TRANSMOGRIFY GUIDE

Mini Project Report

Submitted in partial fulfillment of the Semester-V BE degree as per the requirements of Osmania University, Hyderabad

By

Chennu Vaishnavi Priya	160618733137
Masrath Jahan	160618733154
Sri Momitha Chiluveru	160618733166

Department of Computer Science & Engineering
Stanley College of Engineering and Technology for Women
(Affiliated to OU & Approved by AICTE)

Chapel Road, Abids, Hyderabad - 500001

2020-2021

Stanley College of Engineering & Technology for Women Chapel Road, Hyderabad

(Affiliated to OU & Approved by AICTE)

Ref No: SCETW/CSE Dept/V Semester 2021/MiniProject Date: 8-2-2021

CERTIFICATE

This is to certify that the mini project titled "Transmogrify Guide" is a bonafied work carried over by Ms. Chennu Vaishnavi Priya. (H.T. No 160618733137), Ms. Masrath Jahan (H.T. No 160618733154) and Ms. Sri Momitha Chiluveru (H.T. No. 160618733166) in partial fulfillment of the requirements for the award of the degree Bachelor of Engineering in Computer Science and Engineering from Osmania University during the III/IV Semester-I of their B.E. course during the academic year 2020-2021.

(Dr. B V Ramana Murthy)
Head, Department of CSE

(Ms. Shugufta Fatima)
Project Guide

TABLE OF CONTENTS

Title **Page no.**

LIST OF FIGURES

Stanley College of Engineering and Technology for Women
Chapel Road, Abids, Hyderabad – 500 001

Department of Computer Science and Engineering

Mini Project

Batch No	Roll No	Name	Guide Name
CSEC008	160618733137	Chennu Vaishnavi Priya	Ms. Shugufta Fatima
	160618733154	Masrath Jahan	
	160618733166	Sri Momitha Chiluveru	

TRANSMOGRIFY GUIDE

Abstract

We are in a digital world where having a website is mandatory for a business in any domain and Artificial Intelligence is playing a vital role in reshaping web design and development. The first step towards maintaining a healthy life is to have a healthy diet. Thus the main aim of our project, “Transmogrify Guide” is to provide a useful means to help one eat healthy. It is a real time application that depicts the use of Artificial Intelligence for Web Development. It ensures that its users are provided with a detailed study about various diets and also provides a Virtual Assistant (Chatbot) which is used to increase the efficiency of our webpage and to guide its clients/users by clarifying their queries. Thus it provides a personalized End-User Experience.

With this thought, people no longer have to rely on taking an appointment of a dietician or go to a clinic, instead they can simply clarify their doubts related to diet by using our application. It also provides various diet recipes along with their specified components and understands the users’ necessity and helps them choose a suitable diet. It is completely a responsive application.

Hence it provides a platform that basically acts as a human intervention between the users and the computer and is a great resource to humans, helping them save time and effort.

Guide Signature

CHAPTER 1

INTRODUCTION

1.1 PURPOSE

With the current era of connectivity and technological innovation, we're able to find, share and experience new technologies through our own devices like never before. With services like Siri and Google now becoming widely used, users are no strangers to interacting with machines designed to act-human. Recently, a big step forward happened with the rise of conversational interfaces. As this technology grows and develops, it's reasonable to expect chatbots to be more sophisticated and intelligent. These AI powered technologies will ultimately perform an increasingly impressive job of simulating interaction with a living, breathing virtual assistants.

There are already chatbots serving in certain domain to accomplish complex tasks or support humans to achieve high efficiency in their activity. Chatbots can provide users with health information and diet plans. There are various bots available depending upon the type of diet and exercise plans the user needs. All these bots have the goal of helping users change unhealthy eating habits and give insights into their own eating habits as a way to help them transform. Chatbots are also effective in providing exercise plans, fitness tips and monitoring diet routines to be followed. It's easy to provide this kind of advices without any physical interaction. These bots focus on motivation and providing everyday information to users. They trigger the user into the activity via notifications or rather automatic messages. They're giving a little pieces of inspiration and reminders to help users stay on track. Chatbots can even determine user diet and help them to improve it to achieve their ideal weight. One point to consider is that they don't substitute health professionals, and they almost all tell you to seek the advice of your doctor or a health professional if required.

1.2 SCOPE

With the rise of smartphone, IoT and Messaging platforms, speech and language processing, chatbot systems are starting to appear feasible. However, adapting a one-size-fits-all approach can't achieve much when we deal with specific domains and their specific requirements. Designing patterns for specific domains is essential to increase the possibility that the chatbot can achieve the goal set. Therefore, focusing on the context, content, user-bot interaction and adapting pattern matching techniques to design sophisticated user-bot conversations can enhance the task accomplishment. Even with advanced AI platforms, there is still room for improvement. Chatbot designers have to craft the logical steps to follow so to achieve the goal set for the bot without falling into dead-ends. The logic behind the chatbot system depends a lot on the domain, its expected users, its behavior and the conversational flow between the user and the chatbot. Among the available techniques used in crafting the chatbot logic are simple approaches, such as decision trees, or more complex, such as custom natural language processing (NLP) and machine learning (ML) components.

In future scope of this project, we can include voice based queries in addition to text based inputs. The users will have to give voice input and the system can generate either a text output or even a voice based output. Since the system is scalable, each module can be upgraded by adding more functionalities, and increasing the information content in the knowledge base.

1.3 STUDY OF EXISTING SYSTEM

Health Organizations invest heavily in Health applications to connect with their patients for behavior coaching, medication monitoring, observing their symptoms etc. The results have been mixed. Research shows that some applications fared well while others fell flat. Even for the ones that showed promise, these applications often require a team of clinicians to follow up with patients to make sure that they use the technology solution on a regular basis. This affects the "scalability" factor of such technologies. The average performance of these Health apps lacks a truly engaging user experience. Another way of engaging patients is through Telephone helpline systems that are already available as a part of customer service.

But providing customer service is also a tedious job in terms of financial sense and customer satisfaction.

Thus, there is a need to develop a system that can help to overcome the above mentioned shortcomings and an Ai based chatbot can provide us a solution. A Chatbot is selfless and dedicated to you, it is always there for you and it always has time for yu. The information you provide in a Chabot is secured and that makes it even more incredible about this technology.

1.3.1 LITERATURE

With the help of AI and machine learning algorithms, chatbots are forecasted to save healthcare costs when used in place of a human or assist them as a preliminary step of helping to assess a condition and providing self-care recommendations. A chat bot (also known as a talk bot, Bot, chatterbox, Artificial Conversational Entity) is a computer program which conducts a conversation via auditory or textual methods. Such programs are often designed to convincingly simulate how a human would behave as a conversational partner, thereby passing the Turing test. In 1991, Dr.Sbaitso, a psychologist, with very limited possibilities has created an AI speech synthesis program for MS DOS Personal Computers. Four years later, A.L.I.C.E (Artificial Linguistic Internet Computer Entity) included 40,000 knowledge categories that was later awarded the Loebner Prize thrice. In 2001, SmarterChild was made available as a bot distributed across SMS networks and is now considered as a precursor to Apple's Siri, which was released on iPhones in 2010. Digital personal assistants in the form of Chatbots offer a lot more than simple messaging apps. They can be voice controlled, which makes it possible to use them also when your hands are busy. They come in handy when you want to play music, order food or take notes. The proposed system is a virtual chat assistant that can answer health related questions empathically on the basis of a doctor-patient communication model. The proposed assistant is not only informative but also provides a positive user experience. The patients' queries are all relevant questions, but they usually do not need the response from a physician. Moreover, doing Google searches to find the right answers is also challenging as most patients do not know how to assess the quality of information they find online. Moreover Internet is full of fake news and misleading sites that

want to sell something. Chatbots allow the patient to interact with the health care organization via their platform of choice. A large amount of data is generated by the health care sector. It is kept in a confusing array of different systems, applications and data silos. With the help of Chatbots this information can be accessed safely. Thus helping to initiate interactions with the users which was not possible earlier.

1.4 PROPOSED SYSTEM

The proposed system is a web application that has a chatbot integrated with it. There are several bot development frameworks like Microsoft Bot Framework, Facebook bot Engine, API.ai etc, in this project we are using the 'Dialogflow' API. The chatbot is built with IBM Watson Assistant's DialogFlow are intelligent personal assistants. It abstracts out the Natural Language Processing, Machine Learning and other deeper concepts and gives a clean usable user interface to focus on the conversation flow and build bots.

A Chatbot is an AI based software in a device (for example: Siri, Alexa, Google Assistant etc), application, website or other networks that help and assist customers to perform a particular task. It uses Natural language processing program to analyze the data and generate the results. Natural language processing is a field of artificial intelligence that helps in designing a program to process and analyze natural language data. It allows to establish interactions between computers and humans in a natural language. The proposed system is also referred as Chatterbot, Dialog system or Machine conversation system.

1) New advancements in AI have made today's Chatbot a lot more pleasant than before.

2) A large repository of public chat scripts enable Chatbot to learn and mimic human conversations.

3) The AI based service makes it easy to perform complex image recognition tasks, allowing chat users to send in photos, hand written notes etc.

The proposed system is a chat interface that is based on Retrieval based model of NLP where the bot is trained for a set of questions with a set of possible answers. Such an

intelligent chatbot can guide the concerned patients by understanding and assessing their symptoms that they are experiencing and identify the care that they need.

Features of the Proposed System

Proposed system is a Web Application that incorporates a Chatbot in it.

i) Build a simple and interactive real time chat system.

ii) Dedicated system which is able to solve all the queries regarding a diet.

iii) Suitable diet suggestion based on requirements and calorie intake prediction.

iv) Suggests the need to meet a doctor if necessary.

v) Provides information about diet, its uses and types of diets.

vi) Registration to various weight loss programs.

vii) A payment gateway (dummy) will be there to collect the payment and pay it to the dietician.

CHAPTER 2

REQUIREMENTS

2.1 SOFTWARE REQUIREMENTS

Operating System : Windows 7/8/8.1/10

Front-End : HTML ,CSS , JavaScript / jQuery: For designing the web application, these are the building-blocks of web pages that allows you to put images, text, videos, forms and other pieces of content together into a cohesive webpage.

Bootstrap, Push Notification Dynamic web Changes: Bootstrap is an open source toolkit for developing with HTML, CSS, and JavaScript.

Text Editor : Atom

2.2 HARDWARE REQUIREMENTS

Processor **:** Intel i3 or more

Motherboard **:** Intel Chipset Motherboard

Ram **:** 8 GB

Cache **:** 512 KB

Hard Disk **:** 16 GB Hard Disk recommended

Disk Drive **:** 1.44MB Floppy Disk Drive

Monitor **:** 1024 x 720 Display

Speed **:** 2.7 GHZ

CHAPTER 3

ARCHITECTURE

The architectural design and the system modules of the proposed system are discussed here:

3.1 Web Application

3.1.1 User

1. The web module provides a front end user interface, where the user will get an interface to communicate with the Chatbot.

2. It is a responsive webpage made using Bootstrap. The web application has various features such as, pricing plans, various diet recipes, a dummy payment gateway, comment section and various social platforms for contact.

3. The system is a standalone application where the user can register himself by logging into the application with his username and password. The system provides security by encrypting the password. After sign up the user can chat with the chatbot and can provide his personal information such as: weight, height, age, goal weight etc, which will be stored and accordingly response will be provided.

4. Once a user has registered, he need not register again and can directly sign in using the Login module which will receive username and password from the user, it will then check the database if it is already present and record the user information in its database.

5. The user can chat with the Chatbot by asking a query, the keywords related to the query are fetched and matched with the keywords in the database, which helps to retrieve and process the information and display the results to the user.

6. The system also provides a dummy payment gateway for the users to make the payment.

7.The user can log out of the webpage which will lead to the login page.

3.1.2 Admin

The Admin creates a database and a table to store users' information. He is responsible for updating and monitoring the system. The admin has access to the database and will view the update requests, and makes the required changes and update the database. The admin can perform tasks like: adding, updating, viewing and deleting rows/columns of a table.

3.1.3 Chatbot Module

The chatbot module is an AI ENGINE that detects the users' queries, extracts the requested data from the Knowledge base, Triggers the actions and responds to the users. The proposed Chatbot system design integrates computational algorithms and language model to stimulate a natural language chat communication between a human user and a Computer. In addition to that the system incorporates some personal details like: enquiry about the weight, problems faced, etc or information of a particular diet, so that Chatbot delivers it to the user whenever the user needs it. When the query is asked by the user, keywords are fetched and matched with that of the database with the help of keyword matching algorithm.

3.1.4 Algorithm

Chatbot PSEUDOCODE

i) Step 1: Insert user query in the chatbot window. (INPUT)

ii) Step 2: Pre-processing of the query e.g. suppose there is this query "What is diet?" (Here the words like: diet are given much importance.)

iii) Step 3: Fetch only keywords from the query.

iv) Step 4: Chatbot stores the session information in its log.

v) Step 5: Processes the Command. Matches the fetched keywords with the keywords in Knowledge base, and provides an appropriate response. A keyword matching algorithm is used to process the query.

vi) Step 6: The response is fetched from the knowledge base and returned as an output to the

user.

vii) Step 7: Exit.

The proposed Chatbot system functions based on a Decision Tree algorithm. It generates responses for the user queries based on the following method:

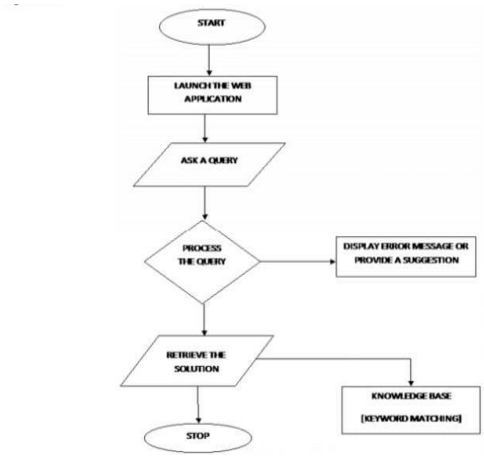


Fig 3.1 Chatbot Flowchart

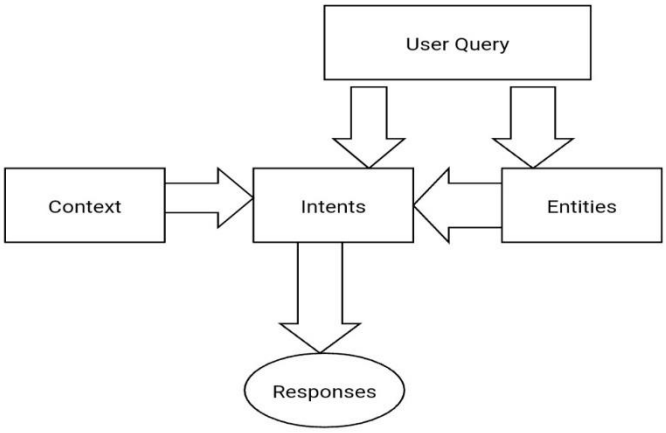
3.1.5 Dialogflow Architecture

1) Built-in natural-language processing (NLP) feature imparts artificial intelligence (AI) to the chatbot, thereby enabling it to process the natural language and carry forward the conversation in a natural way.

