes. The user-uploaded food image will be compared to the food items in the system database for the features obtained in the feature extraction stage in the classification process. The specific food item will be recognised once the ideal match is discovered based on the qualities matched. The detected food item's name with ingredients will be displayed over the food. Here, a convolution neural network approach is employed to classify data.

Nutrition Detection

The food nutrition API receives the image after the model has identified the food category or food type and extracts the food's nutritional data before sending it to the system. The system contrasts the nutritional information with the suggested dietary allowances. If the amount of a specific nutrient, let's say calories, exceeds the recommended dietary allowance, the user will receive a warning message to reduce nutrition intake. In that case, the user will see the food's nutritional value.

4.2   NON-FUNCTIONAL-REQUIREMENTS

Usability

The system shall allow the users to access the system with pc using web application. The system uses a web application as an interface. The system is user friendly which makes the system easy

Availability

The system is available 100% for the user and is used 24 hrs a day and 365 days a year. The system shall be operational 24 hours a day and 7 days a week.

Scalability

Scalability is the measure of a system's ability to increase or decrease in performance and cost in response to changes in application and system processing demands.

Security

A security requirement is a statement of needed security functionality that ensures one of many different security properties of software is being satisfied.

Performance

The information is refreshed depending upon whether some updates have occurred or not in the application. The system shall respond to the member in not less than two seconds from the time of the request submittal. The system shall be allowed to take more time when doing large processing jobs. Responses to view information shall take no longer than 5 seconds to appear on the screen.

Reliability

The system has to be 100% reliable due to the importance of data and the damages that can be caused by incorrect or incomplete data. The system will run 7 days a week. 24 hours a day.

5. PROJECT DESIGN

5.1 DATA FLOW DIAGRAMS

A two-dimensional diagram explains how data is processed and transferred in a system. The graphical depiction identifies each source of data and how it interacts with other data sources to reach a common output. Individuals seeking to draft a data flow diagram must identify external inputs and outputs, determine how the inputs and outputs relate to each other, and explain with graphics how these connections relate and what they result in. This type of diagram helps business development and design teams visualize how data is processed and identify or improve certain aspects.

Data Flow Diagram:
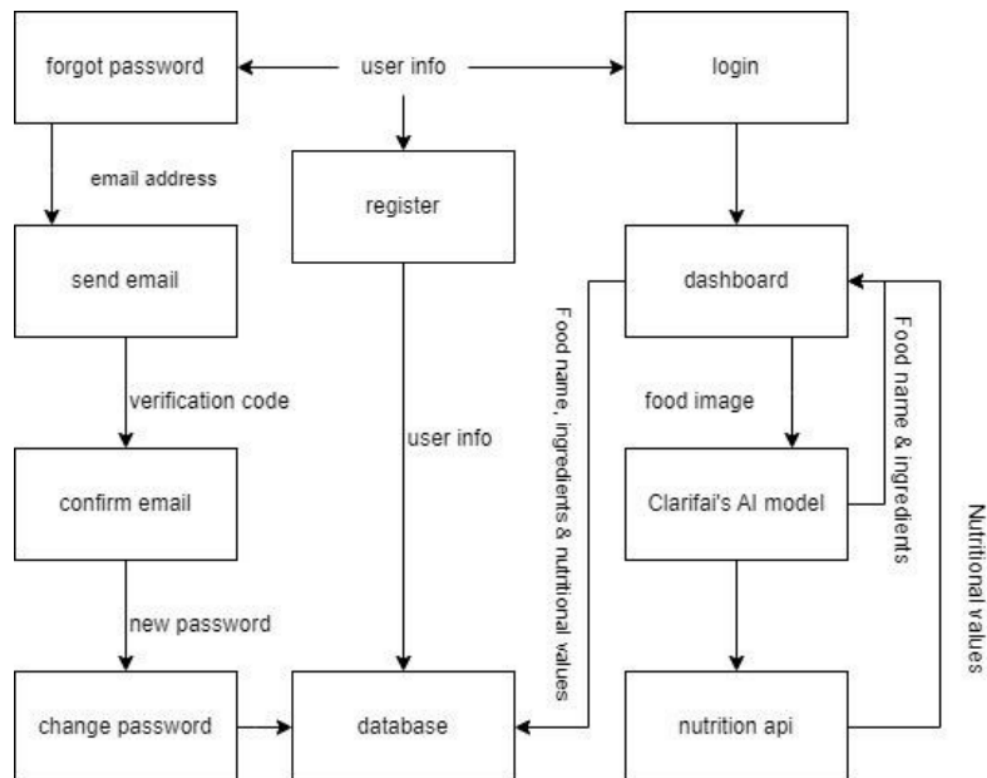
5.2 SOLUTION & TECHNICAL ARCHITECTURE

