

## **Lab Project Submission**

submitted for

### **Software Engineering (UCS503)**

#### **Calorie-Click: Total Nutritional Calculator**

submitted by

**Aryan Garg                      102103748**

**Samarth Paliwal              102103775**

**3 COE 27**

submitted to

**Dr. Nitigya Sambyal**



Computer Science and Engineering Department

Thapar Institute of Engineering and Technology, Patiala

## Software Bid/ Project Teams

### UCS 503- Software Engineering Lab

Group: 3Co27

Dated: 29 August 2023

**Team Name: O554**

**Team ID (will be assigned by Instructor):**

Please enter the names of your Preferred Team Members:

- You are required to form **a three to four person teams'**
- Choose your team members wisely. You will not be allowed to change teams.

Name	Roll No	Project Experience	Programming Language used	Signature
Aryan Garg	102103768	RNN Based Story Generator, Full Stack Data Engineering	Python	
Samarth Paliwal	102103775	Hop Shop Ecommerce Store, News Website	ReactJs, Node, MongoDB, Express, Python	

### Programming Language / Environment Experience

List the languages you are most comfortable developing in, **as a team**, in your order of preference. Many of the projects involve Java or C/C++ programming.

1. Python
2. JavaScript

### Choices of Projects:

Please select **4 projects** your team would like to work on, by order of preference:

First Choice	Image Based Calories and Macro Nutrients tracker Webapp
Second Choice	AI based News delivery Webapp
Third Choice	Ham Spam classifying Chrome Extension
Fourth Choice	Parking validator using image Processing

### **Additional Remarks/ Inputs**

Please tell us about any other factors that we should take into consideration (e.g., if you really would like to work on a project for some particularly convincing reason).

We believe that we can work on this project with ease and will be able to do justice to the problem statement with our relevant skill set.

---

# TABLE OF CONTENTS

Chapter No.	Topic	Page No.
1.	<a href="#">Introduction</a>	5
1.1	<a href="#">Purpose of this Document</a>	5
1.2	<a href="#">Scope of the Development Project</a>	5
1.3	<a href="#">Definitions, abbreviations and acronyms</a>	6
1.4	<a href="#">References</a>	7
1.5	<a href="#">Overview</a>	7
2.	<a href="#">Overall Description</a>	8
2.1	<a href="#">Product Perspective</a>	9
2.2	<a href="#">Product functions</a>	9
2.3	<a href="#">User Characteristics</a>	9
2.4	<a href="#">General Constraints, Assumptions and Dependencies</a>	10
2.5	<a href="#">Apportioning of the requirements</a>	10
3.	<a href="#">Specific Requirements</a>	11
3.1	<a href="#">External Interface Requirements</a>	11
3.2	<a href="#">Detailed Description of Functional Requirements</a>	11
3.2.1	<a href="#">Functional Requirements for Sign-up Screen</a>	12
3.2.2	<a href="#">Functional Requirements for Log-in Screen</a>	12
3.2.3	<a href="#">Functional Requirements for Upload Image Screen</a>	12
3.2.4	<a href="#">Functional Requirements for Statistics Page</a>	12
3.3	<a href="#">Performance requirements</a>	13
4.	<a href="#">Document Approvers</a>	14

# 1.Introduction

## 1.1 Purpose of this Document

The purpose of this SRS document is to provide a detailed overview of our software product, its parameters, and goals. This document describes the project's target audience, the problem it solves and its user interface, hardware, and software requirements. **It defines how our client, team and audience see the product and its functionality.**

## 1.2 Scope of the Development Project

The goal is to design software which can detect various food items on the predictions of a pre-trained ML model and generate nutritional information about them. **In this system, a user will upload images of their food to perform scan on the image provided. The Deep Learning model will study the image and detect the class of food it is. (Which is upto 20 classes). From this, we can extract nutritional information about the food from our static database and add it into our system for further charting. The user can further graph the information and look at their habits developed over a span of a day, a week or even a month. All the details of the user, their images and statistics made will be stored in database server and will also be replicated in a backup server daily to avoid data loss. The ML model will be updated time to time to increase the accuracy of prediction model.**

The software must be able to perform the following operations:

- 1. Identify and authenticate the actor:** It must be able to authenticate the user logging in the system, where each user has its own confidential information and where they can log their own stats.
- 2. Scan and predict the correct meal:** The software must be able to scan the uploaded images and predict the meal correctly based on the training data and classifications applied on the pretrained model.
- 3. Generate Statistical Graphs:** The software will generate nutritional information about the meal. It should at least contain the minimal nutritional information that we are accustomed to, like calories, protein, and fiber.

**6. Update ML model when required:** The software should provide the admin access to update the ML prediction model as and when required to increase the accuracy of predictions. It must authenticate the admin before allowing updating rights.

Initially we plan to implement these functionalities on 10 food items as a part of the **Pilot Phase** to check the authenticity and accuracy of our model on new set of images. After the successful completion of the pilot phase, we plan to make this software publicly available for everyone, with more food items added. In the **Third Phase** we plan to integrate more food items, and instead of creating this as a classification problem, we may treat this as a regressive problem to find the calories in ANY food item, irrespective its data is present in the database.

This will help the users who need specific nutritional needs by allowing them to easily look into their meals and keep a track of their daily intake. Slowly as the accuracy of the model increases, this software can be of great help to all the users to identify all types of food.

### **1.3 Definitions, abbreviations, and acronyms**

#### **Definitions**

Table 1 gives explanation of the most used terms in this SRS document.

**Table 1: Definitions for most used terms**

<b>S.No.</b>	<b>Term</b>	<b>Definition</b>
<b>1</b>	Machine Learning	Machine learning is a branch of artificial intelligence (AI) and computer science which focuses on the use of data and algorithms to imitate the way that humans learn, gradually improving its accuracy. [1]
<b>2</b>	Pre-Trained model	A pre-trained model is a saved network that was previously trained on a large dataset, typically on a large-scale image-

		classification task. [2]
<b>3</b>	CNN	A convolutional Neural Network, is a machine learning model, that uses convolutions to help process images. [3]
<b>4</b>	Authentication	A user authentication policy is a process in which you verify that someone who is attempting to access services and applications is who they claim to be. [4]

## 1.4 References

- [1] Machine Learning, LINK: <https://www.ibm.com/topics/machine-learning>
- [2] Pre-Trained model, LINK: [https://www.tensorflow.org/tutorials/images/transfer\\_learning](https://www.tensorflow.org/tutorials/images/transfer_learning)
- [3] CNN, LINK: <https://www.techtarget.com/searchenterpriseai/definition/convolutional-neural-network>
- [4] Authentication, LINK: <https://www.cisco.com/c/en/us/products/security/identity-services-engine/what-is-user-authentication-policy.html>

## 1.5 Overview

The remaining sections of this document provide a general description, including characteristics of the users of this project, the product's hardware, and the functional and data requirements of the product. General description of the project is discussed in section 2 of this document. Section 2 gives the functional requirements, data requirements and constraints and assumptions made while designing the multi-utility system. It also gives the user viewpoint of product use. Section 3 gives the specific requirements of the product. Section 3.0 also discusses the external interface requirements and gives detailed description of functional requirements

## 2. Overall Description

### 2.1 Product Perspective

The software will run as website which can be accessed by users. The user is

supposed to upload the meal images to know the nutritional information of their meal.

Here the user will have to login using the registered credentials. If the user is a first-time visitor they will have to register before going ahead. Once the registration is completed the user can login to their account. When any user tries to login their login credentials are matched with the already saved data in the user database. If the credentials match "Access Granted" or else "Access Denied" message is displayed.

After the authentication process, the Meal Detection software has 2 more modules:

1. Upload Image
2. Meal Statistics

Once the user is **authenticated**, they can upload the image of their meal to the software. The ML model running in the backend of the product takes the image as input and performs the specified tasks. The model then predicts the meal according to the pre-trained sets and gives an output to the system.

Once the output is generated by the model, the system then **meal specific statistics** which contains all the necessary information about the meal (Calories, fiber, protein etc.). This is published and can be viewed by the user. The same report gets stored in the user database for future reference.

In addition to the two modules, a **Central database server** and a **Backup database server** will also be used in order to read/write data onto the repository. The central database server will periodically update the Backup database server so that in case of server failure it can restore the data by retrieving the records stored in the Backup database server's tables.

## 2.2 Product Functions

The product should be able to perform the following operations:

1. It must be able to authenticate the user by matching the Username and Password which is created at the time of registration and stored in the database.



2. The system should provide appropriate access rights to the user according to the role of the user.
3. The software must allow the user to upload images of the food image and should predict the disorder accurately using the pre-trained ML model.
4. It must provide the user with a detailed statistical information about the meal and provide all the required information including Calories, protein, fiber.
5. It must provide the admin the right to update the ML model as and when needed so as to improve the accuracy of data prediction.

