



NUTRITION ASSISTANT APPLICATION

NALAIYA THIRAN - PROJECT REPORT

PROJECT ID: PNT2022TMID00797

Submitted by

**KONDAPANENI SIVA [211419104141]
ROHIT**

NARENDHIRAN S [211419104176]

NARESH KUMAR D [211419104178]

KARJALA VENKATESH [211419104121]

In partial fulfillment for the award of the degree

of

BACHELOR OF ENGINEERING

IN

COMPUTER SCIENCE AND ENGINEERING

PANIMALAR ENGINEERING COLLEGE, CHENNAI-600123.

(AN AUTONOMOUS INSTITUTION , AFFILIATED TO ANNA UNIVERSITY)

NOVEMBER 2022

PANIMALAR ENGINEERING COLLEGE, CHENNAI-600123.

(AN AUTONOMOUS INSTITUTION , AFFILIATED TO ANNA UNIVERSITY)

BONAFIDE CERTIFICATE

Certified that this project report

“NUTRITION ASSISTANT APPLICATION– PNT2022TMID00797”

is the bonafide work of

**KONDAPANENI SIVA [211419104141]
ROHIT**

NARENDHIRAN S [211419104176]

NARESH KUMAR D [211419104178]

KARJALA VENKATESH [211419104121]

who carried out the NALAIYA THIRAN project work under the supervision.

**SAI PRIYA
INDUSTRY MENTOR
IBM**

**VINMATHI M S
FACULTY MENTOR
Department of CSE
Panimalar Engineering College**

Contents

1. INTRODUCTION.....	2
1.1 Project Overview	3
1.2 Purpose.....	4
2. LITERATURE SURVEY	5
2.1 Existing problem	6
2.2 References.....	7
2.3 Problem Statement Definition	8
3. IDEATION & PROPOSED SOLUTION.....	9
3.1 Empathy Map Canvas	10
3.2 Ideation & Brainstorming	11
3.3 Proposed Solution	12
3.4 Problem Solution fit	13
4. REQUIREMENT ANALYSIS	14
4.1 Functional requirement.....	15
4.2 Non-Functional requirements.....	16
5. PROJECT DESIGN	17
5.1 Data Flow Diagrams	18
5.2 Solution & Technical Architecture	19
5.3 User Stories.....	20
6. PROJECT PLANNING & SCHEDULING	21
6.1 Sprint Planning & Estimation	22
6.2 Sprint Delivery Schedule	24
7. CODING & SOLUTIONING	25
7.1 Feature	26
8. TESTING.....	27
8.1 Test Cases	28
8.2 User Acceptance Testing.....	29
9. RESULTS	30
9.1 Performance Metrics.....	31
10. ADVANTAGES & DISADVANTAGES	33
10.1 Advantages:.....	34
10.2 Disadvantages:	34
11. CONCLUSION	35
11.1 Conclusion	36
12. FUTURE SCOPE.....	37
12.1 Future scope	38
13. APPENDIX.....	39
13.1 Source Code	40
13.2 Appendix.....	53

1. INTRODUCTION

1.1 Project Overview

Better health can be achieved by maintaining a simple lifestyle such as a good night sleep, enough exercises and good nutrition. People spend one third of their lives sleeping however most people do not understand the importance of sleep. Moreover, the lack of sleep can affect a person's memory and emotion. The exercise habit and nutrition can also lead to good health. Daily working life can be affected by lack of sleep such as drowsiness and long-term health problems. Many Researches have shown that not enough sleep or exercise can lead to many health problems such as GERD [3], Alzheimer's disease [4], heart disease [5], sleep apnea [6] and insomnia [7]. In the competitive work environment, nowadays, it is not easy for many people to manage good sleeping and exercise habits. With busy work and personal life schedules, many people indulge themselves in a bad sleeping habit such as sleeping very late or waking up very late especially in young adults and teenagers. A good night sleep can also be affected by the person's exercise habits and nutrition consumption. The sleep, nutrition and exercise have more complicated relationship than many people have realized [8]. Enough exercise helps people sleep better and good nutrition also lead to better mood and better health. Thus, that can automatically record personal information, produce a warning, and give personal advices to its owner in order to maintain good sleeping, exercise and nutrition habits is needed. Today smart phone technology is a good candidate for this project because of its low cost, portability and capability which is similar or close to a personal computer. Moreover, a phone has become a typical device in daily activity. In addition, a current smart phone includes a lot of sensors such as an accelerometer, a microphone and a light sensor. These features make a smart phone suitable for collecting personal data in this work. Android is a Linux-based operating designed for touch screen mobile devices. Lately, Android becomes the world's most widely used smart phone platform [9]. Especially, its customizable features allow Androids to be the software of choice for many developers. Thus, this project aims to develop an application for Android smart phones that has a capability (1) to record the information related to the duration and quality of sleeping, the types and duration. Of exercise activities and the amount of nutrition consumed, (2) to analyse the collected data and provide a notification or an alarm in order to suggest or remind the user in taking care of his/her health; and (3) to present the analysed results in a format that is easy to understand without the need of a deep medical knowledge. The rest of this document is organized as follows. Next, the design is given.

1.2 Purpose

By using machine learning techniques, the problem can be solved with minimal error rate. The voice dataset of Parkinson's disease from the UCI Machine learning library is used as input. Also our proposed system provides accurate results by integrating spiral drawing inputs of normal and Parkinson's affected patients. We propose a hybrid and accurate results analyzing patient both voice and spiral drawing data's. Thus combining both the results, the doctor can conclude normality or abnormality and prescribe the medicine based on the affected stage.

2.LITERATURE SURVEY

2.1 Existing problem

Jie Mei et al used all basic algorithms of deep learning techniques for the detection of PD. Like SVM, RF, Decision Tree, ANN, KNN, Radial Basis Function Networks (RBF) and Deep Belief Networks (DBN) etc. The early identification of Parkinson's disease is critical. The identification can be performed with the use of a data mining technique. The techniques for detecting PD, such as Naive Bayes, support vector machine, multilayer perceptron neural network, and decision tree, are theoretically explained in this study. This study uses speech input from acoustic devices to predict Parkinson's disease. People from various areas and speech factors are investigated in this article in order to predict Parkinson's disease among patients. The speech dataset was used to recognize Parkinson's illness using Multi-layer Perceptron and Logistic Regression(LR) frameworks.

Gabriel Solana-Lavalle et al. uses the algorithms such as Multilayer Perceptron (MLP), Random Forest (RF), K-Nearest Neighbour (KNN). For the prediction of Parkinson disease, three set of experiences were conducted to obtain the features with highest contribution to PD. This three sets are 1.a population with male and female subjects (balanced), 2.male subjects (balanced and unbalanced), and 3. Female subjects (balanced and unbalanced). In this study, the researchers used acoustic devices to collect speech parameters from 50 persons with Parkinson's disease and fifty healthy people. They employed the kfold cross validation method for testing and claim that it can deliver 85 percent accuracy.