A system architecture or systems architecture is the conceptual model that defines the structure, behavior, and more views of a system. An architecture description is a formal description and representation of a system, organized in a way that supports reasoning about the structures and behaviors of the system.

Technical Architecture:

System architecture can comprise system components, the externally visible properties of those components, the relationships (e.g. the behavior) between them. It can provide a plan from which products can be procured, and systems developed, that will work together to implement the overall system. There have been efforts to formalize languages to describe system architecture, collectively these are called architecture description languages (ADLs).
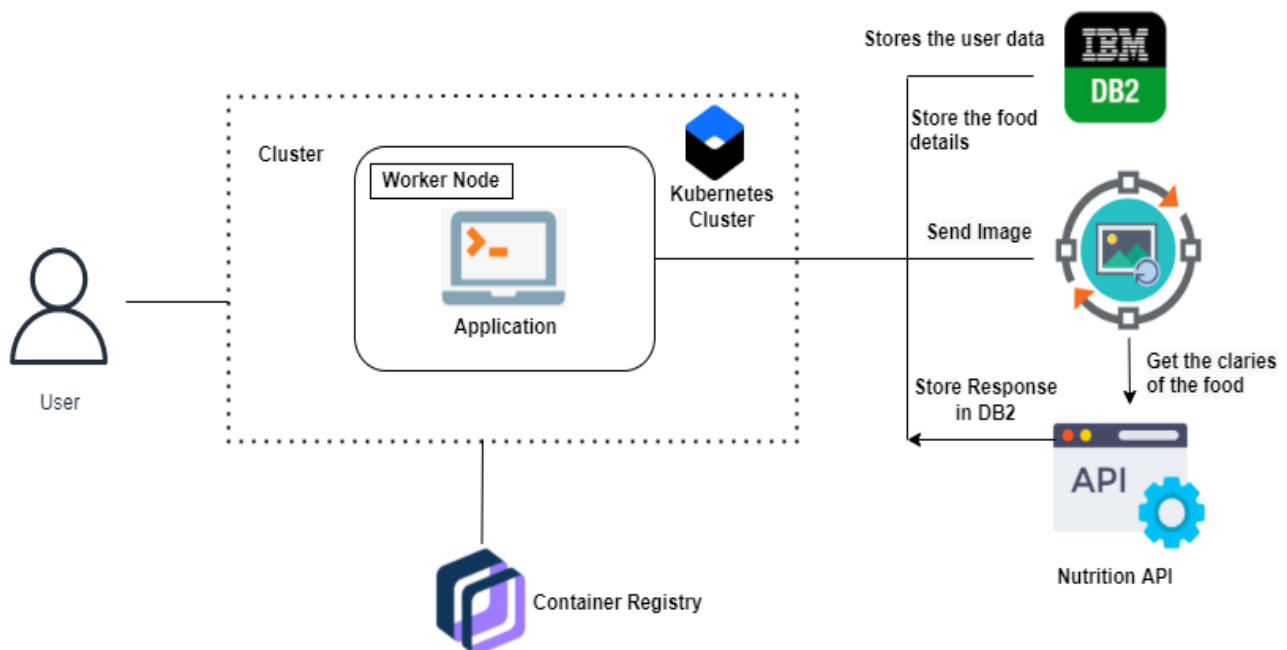
5.3 USER STORIES

As a User, I can register the application by entering of user's name, email id and password.