2) Dialogflow translates end-user text or audio during a conversation to structured data that your apps and services can understand. Watson Assistant is IBM's AI product that lets you build, train, and deploy conversational interactions into any application, device, or channel.

3) Machine learning makes Dialogflow intelligent enough to predict the hidden intention expressed in the natural input language.

4) Works with an array of platforms.



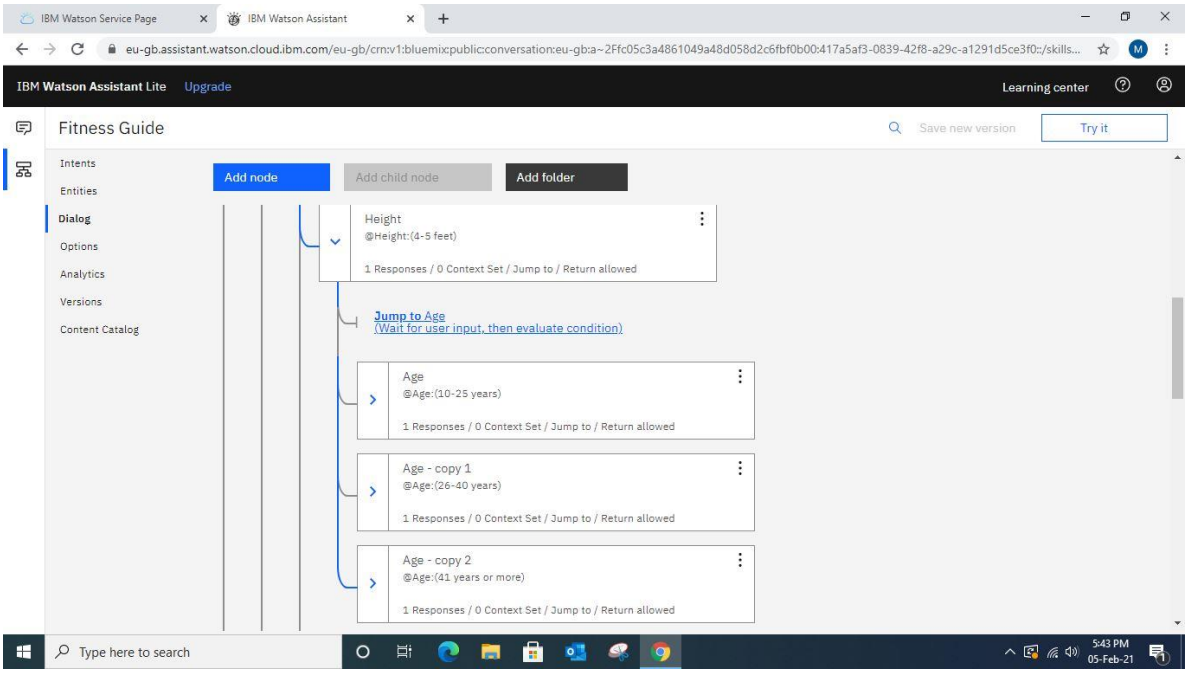


Fig 3.2 Dialogflow Architecture

1) User Query: The bot provides various categories of generally asked queries from which a user can choose his query related category or can even initiate a conversation with the bot by asking it questions of his own.

a) Intents: are the classes that help to design or create meaningful user queries, basically helping to create categories for different user queries.

b) Entities: are the parameters of the user query that help the bot to extract responses based on the keywords.

c) Responses: After the intents are processed by the bot, it generates responses and displays them to the users.