Yi Xia et al. they have considered approaches, they include four DL-based models (DCNN, DALSTM, DCLSTM, and CNN-LSTM) and also used two traditional classifications for extraction. In the DL-based model DCNN gives less accuracy than other DL models. Parkinson's disease affects people all around the world. People and people with Parkinson's disease could be classified using machine learning approach. This paper provides a comprehensive overview of machine learning-based approaches for Parkinson disease prediction. A comprehensive overview of various computational system-based techniques for Parkinson disease prediction is presented. This report also includes an overview of the results obtained by several scientists from publicly available data in order to forecast Parkinson's disease.

Rahul R. Chakre et al. According to the hybrid approach, which is a combination of supervised and unsupervised techniques, is also beneficial for classification and feature extraction. Support vector machine is employed as the supervised technique for classification, and ICA is used as unsupervised technique for the feature extraction in multiclass data set.

Rahul Ramesh Chakre et al. According to the field of medical diagnosis, bioinspired computing is also a novel technique. Swarm intelligence and immune computing algorithms, two major subsets of bio-inspired computation, are presented for a wide range of issues.

2.2 References

- [1] Machine learning models for mathematical symbol recognition: A stem to stern literature analysis - Vinay Kukreja & Sakshi**

- [2] A Novel Method for the Recognition of Isolated Handwritten Arabic Characters - Ahmed Sahlol, Cheng Suen**

- [3] A novel method for offline handwriting-based writer identification - Zhenyu He; Bin Fang; Jianwei Du; Yuan Yan Tang; Xinge You**

- [4] A Novel Method for Recognition of Persian Alphabet by Using Fuzzy NeuralNetwork - Mohammad Mehdi Motahari Kia; Jafar A. Alzubi; Mehdi Gheisari; Xiaobo Zhang; Mohamadtaghi Rahimi; Yongrui Qin.**

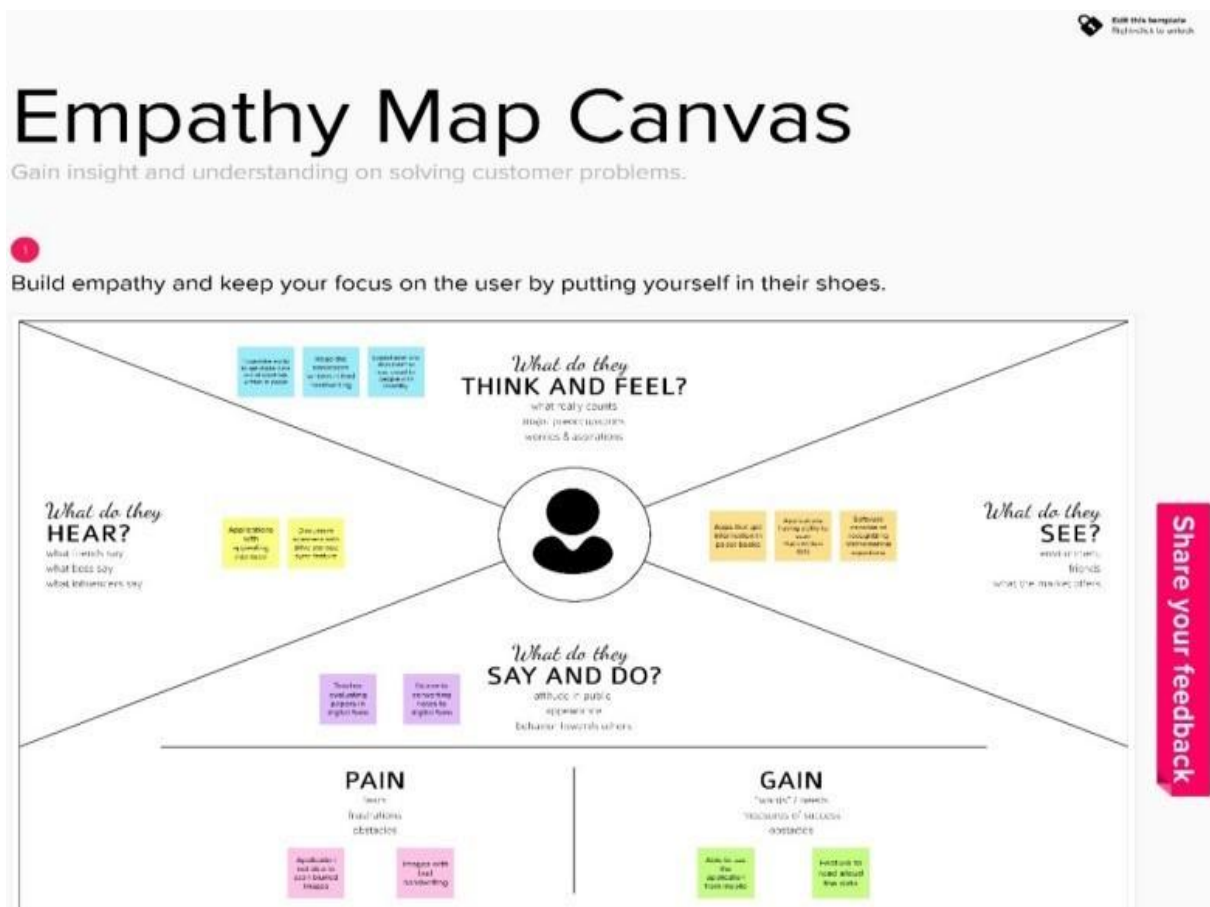
2.3 Problem Statement Definition

Obesity is one of the biggest health problems associated with junk food and People want 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.