### **2.3 User Characteristics**

The goal is to design software for meal detection using ML and providing users check the calorific values of meals. These user types are listed below as follows:

1. General User
2. Admin/ Developer

As one can see that the patients can belong to different educational background and expertise level in using the system. Our goal is to develop a software that should be easy to use for all types of users, including patients from rural India as well. Thus, while designing the software one can assume that each user has the following characteristics:

- The user has the ability to take photos on the website, either by using their phone, or uploading them later on another device
- The user has a direct connection with the internet at a minimal speed of at least 0.5 MBPS.

### **2.4 General Constraints, Assumptions and Dependencies**

The following list presents the constraints, assumptions, dependencies, or guidelines that are imposed upon implementation of the Meal Detection System:

- As images of high resolution will be uploaded by user the system and the

model should be ready enough to process these images

- There are no memory requirements
- The product must have a user-friendly interface that is simple enough for all types of users to understand.
- Response time for loading the software and for processing a transaction should be no longer than 10 seconds.
- A general knowledge of basic computer skills is required to use the product.
- Strong Authentication and Encryption techniques are required to keep the data secure from any user data leak.
- The central database server and backup database servers should be updated regularly. This updating and replication of data from central database server to the backup database server can introduce additional latency in the working of the system.
- The replication of data from central to the backup server has to be Asynchronous as it also provides a greater amount of protection by extending the distance between the primary and secondary locations of the data. Increased distances can provide protection from local events as the loss of a power grid, as well as natural disasters such as earthquakes and hurricanes.

## 2.5 Apportioning of requirements

The Meal nutrition Detection is to be implemented in the following three phases:

1. **Pilot Phase:** Here the meal nutrition detection system will be first tested on images collected from the internet as well as from friends and family to maintain the accuracy of the pre-trained ML model. This will help us ensure that the model is not providing any wrong predictions.
2. **Public Awareness:** Following the successful completion of the pilot phase, we plan to make this software available for the use of public first. This will give them the general idea of the new technology to detect food items from images and give detailed information about their nutrients.
3. **Adding Additional Food Items:** Following the general like towards the product, we can add new food categories to increase the usability of the product.

### 3. Specific Requirements

#### 3.1 External Interface Requirements

The following list represents the external interface requirements:

- **User Interface:** The app should have a user-friendly interface to capture food photos. It should also display the predicted calorie count and any additional relevant information to users.
- **Camera Integration:** The app should have access to the device's camera to capture food images. Integration with the camera API on both mobile and desktop z platforms is essential
- **Calorie Database:** Access to an external calorie database or API to retrieve nutritional information for various food items. This database should be regularly updated and accurate.
- **Security of user data** is a big feature the software should be capable of handling.
- **As such no hardware devices are required** for the software but can later be integrated with the development in the project.

#### 3.2 Detailed Description of Functional Requirements

Table 3 shows a template that we'll be using to describe functional requirements for two users, General User and Admin/Developer

**Table 3: Template for describing functional requirements**

<b>Purpose</b>	A description of the functional requirements and its reasons
<b>Inputs</b>	What are the inputs; in what form will they arrive; from what sources can the inputs come; what are the legal domains of each input.
<b>Processing</b>	Describes the outcome rather than the implementation; includes any validity checks on the data, exact timing of operation (if needed), how to handle unexpected or abnormal situations
<b>Outputs</b>	The form, shape, destination, and volume of output; output timing; range of parameters in the output; unit of measure of the output; process by which output is stored or destroyed; process for handling error message produced as output.

### 3.2.1 Functional Requirements for Sign-in Screen

<b>Purpose</b>	Users can enter their information and create UserID and Password, which is stored in the database for future logins
<b>Inputs</b>	Name, Date of Birth, UserID, Password, Confirm Password
<b>Processing</b>	All the entered information is stored in the database, and passwords will be hashed for security
<b>Outputs</b>	A new account is created for the user

### 3.2.2 Functional Requirements for Log-in Screen

<b>Purpose</b>	To authenticate users, and validate their credentials
<b>Inputs</b>	UserID, Password
<b>Processing</b>	The ID and password are sent to the database and compared with the stored value. When validated users are directed to the home screen. On failure the process is repeated
<b>Outputs</b>	The user is logged in to the account

### 3.2.3 Functional Requirements for Food Upload Page

<b>Purpose</b>	To upload Food Image
<b>Inputs</b>	Image of Food on the device, can also take a photo on the spot with a camera.
<b>Processing</b>	The food image is fed into the pre-trained ML model. The model evaluates the image based on training data and determines which class the food belongs to
<b>Outputs</b>	The page determines which food the class belongs to

### 3.2.4 Functional Requirements for Statistics Page

<b>Purpose</b>	To check food logs
<b>Inputs</b>	None
<b>Processing</b>	Process previous data and generate meaningful data to the end user
<b>Outputs</b>	Tables or graphs of the food consumed

### 3.3 Performance Requirements

- The software is designed as a website and can run from a standalone desktop PC.
- The software will support simultaneous user access **only if there are multiple-terminals.**
- **The software can handle both pictorial and textual data.** Amount of information to be handled vary from user to user.
- **For normal conditions, 95% of the transactions should be processed in less than 10 seconds.**

### 4. Document Approvers

SRS for Calorie Click: Image Based food recognition and nutrient information web-app approved by: \_\_\_\_\_

Designation:

Date:

# **Feasibility Report**

## **Project Overview:**

The "Calorie Click" project aims to revolutionize the way individuals access nutritional information by leveraging image recognition technology. The primary objective is to develop a user-friendly mobile application that allows users to obtain detailed nutritional information about food items simply by capturing a photo.

## **Technical Feasibility:**

- **Technical Expertise:** Our team possesses the necessary technical expertise in Web Development, Machine Learning and Database Management.
- **Technology Stack:** We plan to use industry-standard web development technologies such as HTML, CSS, JavaScript, ML Models like CNN, and a backend framework.
- **Infrastructure:** We will need dedicated servers or cloud hosting to handle system scalability and performance requirements.
- **Data Privacy and Security:** Address data privacy and security concerns, including compliance with relevant regulations.

## **Operational Feasibility:**

- **User Acceptance:** Data will be tested on dataset to achieve the best accuracy.
- **Consultants Support:** Engaging with a multidisciplinary team of consultants, including machine learning specialists, medical experts, and regulatory advisors, to ensure comprehensive support for developing the Automated Meal detection system.
- **Resource Availability:** Necessary resources, including personnel and technology, are available or can be acquired within the project timeline.

## **Economic Feasibility:**

- **Cost Estimation:** The project will require initial investments in software development, machine learning model development, and data acquisition. Ongoing operational costs include maintenance, cloud computing expenses, and potential personnel costs. A detailed budget analysis is necessary to determine the economic feasibility.

## **Scheduling Feasibility:**

- **Project Timeline:** The project timeline is reasonable, with development, testing, and deployment expected to take approximately 2-3 months for the website along with a working model, and require 9-12 months for full implementation.
- **Resource Allocation:** Resources, including human resources and technology infrastructure, can be allocated according to the project timeline.

### **Legal Feasibility:**

- **Regulatory Compliance:** The project will adhere to all relevant data protection regulations (e.g., GDPR)

### **Security Feasibility:**

- **Secured Software:** The project will apply SHA256 encryption to hash and save the passwords. The payment gateway will also be made secure to prevent any attack.

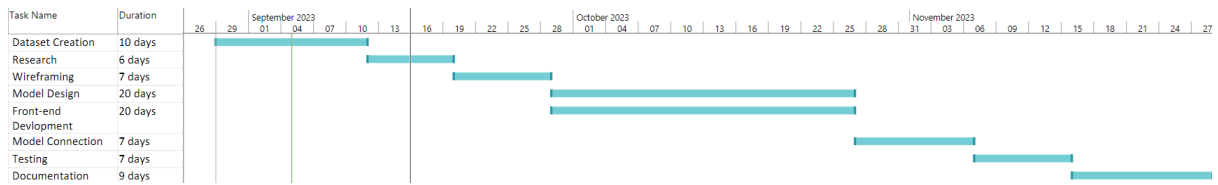
### **Cultural Feasibility:**

- **Diverse Cultural Data:** Collecting a wide range of cultural data to avoid biases and stereotypes.
- **User Preferences:** Allowing users to self-identify their cultural backgrounds and preferences to ensure personalized
- **Ethical Guidelines:** Adhering to ethical guidelines and involving cultural experts in the development process to avoid cultural insensitivity.