CHAPTER 4

SOFTWARE DESIGN

4.1 UML Diagrams

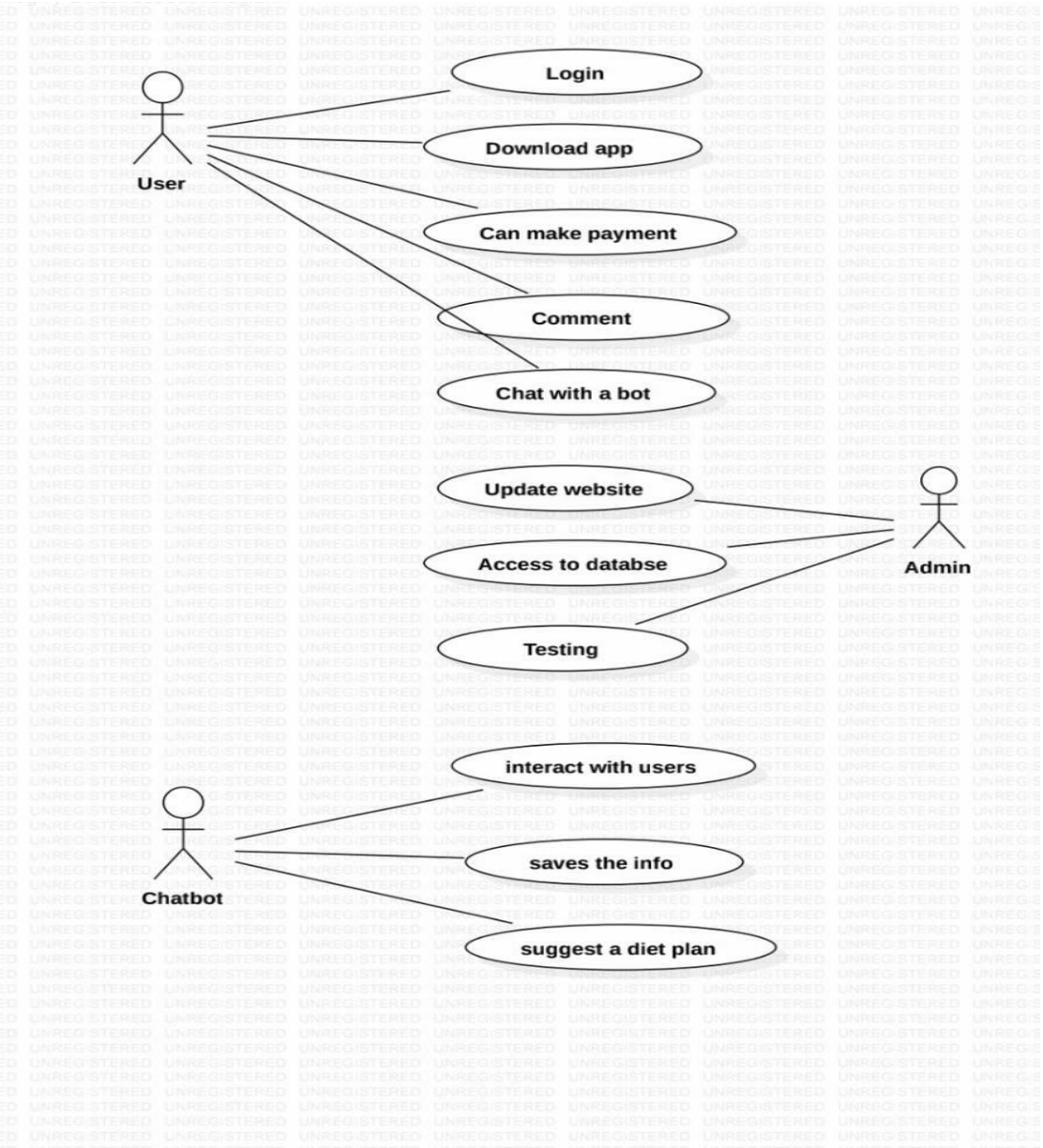


Fig 4.1 Use-case diagram for Transmogrify Guide

4.2 E-R Diagrams

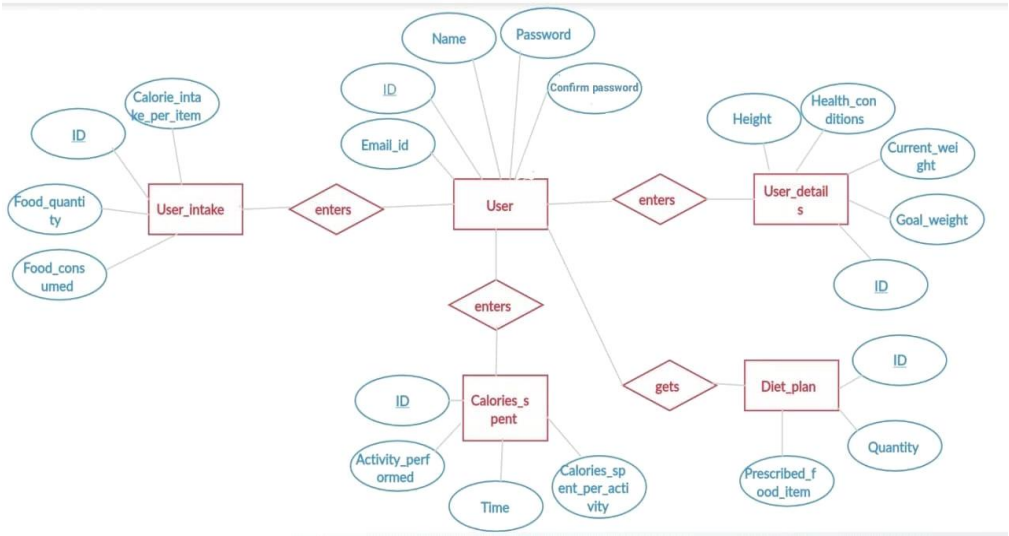


Fig 4.2.1 E-R Diagram of Transmogrify Guide

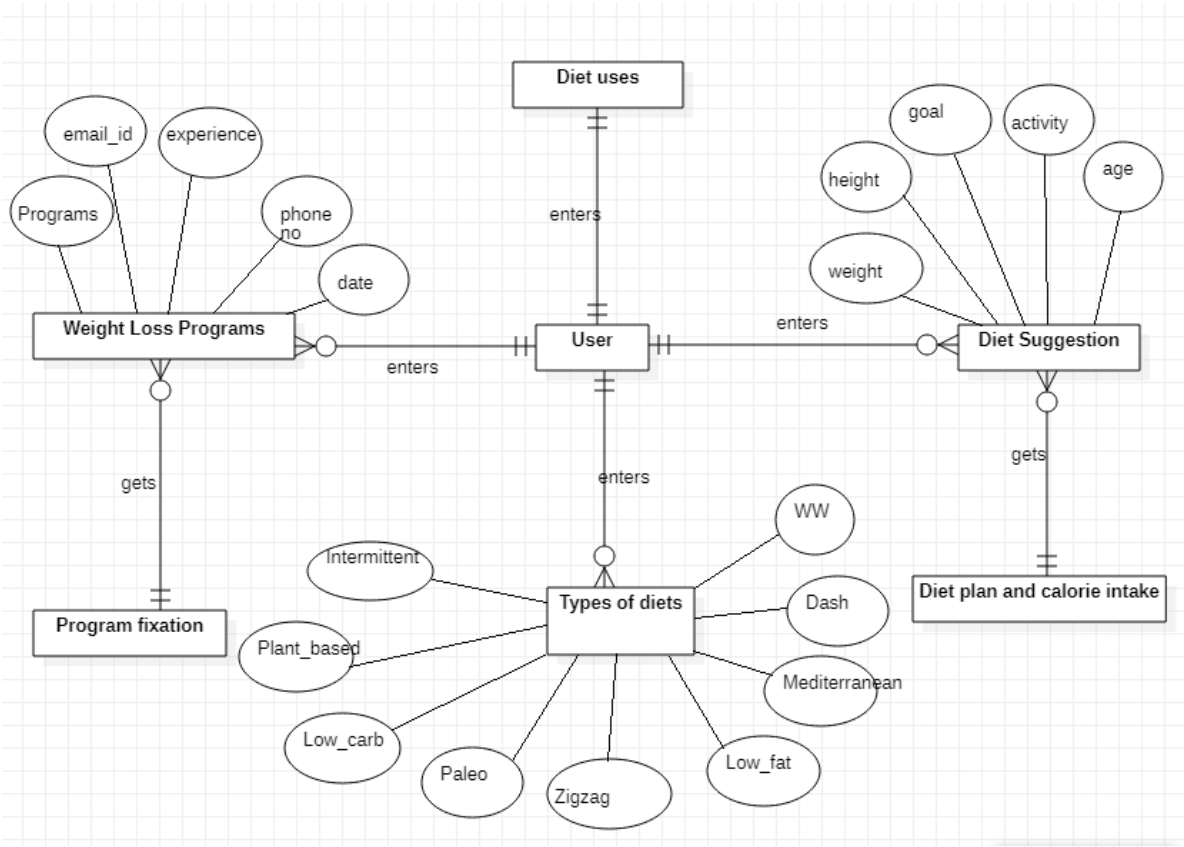


Fig 4.2.2 E-R Diagram of Transmogrify Guide Chatbot

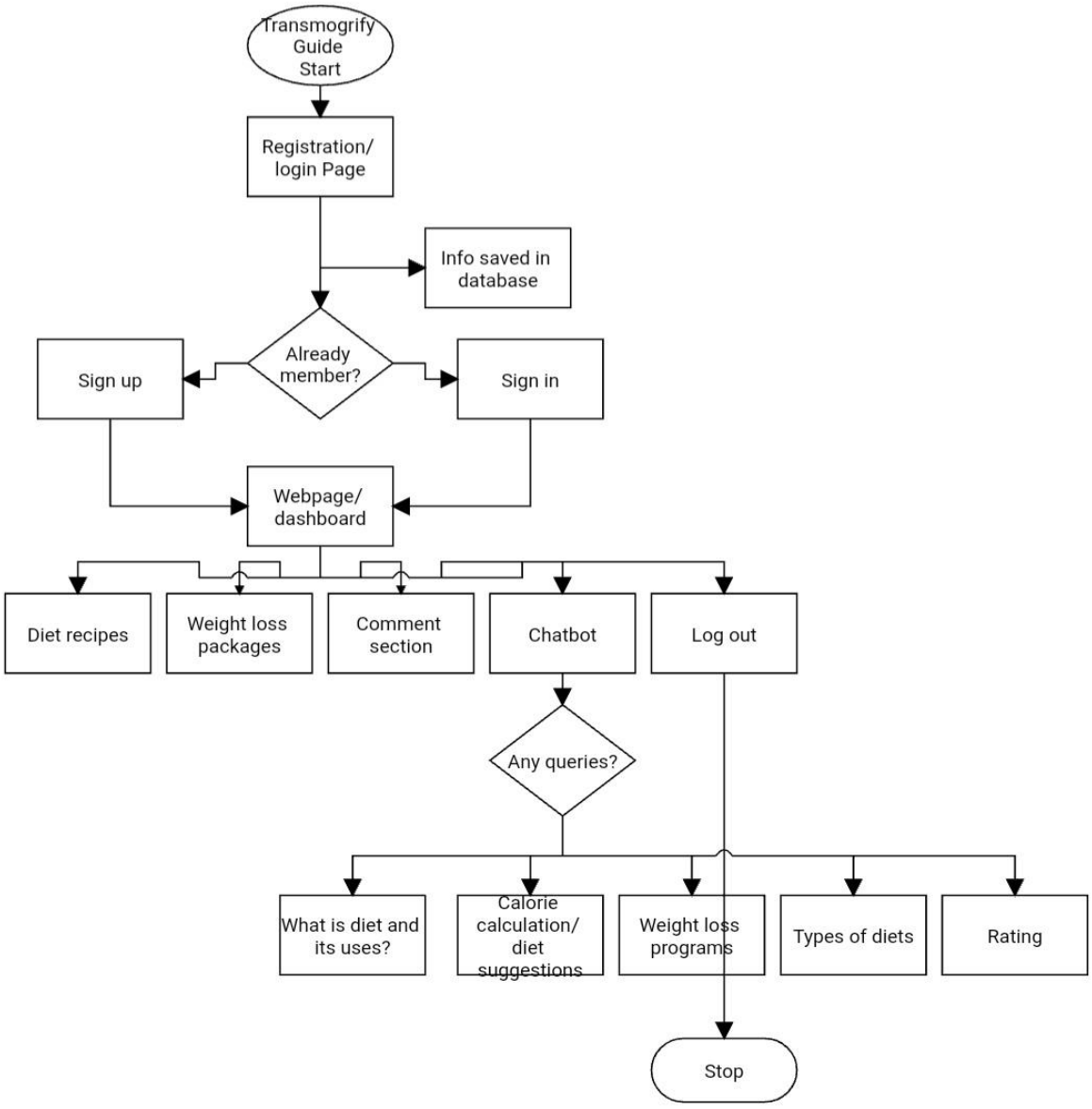


Fig 4.2.3 Flow chart of Transmogrify Guide

CHAPTER 5

CODING / CODING TEMPLATES

5.1 CODING

index.php

```
<?php include('server.php');

if(empty($_SESSION['username'])){

header('location: login.php');

}

?>

<!DOCTYPE html>

<html>

<head>

<meta charset="utf-8">

<link rel="stylesheet"
href="https://stackpath.bootstrapcdn.com/bootstrap/4.5.0/css/bootstrap.min.css"
integrity="sha384-
9aIt2nRpC12Uk9gS9baDl411NQApFmC26EwAOH8WgZl5MYYxFfc+NcPb1dKGj7Sk"
crossorigin="anonymous">

<title>Transmogrify Guide</title>

<!-- google fonts -->
```

```
<link
href="https://fonts.googleapis.com/css2?family=Montserrat:wght@600&family=Ubuntu:wght@300&display=swap" rel="stylesheet">

<link rel="stylesheet" href="css/styles.css">

<script src="https://kit.fontawesome.com/e5c73e334e.js"
crossorigin="anonymous"></script>

<script src="https://code.jquery.com/jquery-3.5.1.slim.min.js" integrity="sha384-DfXdz2htPH0lsSSs5nCTpuj/zy4C+OGpamoFVy38MVBnE+IbbVYUew+OrCXaRkfj"
crossorigin="anonymous"></script>

<script src="https://cdn.jsdelivr.net/npm/popper.js@1.16.0/dist/umd/popper.min.js"
integrity="sha384-Q6E9RHvbIyZFJoft+2mJbHaEWldlvI9IOYy5n3zV9zzTtmI3UksdQRVvoxMfooAo"
crossorigin="anonymous"></script>

<script src="https://stackpath.bootstrapcdn.com/bootstrap/4.5.0/js/bootstrap.min.js"
integrity="sha384-OgVRvuATP1z7JjHLkuOU7Xw704+h835Lr+6QL9UvYjZE3Ipu6Tp75j7Bh/kR0JKI"
crossorigin="anonymous"></script>

<title>User Registration</title>

<link rel="stylesheet" type="text/css" href="style.css">

<style>

  @keyframes slideInRight {

    from {

      transform: translate3d(100%, 0, 0);

      visibility: visible;
```

```
  }
  to {
    transform: translate3d(0, 0, 0);
  }
}
  button.chatLauncher {
  animation-duration: 0.5s;
  transition-duration: 0.5s;
  position: fixed;
  bottom: 32px;
  right: 32px;
  z-index: 9999;
  border: 4px solid #075cc2;
  padding: 1em;
  border-radius: 8px;
  margin: 0;
  text-decoration: none;
  background-color: #ffffff;
  color: #454545;
  font-family: sans-serif;
```

```
  font-size: 1rem;
  cursor: pointer;
  text-align: left;
  -webkit-appearance: none;
  -moz-appearance: none;
  width: 264px;
  opacity: 0;
}
button.chatLauncher.open {
  animation-name: slideInRight;
  opacity: 1;
}
button.chatLauncher:hover,
button.chatLauncher:focus {
  background-color: rgb(225, 225, 254);
  border: 4px solid #0053ba;
}
button.chatLauncher:focus {
  outline: 1px solid #0053ba;
  outline-offset: -4px;
```

```
    }
</style>
<section id="title">
    <div class="container-fluid">
    <!-- Nav Bar -->
    <nav class="navbar navbar-expand-lg navbar-dark">
        <a class="navbar-brand" href="">Transmogrify Guide <i class="fas fa-fire-alt"></i></a>
        <button class="navbar-toggler" type="button" data-toggle="collapse" data-target="#navbarSupportedContent" aria-controls="navbarSupportedContent" aria-expanded="false" aria-label="Toggle navigation">
      <span class="navbar-toggler-icon"></span>
      </button>
     <div class="collapse navbar-collapse" id="navbarSupportedContent">
        <ul class="navbar-nav ml-auto">
            <li class="nav-item">
                <a class="nav-link" href="#features">About</a>
            </li>
            <li class="nav-item">
                <a class="nav-link" href="#testimonials">Testimonials</a>
            </li>
```

```
            <li class="nav-item">
              <a class="nav-link" href="#pricing">Our Plans</a>
            </li>
            <li class="nav-item">
              <a class="nav-link" href="#cta">Contact Us</a>
            </li>
            <li class="nav-item">
              <a class="nav-link" href="#diet">Diet Recipes</a>
            </li>
          </ul>
        </div>
      </nav>
      <!-- Title -->
      <div class="row">
        <div class="col-lg-6">
        <h1 class="typing">All progess takes outside the comfort zone.</h1>
        <button type="button" class="dn btn btn-dark btn-lg download-button"><i class="fab fa-google-          play"></i>Download</button>
        <button type="button" class="dn btn btn-light btn-lg download-button"><i class="fab fa-apple"></i>Download</button>
      </div>
```

```
  <div class="col-lg-6">
  </div>
</div>
</div>
</section>
<!-- Features -->
<section id="features">
  <div class="row">
    <div class="col-lg-4">
      <i class="icon fas fa-check fa-4x"></i>
          <h3>Organic Diet</h3>
          <p class="fea">So easy to prepare, even your a busy person.</p>
  </div>
  <div class="col-lg-4">
    <i class="icon fas fa-heart fa-4x"></i>
        <h3>Safe Workouts</h3>
        <p class="fea">We have all the equipments and the greatest coaches.</p>
  </div>
  <div class="col-lg-4">
  <i class="icon fas fa-bullseye fa-4x"></i>
```

```
    <h3>Guaranteed Results</h3>
    <p class="fea">Find your new self or your money back.</p>
    </div>
   </div>
  </section>
  <!-- Testimonials -->
  <section id="testimonials">
    <div id="carouselExampleControls" class="carousel slide" data-ride="carousel">
      <div class="carousel-inner">
        <div class="carousel-item active">
          <h2>I no longer have to be worried about my food intake, because Transmogrify Guide has made my life simpler providing a guide and good recipes.</h2>
          <img class="test-pic" src="img\3.jpg" alt="dog-profile">
          <em>Pebbles, New York</em>
        </div>
        <div class="carousel-item">
          <h2 class="testimonial-text">As am a working woman, it was difficult for me to prepare a meal everyday, but with Transmogrify Guide, I've found the secret to a healthy life. It provides me a perfect advice and easy to go foods.</h2>
          <img class="testimonial-image" src="img\4.jpg" alt="lady-profile">
          <em>Beverly, Illinois</em>
```

```
</div>
<div class="carousel-item">
<h2 class="testimonial-text">There was a time when I used to eat so much of junk and when I realised I needed a lifestyle change, Transmogrify Guide has helped me a lot.</h2>
<img class="testimonial-image" src="img\5.jpg" alt="lady-profile">
<em>Hyderabad, India</em>
</div>
<div class="carousel-item">
<h2 class="testimonial-text">Transmogrify Guide has helped me change my food habits and I would like to highly recommend this to everyone</h2>
<img class="testimonial-image" src="img\6.jpg" alt="lady-profile">
<em>Posco, LA</em>
</div>
</div>
<a class="carousel-control-prev" href="#carouselExampleControls" role="button" data-slide="prev">
<span class="carousel-control-prev-icon" aria-hidden="true"></span>
<span class="sr-only">Previous</span>
</a>
<a class="carousel-control-next" href="#carouselExampleControls" role="button" data-slide="next">
```

```
      <span class="carousel-control-next-icon" aria-hidden="true"></span>
      <span class="sr-only">Next</span>
    </a>
  </div>
</section>
<!-- Press -->
<section id="press">
  <img class="press-pic" src="images/techcrunch.png" alt="tc-logo">
  <img class="press-pic" src="images/tnw.png" alt="tnw-logo">
  <img class="press-pic" src="images/bizinsider.png" alt="biz-insider-logo">
  <img class="press-pic" src="images/mashable.png" alt="mashable-logo">
</section>
<!-- Pricing -->
<section id="pricing">
  <h2>A Plan for every soul in need</h2>
  <p>Simple and affordable price plans for you.</p>
<div class="row">
<div class="pricing-col col-lg-4 col-md-6">
<div class="card">
  <div class="card-header">
```

```
    <h3>Muscle up, Butter cup</h3>
    </div>
    <div class="card-body">
      <h2>Free</h2>
      <p>Basic Workouts</p>
      <p>Diet suggestions</p>
      <p>Regular guidance</p>
      <button class="dn btn btn-lg btn-block btn-outline-dark" type="button">Sign Up</button>
    </div>
  </div>
</div>
<div class="pricing-col col-lg-4 col-md-6">
  <div class="card">
    <div class="card-header">
      <h3>Premium</h3>
    </div>
    <div class="card-body">
      <h2>$ 49 / mo</h2>
      <p>HIIT Workouts</p>
      <p>Meet a Nutrionist</p>
```

```
      <p>Grocery List</p>
      <p>1-1 Interaction</p>
      <button class="dn btn btn-lg btn-block btn-dark" type="button">Sign Up</button>
    </div>
  </div>
</div>
<div class="pricing-col col-lg-4">
  <div class="card">
    <div class="card-header">
    <h3>Platinum</h3>
    </div>
    <div class="card-body">
        <h2>$99 / mo</h2>
        <p>Protien based Meal</p>
        <p>Dietician</p>
        <p>First 2 months free</p>
        <p>HIIT Workouts</p>
        <button class="dn btn btn-lg btn-block btn-dark" type="button">Sign Up</button>
    </div>
    </div>
```

```
  </div>
</div>
  </section>
  <section id="diet">
  <h2>Diet Recipes</h2>
  <div class="container">
     <div class="box">
        <div class="imgBx">
           <img src="img\8.jpg" alt="" width="450px" height="300px">
        </div>
        <div class="contentBx">
           <h2>BBQ Protein Bowl</h2>
           <p>595 kcal kcal <br>14g
  Protein <br>
  35g
  Carbs <br>
  17g
  Fats <br>
  17g
  Fibre</p>
```

```
  <p>Ingredients:<br> Fajita Veggies, Barbeque Proteins,Millets, Lettuce, Spicy Broccoli, Spicy and Tangy Barbeque sauce   </p>

        </div>

    </div>

    <div class="box">

        <div class="imgBx">

            <img src="img\7.jpg" alt="" width="450px" height="300px">

        </div>

        <div class="contentBx">

            <h2>Shroomy Tzatziki</h2>

            <p>432 kcal <br>14g

  Protein <br>

  8g

  Carbs <br>

  8g

  Fats <br>

  8g

  Fibre</p>

  <p>Ingredients:<br> Fajita Veggies, Organic Brown Rice, Mushrooms, Lettuce, Spicy Broccoli, chickpeas spiced, Tzatziki   sauce </p>

        </div>
```

```
    </div>
    <div class="box">
        <div class="imgBx">
            <img src="img\9.webp" alt="" width="450px" height="300px">
        </div>
        <div class="contentBx">
            <h2>Chimichurri Zoodles</h2>
            <p>
  524 kcal <br>17g
  Protein <br>
  15g
  Carbs <br>
  15g
  Fats <br>
  15g
  Fibre</p>
  <p>Ingredients:<br> Fajita Veggies, Arugula, Candy Tomatoes, Caramel Onions, Chimichurri Sauce, Feta cheese,   Zoodles,spiced chickpeas. </p>
        </div>
    </div>
  </div>
```

```
</section>

<!-- Call to Action -->

<section id="cta">

<h3>Start your healthy hustle today</h3>

  <button type="button" class="dn cta-button btn btn-dark btn-lg download-button"><i class="fab fa-google-  play"></i>Download</button>

  <button type="button" class="dn cta-button btn btn-light btn-lg download-button"><i class="fab fa-  apple"></i>Download</button><br>

  <a href="mailto:vaishnavi.chennu@gmail.com" class="icons"><i class="fa fa-envelope" aria-hidden="true"></i></a>

  <a href="https://www.instagram.com/accounts/login/" class="icons"><i class="fa fa-instagram" aria-  hidden="true"></i></a>

  <a href="mailto:vaishnavi.chennu@gmail.com" class="icons"><i class="fa fa-facebook-square" aria-hidden="true"></i></a>

  <a href="tel:8501072588" class="icons"><i class="fa fa-phone" aria-hidden="true"></i></a>

  <a href="mailto:vaishnavi.chennu@gmail.com" class="icons"><i class="fa fa-twitter" aria-hidden="true"></i></a>

</section>

<!-- Footer -->

<footer id="footer">

  <p class="copyright-text">© Copyright 2021 Transmogrify Guide</p>

</footer>
```

```
  <script type="text/javascript" src="app.js"></script>
  <script type="text/javascript">
     VanillaTilt.init(document.querySelectorAll(".box"), {
        max: 25,
        speed: 400
     });
  </script>
<style>
body {font-family: Arial, Helvetica, sans-serif;}
* {box-sizing: border-box;}
/* Button used to open the chat form - fixed at the bottom of the page */
.open-button {
  background-color: #555;
  color: white;
  padding: 16px 20px;
  cursor: pointer;
  opacity: 0.7;
}
/* The popup chat - hidden by default */
.chat-popup {
```

```
  display: none;
}
/* Add styles to the form container */
.form-container {
  max-width: 300px;
  padding: 10px;
  background-color: white;
}
/* Full-width textarea */
.form-container textarea {
  width: 100%;
  padding: 15px;
  margin: 5px 0 22px 0;
  border: none;
  background: #f1f1f1;
  resize: none;
  min-height: 200px;
}
/* When the textarea gets focus, do something */
.form-container textarea:focus {
```

```
  background-color: #ddd;
  outline: none;
}
/* Set a style for the submit/send button */
.form-container .btn {
  background-color: #4CAF50;
  color: white;
  padding: 16px 20px;
  border: none;
  cursor: pointer;
  width: 100%;
  margin-bottom:10px;
  opacity: 0.8;
}
/* Add a red background color to the cancel button */
.form-container .cancel {
  background-color: red;
}
/* Add some hover effects to buttons */
.form-container .btn:hover, .open-button:hover {
```

```
  opacity: 1;
}
</style>
</head>
<body>
<style>
  #title {
    background-image: url('img/2.jpg');
    background-repeat: no-repeat;
    opacity:1;
    background-size: cover;
  }
</style>
<button class="open-button" onclick="openForm()">COMMENT SECTION</button>
<div class="chat-popup" id="myForm">
  <form action="mailto:srimomitha@gmail.com" method="post" enctype="text/plain" class="form-container">
    <h4>Comment here</h4>
    <label for="comment">Message</label>
    <textarea placeholder="Type message.." name="comment" required></textarea>
    <button type="submit" name="send" class="btn">Send</button>
```

```
    <button type="button" class="btn cancel" onclick="closeForm()">Close</button>
  </form>
</div>
<script>
function openForm() {
  document.getElementById("myForm").style.display = "block";
}
function closeForm() {
  document.getElementById("myForm").style.display = "none";
}
</script>
<button type="button" class="chatLauncher" style="display:none;">
  <div>
    <strong>Have questions?</strong> Chat with our Virtual Assistant.
  </div>
</button>
<script>
  window.watsonAssistantChatOptions = {
      integrationID: "9170a879-cff2-43ea-8d0d-8515933bedd8", // The ID of this integration.
      region: "eu-gb", // The region your integration is hosted in.
```

```
    serviceInstanceID: "417a5af3-0839-42f8-a29c-a1291d5ce3f0", // The ID of your service instance.
   showLauncher: false,
   onLoad: function(instance) {
    // Select the button element from the page.
    const button = document.querySelector('.chatLauncher');
     // Add the event listener to open your web chat.
    button.addEventListener('click', function clickListener() {
     instance.openWindow();
    });
     // Render the web chat. Nothing appears on the page, because the launcher is
    // hidden and the web chat window is closed by default.
    instance.render().then(function() {
     // Now that web chat has been rendered (but is still closed), we make the
     // custom launcher button visible.
     button.style.display = 'block';
     button.classList.add('open');
    });
   }
  };
  setTimeout(function(){
```

```
    const t=document.createElement('script');
    t.src="https://web-chat.global.assistant.watson.appdomain.cloud/loadWatsonAssistantChat.js";
    document.head.appendChild(t);
  });
</script>
<?php if (isset($_SESSION["username"])): ?>
<center><p>Thank you <strong><?php echo $_SESSION['username']; ?></strong> for visiting. Do visit again.</p></center>
<center><p><a href="index.php?logout='1'" style="color: red;">Logout</a></p></center>
<?php endif ?>
</body>
</html>
```