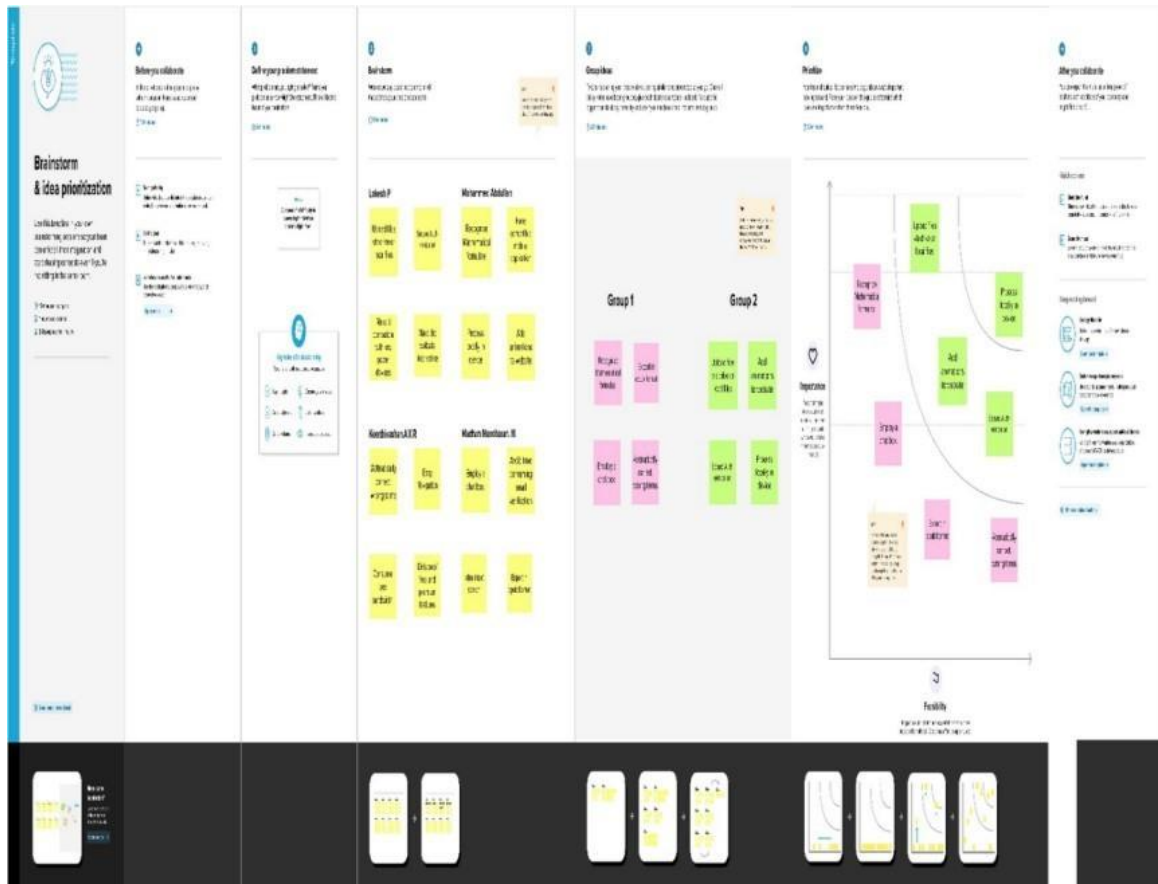
3. IDEATION & PROPOSED SOLUTION

3.1 Empathy Map Canvas

Nutrition Assistant Application



3.2 Ideation & Brainstorming



3.3 Proposed Solution

S. No.	Parameter	Description
1.	Problem Statemnt	The handwritten digit recognition is the capability of computer applications to recognize the human handwritten digits. It is a Hard task for the machine because handwritten digits are not perfect and can be made with many different shapes and sizes. The handwritten digit recognition system is a way to tackle this problem which uses the image of a digit and recognizes the digit present in the image.
2.	Idea /Solution description	We came up with a solution that recognizes the handwritten digits by means of a deep learning model.
3.	Novelty Uniqueness	The system recognizes input given by userin a precise and efficient manner.
4.	Social Impact / Customer	The aim of a handwriting digit recognition system is to convert handwritten digits into machine readable formats. The main objective of this work is to ensure effective and reliable approaches for recognition of handwritten digits.
5.	Business Model	Pay per use – each time a person needs the service he can avail it by paying forthe use.
6.	Scalability of the Solution	It can be implemented using any we framework and can be made available to everyone in need.

3.4 Problem Solution fit

Problem-Solution fit canvas 2.0 Purpose / Vision

Define CS, fit into CC	1. CUSTOMER SEGMENT(S) CS It is useful for <ul style="list-style-type: none"> Children to understand the digits Person who are at industry side for recognising various handwriting digits. People working in bank, post offices 	6. CUSTOMER CONSTRAINTS CC <ul style="list-style-type: none"> Time Accuracy Ease to access Imperfect findings 	5. AVAILABLE SOLUTIONS AS <ul style="list-style-type: none"> In past they get trouble in finding handwritten digits Using this system, they can resolve this type of problems Pros of this system is quick recognition and Accurate prediction Cons are network connection is mandatory for using this system For using this system knowledge about the system is required 	Explore AS, differentiate
	2. JOBS-TO-BE-DONE / PROBLEMS J&P There are different types of handwriting are in world. Each and every handwriting has its own characteristics and uniqueness. Its difficult to understand the different people's handwriting digit.	9. PROBLEM ROOT CAUSE RC <ul style="list-style-type: none"> Not everyone can understand everyone's handwriting The handwriting is differed from person to person So, it is difficult to recognise the digits To solve this problem this system has developed 	7. BEHAVIOUR BE To address the problem, they can take a snap of the handwritten digit and upload it in the software.	
Identify strong TR & EM	3. TRIGGERS TR <ul style="list-style-type: none"> By word of mouth Good user experience 	10. YOUR SOLUTION SL <ul style="list-style-type: none"> A novel method for handwritten digit recognition system helps in recognizing the handwritten digits that uses MNIST dataset for training the model. The model gets the image of the handwritten digits and recognizes the handwritten digits. CNN algorithm is used over the MNIST dataset to recognise the handwritten digits. 	8. CHANNELS of BEHAVIOUR CH 8.1 ONLINE In online they can upload the handwritten picture and yield output 8.2 OFFLINE In offline they can ask their neighbors to scribe the digits to find them	Extract online & offline CH of BE
	4. EMOTIONS: BEFORE / AFTER EM <ul style="list-style-type: none"> It is a quite irritating and frustrating while manually convert the handwritten digits By using our system, user can save the time and reduce the error occur on recognition 			

 Problem-Solution fit canvas is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 license
 Created by Daria Negrahina / Amaltama.com



4. REQUIREMENT ANALYSIS

4.1 Functional requirement

FR No.	Functional Requirement(Epic)	Sub Requirement (Story / Sub-Task)
FR-1	User Registration	It is necessary to register
FR-2	Uploading the image	Please upload a handwritten digitimage in the format provided
FR-3	Using a web browser	Digit recognition requires a desktopor mobile browse

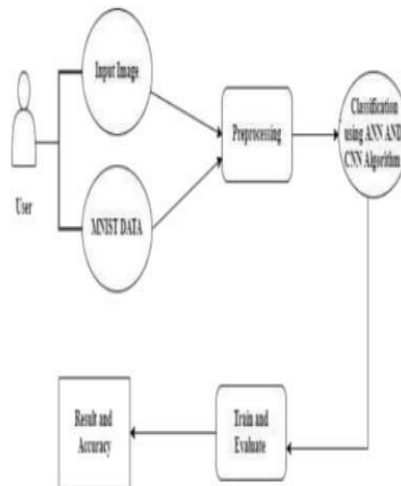
4.2 Non-Functional requirements

FR No.	Non-Functional Requirement	Description
NFR-1	Usability	Handwritten recognition digits are easy to get and simple to understand.
NFR-2	Security	Our application does not take any security measures.
NFR-3	Reliability	Be able to endure the long periods of time without errors.
NFR-4	Performance	The performance of a lightweight application.