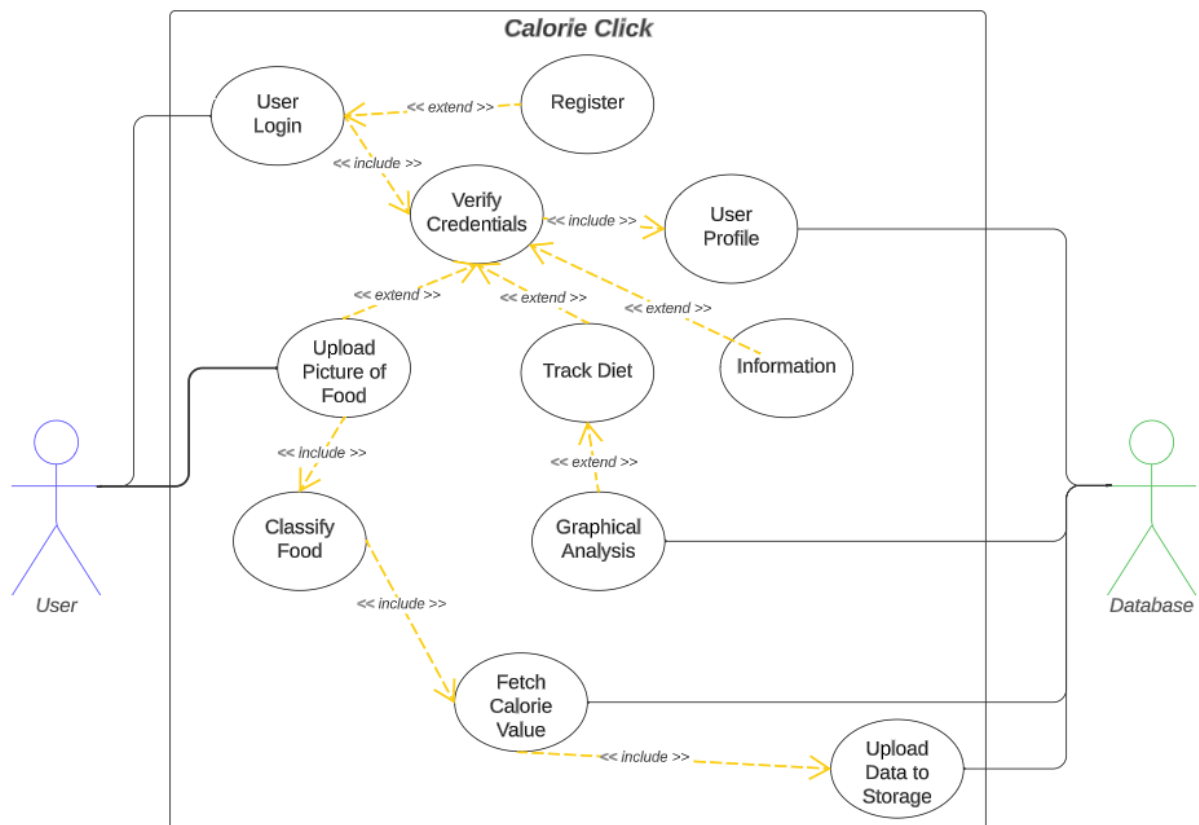
### **Conclusion:**

The development of Calorie Click (Meal Tracking System) is technically and economically feasible. However, it requires substantial data, technical expertise, and financial resources. Ensuring data privacy and security, scalability, and user adoption will be key operational challenges. By addressing these challenges and maintaining a focus on building credibility, the system can provide accurate predictions based on calorific values, enhancing our authenticity.

## GAANT CHART



## Use Case Diagram





## Use Case Template

1. Use Case Title	User Login
2. Abbreviated Title	User Login
3. Use Case ID	1
4. Actors	User
5. Description:	Login by filling in user email id and password to access the software.
5.1. Pre-Conditions:	User must be signed up and verified
5.2. Task Sequence:	<ul style="list-style-type: none"> <li>Enter verified email id.</li> <li>Enter set user password.</li> <li>Verify by clicking on “Log In”.</li> </ul>
5.3. Post Conditions:	User is logged in and can now access the software.
6. Modification History:	Date 17-September-2023
7. Author:	Aryan, Samarth

1. Use Case Title	Upload Food Image
2. Abbreviated Title	Upload Picture of Food
3. Use Case ID	2
4. Actors	User
5. Description:	User can upload any food item image and he/she will get the various nutrients information like protein, calories etc. present in food item.
5.1. Pre-Conditions:	User must be logged in to the software.
5.2. Task Sequence:	<ul style="list-style-type: none"> <li>Click on ‘Upload Image’ to upload an image.</li> </ul>
5.3. Post Conditions:	User will get a confirmation message about the food item image he/she uploads.
6. Modification History:	Date 17-September-2023
7. Author:	Aryan, Samarth

1. Use Case Title	Full Day Diet Nutrient log
2. Abbreviated Title	Track Diet
3. Use Case ID	3
4. Actors	User
5. Description:	The user can review their entire day's dietary intake, including an analysis of various nutrients consumed, to determine whether they have met their daily nutritional requirements.
5.1. Pre-Conditions:	<ul style="list-style-type: none"> <li>User must have an active account and must be logged in.</li> </ul>
5.2. Task Sequence:	<ul style="list-style-type: none"> <li>User need to click on “Nutrient Log”.</li> </ul>

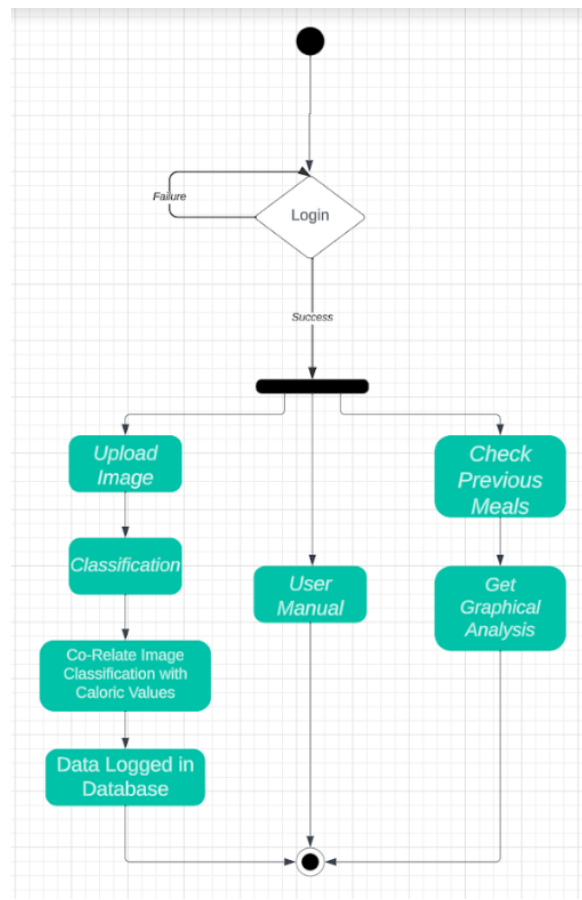
**5.3. Post Conditions:**

The user will receive nutritional information for their different meals.

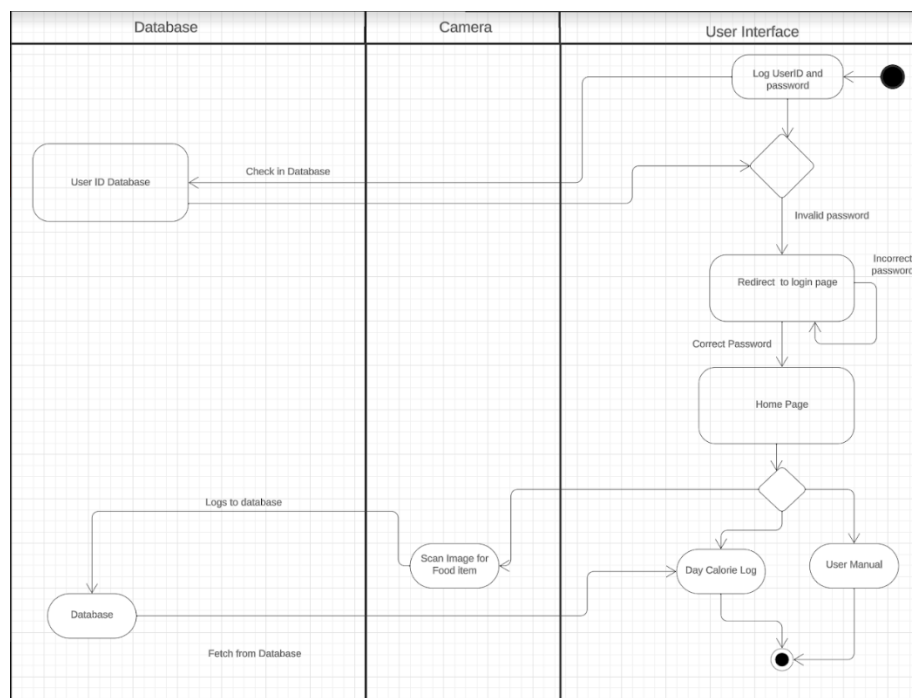
**6. Modification History:** Date 17-September-2023**7. Author:** Aryan, Samarth