As a User. I can login the application by entering the user name and password.

As a User, If I forgot my password, I can change my password.

As a User, I can upload the food image to the application.

As a User, I can analyse and gather a knowledge about the ingredients values and nutritional values of the uploaded food image.

6. PROJECT PLANNING & SCHEDULING

6.1 SPRINT PLANNING & ESTIMATION

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority
Sprint-1	Registration	USN-1	As a user, I can register for the application by entering my email, password, and confirming my passwor and will receive confirmation email once I have registered for the application	2	High
Sprint-1	Login	USN-2	As a user, I can log into the application by entering email & password	2	High
Sprint-2	Forgot Password	USN-3	As a user i can change password if i forgot my password	1	Low
Sprint-2	Upload image	USN-4	As a user after loggin in i can upload food images in dashboard	2	High
Sprint-3	Food regognition	USN-5	I get food name recognized by AI model	3	High
Sprint-3	Nutrition API	USN-6	I get nutrition details from APIs	2	High

6.2 SPRINT DELIVERY SCHEDULE

Sprint	Total Story Points	Duration	Sprint Start Date	Sprint End Date (Planned)	Story Points Completed (as on Planned End Date)	Sprint Release Date (Actual)
Sprint-1	20	6 Days	24 Oct 2022	29 Oct 2022	20	01 Novt 2022
Sprint-2	20	6 Days	31 Oct 2022	05 Nov 2022	20	05 Nov 2022
Sprint-3	20	6 Days	07 Nov 2022	12 Nov 2022	20	11 Nov 2022

6.3 REPORTS FROM JIRA

Sprints		NOV
⚡ NA-5 Create a Flask Project	▮	
⚡ NA-6 Create IBM Cloud Account	▮	
⚡ NA-7 Install IBM Cloud CLI	▮	
⚡ NA-8 Docker CLI Installation	▮	
⚡ NA-9 Create An Account In SendGrid	▮	
⚡ NA-10 Create An Account In Nutrition API	▮	
⚡ NA-11 Create UI To Interact with Application	▮	
⚡ NA-12 Create IBM DB2 And Connect With Python		▮
⚡ NA-13 Integrate Nutrition API		▮
⚡ NA-14 SendGrid Integration With Python Code		▮
⚡ NA-15 Containerize The App		▮
⚡ NA-16 Upload Image To IBM Container Registry		▮
⚡ NA-17 Deploy In Kubernetes Cluster		▮

7. CODING & SOLUTIONING

Workflow

> ➢ User can enter the web app by registration and login.
> ➢ User interacts with the Web App to Load an image.
> ➢ The image is passed to the server application, which uses Clarifai's AI-Driven Food Detection Model Service to analyze the image and Nutrition API to provide nutritional information about the food.
> ➢ Nutritional information of the analyzed image is returned to the app for display.

7.1 FEATURE 1

In Dashboard page, It having the food image upload section and food details section in it. This page displayed the User name on dashboard and also having logout button for logging off of the user. This page can having to upload the food image to web app. The food image can recongize by using ***Clarifai's AI-Driven Food Detection Model*** and food is analysed by Intregrated Nutrition Food API. The Ingredients values and Nutritional values of food is displayed on the dashboard page. User can upload more images for analyse of its nutritional values.