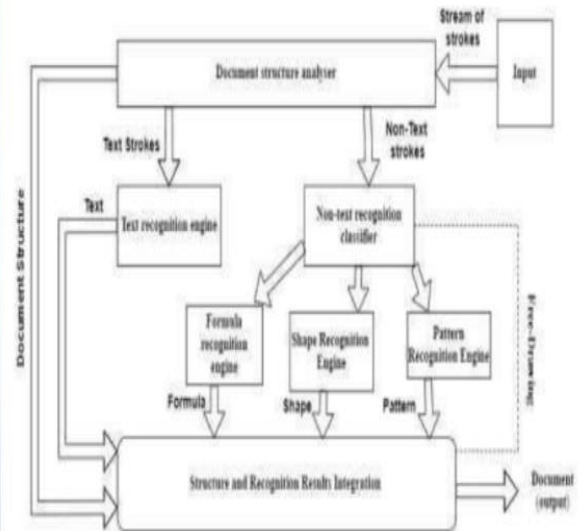
5. PROJECT DESIGN

5.1 Data Flow Diagrams

Simplified:



Example: DFD Level 0 (Industry Standard)

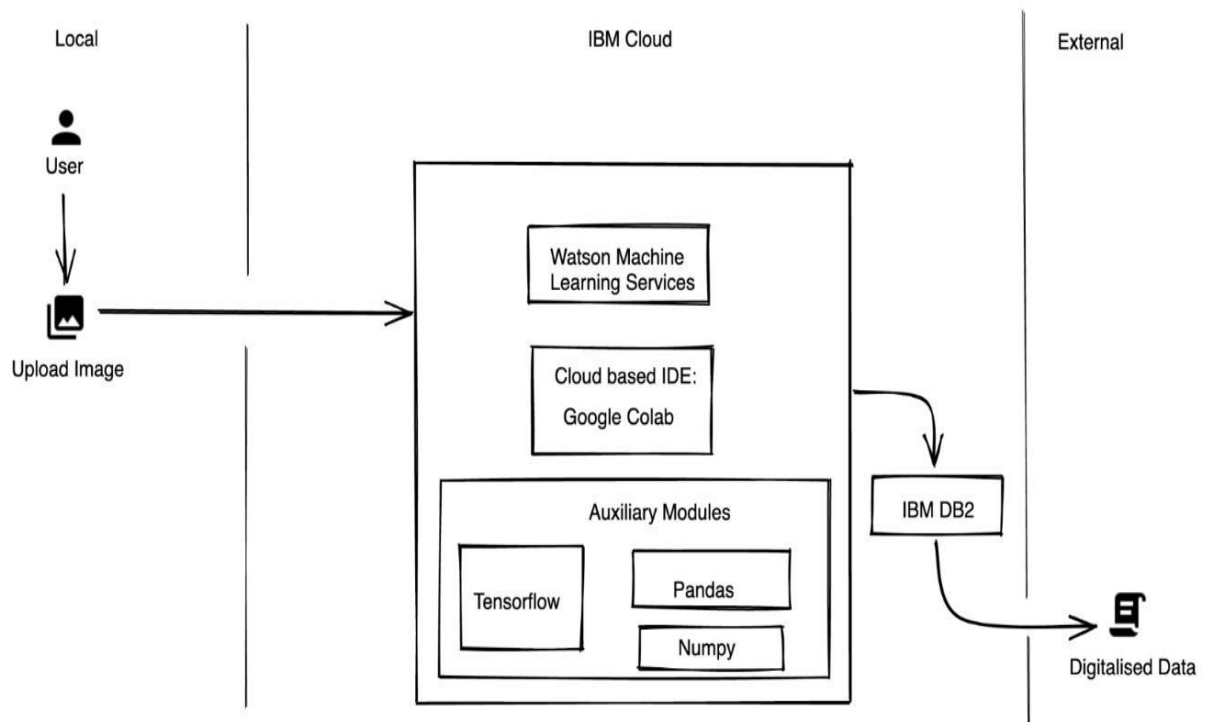


5.2 Solution & Technical Architecture

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.

Example - Solution Architecture Diagram :



5.3 User Stories

To list all the user stories for the product.

User Type	Functional Requirement(Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority	Release
User (subject that submits the photograph)	Upload image	USN-1	Users can upload the pictures to the website to achieve a desired result	We can upload an image	High	Sprint-1
Examination	Image process	USN-1	Users may examine the projections and outcomes with accuracy.	We can get results instantly.	High	Sprint-1

6. PROJECT PLANNING & SCHEDULING

6.1 Sprint Planning & Estimation

Product Backlog, Sprint Schedule, and Estimation

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-1	Data Collection	USN-1	As a user, the dataset can be collected from a variety of sources with different handwritings.	10	Low	Siva Rohit
Sprint-1	Data Preprocessing	USN-2	As a user, I can Load the dataset, handling the missing data scaling split data into train and test.	10	Medium	Naredhiran
Sprint-2	Model Building	USN-3	As a user, I will get an application which provides high accuracy of recognized handwritten digit.	5	High	Naresh Kumar
Sprint-2	Add CNN layersx	USN-4	Creating the model and adding the input, hidden, and output layers to it.	5	High	Siva Rohit, Venkatesh, Narendhiran, Naresh

Velocity:

Imagine we have a 6-day sprint duration, and the velocity of the team is 20 (points per sprint). Let's calculate the team's average velocity (AV) per iteration unit (story points per day)

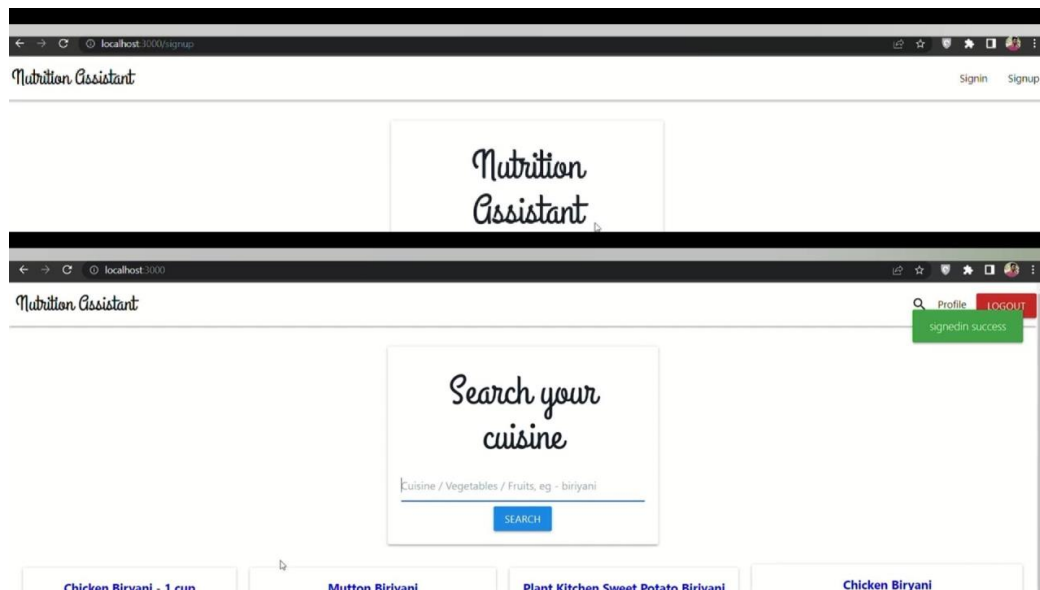
$$AV = (\text{Sprint Duration} / \text{Velocity}) = 20/6 = 3.33$$

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	29 Oct 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	12 Nov 2022
Sprint-4	20	6 Days	14 Nov 2022	19 Nov 2022	20	19 Nov 2022

7. CODING & SOLUTIONING

7.1 Feature



8. TESTING

8.1 Test Cases

Test Case No.	Action	Expected Output	Actual Output	Result
1	Entering details in sign up page	Sign up button is enable	Sign up button is enable	Pass
2	Entering details in sign up page	Profile pic button is enable	Profile pic button is enable	Pass
3	Entering details in sign up page	Already have an account Button is enable	Already have an account Button is enable	Pass
4	Entering details in login page	Login button enable	Login button enable	Pass
5	Entering details in login page	Do you have account button enable	Do you have account button enable	Pass
6	Watson chat bot	Entering queries enable	Entering queries enable	Pass
7	Watson chatbot	Automated replies enable	Automated replies enable	Pass
8	Entering wrong login details	Invalid reply	Invalid reply	Pass
9	Entering wrong password	Invalid reply	Invalid reply	Pass

8.2 User Acceptance Testing

Purpose of Document:

The purpose of this document is to briefly explain the test coverage and open issues of the Nutrition Assistant Application project at the time of the release to User Acceptance Testing (UAT).