<b>1. Use Case Title</b>	User Manual
<b>2. Abbreviated Title</b>	Information
<b>3. Use Case ID</b>	4
<b>4. Actors</b>	User
<b>5. Description:</b> Users can learn how to use the software	
<b>5.1. Pre-Conditions:</b> User must be logged in to the software.	
<b>5.2. Task Sequence:</b> <ul style="list-style-type: none"><li>Click on “User Manual” button</li></ul>	
<b>5.3. Post Conditions:</b> User will get Information about how the website works	
<b>6. Modification History:</b> Date 17-September-2023	
<b>7. Author:</b> Aryan, Samarth	

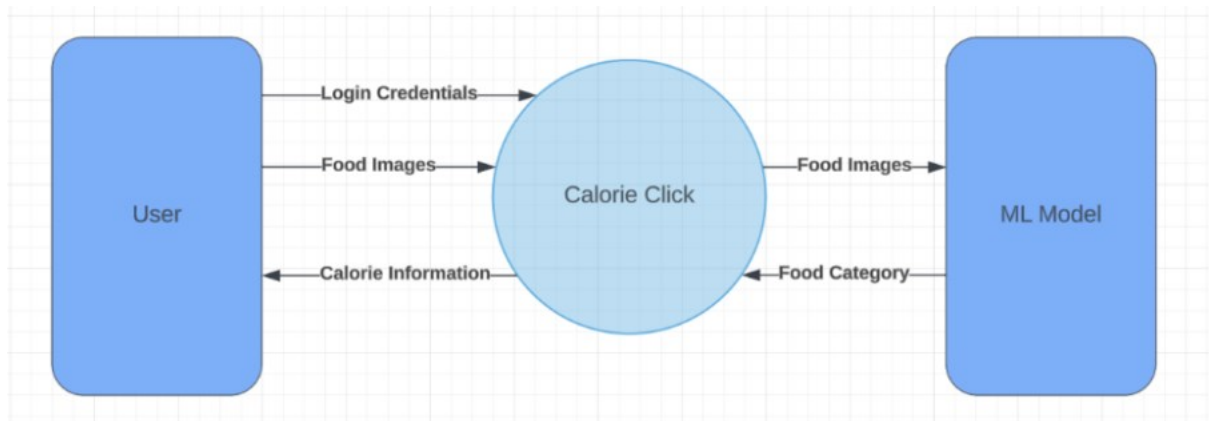
## Activity Diagram



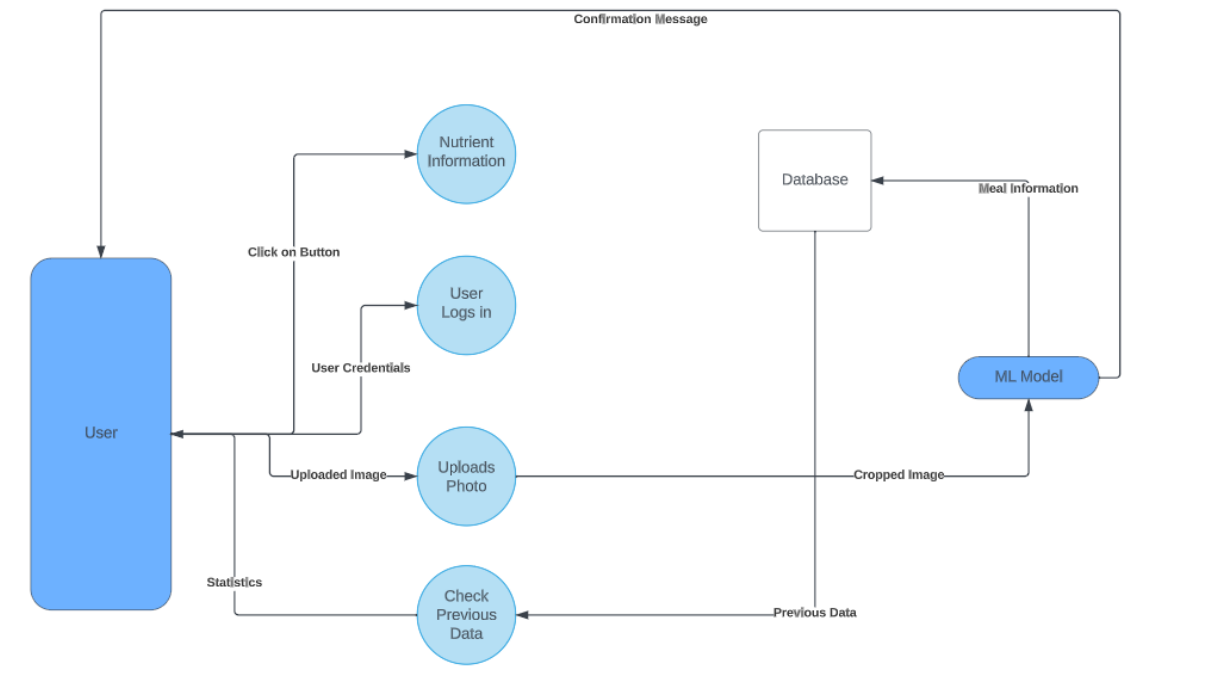
## Swimlane



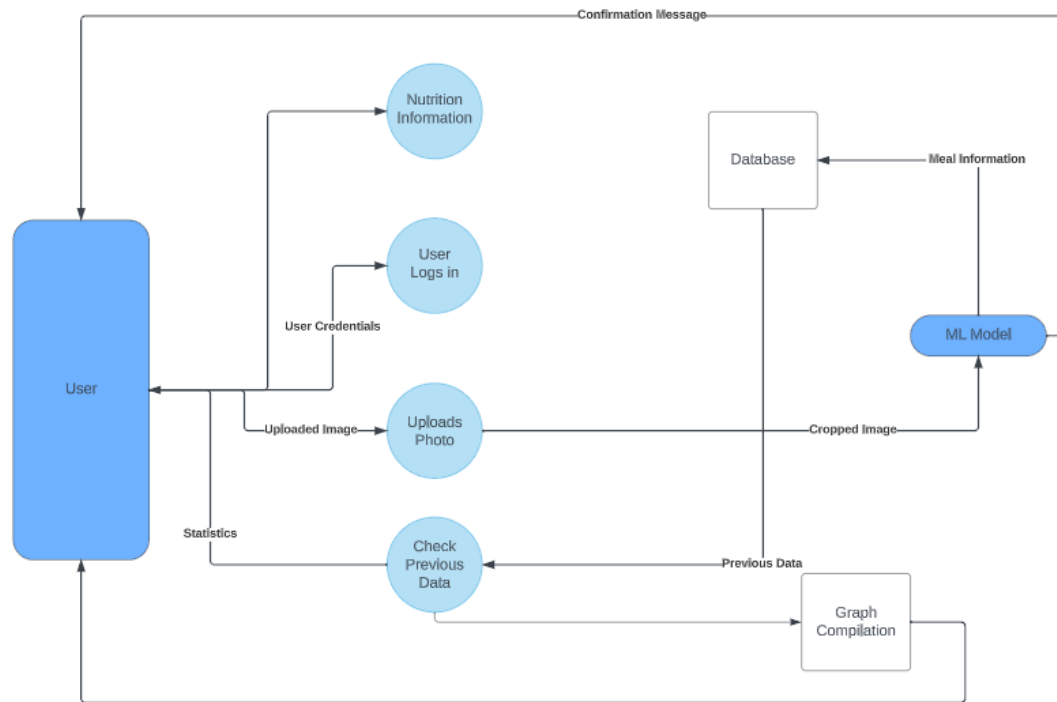
## Data Flow Diagram-0



## Data Flow Diagram-1



## Data Flow Diagram-2



#001	User Login	
As a [registered user], I want to [login],so I can [access my nutritional information]		

SIGN IN

UserID: Validate Format

Username

Password

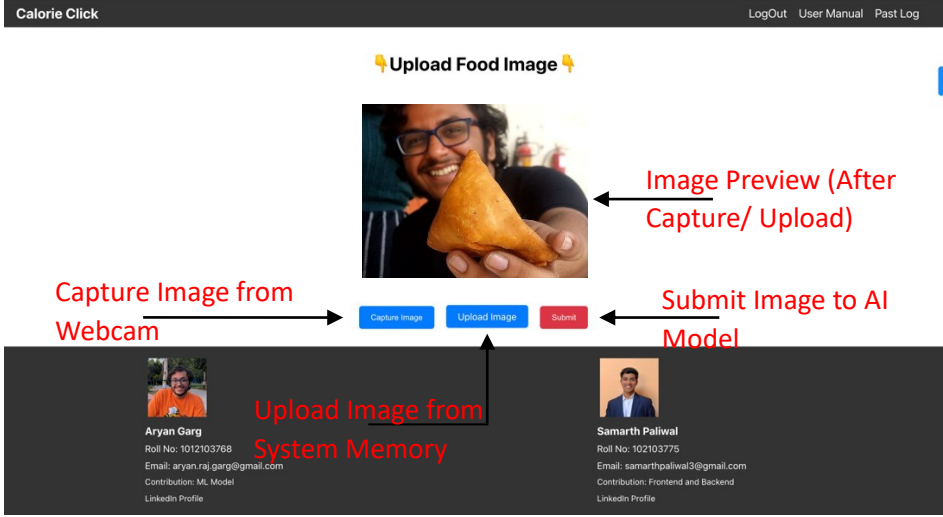
Authentication against  
JWT Service

LOGIN

Something went wrong...
← Error Messages (if any)