Python Code:

```python
@app.route("/dashboard", methods=["GET", "POST"])
def dashboard():
 global request
 if flask.request.method == "POST" and session['LoggedIn']:
   if 'file' not in flask.request.files:
     flash('No file part')
     return redirect(flask.request.url)
   file = flask.request.files['file']
   if file.filename == '':
     flash('No image selected')
     return redirect(flask.request.url)
   if file and allowed_file(file.filename):
     filename = secure_filename(file.filename)
     file.save(os.path.join(app.config['UPLOAD_FOLDER'], filename))
     flash('Image successfully uploaded')

     with open(os.path.join(app.config['UPLOAD_FOLDER'], filename), "rb") as f:
       file_bytes = f.read()

     request = service_pb2.PostModelOutputsRequest(
         model_id="food-item-v1-recognition",
         user_app_id=resources_pb2.UserAppIDSet(app_id=YOUR_APPLICATION_ID),
```

```python
  inputs=[
    resources_pb2.Input(
      data=resources_pb2.Data(image=resources_pb2.Image(
        base64=file_bytes
      )
    )
  )
  ],
)
response = stub.PostModelOutputs(request, metadata=metadata)

if response.status.code != status_code_pb2.SUCCESS:
    print(response)
    raise Exception(f"Request failed, status code: {response.status}")

foodname = response.outputs[0].data.concepts[0].name

ingredients = ''
for concept in response.outputs[0].data.concepts:
    ingredients += f"{concept.name}: {round(concept.value, 2)}, "

nutritionValues = ''
nutritionApiUrl = "https://spoonacular-recipe-food-nutrition-v1.p.rapidapi.com/recipes/guessNutrition"
querystring = {"title":foodname}
headers = {
 "X-RapidAPI-Key": RAPIDAPI_KEY,
  "X-RapidAPI-Host": "spoonacular-recipe-food-nutrition-v1.p.rapidapi.com"
}
response = requests.request("GET", nutritionApiUrl, headers=headers, params=querystring)
nutritions = response.text
nutritions.pop('recipesUsed')
for i in nutritions:
    nutritionValues += f"{i}: {nutritions[i]['value']} {nutritions[i]['unit']}, "


sql = "INSERT INTO foods VALUES(?,?,?,?,?)"
stmt=ibm_db.prepare(conn, sql)
ibm_db.bind_param(stmt, 1, session['userid'])
ibm_db.bind_param(stmt, 2, datetime.datetime.now().strftime('%Y-%m-%d %H:%M:%S'))
ibm_db.bind_param(stmt, 3, foodname)
ibm_db.bind_param(stmt, 4, ingredients)
```

```python
            ibm_db.bind_param(stmt, 5, nutritionValues)
            ibm_db.execute(stmt)

            # os.remove(os.path.join(app.config['UPLOAD_FOLDER'], filename))
            return render_template("dashboard.html",
                filename = filename,
                username = session['username'],
                foodname = foodname,
                ingredients = ingredients,
                nutritionValues = nutritionValues,
            )
        else:
            flash('Allowed image formats - png, jpg, jpeg')
            return redirect(flask.request.url)

    elif session['LoggedIn']:
        return render_template("dashboard.html", username=session['username'])
    else:
        return redirect(url_for("login"))
```

7.2 FEATURE 2

In Nutrition Assistant Application, The Index page have two specified button for login and register. In Registration Page, It has a text box for getting user necessary details for register to web app. Necessary details are user email id, user name, password. This page having Register button to register the user input. In Login page, It has a text box for entering the registered input of user to enter into dashboard of web app. This page also having the forgotten password option for registered user of web app which is used to change the password by entering registered email id.