Defect Analysis:

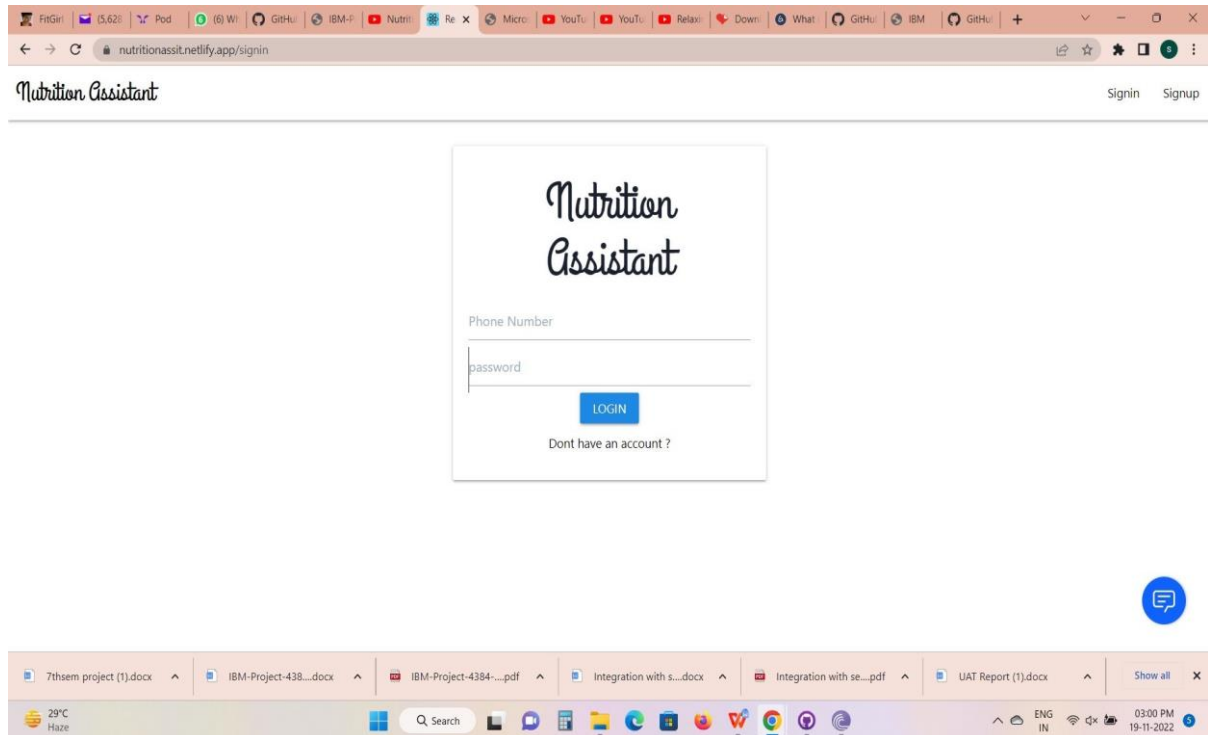
This report shows the number of resolved or closed bugs at each severity level, and how they were resolved.

Resolution	Severity 1	Severity 2	Severity 3	Severity 4	Subtotal
By Design	5	4	6	5	20
Duplicate	0	1	3	0	4
External	1	1	0	4	6
Fixed	2	5	20	10	37
Not Reproduced	0	0	1	0	1
Skipped	0	0	1	1	2
Won't Fix	0	5	2	1	8
Totals	8	16	33	21	77

9. RESULTS

9.1 Performance Metrics

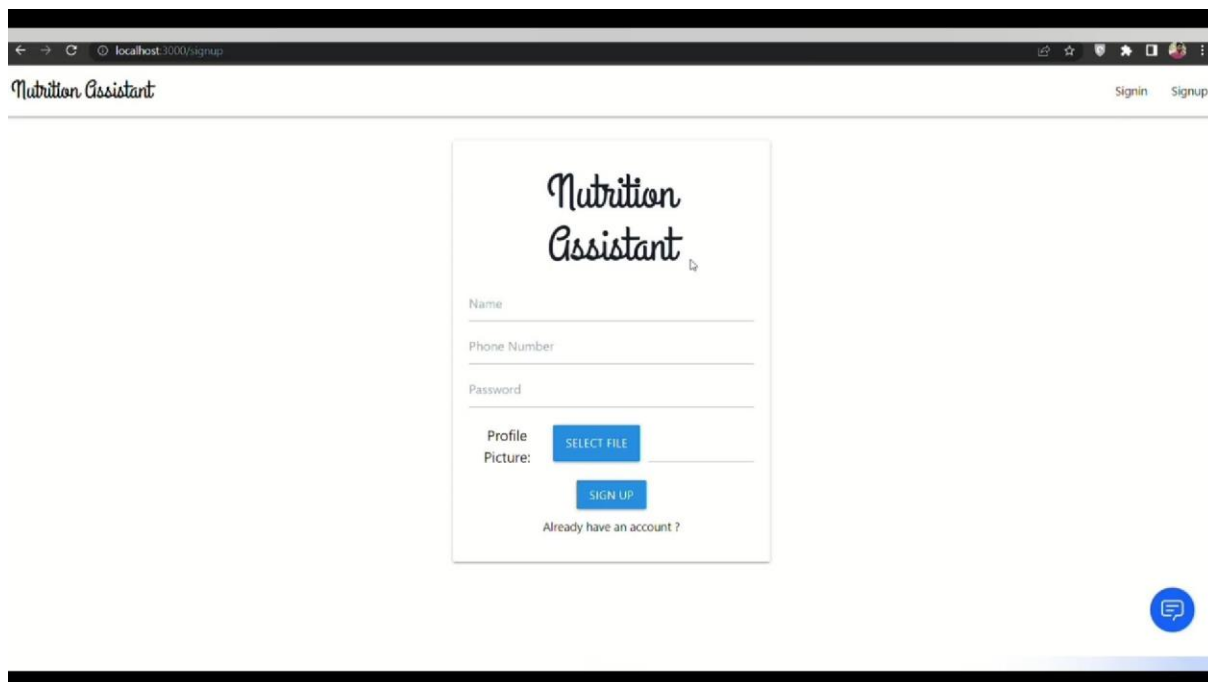
RESULTS:



The screenshot shows a web browser window with the address bar displaying "nutritionassit.netlify.app/signin". The page title is "Nutrition Assistant". The login form is centered and contains the following elements:

- Phone Number input field
- password input field
- LOGIN button
- Link: "Dont have an account ?"