app.js

```
var VanillaTilt = (function () {
'use strict';
class VanillaTilt {
  constructor(element, settings = {}) {
    if (!(element instanceof Node)) {
      throw ("Can't initialize VanillaTilt because " + element + " is not a Node.");
    }
```

```
this.width = null;

this.height = null;

this.clientWidth = null;

this.clientHeight = null;

this.left = null;

this.top = null;

// for Gyroscope sampling

this.gammazero = null;

this.betazero = null;

this.lastgammazero = null;

this.lastbetazero = null;

this.transitionTimeout = null;

this.updateCall = null;

this.event = null;

this.updateBind = this.update.bind(this);

this.resetBind = this.reset.bind(this);

this.element = element;

this.settings = this.extendSettings(settings);

this.reverse = this.settings.reverse ? -1 : 1;

this.glare = VanillaTilt.isSettingTrue(this.settings.glare);
```

```
  this.glarePrerender = VanillaTilt.isSettingTrue(this.settings["glare-prerender"]);
  this.fullPageListening = VanillaTilt.isSettingTrue(this.settings["full-page-listening"]);
  this.gyroscope = VanillaTilt.isSettingTrue(this.settings.gyroscope);
  this.gyroscopeSamples = this.settings.gyroscopeSamples;
  this.elementListener = this.getElementListener();
  if (this.glare) {
    this.prepareGlare();
  }
  if (this.fullPageListening) {
    this.updateClientSize();
  }
  this.addEventListeners();
  this.updateInitialPosition();
}
static isSettingTrue(setting) {
  return setting === "" || setting === true || setting === 1;
}
/**
 * Method returns element what will be listen mouse events
 * @return {Node}
```

```
 */
 getElementListener() {
  if (this.fullPageListening) {
   return window.document;
  }
  if (typeof this.settings["mouse-event-element"] === "string") {
   const mouseEventElement = document.querySelector(this.settings["mouse-event-element"]);
   if (mouseEventElement) {
    return mouseEventElement;
   }
  }
  if (this.settings["mouse-event-element"] instanceof Node) {
   return this.settings["mouse-event-element"];
  }
  return this.element;
 }
 /**
  * Method set listen methods for this.elementListener
  * @return {Node}
  */
```

```
addEventListeners() {
  this.onMouseEnterBind = this.onMouseEnter.bind(this);
  this.onMouseMoveBind = this.onMouseMove.bind(this);
  this.onMouseLeaveBind = this.onMouseLeave.bind(this);
  this.onWindowResizeBind = this.onWindowResize.bind(this);
  this.onDeviceOrientationBind = this.onDeviceOrientation.bind(this);
  this.elementListener.addEventListener("mouseenter", this.onMouseEnterBind);
  this.elementListener.addEventListener("mouseleave", this.onMouseLeaveBind);
  this.elementListener.addEventListener("mousemove", this.onMouseMoveBind);
  if (this.glare || this.fullPageListening) {
    window.addEventListener("resize", this.onWindowResizeBind);
  }
  if (this.gyroscope) {
    window.addEventListener("deviceorientation", this.onDeviceOrientationBind);
  }
}
/**
 * Method remove event listeners from current this.elementListener
 */
removeEventListeners() {
```

```
  this.elementListener.removeEventListener("mouseenter", this.onMouseEnterBind);
  this.elementListener.removeEventListener("mouseleave", this.onMouseLeaveBind);
  this.elementListener.removeEventListener("mousemove", this.onMouseMoveBind);
  if (this.gyroscope) {
    window.removeEventListener("deviceorientation", this.onDeviceOrientationBind);
  }
  if (this.glare || this.fullPageListening) {
    window.removeEventListener("resize", this.onWindowResizeBind);
  }
 }
 destroy() {
  clearTimeout(this.transitionTimeout);
  if (this.updateCall !== null) {
    cancelAnimationFrame(this.updateCall);
  }
  this.reset();
  this.removeEventListeners();
  this.element.vanillaTilt = null;
  delete this.element.vanillaTilt;
  this.element = null;
```

```
  }
  onDeviceOrientation(event) {
    if (event.gamma === null || event.beta === null) {
      return;
    }
    this.updateElementPosition();
    if (this.gyroscopeSamples > 0) {
      this.lastgammazero = this.gammazero;
      this.lastbetazero = this.betazero;
      if (this.gammazero === null) {
        this.gammazero = event.gamma;
        this.betazero = event.beta;
      } else {
        this.gammazero = (event.gamma + this.lastgammazero) / 2;
        this.betazero = (event.beta + this.lastbetazero) / 2;
      }
      this.gyroscopeSamples -= 1;
    }
    const totalAngleX = this.settings.gyroscopeMaxAngleX -
this.settings.gyroscopeMinAngleX;
```

```
    const totalAngleY = this.settings.gyroscopeMaxAngleY -
this.settings.gyroscopeMinAngleY;
    const degreesPerPixelX = totalAngleX / this.width;
    const degreesPerPixelY = totalAngleY / this.height;
    const angleX = event.gamma - (this.settings.gyroscopeMinAngleX + this.gammazero);
    const angleY = event.beta - (this.settings.gyroscopeMinAngleY + this.betazero);
    const posX = angleX / degreesPerPixelX;
    const posY = angleY / degreesPerPixelY;
    if (this.updateCall !== null) {
      cancelAnimationFrame(this.updateCall);
    }
    this.event = {
      clientX: posX + this.left,
      clientY: posY + this.top,
    };
    this.updateCall = requestAnimationFrame(this.updateBind);
  }
  onMouseEnter() {
    this.updateElementPosition();
    this.element.style.willChange = "transform";
    this.setTransition();
```

```
}
onMouseMove(event) {
  if (this.updateCall !== null) {
    cancelAnimationFrame(this.updateCall);
  }
  this.event = event;
  this.updateCall = requestAnimationFrame(this.updateBind);
}
onMouseLeave() {
  this.setTransition();
  if (this.settings.reset) {
    requestAnimationFrame(this.resetBind);
  }
}
reset() {
  this.event = {
    clientX: this.left + this.width / 2,
    clientY: this.top + this.height / 2
  };
  if (this.element && this.element.style) {
```

```
    this.element.style.transform = `perspective(${this.settings.perspective}px) ` +
      `rotateX(0deg) ` +
      `rotateY(0deg) ` +
      `scale3d(1, 1, 1)`;
  }
  this.resetGlare();
}
resetGlare() {
  if (this.glare) {
    this.glareElement.style.transform = "rotate(180deg) translate(-50%, -50%)";
    this.glareElement.style.opacity = "0";
  }
}
updateInitialPosition() {
  if (this.settings.startX === 0 && this.settings.startY === 0) {
    return;
  }
  this.onMouseEnter();
  if (this.fullPageListening) {
    this.event = {
```

```
      clientX: (this.settings.startX + this.settings.max) / (2 * this.settings.max) * this.clientWidth,
      clientY: (this.settings.startY + this.settings.max) / (2 * this.settings.max) * this.clientHeight
    };
  } else {
    this.event = {
      clientX: this.left + ((this.settings.startX + this.settings.max) / (2 * this.settings.max) * this.width),
      clientY: this.top + ((this.settings.startY + this.settings.max) / (2 * this.settings.max) * this.height)
    };
  }
  let backupScale = this.settings.scale;
  this.settings.scale = 1;
  this.update();
  this.settings.scale = backupScale;
  this.resetGlare();
 }
 getValues() {
  let x, y;
  if (this.fullPageListening) {
```

```
      x = this.event.clientX / this.clientWidth;
      y = this.event.clientY / this.clientHeight;
    } else {
      x = (this.event.clientX - this.left) / this.width;
      y = (this.event.clientY - this.top) / this.height;
    }
    x = Math.min(Math.max(x, 0), 1);
    y = Math.min(Math.max(y, 0), 1);
    let tiltX = (this.reverse * (this.settings.max - x * this.settings.max * 2)).toFixed(2);
    let tiltY = (this.reverse * (y * this.settings.max * 2 - this.settings.max)).toFixed(2);
    let angle = Math.atan2(this.event.clientX - (this.left + this.width / 2), -(this.event.clientY
- (this.top + this.height / 2))) * (180 / Math.PI);
    return {
      tiltX: tiltX,
      tiltY: tiltY,
      percentageX: x * 100,
      percentageY: y * 100,
      angle: angle
    };
  }
  updateElementPosition() {
```

```
    let rect = this.element.getBoundingClientRect();
    this.width = this.element.offsetWidth;
    this.height = this.element.offsetHeight;
    this.left = rect.left;
    this.top = rect.top;
  }
  update() {
    let values = this.getValues();
    this.element.style.transform = "perspective(" + this.settings.perspective + "px) " +
      "rotateX(" + (this.settings.axis === "x" ? 0 : values.tiltY) + "deg) " +
      "rotateY(" + (this.settings.axis === "y" ? 0 : values.tiltX) + "deg) " +
      "scale3d(" + this.settings.scale + ", " + this.settings.scale + ", " + this.settings.scale +
")";
    if (this.glare) {
      this.glareElement.style.transform = `rotate(${values.angle}deg) translate(-50%, -50%)`;
      this.glareElement.style.opacity = `${values.percentageY * this.settings["max-glare"] /
100}`;
    }
    this.element.dispatchEvent(new CustomEvent("tiltChange", {
      "detail": values
    }));
```

```
    this.updateCall = null;
  }
  /**
   * Appends the glare element (if glarePrerender equals false)
   * and sets the default style
   */
  prepareGlare() {
    // If option pre-render is enabled we assume all html/css is present for an optimal glare effect.
    if (!this.glarePrerender) {
      // Create glare element
      const jsTiltGlare = document.createElement("div");
      jsTiltGlare.classList.add("js-tilt-glare");
      const jsTiltGlareInner = document.createElement("div");
      jsTiltGlareInner.classList.add("js-tilt-glare-inner");
      jsTiltGlare.appendChild(jsTiltGlareInner);
      this.element.appendChild(jsTiltGlare);
    }
    this.glareElementWrapper = this.element.querySelector(".js-tilt-glare");
    this.glareElement = this.element.querySelector(".js-tilt-glare-inner");
    if (this.glarePrerender) {
```

```
      return;
    }
    Object.assign(this.glareElementWrapper.style, {
      "position": "absolute",
      "top": "0",
      "left": "0",
      "width": "100%",
      "height": "100%",
      "overflow": "hidden",
      "pointer-events": "none"
    });
    Object.assign(this.glareElement.style, {
      "position": "absolute",
      "top": "50%",
      "left": "50%",
      "pointer-events": "none",
      "background-image": `linear-gradient(0deg, rgba(255,255,255,0) 0%,
rgba(255,255,255,1) 100%)`,
      "width": `${this.element.offsetWidth * 2}px`,
      "height": `${this.element.offsetWidth * 2}px`,
      "transform": "rotate(180deg) translate(-50%, -50%)",
```

```
    "transform-origin": "0% 0%",
    "opacity": "0",
  });
}
updateGlareSize() {
  if (this.glare) {
    Object.assign(this.glareElement.style, {
      "width": `${this.element.offsetWidth * 2}`,
      "height": `${this.element.offsetWidth * 2}`,
    });
  }
}
updateClientSize() {
  this.clientWidth = window.innerWidth
    || document.documentElement.clientWidth
    || document.body.clientWidth;
  this.clientHeight = window.innerHeight
    || document.documentElement.clientHeight
    || document.body.clientHeight;
}
```

```
  onWindowResize() {
    this.updateGlareSize();
    this.updateClientSize();
  }
  setTransition() {
    clearTimeout(this.transitionTimeout);
    this.element.style.transition = this.settings.speed + "ms " + this.settings.easing;
    if (this.glare) this.glareElement.style.transition = `opacity ${this.settings.speed}ms
${this.settings.easing}`;
    this.transitionTimeout = setTimeout(() => {
      this.element.style.transition = "";
      if (this.glare) {
        this.glareElement.style.transition = "";
      }
    }, this.settings.speed);
  }
  /**
   * Method return patched settings of instance
   * @param {boolean} settings.reverse - reverse the tilt direction
   * @param {number} settings.max - max tilt rotation (degrees)
   * @param {startX} settings.startX - the starting tilt on the X axis, in degrees. Default: 0
```

* @param {startY} settings.startY - the starting tilt on the Y axis, in degrees. Default: 0

* @param {number} settings.perspective - Transform perspective, the lower the more extreme the tilt gets

* @param {string} settings.easing - Easing on enter/exit

* @param {number} settings.scale - 2 = 200%, 1.5 = 150%, etc..