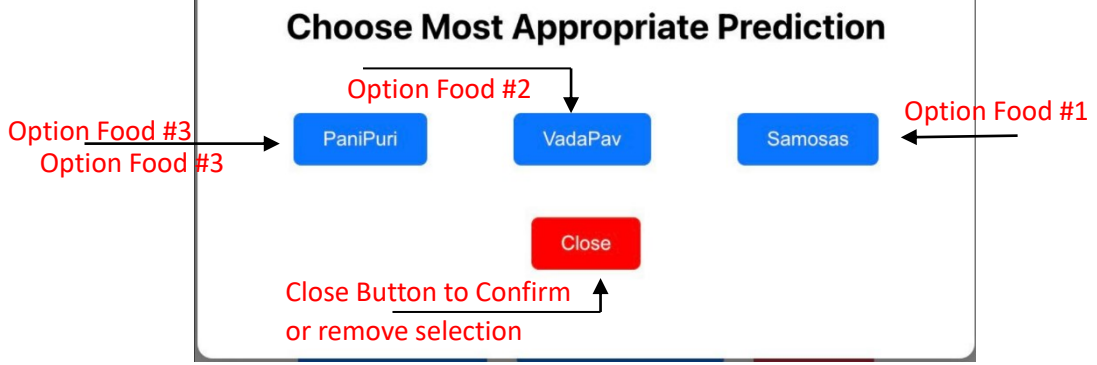
**Confirmation (Login):**

1. **Success:** valid user logged in and directed to home page.
2. **Failure:** display message
  - a. "Something Went Wrong" in case where Username or Password is missing

#002	Capturing/Uploading Meal Image	
As a [logged in user], I want to [add a meal],so that I can [track my diet]		
		

### **Confirmation (Image):**

1. **Success:** Preview of food image is available on home screen.
  - a. **Upload image button clicked:** Upload image dialog box is opened.
  - b. **Capture image button clicked:** Uploading image via webcam.
2. **Failure:** No preview if unsupported file format.

#003	Opting for best suited meal	
As a [logged in user], I want to [verify my meal],so that I can [track my diet]		
 <p style="text-align: center;"><b>Choose Most Appropriate Prediction</b></p> <p>Option Food #3 → PaniPuri      Option Food #2 → VadaPav      Samosas ← Option Food #1</p> <p style="text-align: center;">Close</p> <p style="text-align: center;">Close Button to Confirm or remove selection</p>		

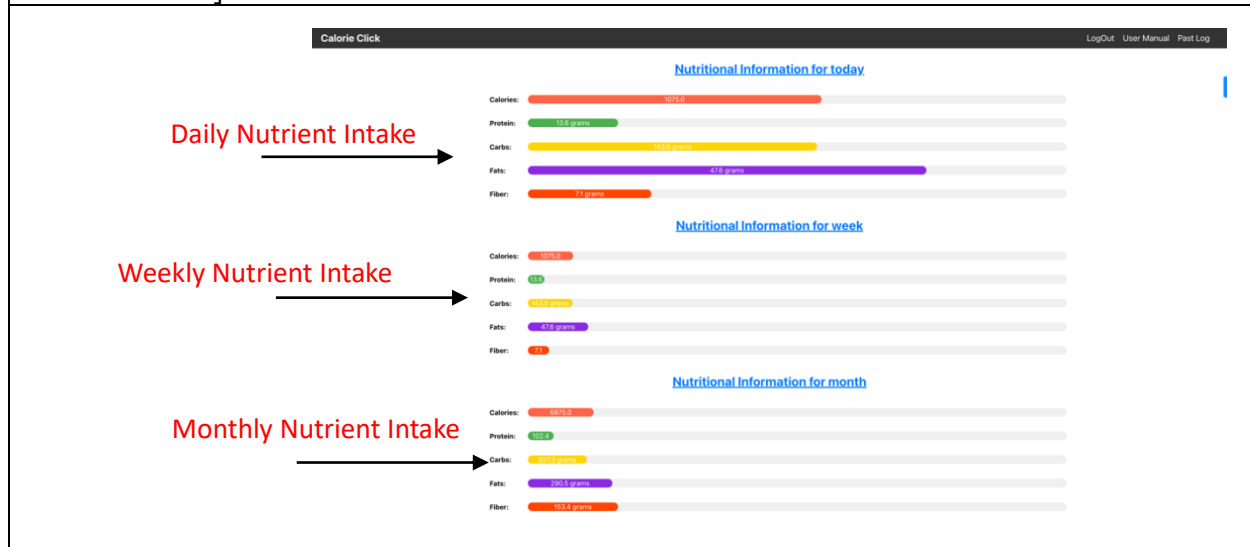
### **Confirmation (Meal Choice):**

1. **Success:** Choosing the most appropriate prediction.
  - a. **Food option #1:** Confirm image to be of meal #1.
  - b. **Food option #2:** Confirm image to be of meal #2.
  - c. **Food option #3:** Confirm image to be of meal #3.
  - d. **Close Button:** Dialogue box close with selection.
2. **Failure:** Back to home screen with no selection.
  - a. **Close Button:** Dialogue box close with no selection.



#004	Past Log	
------	----------	--

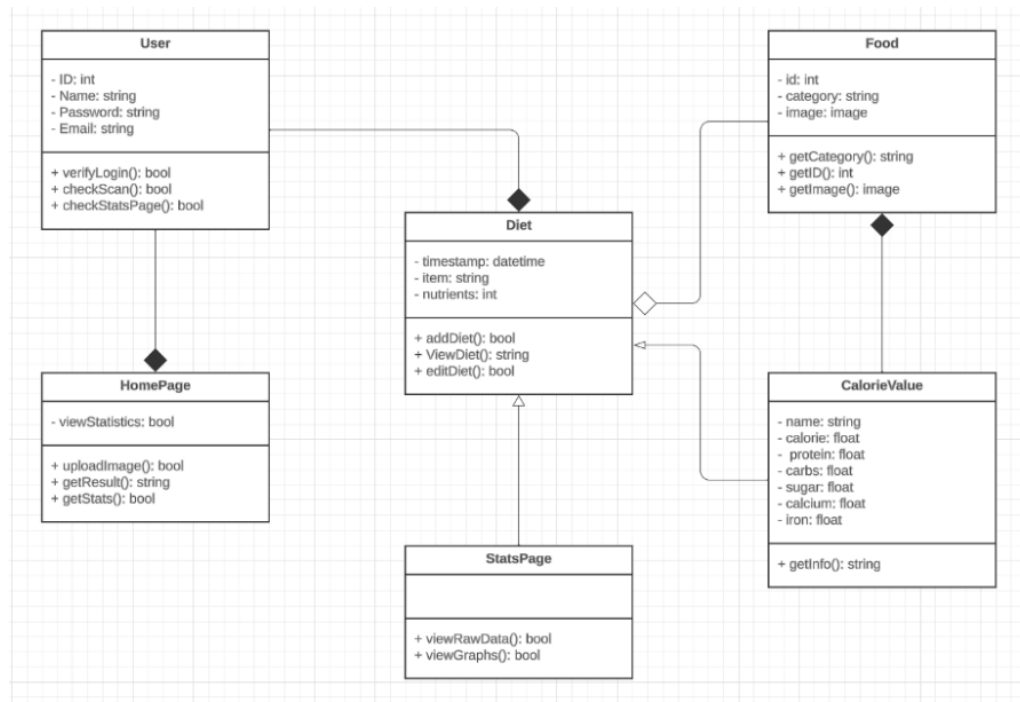
As a [logged in user], I want to [check my nutrients stats], so that I can [have a healthy mind and soul].