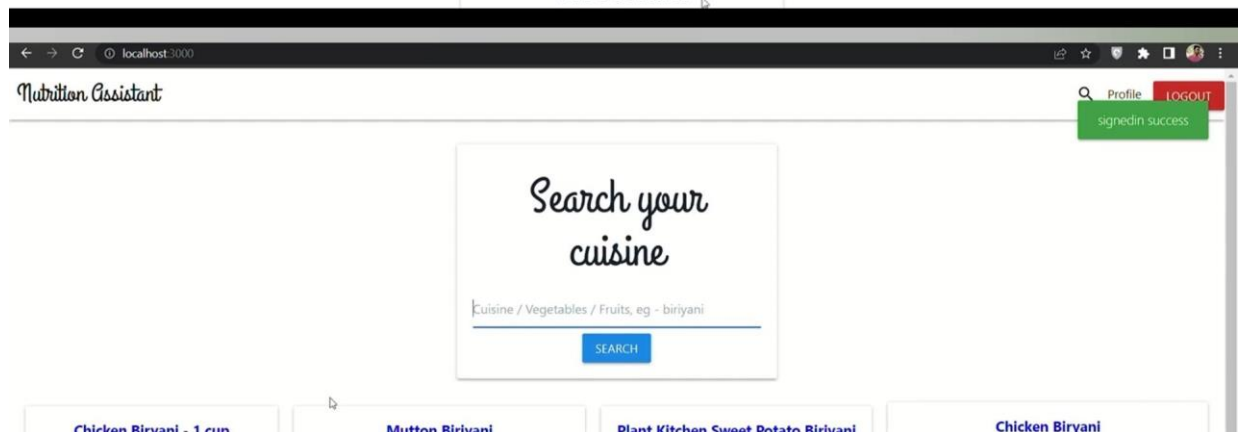
The browser's taskbar at the bottom shows several open documents, including "7thsem project (1).docx", "IBM-Project-438....docx", "IBM-Project-4384-....pdf", "Integration with s....docx", "Integration with se...pdf", and "UAT Report (1).docx". The system tray shows the date and time as "03:00 PM 19-11-2022".



The screenshot shows a web browser window with the address bar displaying "localhost:3000/signup". The page title is "Nutrition Assistant". The signup form is centered and contains the following elements:

- Name input field
- Phone Number input field
- Password input field
- Profile Picture input field with a "SELECT FILE" button
- SIGN UP button
- Link: "Already have an account ?"

The browser's taskbar at the bottom shows several open documents, including "7thsem project (1).docx", "IBM-Project-438....docx", "IBM-Project-4384-....pdf", "Integration with s....docx", "Integration with se...pdf", and "UAT Report (1).docx". The system tray shows the date and time as "03:00 PM 19-11-2022".



10. ADVANTAGES & DISADVANTAGES

10.1 Advantages:

- **Works Under Low Data Connection.**
- **Low Energy Consumption.**
- **User Friendly Web Application.**
- **Data Privacy.**
- **Easy to Understand.**

10.2 Disadvantages:

- **It Cannot be Used Without Internet Connection.**
- **Usage of 3rd party API may cause the time delay.**

11. CONCLUSION

11.1 Conclusion

Nutritional support is the provision of adequate nutrients to maintain a healthy body weight and avoid malnutrition. The continuous delivery of high-quality and cost-effective nutritional care to patients has been shown to be an increasingly difficult task. We developed a cloud based nutrition application which detects the nutrition in food. It clarifies the calories in the food which affects our health. It is observed that dietitians are requested to carry out the nutritional assessment, to manually calculate the nutritional needs and to design the everyday meal plan for each patient. In most cases, these time-consuming tasks are not completed due to lack of time or inadequate number of a person.

12. FUTURE SCOPE

12.1 Future scope

Project scope is a way to set boundaries on your project and define exactly what goals, deadlines, and project deliverables you'll be working towards. By clarifying your project scope, you can ensure you hit your project goals and objectives without delay or overwork. Defining your project scope isn't a one-person job.

You can work as a Nutritionist/Dietitian there and take control of the food intake and also the food quality consumed by the people. With a degree in food and nutrition, you can act as a Public

Health Nutritionist in non governmental organizations and play your part in spreading some good in the world.

Future Scope is for the Undergraduates, Graduates and the Working Professionals. They may want to review or reconsider their future options and goals in terms of its suitability now; may be with a different perspective of their options in terms of time, resources, inclination etc.

13. APPENDIX

13.1 Source Code

```
import React, { useState, useContext } from "react";
import { Link, useHistory } from "react-router-dom";
import { UserContext } from "../App";
import M from "materialize-css";

const SignIn = () => {
  const { state, dispatch } = useContext(UserContext);
  const history = useHistory();
  const [password, setPassword] = useState("");
  const [email, setEmail] = useState("");
  const postData = () => {
    if (!email) {
      M.toast({ html: "Invalid Phone Number", classes: "#c62828 red darken-3"
    });
      return;
    }
    fetch("/signin", {
      method: "post",
      headers: {
        "Content-Type": "application/json",
      },
      body: JSON.stringify({
        password,
        phoneNumber: email,
      }),
    })
      .then((res) => res.json())
      .then((data) => {
        console.log(data);
        if (data.error) {
          M.toast({
            html: data.error,
            classes: "#c62828 red darken-3",
          });
        } else {
          localStorage.setItem("jwt", data.token);
          localStorage.setItem("user", JSON.stringify(data.user));
          dispatch({ type: "USER", payload: data.user });
          M.toast({
            html: "signedin success",
            classes: "#43a047 green darken-1",
          });
          history.push("/");
        }
      })
      .catch((err) => {
        console.log(err);
      });
  };
};
```

```

    };
    return (
      <div className="mycard">
        <div className="card auth-card input-field">
          {/*  */}
          <h2 style={{ fontSize: 60, fontWeight: "500", margin: 15 }}>
            Nutrition Assistant
          </h2>
          <input
            type="number"
            placeholder="Phone Number"
            value={email}
            onChange={(e) => setEmail(e.target.value)}
          />
          <input
            type="password"
            placeholder="password"
            value={password}
            onChange={(e) => setPasword(e.target.value)}
          />
          <button
            className="btn waves-effect waves-light #64b5f6 blue
darken-1"
            onClick={() => PostData()}
          >
            Login
          </button>
          <h5 style={{ margin: 15 }}>
            <Link to="/signup">Dont have an account ?</Link>
          </h5>
        </div>
      </div>
    );
  };

```

```

export default SignIn;
import React, { useState, useEffect } from "react";
import { Link, useHistory } from "react-router-dom";
import M from "materialize-css";
import { storage } from "../utils/firebase";