* @param {number} settings.speed - Speed of the enter/exit transition

* @param {boolean} settings.transition - Set a transition on enter/exit

* @param {string|null} settings.axis - What axis should be disabled. Can be X or Y

* @param {boolean} settings.glare - What axis should be disabled. Can be X or Y

* @param {number} settings.max-glare - the maximum "glare" opacity (1 = 100%, 0.5 = 50%)

* @param {boolean} settings.glare-prerender - false = VanillaTilt creates the glare elements for you, otherwise

* @param {boolean} settings.full-page-listening - If true, parallax effect will listen to mouse move events on the whole document, not only the selected element

* @param {string|object} settings.mouse-event-element - String selector or link to HTML-element what will be listen mouse events

* @param {boolean} settings.reset - false = If the tilt effect has to be reset on exit

* @param {gyroscope} settings.gyroscope - Enable tilting by deviceorientation events

* @param {gyroscopeSensitivity} settings.gyroscopeSensitivity - Between 0 and 1 - The angle at which max tilt position is reached. 1 = 90deg, 0.5 = 45deg, etc..

* @param {gyroscopeSamples} settings.gyroscopeSamples - How many gyroscope moves to decide the starting position.

```
 */
extendSettings(settings) {
  let defaultSettings = {
    reverse: false,
    max: 15,
    startX: 0,
    startY: 0,
    perspective: 1000,
    easing: "cubic-bezier(.03,.98,.52,.99)",
    scale: 1,
    speed: 300,
    transition: true,
    axis: null,
    glare: false,
    "max-glare": 1,
    "glare-prerender": false,
    "full-page-listening": false,
    "mouse-event-element": null,
    reset: true,
    gyroscope: true,
```

```
  gyroscopeMinAngleX: -45,
  gyroscopeMaxAngleX: 45,
  gyroscopeMinAngleY: -45,
  gyroscopeMaxAngleY: 45,
  gyroscopeSamples: 10
};
let newSettings = {};
for (var property in defaultSettings) {
  if (property in settings) {
    newSettings[property] = settings[property];
  } else if (this.element.hasAttribute("data-tilt-" + property)) {
    let attribute = this.element.getAttribute("data-tilt-" + property);
    try {
      newSettings[property] = JSON.parse(attribute);
    } catch (e) {
      newSettings[property] = attribute;
    }
  } else {
    newSettings[property] = defaultSettings[property];
  }
```

```
    }
    return newSettings;
  }
  static init(elements, settings) {
    if (elements instanceof Node) {
      elements = [elements];
    }
    if (elements instanceof NodeList) {
      elements = [].slice.call(elements);
    }
    if (!(elements instanceof Array)) {
      return;
    }
    elements.forEach((element) => {
      if (!("vanillaTilt" in element)) {
        element.vanillaTilt = new VanillaTilt(element, settings);
      }
    });
  }
}
```

```
if (typeof document !== "undefined") {
  /* expose the class to window */
  window.VanillaTilt = VanillaTilt;
  /**
   * Auto load
   */
  VanillaTilt.init(document.querySelectorAll("[data-tilt]"));
}
return VanillaTilt;
}());
```

5.2 FUNCTIONALITY EXPLANATION

Login Page

Php login script is used to provide the authentication for our website. The Script executes after submitting the user register button. Login page works based on session. If the user closes the session, it will not erase the session data unless the user logs out of the session.

Web Page

After successful logging in the user is taken to the website (main page), here we Integrated Chatbot with the Web Application. The Website has been made responsive using Bootstrap which means all device users can use the website without any compromise. It adjusts the screen according to users' device measurements. In order to use Bootstrap we had to insert the Bootstrap CDN in the html file.

```
5    <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/
     bootstrap/3.3.7/css/bootstrap.min.css" integrity="
     sha384-BVYiiSIFeK1dGmJRAkycuHAHRg32OmUcww7on3RYdg4Va+PmSTsz/
     K68vbdEjh4u" crossorigin="anonymous">
```

Content Delivery Network (CDN) is a system of distributed servers (network) that deliver webpages and other Web content to a user based on the geographic locations of the user, the origin of the webpage and a content delivery server.

JavaScript code was also used to provide features to the Website for presentation like Carousels and gyrosopic view of the diet recipes in the website.

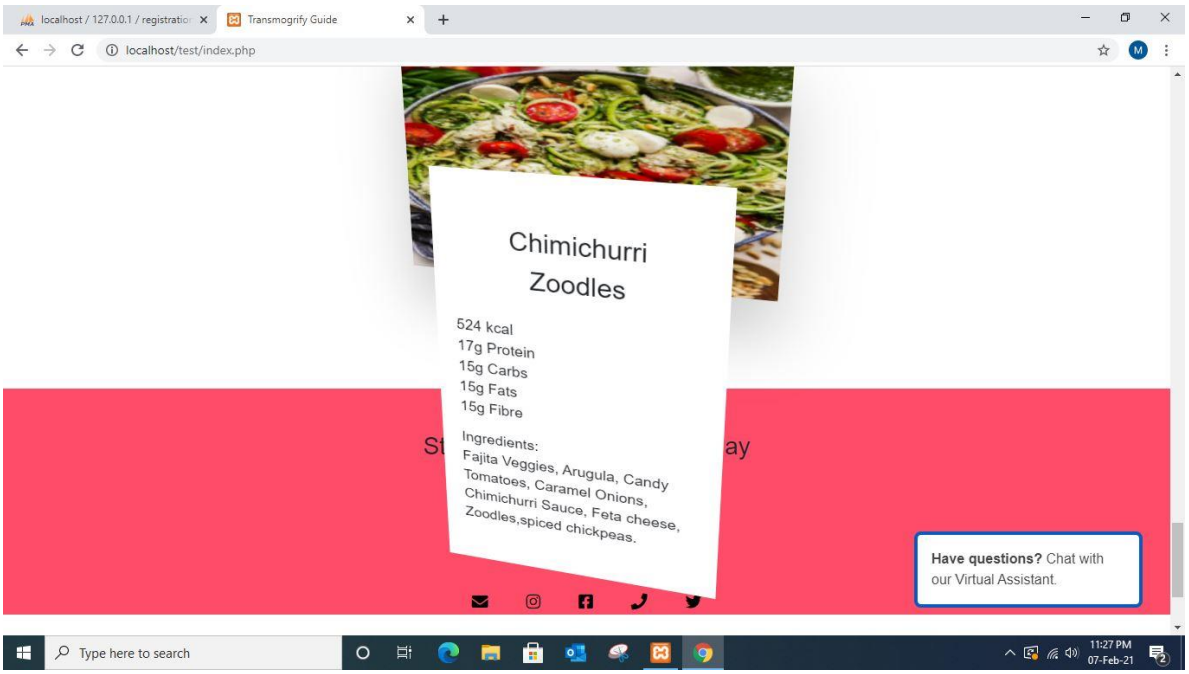


Fig 5.1 Gyroscopic View

Chatbot Functionality

We created a bot and configured it with one or more intents that we want to support. Configured the bot so it understands the user's goal (intent), engages in conversation with the user to elicit information, and fulfills the user's intent. Deployed the bot. You can deploy the bot on various platforms such as Facebook Messenger, mobile applications or messaging applications. Since the proposed system is a web application, we deployed the bot on the

website. The following code snippet is added in the program to integrate the bot to the web application:

```
window.watsonAssistantChatOptions = {
    integrationID: "9170a879-cff2-43ea-8d0d-8515933bedd8", // The ID of this integration.
    region: "eu-gb", // The region your integration is hosted in.
    serviceInstanceID: "417a5af3-0839-42f8-a29c-a1291d5ce3f0", // The ID of your service instance.
    onLoad: function(instance) { instance.render(); }
  };
setTimeout(function(){
```

Fig 5.2 Chatbot Code Snippet

5.3 DATABASE TABLE

We created a PHP Login System with the help of PHP database. There are few steps given for creating a login system with PHP database. The main pre-requisites to create the login module is XAMPP - XAMPP is a cross-platform software, which stands for Cross-platform(X) Apache server (A), MySQL (M), PHP (P), Perl (P). XAMPP is a complete software package, so, we don't need to install all these separately.

To create the database and the login table inside the database. Access the phpMyAdmin on the browser using localhost/phpmyadmin/ and create a table in the database. Here we created a database and table using GUI based phpMyAdmin rather than queries execution.

To run the register/login form, open the xampp control panel and start the Apache server and MySQL and click on MySQL admin. Now, you will be led to http://localhost/phpmyadmin. All setup is done now. Enter the user's details in the register/login form and click the register/login button. Select the particular database name and table in it to check the database updates.

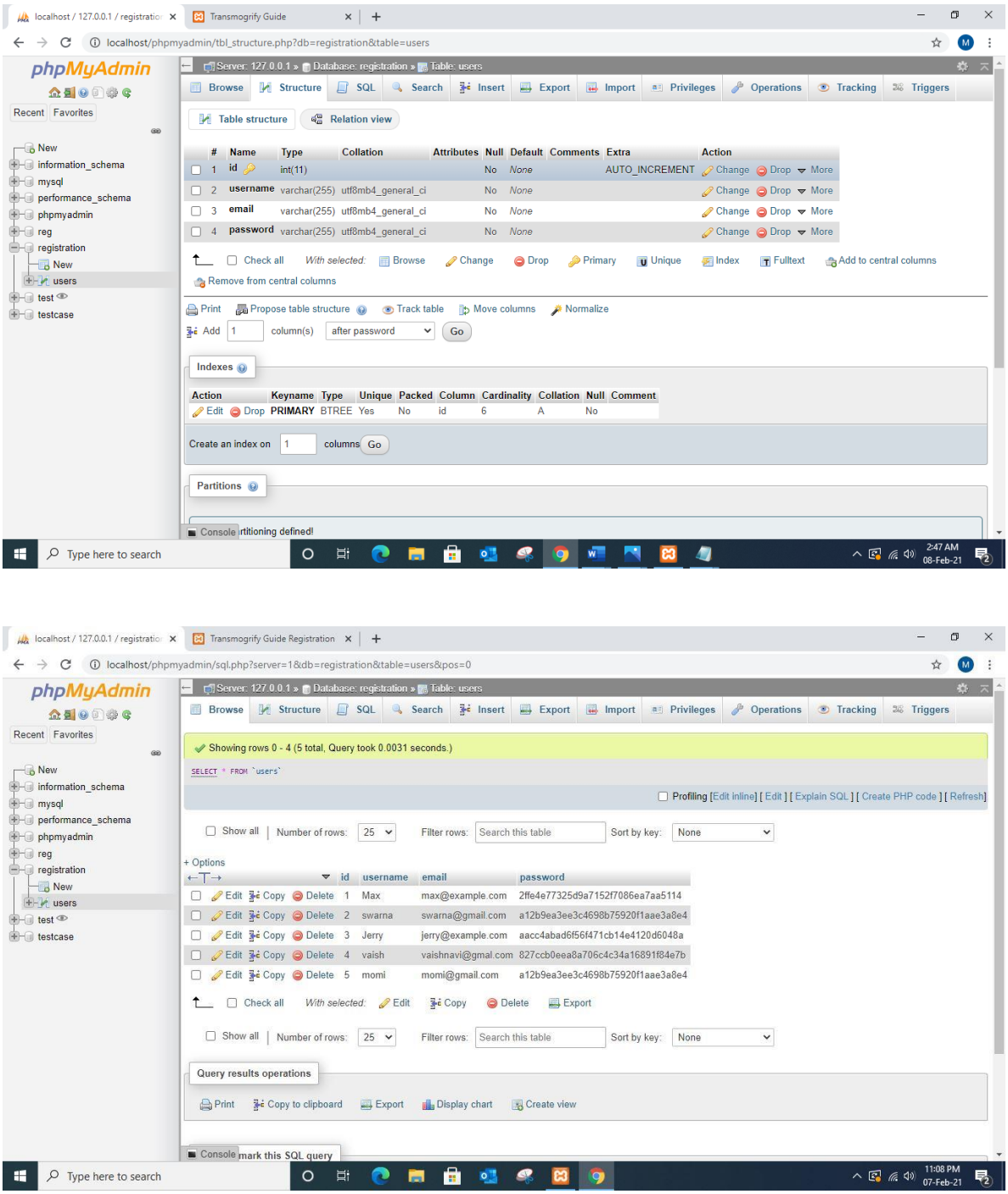


Fig 5.3 Database Table

CHAPTER 6

TESTING CASES

Web application testing, a software testing technique is exclusively adopted to test the applications that are hosted on web in which the application interfaces and other functionalities are tested.

Web Application Testing - Techniques:

1. **Functionality Testing** - The below are some of the checks that are performed but not limited to the below list:

- Verify there is no dead page or invalid redirects.
- First check all the validations on each field.
- Wrong inputs are to perform negative testing.
- Verify the workflow of the system.
- Verify the data integrity.

2. **Usability testing -** To verify how the application is easy to use.

- Test the navigation and controls.
- Content checking.
- Check for user intuition.

3. **Interface testing -** Performed to verify the interface and the dataflow from one system to other.

4. **Compatibility testing-** Compatibility testing is performed based on the context of the application.

- Browser compatibility
- Operating system compatibility
- Compatible to various devices like notebook, mobile, etc.

5. **Performance testing -** Performed to verify the server response time and throughput under various load conditions.