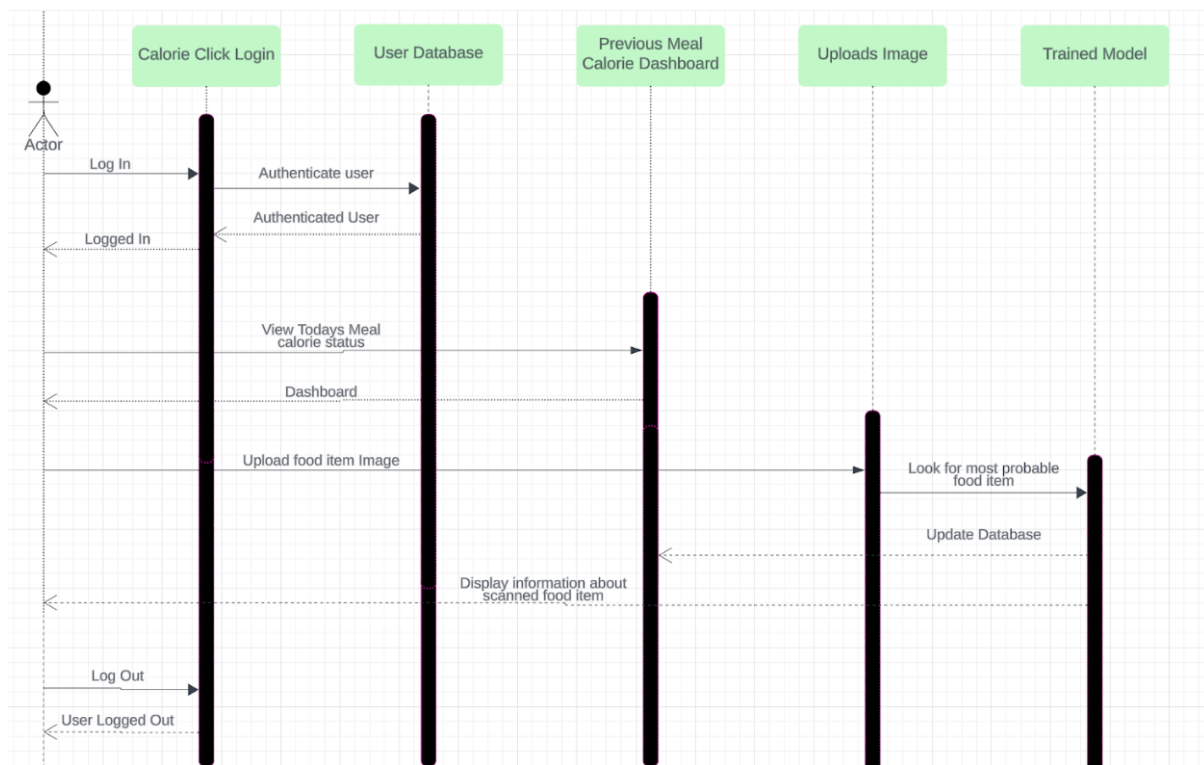
### **Confirmation (Statistics):**

- 1. Success:** View daily, weekly, and monthly nutrient's goal.