Python Code:

```
def sendMail(to, title, text):
  sg = sendgrid.SendGridAPIClient(api_key=SENDGRID_API_KEY)
  from_email = Email("nsnandhaa1@gmail.com")
  to_email = To(to)
  subject = title
  content = Content("text/plain", text)
  mail = Mail(from_email, to_email, subject, content)
  response = sg.client.mail.send.post(request_body=mail.get())
  print(response.status_code)
  print(response.body)
  print(response.headers)


@app.route("/forgot-pw", methods=["GET", "POST"])
def forgotpw():
  if flask.request.method == "POST":
    data = flask.request.form
    username=data['username']
    code = ''.join(random.choices(string.ascii_letters, k=6))
    sql= "SELECT * FROM users WHERE username=?"
    stmt=ibm_db.prepare(conn,sql)
    ibm_db.bind_param(stmt,1,username)
    ibm_db.execute(stmt)
    account=ibm_db.fetch_assoc(stmt)
    print(account)
    session['userid'] = account['USERID']
    insert_sql = "INSERT INTO VERIFY VALUES(?,?)"
    prep_stmt=ibm_db.prepare(conn, insert_sql)
    ibm_db.bind_param(prep_stmt, 1, account['USERID'])
    ibm_db.bind_param(prep_stmt, 2, code)
    ibm_db.execute(prep_stmt)
    sendMail(account['EMAIL'], "Verification Code", code)
    flash("We have sent a code to your registered email. please check spam folder also.")
```

```python
    return redirect(url_for("confirmMail"))
  flash("We will send you a confirmation code to your registered email")
  return render_template("forgot-pw.html")


@app.route("/confirm-mail", methods=["GET", "POST"])
def confirmMail():
  session['LoggedIn'] = False
  if flask.request.method == "POST":
    data = flask.request.form
    usercode=data['code']
    sql= "SELECT * FROM verify WHERE userid=?"
    stmt=ibm_db.prepare(conn,sql)
    ibm_db.bind_param(stmt,1,session['userid'])
    ibm_db.execute(stmt)
    verify=ibm_db.fetch_assoc(stmt)
    dbcode = verify['CODE']
    if usercode == dbcode:
      session['LoggedIn'] = True
      delete_sql = "DELETE FROM verify WHERE CODE=?"
      prep_stmt=ibm_db.prepare(conn, delete_sql)
      ibm_db.bind_param(prep_stmt, 1, dbcode)
      ibm_db.execute(prep_stmt)
      flash("Email verified. Enter new password")
      return redirect(url_for("changepw"))
    else:
      flash("Error")
      return render_template("confirm-mail")
  return render_template("confirm-mail.html")


@app.route("/change-pw", methods=["GET", "POST"])
def changepw():
  if flask.request.method == "POST" and session['LoggedIn']:
    data = flask.request.form
    password=data['pw']
    sql = "UPDATE users SET PASSWORD=? WHERE USERID=?"
    prep_stmt=ibm_db.prepare(conn, sql)
    print(password, session['userid'])
    ibm_db.bind_param(prep_stmt, 1, password)
    ibm_db.bind_param(prep_stmt, 2, session['userid'])
    ibm_db.execute(prep_stmt)
    flash("Password changed.")
```

```python
      return redirect(url_for("login"))
    else:
      flash("verification error")
      redirect(url_for("confirmMail"))
    return render_template("change-pw.html")


@app.route("/register", methods=["GET", "POST"])
def reg():
  if flask.request.method == "POST":
    data = flask.request.form
    email=data['email']
    username=data['username']
    password=data['pw']
    sql= "SELECT * FROM users WHERE username=?"
    stmt=ibm_db.prepare(conn,sql)
    ibm_db.bind_param(stmt,1,username)
    ibm_db.execute(stmt)
    account=ibm_db.fetch_assoc(stmt)
    print(account)
    if account:
      flash("Account already exists!")
    elif not re.match(r'[^@]+@[^@]+\.[^@]+', email):
      flash("invalid email address")
    elif not re.match(r'[A-Za-z0-9]+', username):
      flash("name must contain only characters and numbers")
    else:
      insert_sql = "INSERT INTO users VALUES(?,?,?,?)"
      prep_stmt=ibm_db.prepare(conn, insert_sql)
      ibm_db.bind_param(prep_stmt, 1, username)
      ibm_db.bind_param(prep_stmt, 2, email)
      ibm_db.bind_param(prep_stmt, 3, password)
      ibm_db.bind_param(prep_stmt, 4, ''.join(random.choices(string.ascii_letters, k=16)))
      ibm_db.execute(prep_stmt)
      flash("logged in")
    return redirect(url_for("dashboard"))
  return render_template("reg.html")


@app.route("/login", methods=["GET", "POST"])
def login():
  if flask.request.method == "POST":
    data = flask.request.form
```

```python
username=data['username']
password=data['pw']
sql = "SELECT * FROM users WHERE username=? AND password=?"
stmt = ibm_db.prepare(conn,sql)
ibm_db.bind_param(stmt, 1, username)
ibm_db.bind_param(stmt, 2, password)
ibm_db.execute(stmt)
account = ibm_db.fetch_assoc(stmt)
print(account)
if account:
    session['LoggedIn'] = True
    session['userid'] = account['USERID']
    session['username'] = account['USERNAME']
    userid = account['USERID']
    flash("logged in")
    return redirect(url_for("dashboard"))
else:
    flash("error")
return render_template("login.html")
```