```

```

const SignIn = () => {
  const history = useHistory();
  const [name, setName] = useState("");
  const [password, setPasword] = useState("");
  const [email, setEmail] = useState("");
  const [image, setImage] = useState("");
  const [url, setUrl] = useState(undefined);
  const [userType, setUserType] = useState("PASSENGER");
  const [aadharImg, setAadharImg] = useState(null);
  const [licenseImg, setLicenseImg] = useState(null);

```

```

const [currRel, setCurrRel] = useState("");
const [currFamNum, setFamNum] = useState("");
const [vehicleNumber, setVehicleNumber] = useState("");
const [isSubmitDis, setIsSubmitDis] = useState(false);

useEffect(() => {
  if (url) {
    uploadFields();
  }
}, [url]);
const uploadPic = () => {
  const data = new FormData();
  data.append("file", image);
  data.append("upload_preset", "insta-abi");
  data.append("cloud_name", "insta-07");
  fetch("https://api.cloudinary.com/v1_1/insta-07/image/upload", {
    method: "post",
    body: data,
  })
    .then((res) => res.json())
    .then((data) => {
      setUrl(data.url);
    })
    .catch((err) => {
      console.log(err);
    });
};

const uploadImg = (imgData, isAadhar) => {
  setIsSubmitDis(true);
  const data = new FormData();
  data.append("file", imgData);
  data.append("upload_preset", "insta-abi");
  data.append("cloud_name", "insta-07");
  fetch("https://api.cloudinary.com/v1_1/insta-07/image/upload", {
    method: "post",
    body: data,
  })
    .then((res) => res.json())
    .then((data) => {
      if (isAadhar) {
        setAadharImg(data.url);
      } else {
        setLicenseImg(data.url);
      }
      setIsSubmitDis(false);
    })
    .catch((err) => {
      console.log(err);
      setIsSubmitDis(false);
    });
};

```

```

const uploadFields = () => {
  if (!email) {
    M.toast({
      html: "Invalid Phone Number",
      classes: "#c62828 red darken-3",
    });
    return;
  }
  console.log({ email });
  fetch("/signup", {
    method: "post",
    headers: {
      "Content-Type": "application/json",
    },
    body: JSON.stringify({
      name,
      password,
      phoneNumber: email,
      pic: url,
      userType,
      aadharImg,
      vehicleNumber,
      licenseImg,
    }),
  })
    .then((res) => res.json())
    .then((data) => {
      if (data.error) {
        M.toast({
          html: data.error,
          classes: "#c62828 red darken-3",
        });
      } else {
        M.toast({
          html: data.message,
          classes: "#43a047 green darken-1",
        });
        history.push("/signin");
      }
    })
    .catch((err) => {
      console.log(err);
    });
};

const PostData = () => {
  if (image) {
    uploadPic();
  } else {
    uploadFields();
  }
};

```

```

return (
  <div className="mycard">
    <div className="card auth-card input-field">
      {/*  */}
      <h2 style={{ fontSize: 60, fontWeight: "500", margin: 15 }}>
        Nutrition Assistant
      </h2>
      <input
        type="text"
        placeholder="Name"
        value={name}
        onChange={(e) => setName(e.target.value)}
      />
      <input
        type="number"
        placeholder="Phone Number"
        value={email}
        onChange={(e) => setEmail(e.target.value)}
      />
      <input
        type="password"
        placeholder="Password"
        value={password}
        onChange={(e) => setPasword(e.target.value)}
      />
      <div style={{ display: "flex", alignItems: "center" }}>
        <p style={{ marginRight: 10, fontSize: 18 }}>
          Profile Picture:
        </p>
        <div className="file-field input-field">
          <div className="btn #64b5f6 blue darken-1">
            <span>Select file</span>
            <input
              type="file"
              onChange={(e) =>
                setImage(e.target.files[0])
              }
            />
          </div>
          <div className="file-path-wrapper">
            <input className="file-path validate"
              type="text" />
          </div>
        </div>
      </div>
      <button
        className="btn waves-effect waves-light #64b5f6 blue
          darken-1"
        onClick={() => {
          if (!isSubmitDis) {
            PostData();
          }
        }}
      />
    </div>
  </div>
)

```

```

        }
      }}
    >
      Sign up
    </button>
    <h5 style={{ margin: 15 }}>
      <Link to="/signin">Already have an account ?</Link>
    </h5>
  </div>
</div>
);
};

```

Nutrition Assistant App:

```

import React, { useState, useContext, useEffect } from "react";
import { Link, useHistory } from "react-router-dom";
import { UserContext } from "../App";
import M from "materialize-css";
import axios from "axios";

```

```

const defaultVal = [
  {
    item_id: "54bd4f7ba09dabf70a684d02",
    item_name: "Chicken Biryani - 1 cup",
    brand_name: "Nutritionix",
    nf_calories: 291.95,
    nf_total_fat: 9.44,
    nf_serving_size_qty: 1,
    nf_serving_size_unit: "serving",
  },
  {
    item_id: "5de60a3fa3fba413474c82b7",
    item_name: "Mutton Biryani",
    brand_name: "Delicious Delights",
    nf_calories: 350,
    nf_total_fat: 12,
    nf_serving_size_qty: 1,
    nf_serving_size_unit: "serving",
  },
  {
    item_id: "5d244485765eef965d6bde5c",
    item_name: "Plant Kitchen Sweet Potato Biryani",
    brand_name: "M&S Food",
    nf_calories: 397.44,
    nf_total_fat: 12.8,
    nf_serving_size_qty: 1,
    nf_serving_size_unit: "serving",
  },
  {
    item_id: "5fa01232743c18841a132a4a",
    item_name: "Chicken Biryani",

```

```

    brand_name: "Great To Go By Market District",
    nf_calories: 660,
    nf_total_fat: 8,
    nf_serving_size_qty: 1,
    nf_serving_size_unit: "serving",
  },
  {
    item_id: "5500261d80dab5f6622012e3",
    item_name: "Chicken Biryani",
    brand_name: "Tandoor Chef",
    nf_calories: 480,
    nf_total_fat: 15,
    nf_serving_size_qty: 1,
    nf_serving_size_unit: "serving",
  },
  {
    item_id: "5e4f81fb705315b12e4a5255",
    item_name: "Sindhi Biryani",
    brand_name: "Malka",
    nf_calories: 26,
    nf_total_fat: 0.6,
    nf_serving_size_qty: 1,
    nf_serving_size_unit: "serving",
  },
  {
    item_id: "5e50d1e2fdd5a2bc5798cada",
    item_name: "Chicken Biryani",
    brand_name: "Mother's Recipe",
    nf_calories: 100,
    nf_total_fat: 5.5,
    nf_serving_size_qty: 1,
    nf_serving_size_unit: "serving",
  },
  {
    item_id: "60b399b71676c90008f388cf",
    item_name: "Veg. Biryani",
    brand_name: "Hyderabadi",
    nf_calories: 521,
    nf_total_fat: 27.6,
    nf_serving_size_qty: 1,
    nf_serving_size_unit: "serving",
  },
  {
    item_id: "630757beaac4c90008189661",
    item_name: "Delhi Biryani",
    brand_name: "The Spice Tailor",
    nf_calories: 129.22,
    nf_total_fat: 2.9,
    nf_serving_size_qty: 1,
    nf_serving_size_unit: "serving",
  },