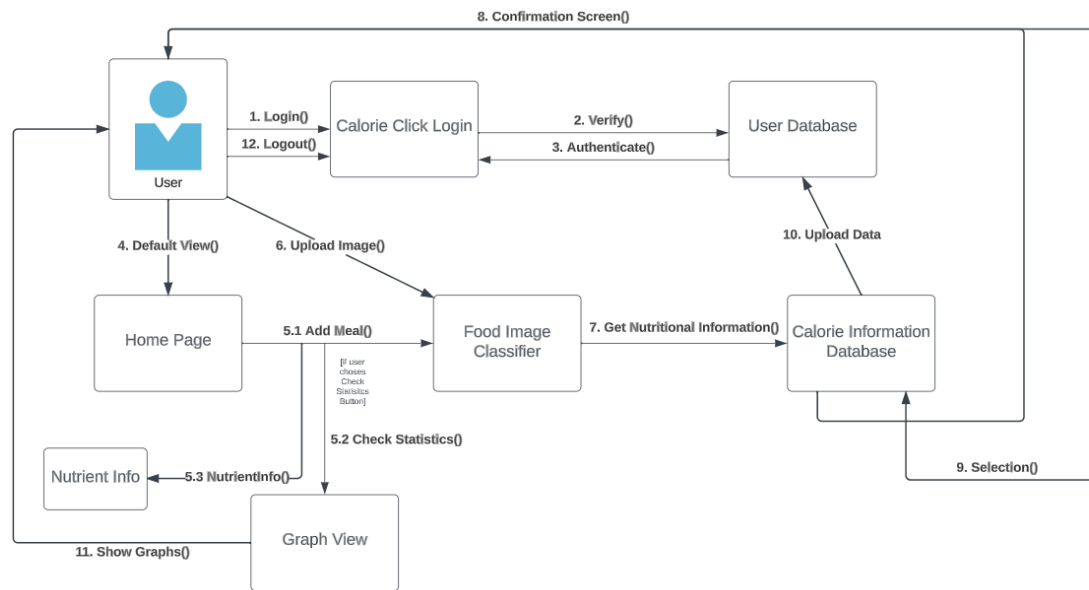
## Class Diagram



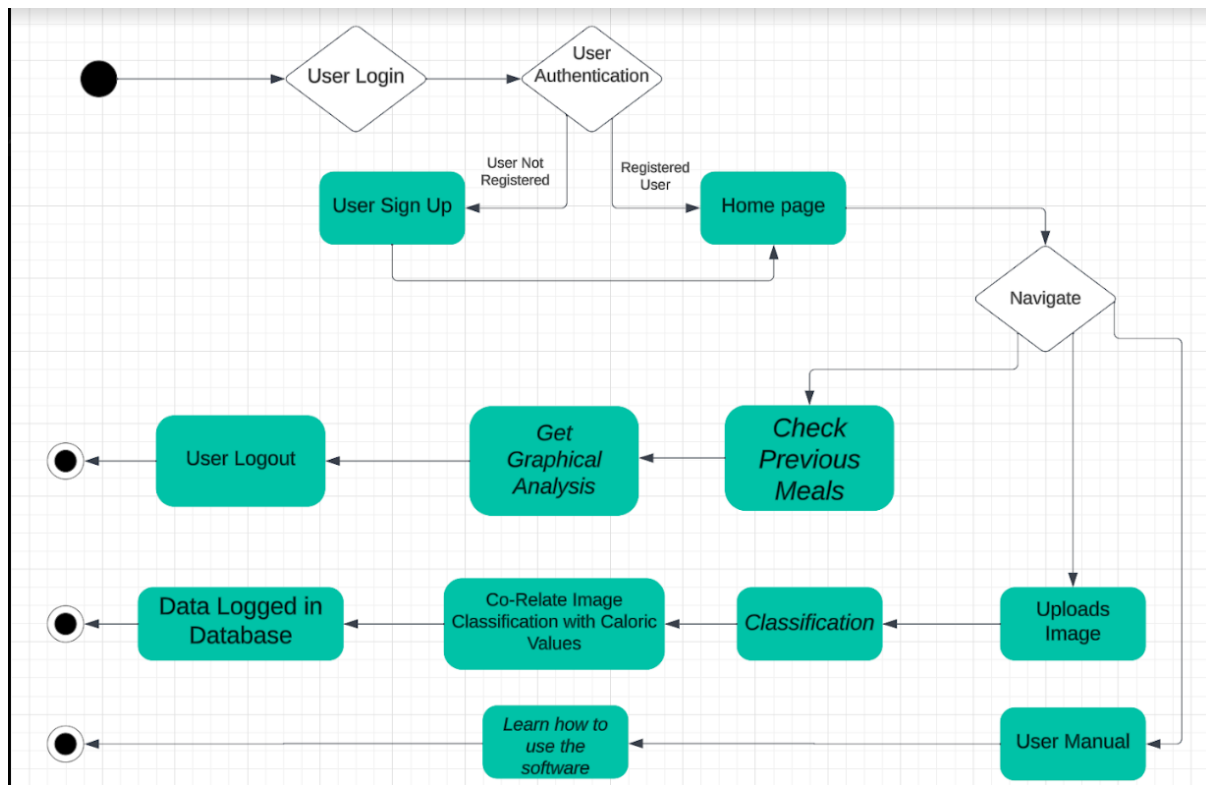
## Sequence Diagram



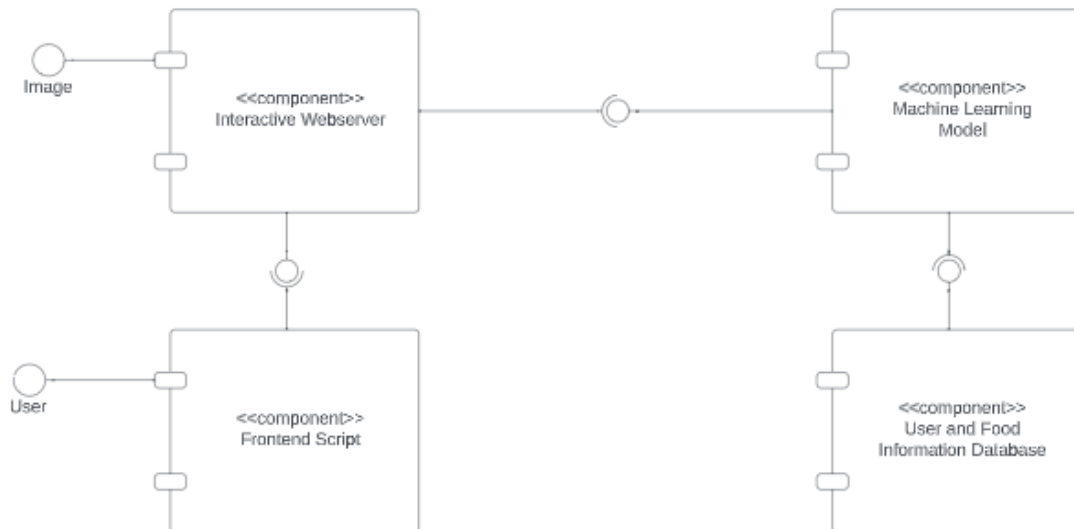
## Collaboration Diagram



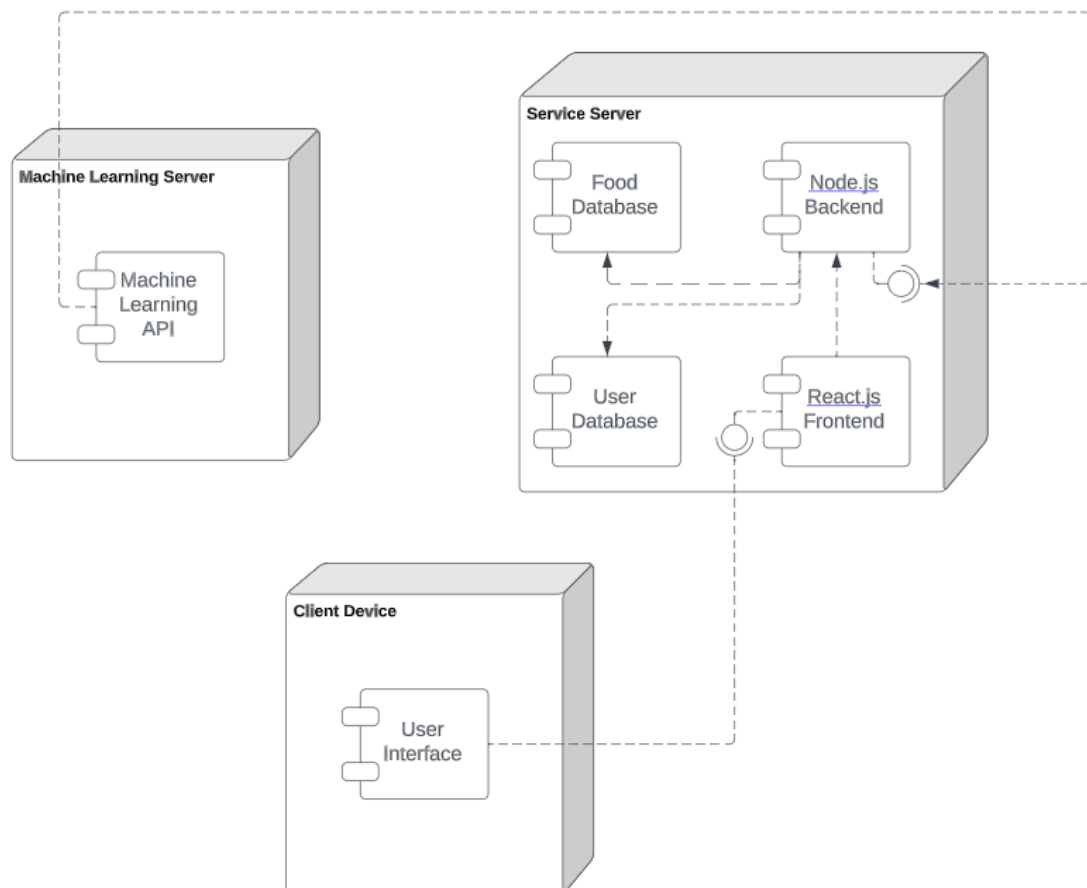
## State Chart Diagram



## Component Diagram



## Deployment Diagram



## Screenshots of Software

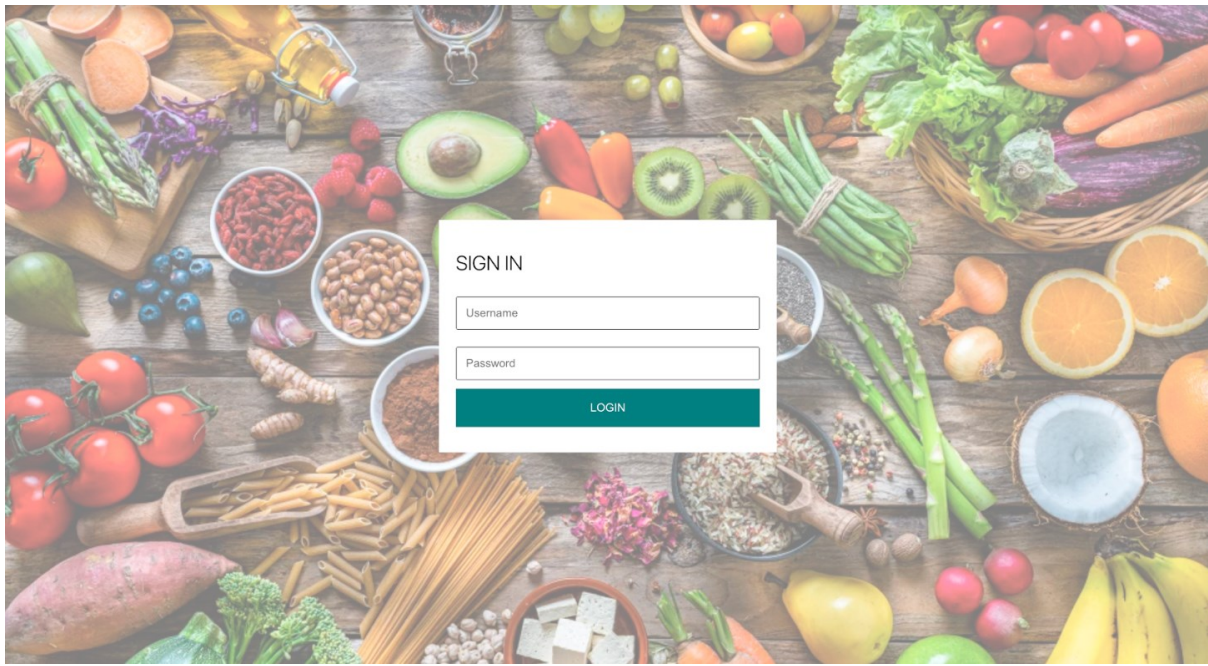


Figure 1: Login Screen

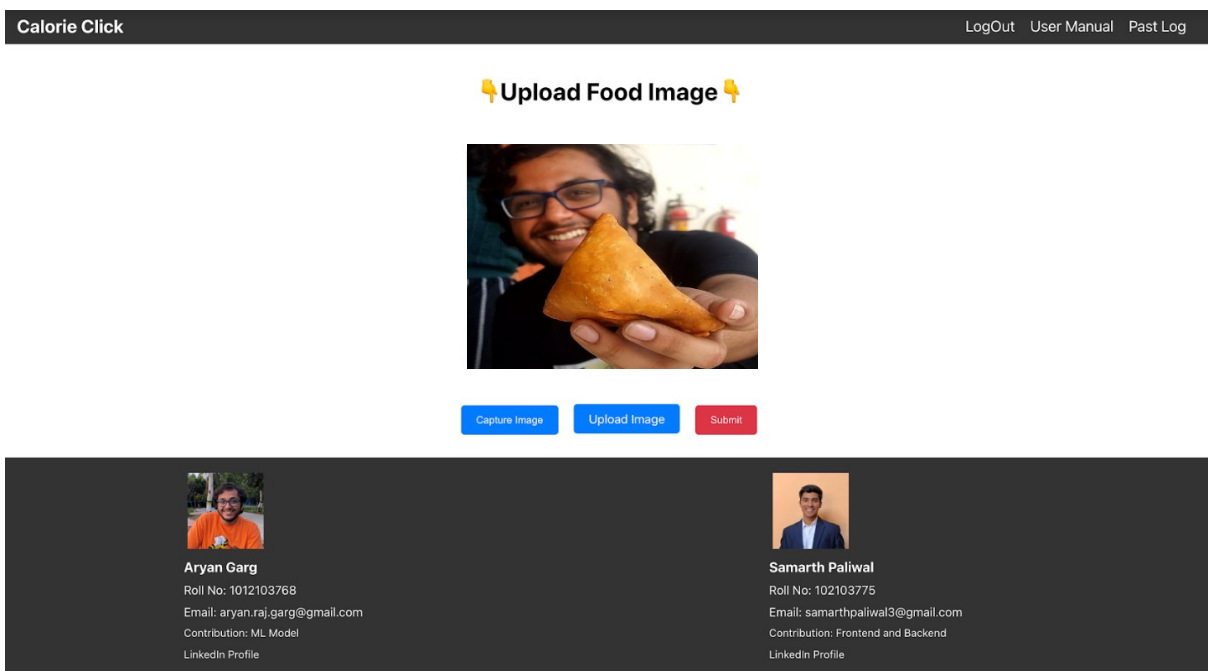


Figure 2: Front Page

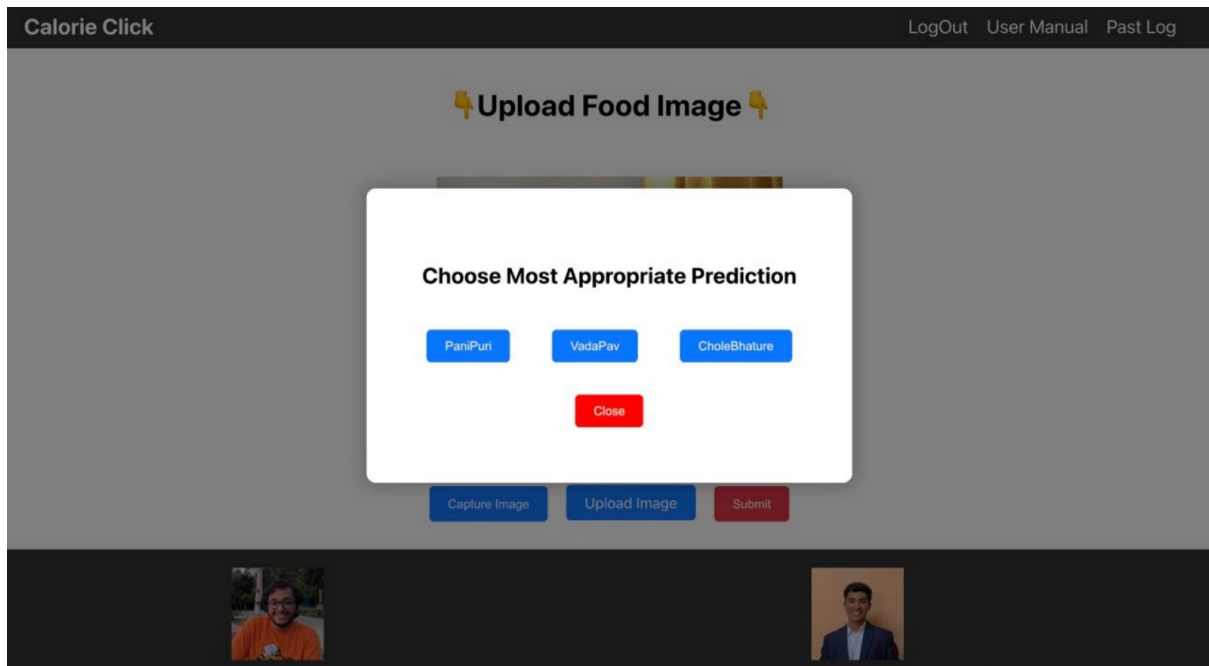


Figure 3: Selection of Item

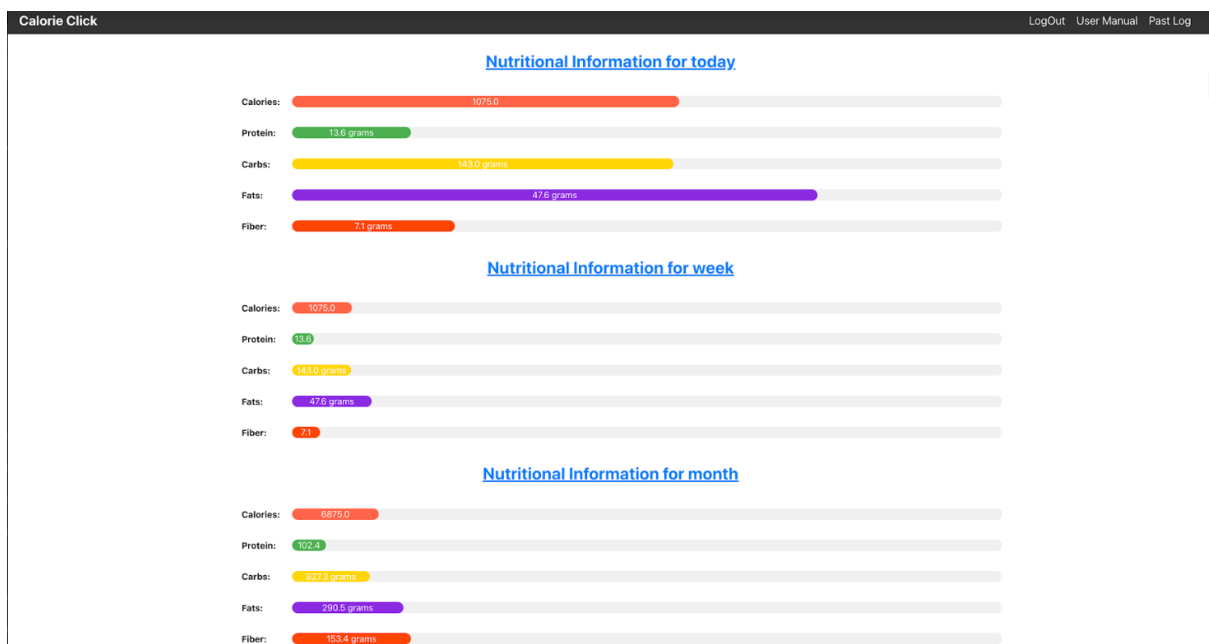


Figure 4: Nutritional Information

### How to use Calorie Click

**Image Upload:** Capture or upload an image of your meal using the designated button on the homepage and then click on Submit Button.

**Receive Recommendations:** After uploading, our machine learning model will provide you with three food combinations. Choose the one that matches your meal.

**Log Nutritional Information:** Select the chosen food combination, and the system will log the nutritional information for your records.

**View Past Logs:** Check your past nutritional logs by clicking on the 'Past Log' button to view your historical data.

**Logout:** To end your session, click on the 'Logout' button. You'll be redirected to the login page.



**Aryan Garg**

Roll No: 1012103768

Email: aryan.raj.garg@gmail.com

Contribution: ML Model

[LinkedIn Profile](#)



**Samarth Paliwal**

Roll No: 102103775

Email: samarthpaliwal3@gmail.com

Contribution: Frontend and Backend

[LinkedIn Profile](#)

*Figure 5: User Manual*

## **Test Cases**

Test Case # 1

Test Case Name: User Login

Page:

1 of 10

System: Calorie Click

Subsystem: Authentication

Designed By: Samarth Paliwal

Design Date: 17 Nov 2023

Executed By: Aryan Garg

Execution Date: 17 Nov 2023

Short Description: User try to login to the software