- **Load testing -** It is the simplest form of testing conducted to understand the behaviour of the system under a specific load. Load testing will result in measuring important business critical transactions and load on the database, application server, etc. are also monitored.
- **Stress testing -** It is performed to find the upper limit capacity of the system and also to determine how the system performs if the current load goes well above the expected maximum.
- **Soak testing -** Soak Testing also known as endurance testing, is performed to determine the system parameters under continuous expected load. During soak tests the parameters such as memory utilization is monitored to detect memory leaks or other performance issues. The main aim is to discover the system's performance under sustained use.
- **Spike testing -** Spike testing is performed by increasing the number of users suddenly by a very large amount and measuring the performance of the system. The main aim is to determine whether the system will be able to sustain the work load.

6. **Security testing -** Performed to verify if the application is secured on web as data theft and unauthorized access are more common issues and below are some of the techniques to verify the security level of the system.

- Injection
- Broken Authentication and Session Management
- Cross-Site Scripting (XSS)
- Insecure Direct Object References
- Security Misconfiguration
- Sensitive Data Exposure

- Missing Function Level Access Control
- Cross-Site Request Forgery (CSRF)
- Using Components with Known Vulnerabilities
- Unvalidated Redirects and Forwards.

Chatbot Testing Cases

1.Manually modify the session context: The tool saves the entire session context to a file session_context.json after receiving the response from Watson Assistant. It reads that file before sending the next message request. By using an editor, you could modify the session context and add, change or delete values. The new request would pick up those values and send them to Watson Assistant.

The process works well in general, but does not strictly follow the protocol for the client actions.

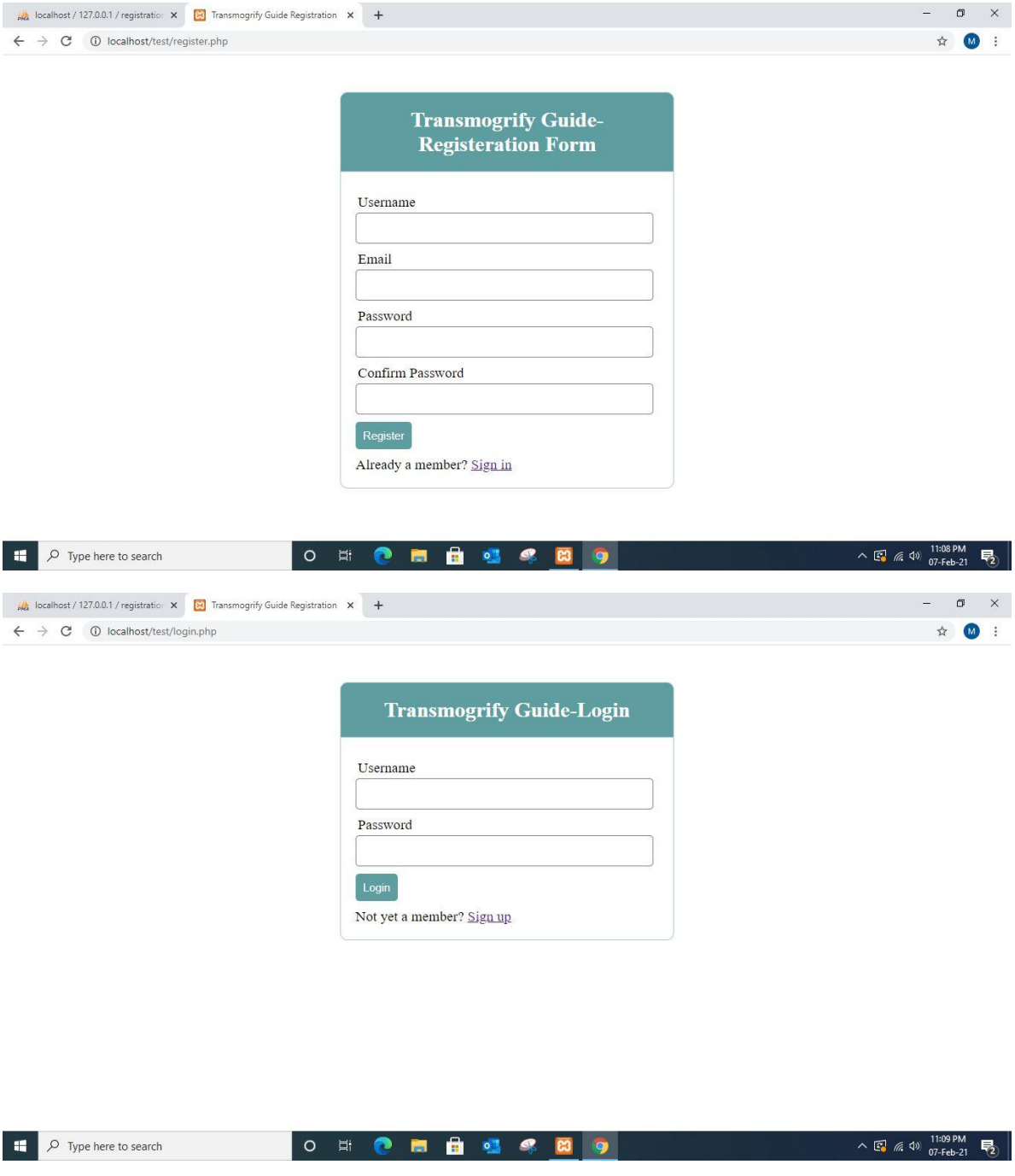
2.Provide code for / to simulate client action: As a new feature, the tool now accepts a parameter "-actionmodule name". That module is then dynamically loaded and a function handleClientActions called. That function either provides the regular logic for the client action or could be some code to simulate the action by assigning test values.

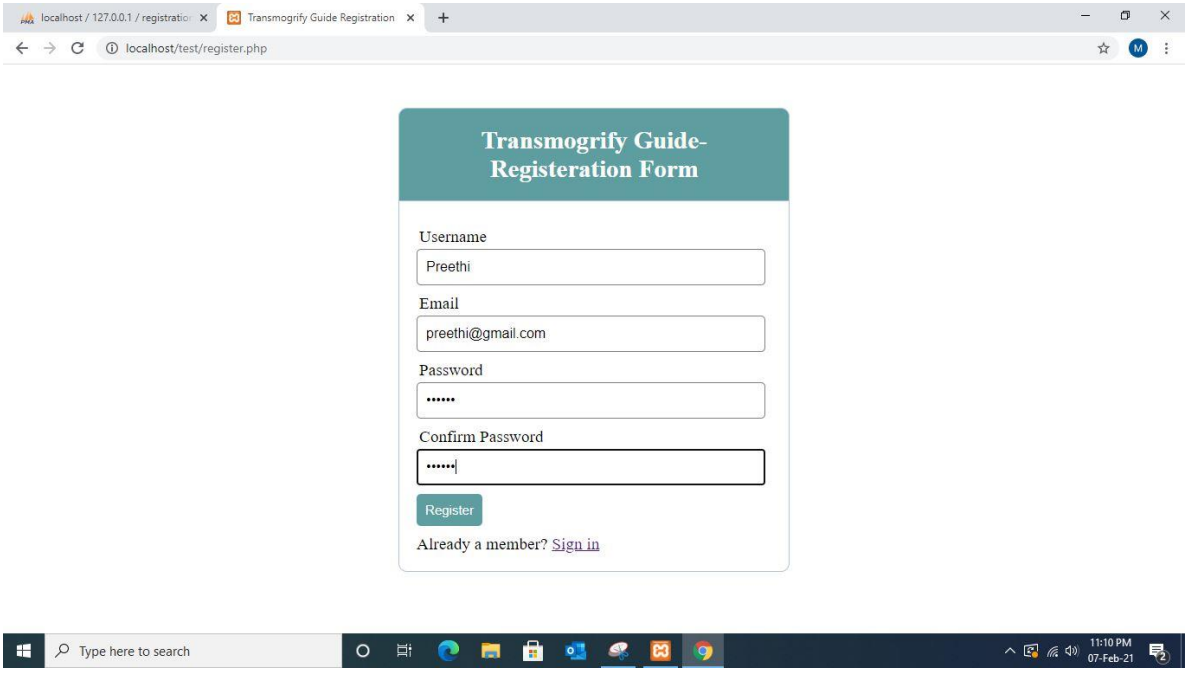
I have successfully used it to test some client actions with a Watson Assistant workspace. This way, you could test Watson Assistant independently of the actual client application.