```



```

    {
      item_id: "5a3618048a188be6561496b6",
      item_name: "Chicken Biryani",
      brand_name: "Lidl",
      nf_calories: 410,
      nf_total_fat: 19,
      nf_serving_size_qty: 1,
      nf_serving_size_unit: "serving",
    },
  ];

const Home = () => {
  const { state, dispatch } = useContext(UserContext);
  const history = useHistory();
  const [password, setPassword] = useState("");
  const [email, setEmail] = useState(null);
  const [existingOrder, setExistingOrder] = useState(defaultVal);

  const postData = () => {
    console.log({ state });
    if (!email) {
      M.toast({
        html: "Cuisine or Query input needed.",
        classes: "#c62828 red darken-3",
      });
      return;
    }

    const options = {
      method: "GET",
      url: `https://localhost:3001/v1_1/search/${encodeURIComponent(
        email
      )}`,
      params: {
        fields: "item_name,item_id,brand_name,nf_calories,nf_total_fat",
      },
    };

    axios
      .request(options)
      .then(function (response) {
        const data = response.data;
        console.log(response.data);
        if (data.error) {
          M.toast({
            html: data.message || "Something went wrong.",
            classes: "#c62828 red darken-3",
          });
        } else {
          M.toast({
            html: "Order placed successfully",

```

```

        classes: "#c62828 green darken-3",
    });
    const { hits = [] } = data || {};
    const resultArr = hits.map(({ fields = {} }) => fields);
    if (resultArr) {
        setExistingOrder(resultArr);
    }
}
})
.catch(function (err) {
    console.error(err);
    M.toast({
        html: err.message,
        classes: "#c62828 red darken-3",
    });
});
};

return (
    <div className="mycard">
        <div className="card auth-card input-field">
            {/*  */}
            <h2 style={{ fontSize: 60, fontWeight: "500", margin: 15 }}>
                Search your cuisine
            </h2>
            <input
                type="text"
                placeholder="Cuisine / Vegetables / Fruits, eg - biriyani"
                value={email}
                onChange={(e) => setEmail(e.target.value)}
            />
            <button
                className="btn waves-effect waves-light #64b5f6 blue
darken-1"
                onClick={() => PostData()}
            >
                Search
            </button>
        </div>
    </div></div>
    {existingOrder?.length ? (
        <>
            <div
                style={{
                    display: "flex",
                    alignItems: "center",
                    justifyContent: "center",
                    flexWrap: "wrap",
                    padding: 10,
                }}
            >

```

```

{existingOrder?.map(
  ({
    item_name: name,
    item_id,
    brand_name,
    nf_calories,
    nf_total_fat,
    nf_serving_size_qty,
    nf_serving_size_unit,
  }) => {
    return (
      <div
        key={item_id}
        className="card auth-card"
        style={{
          display: "flex",
          justifyContents:
"center",
          margin: 10,
          flexDirection:
"column",
        }}
        >
          <h3
            style={{
              fontSize:
"18px",
              fontWeight: 700,
              color: "blue",
              marginBottom:
"5px",
            }}
            >
            {name}
          </h3>
          <div
            style={{
              display: "flex",
              alignItems:
"center",
            }}
            >
            <div style={{
              marginRight: 10 }}>
              
Brand: {brand_name}</p>
{nf_calories}</p>
{nf_total_fat}</p>

    Quantity:{" "}
    {nf_serving_size_qty}{" "}
    {nf_serving_size_unit}

    </p>
  </div>
</div>
</div>
);
}
) || null}
</div>
</>
) : null}
</div>
);
};

export default Home;

IBM Watson:

<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="utf-8" />
    <link rel="icon" href="%PUBLIC_URL%/favicon.ico" />
    <meta name="viewport" content="width=device-width, initial-scale=1" />
    <meta name="theme-color" content="#000000" />
    <meta
      name="description"
      content="Web site created using create-react-app"
    />
    <link rel="apple-touch-icon" href="%PUBLIC_URL%/logo192.png" />
    <!--
      manifest.json provides metadata used when your web app is installed on a
      user's mobile device or desktop. See
      https://developers.google.com/web/fundamentals/web-app-manifest/

```

```
-->
<link rel="manifest" href="%PUBLIC_URL%/manifest.json" />
<!--
```

Notice the use of %PUBLIC_URL% in the tags above.
It will be replaced with the URL of the `public` folder during the build.
Only files inside the `public` folder can be referenced from the HTML.

Unlike "/favicon.ico" or "favicon.ico", "%PUBLIC_URL%/favicon.ico" will
work correctly both with client-side routing and a non-root public URL.
Learn how to configure a non-root public URL by running `npm run build`.

```
-->
<!-- Compiled and minified CSS -->
<link href="https://fonts.googleapis.com/icon?family=Material+Icons" rel="stylesheet">
<link rel="stylesheet"
href="https://cdnjs.cloudflare.com/ajax/libs/materialize/1.0.0/css/materialize.min.css">
<title>React App</title>
</head>
<body>
<noscript>You need to enable JavaScript to run this app.</noscript>
<div id="root"></div>
<!--
This HTML file is a template.
If you open it directly in the browser, you will see an empty page.

You can add webfonts, meta tags, or analytics to this file.
The build step will place the bundled scripts into the <body> tag.

To begin the development, run `npm start` or `yarn start`.
To create a production bundle, use `npm run build` or `yarn build`.
-->
<script>
window.watsonAssistantChatOptions = {
  integrationID: "sdsdsdd", // The ID of this integration.
  region: "us-east", // The region your integration is hosted in.
  serviceInstanceID: "dufhusdh", // The ID of your service instance.
  onLoad: function(instance) { instance.render(); }
};
setTimeout(function(){
  const t=document.createElement('script');
  t.src="https://web-chat.global.assistant.watson.appdomain.cloud/versions/" +
(window.watsonAssistantChatOptions.clientVersion || 'latest') +
"/WatsonAssistantChatEntry.js";
  document.head.appendChild(t);
});
</script>
</body>
</html>
```

Docker-compose:

version: "3.7"

services:

db:

image: postgres

environment:

POSTGRES_PASSWORD: postgres

POSTGRES_USER: postgres

POSTGRES_DB: xmeme

volumes:

- ./pgdata:/var/lib/postgresql/data

ports:

- "5432:5432"

server:

build:

context: ./

dockerfile: Dockerfile.Development

depends_on:

- db

ports:

- "4500:4500"

environment:

- DEV_DATABASE_URL=postgres://postgres:postgres@db:5432/xmeme

- PORT=4500

- NODE_ENV=development

volumes:

- ./app

command: npm run dev

13.2 Appendix

Source Code

<https://github.com/IBM-EPBL/IBM-Project-4262-1658727002>

Demo Link

https://www.youtube.com/watch?v=llvRMgcQMH4&ab_channel=CNBC

..