7. TESTING

8.1 TEST CASES

A test case has components that describe input, action and an expected response, in order to determine if a feature of an application is working correctly. A test case is a set of instructions on "HOW" to validate a particular test objective/target, which when followed will tell us if the expected behavior of the system is satisfied or not.
Characteristics of a good test case:

- Accurate: Exacts the purpose.
- Economical: No unnecessary steps or words.
- Traceable: Capable of being traced to requirements.
- Repeatable: Can be used to perform the test over and over.
- Reusable: Can be reused if necessary.

S.NO	Scenario	Input	Output
1	User Registration	User Name, Email id, Password, Confirmation Password	Registeration success
2	User Login	Username, , Password	Login success
3	Forgot Password (If password is forgotten)	User email id, New password	New password is updated
4	Upload Food image	Food image	Uploaded food image in dashboard.
5	Analyse the food	Food image	Predicted ingredients values and nutritional values of given food image are shown in dashboard of user.

8.2 USER ACCEPTANCE TESTING

This sort of testing is carried out by users, clients, or other authorised bodies to identify the requirements and operational procedures of an application or piece of software. The most crucial stage of testing is acceptance testing since it determines whether or not the customer will accept the application or programme. It could entail the application's U.I., performance, usability, and usefulness. It is also referred to as end-user testing, operational acceptance testing, and user acceptance testing (UAT).