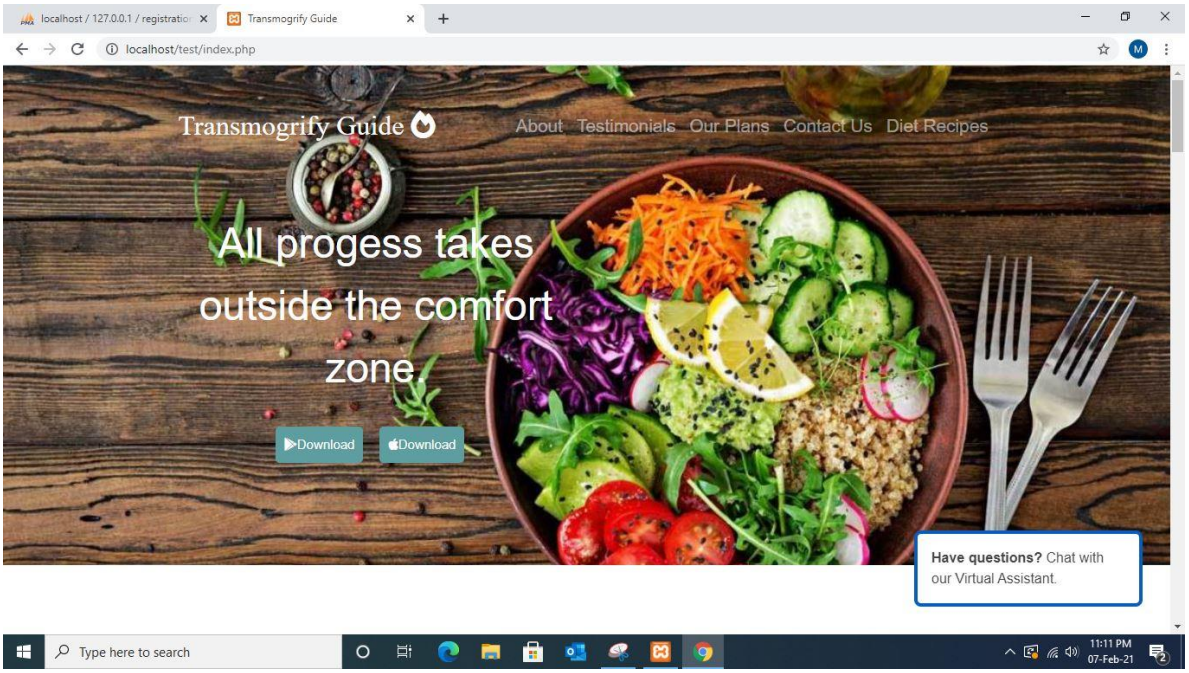
CHAPTER 7

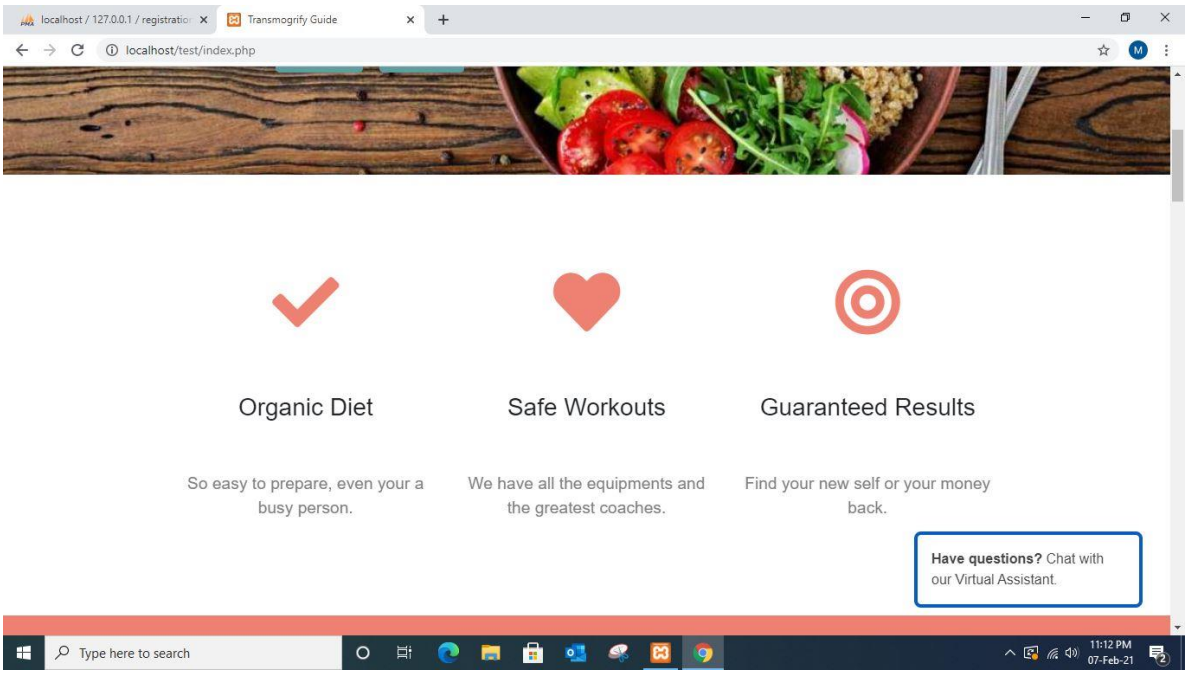
OUTPUT SCREENS

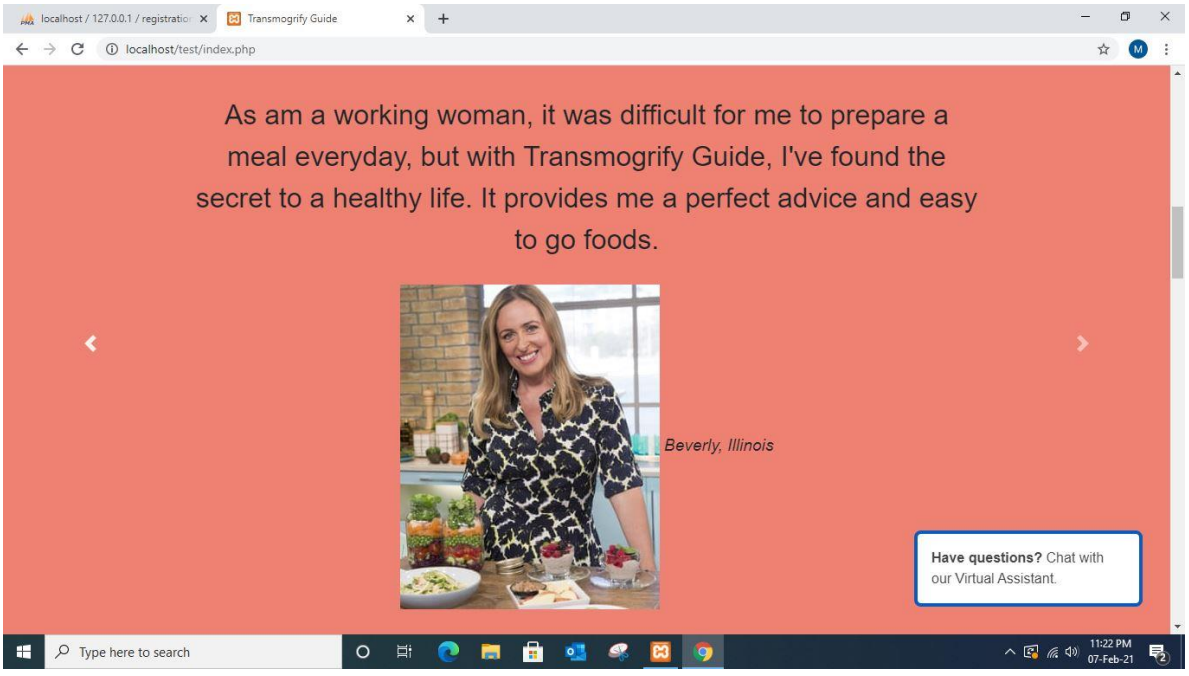
Fig 7.1 Output Screens

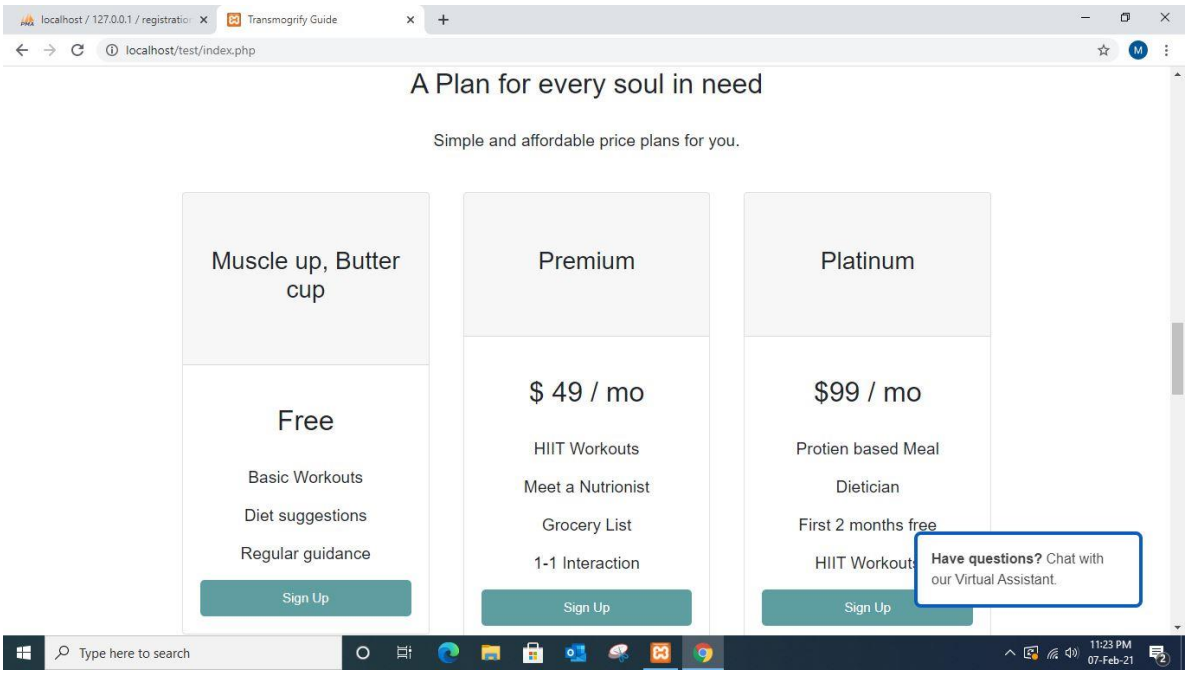


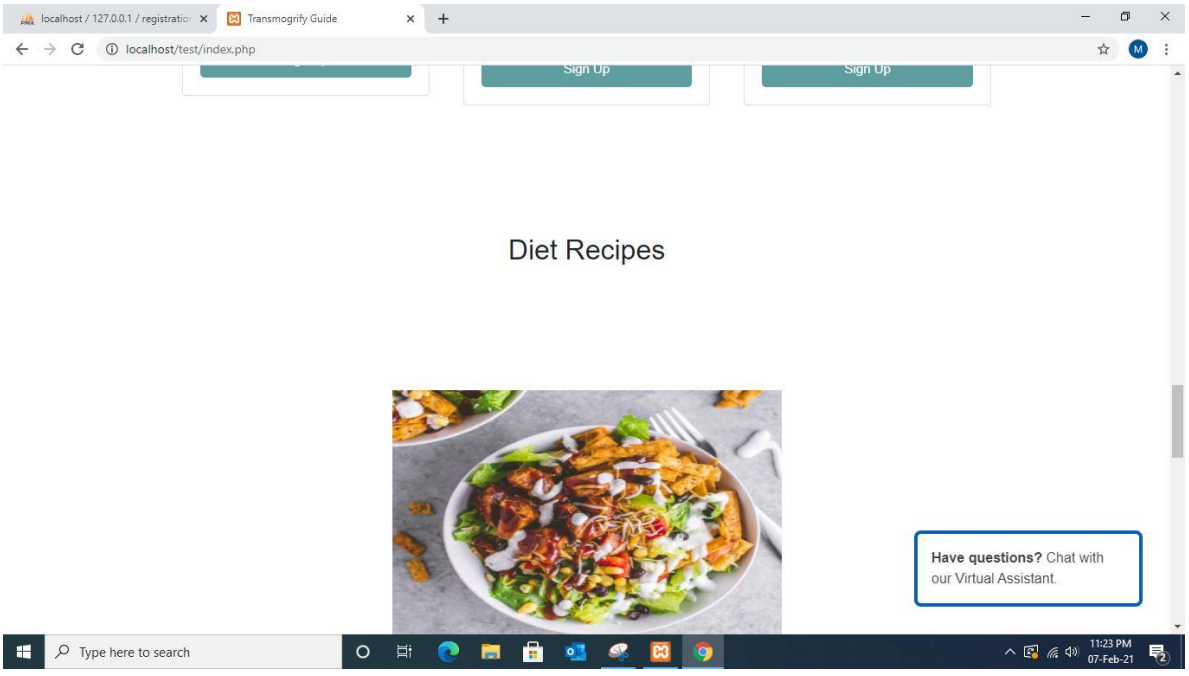

localhost / 127.0.0.1 / registration
Transmogrify Guide Registration
localhost/test/register.php
Transmogrify Guide-
Registeration Form
Username
Preethi
Email
preethi@gmail.com
Password
Confirm Password
Register
Already a member? Sign in
Type here to search
11:10 PM
07-Feb-21

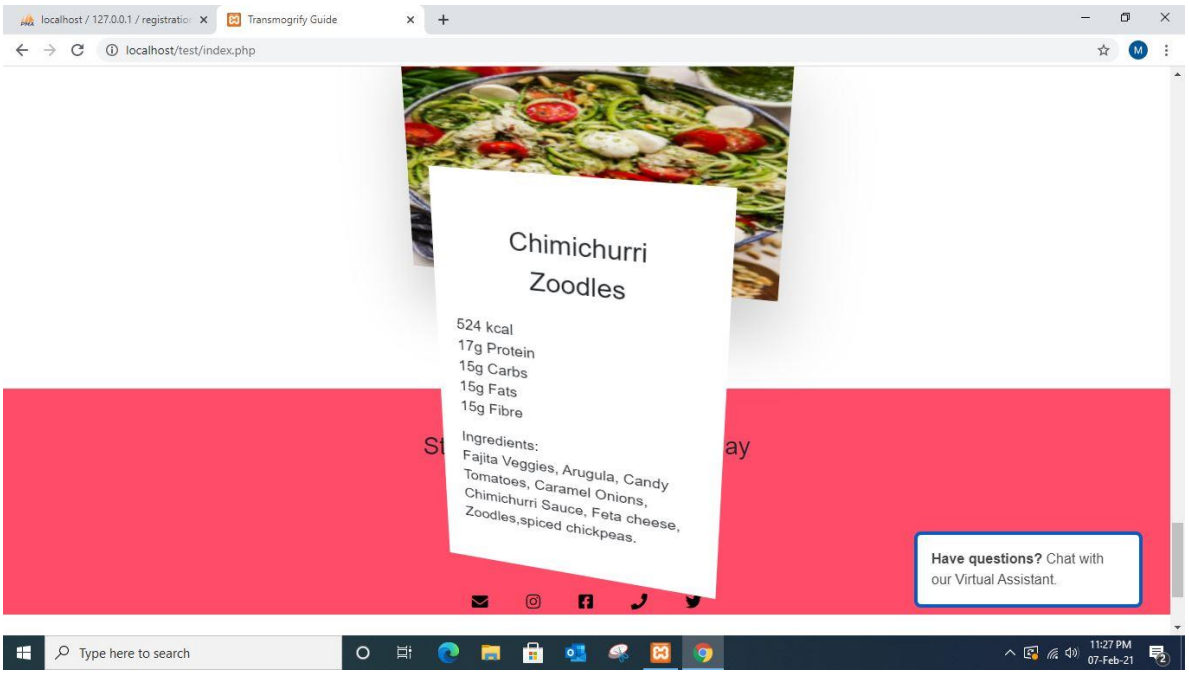

localhost / 127.0.0.1 / registration
Transmogrify Guide
localhost/test/index.php
Transmogrify Guide
About
Testimonials
Our Plans
Contact Us
Diet Recipes
All progess takes outside the comfort zone.
Download
Download
Have questions? Chat with our Virtual Assistant.
Type here to search
11:11 PM
07-Feb-21

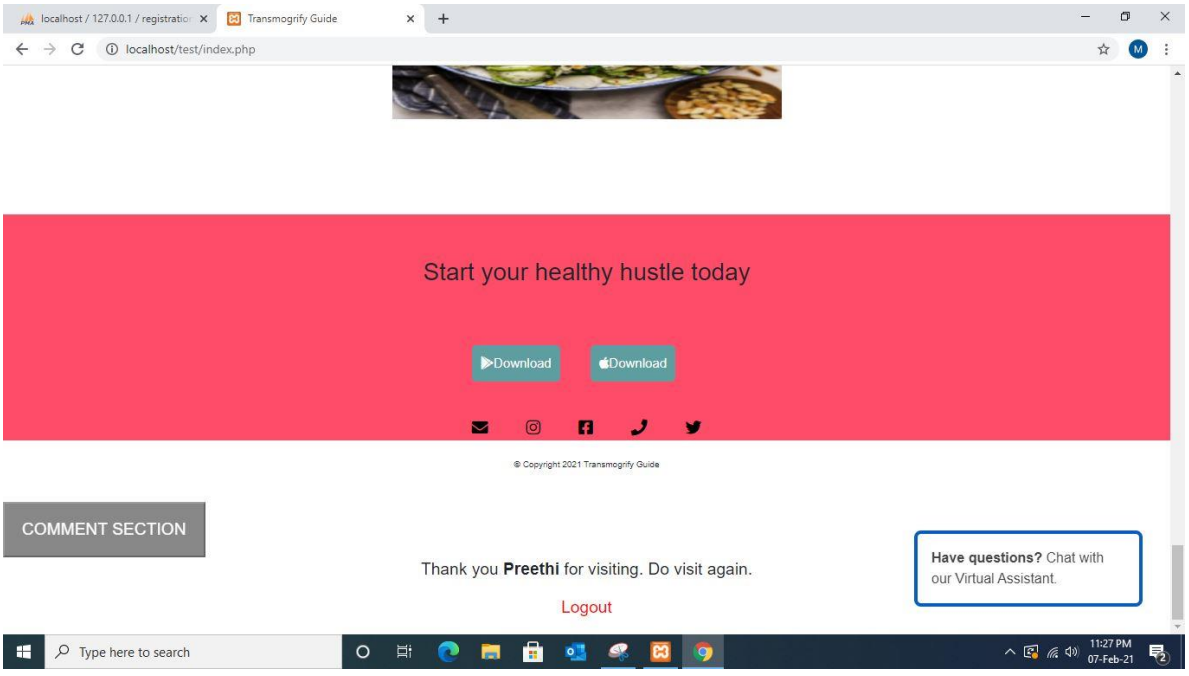

localhost / 127.0.0.1 / registratio
Transmogrify Guide
localhost/test/index.php
Organic Diet
So easy to prepare, even your a busy person.
Safe Workouts
We have all the equipments and the greatest coaches.
Guaranteed Results
Find your new self or your money back.
Have questions? Chat with our Virtual Assistant.
Type here to search
11:12 PM
07-Feb-21

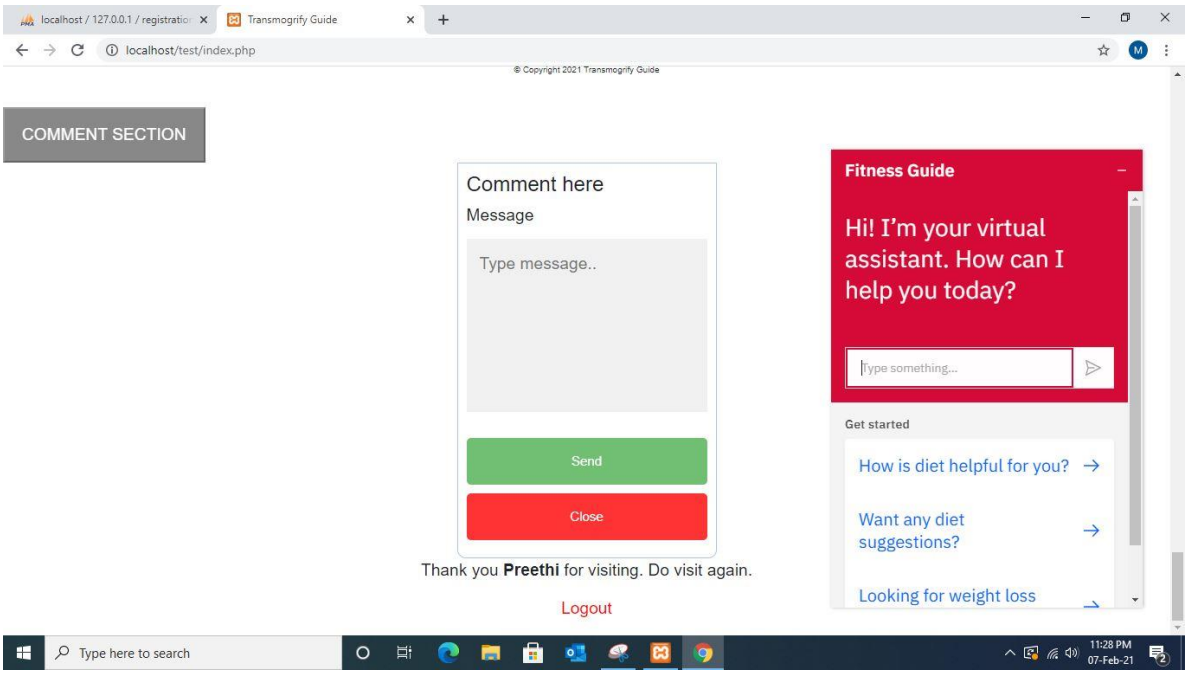

localhost / 127.0.0.1 / registratio
Transmogrify Guide
localhost/test/index.php
As am a working woman, it was difficult for me to prepare a meal everyday, but with Transmogrify Guide, I've found the secret to a healthy life. It provides me a perfect advice and easy to go foods.
Beverly, Illinois
Have questions? Chat with our Virtual Assistant.
Type here to search
11:22 PM
07-Feb-21

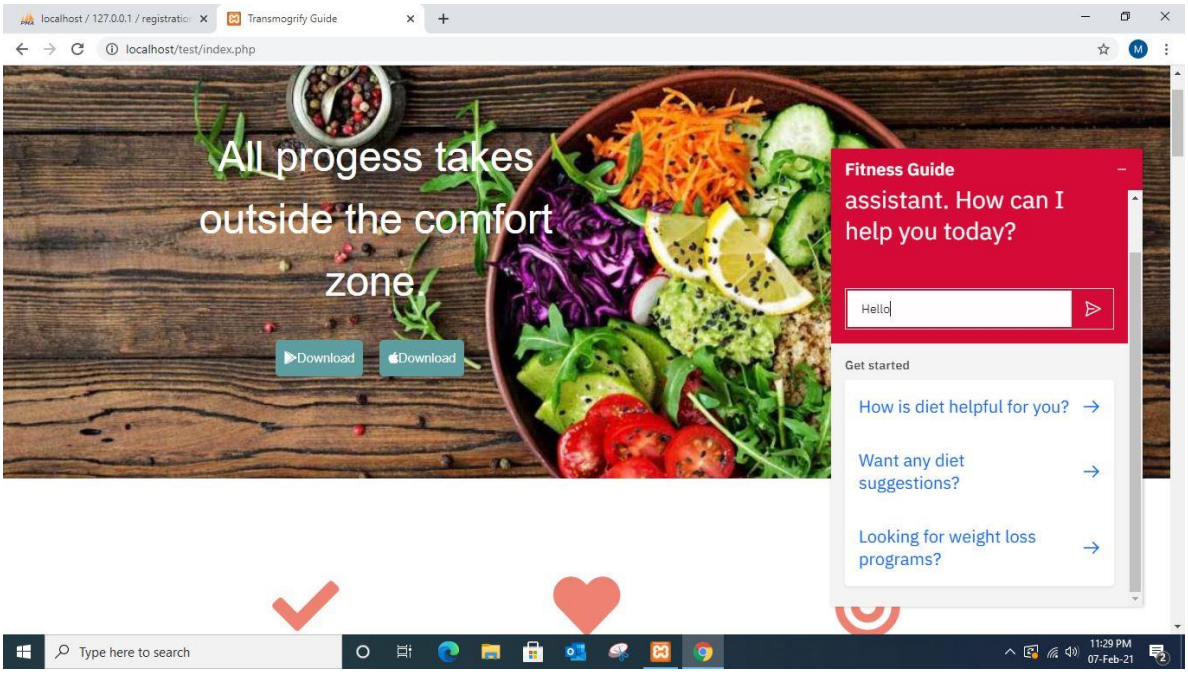

localhost / 127.0.0.1 / registratio
Transmogrify Guide
localhost/test/index.php
A Plan for every soul in need
Simple and affordable price plans for you.
Muscle up, Butter cup
Free
Basic Workouts
Diet suggestions
Regular guidance
Sign Up
Premium
$ 49 / mo
HIIT Workouts
Meet a Nutrionist
Grocery List
1-1 Interaction
Sign Up
Platinum
$99 / mo
Protien based Meal
Dietician
First 2 months free
HIIT Workout
Sign Up
Have questions? Chat with our Virtual Assistant.
Type here to search
11:23 PM
07-Feb-21

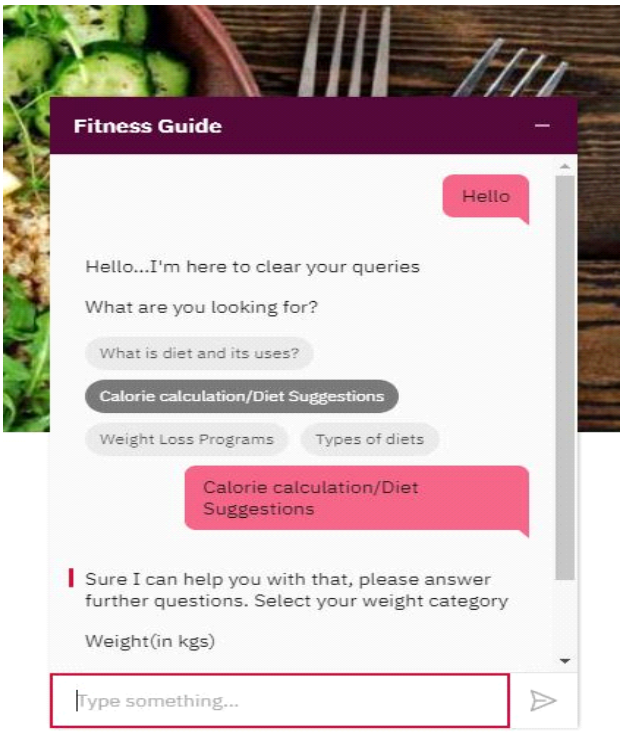

localhost / 127.0.0.1 / registratio
Transmogrify Guide
localhost/test/index.php
Sign Up
Sign Up
Diet Recipes
Have questions? Chat with our Virtual Assistant.
Type here to search
11:23 PM
07-Feb-21

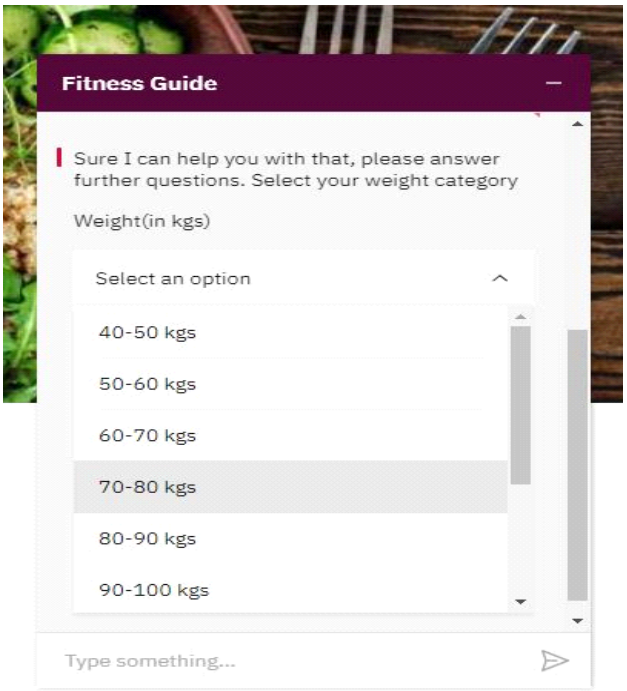

localhost / 127.0.0.1 / registratio
Transmogrify Guide
localhost/test/index.php
Chimichurri Zoodles
524 kcal
17g Protein
15g Carbs
15g Fats
15g Fibre
Ingredients:
Fajita Veggies, Arugula, Candy Tomatoes, Caramel Onions, Chimichurri Sauce, Feta cheese, Zoodles,spiced chickpeas.
Have questions? Chat with our Virtual Assistant.
Type here to search
11:27 PM
07-Feb-21

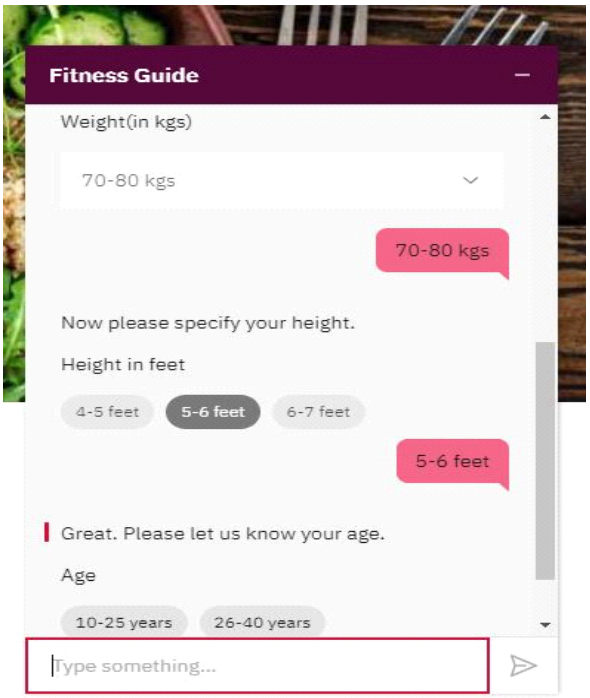

localhost / 127.0.0.1 / registratio
Transmogrify Guide
localhost/test/index.php
Start your healthy hustle today
Download
Download
© Copyright 2021 Transmogrify Guide
COMMENT SECTION
Thank you Preethi for visiting. Do visit again.
Logout
Have questions? Chat with our Virtual Assistant.
Type here to search
11:27 PM
07-Feb-21

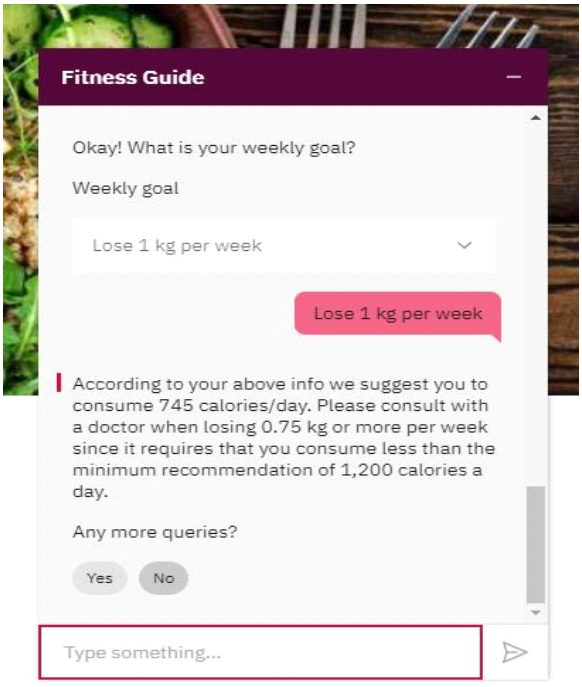

localhost / 127.0.0.1 / registratio
Transmogrify Guide
localhost/test/index.php
© Copyright 2021 Transmogrify Guide
COMMENT SECTION
Comment here
Message
Type message..
Send
Close
Thank you Preethi for visiting. Do visit again.
Logout
Fitness Guide
Hi! I'm your virtual assistant. How can I help you today?
Type something...
Get started
How is diet helpful for you?
Want any diet suggestions?
Looking for weight loss
Type here to search
11:28 PM
07-Feb-21


localhost / 127.0.0.1 / registratio
Transmogrify Guide
localhost/test/index.php
All progess takes outside the comfort zone.
Download
Download
Fitness Guide
assistant. How can I help you today?
Hello
Get started
How is diet helpful for you?
Want any diet suggestions?
Looking for weight loss programs?
Type here to search
11:29 PM
07-Feb-21


Fitness Guide
Hello
Hello...I'm here to clear your queries
What are you looking for?
What is diet and its uses?
Calorie calculation/Diet Suggestions
Weight Loss Programs
Types of diets
Calorie calculation/Diet Suggestions
Sure I can help you with that, please answer further questions. Select your weight category
Weight(in kgs)
Type something...


Fitness Guide
Sure I can help you with that, please answer further questions. Select your weight category
Weight(in kgs)
Select an option
40-50 kgs
50-60 kgs
60-70 kgs
70-80 kgs
80-90 kgs
90-100 kgs
Type something...


Fitness Guide
Weight(in kgs)
70-80 kgs
70-80 kgs
Now please specify your height.
Height in feet
4-5 feet
5-6 feet
6-7 feet
5-6 feet
Great. Please let us know your age.
Age
10-25 years
26-40 years
Type something...


Fitness Guide
Okay! What is your weekly goal?
Weekly goal
Lose 1 kg per week
Lose 1 kg per week
According to your above info we suggest you to consume 745 calories/day. Please consult with a doctor when losing 0.75 kg or more per week since it requires that you consume less than the minimum recommendation of 1,200 calories a day.
Any more queries?
Yes
No
Type something...

CHAPTER 8

REFERENCES

[1] Bayu Setiaji, Ferry Wahyu Wibowo, “Chatbot Using A Knowledge in Database Human-to-Machine Conversation Modeling”, Published in: IEEE 7th International Conference on Intelligent Systems, Modeling and Simulation, 2016.

[2] Ayah Atiyah, Shaidah Jusoh, Sufyan Almajali, “An Efficient Search for Context-Based Chatbots”, Published in: 8th International Conference on Computer Science and Information Technology (CSIT), 2018.

[3] Ashay Argal ; Siddharth Gupta ; Ajay Modi ; Pratik Pandey ; Simon Shim ; Chang Choo , “Intelligent travel chatbot for predictive recommendation in echo platform”, Published in: IEEE 8th Annual Computing and Communication Workshop and Conference (CCWC), 2018.

[4] Hyo Jin Do and Wai-Tat Fu, Department of Computer Science: “Empathic Virtual Assistant for Healthcare Information with Positive Emotional Experience”, Published in:IEEE International Conference on Healthcare Informatics, 2016.

[5] BayuSetiaji, Ferry WahyuWibowo: “Chatbot Using a Knowledge in Database: Human-to-Machine Conversation Modeling“, Published in: Intelligent Systems, Modelling and Simulation (ISMS), 2016 7th International Conference, 2016.

[6] “Dialogflow,” https://dialogflow.com/, accessed: 2017

[7] Divya S, Indumathi V, Ishwarya S, Priyasankari M, Kalpana Devi S,”A Self-Diagnosis Chatbot Using Artificial Intelligence”, Published in: Journal of Web Development and Web Designing Volume 3 Issue 1, 2018