8. RESULTS

9.1 PERFORMANCE METRICS
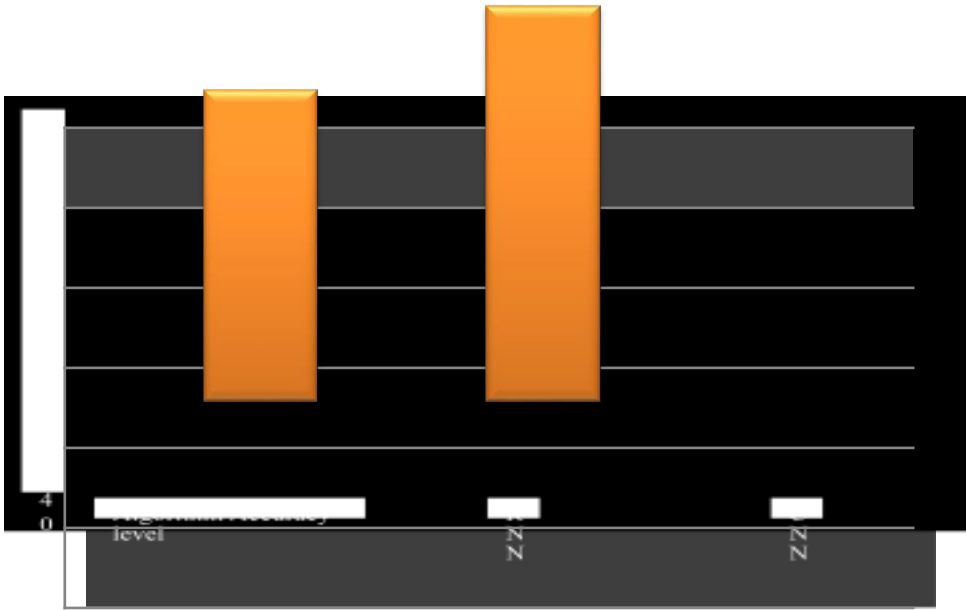
9. ADVANTAGES & DISADVANTAGES

ADVANTAGE

- Provide the nutrition content and ingredients of Multifoods
- Helps for fitness people to maintain and know the proteins and calories of the food
- Gives accurate results in real-time application

DISADVANTAGE

- Hard to know the details of nutrition and calories of food
- Doesn't ask to provide the users health condition
- Required more time to know the Multifoods

10. CONCLUSION

This project study suggests a technique for an automated food nutrition detection system that can estimate the quantity of nutrients in food. As of now, the machine can classify the meal into one of the numerous categories provided in the dataset. The categorization made use of the popular food dataset. Using a deep learning approach, the food photographs were categorised into the proper categories. It may be possible to improve the classification process by removing noise from the dataset. The same study can be conducted using a larger dataset, more classes, and more images in each class because doing so improves accuracy by teaching the algorithm new features and reduces loss rates. The weights of the model can be stored and used to develop designs for calorie extraction, food categorization, and image classification.

\

11. FUTURE SCOPE

Using a deep learning approach, the research study's food images are categorised into the relevant groupings. The classification task might be improved in the future by removing noise from the dataset. The same study may be conducted using a larger dataset, more classes, and more images in each class because doing so improves accuracy by teaching the algorithm new characteristics and reduces loss rates. A web or mobile application that categorises images and also extracts the calories from the food that has been identified may be created using the model's weights, which can be saved and used later.

12. APPENDIX

SOURCE CODE

https://github.com/IBM-EPBL/IBM-Project-41013-1660638581/tree/main/Final%20Deliverables/Code

DEMO LINK

https://drive.google.com/file/d/10-URH48XKv0ZT1I8L_rItFDMqfZyoCDb/view?usp=share_link

SCREENSHOTS:

INDEX PAGE

Nutrition Assistant

Login Register

REGISTER PAGE

Registeration

Email

Username

Password

Register

Login

LOGIN PAGE

Login

Username

Password

Login

Register

Forgot Password

DASHBOARD

Nutrition Assistant   Baskaraboopathy P   Logout

- logged in

Upload food image

Choose File   No file chosen

Analyze

Food details

Food Name:

Ingredients:

Nutrition Values:

- Image successfully uploaded

Upload food image

Upload Another Image

Food details

Food Name: rice

Ingredients: rice: 1.0, meat: 1.0, vegetable: 0.99, chicken: 0.99, pepper: 0.99, pork: 0.99, pilaf: 0.98, sauce: 0.94, fried rice: 0.94, risotto: 0.85, beef: 0.77, tomato: 0.52, chili: 0.48, onion: 0.43, cereal: 0.42, mutton: 0.39, saffron: 0.37, chicken curry: 0.35, curry: 0.35, paella: 0.26,

Nutrition Values: calories: 470 calories, fat: 17 g, protein: 15 g, carbs: 65 g,

RESPONSIVE IN ALL DISPLAY

Upload Another Image

Food details

Food Name: rice

Ingredients: rice: 0.94, egg: 0.94, pork: 0.83, chicken: 0.81, beef: 0.79, meat: 0.78, sweet: 0.74, ramen: 0.71, chocolate: 0.66, quinoa: 0.66, banana: 0.64, corn: 0.6, fried rice: 0.59, vegetable: 0.59, cake: 0.58, coconut: 0.57, sauce: 0.49, onion: 0.44, beans: 0.43, ground beef: 0.42,