### **Pre-Conditions:**

- The user has a registered account on Calorie Click.

Step	Action	Expected System Response	Pass/Fail	Comment
1.	Enter username	NA	Pass	
2.	Enter Password	NA	Pass	
3.	Click on "Login" button	Credentials are cross checked with Database.	Pass	Works

### **Post-Conditions**

- The user is either redirected to the home page upon successful login or remains on the login page with an error message.



Test Case # 2

Test Case Name: Web Cam  
Access

Page 2 of  
10

System: Calorie Click

Subsystem: Website

Designed By: Samarth Paliwal

Design Date: 17 Nov 2023

Executed By: Aryan Garg

Execution Date: 17 Nov 2023

Short Description: Webcam Access Permission Prompt on Home  
Screen

### Pre-Conditions:

- The user is successfully logged into the Calorie Click.
- The device has a webcam.

Step	Action	Expected System Response	Pass/Fail	Comment
1.	Navigate to Home page	Displays a prompt requesting access to the webcam.	Pass	
2.	Confirm Webcam Access	The system detects and confirms that webcam access has been granted.	Pass	
3.	Deny Webcam Access	The system handles denied access successfully, displaying a white screen in the preview area.	Pass	Works

### Post-Conditions

- The user has either granted or denied webcam access on the home screen.

Test Case # 3

Test Case Name: User  
Manual

Page  
3 of 10

System: Calorie-Click

Subsystem: Website

Designed By: Samarth Paliwal

Design Date: 17 Nov 2023

Executed By: Aryan Garg

Execution Date: 17 Nov 2023

Short Description: User can get information about the nutrients being tracked

### Pre-Conditions:

- The user is successfully logged into the Calorie Click.

Step	Action	Expected System Response	Pass/Fail	Comment
1.	Click on 'User Manual' button	System will navigate the user to Instruction page where user can see how to use the software.	Pass	Works

### Post-Conditions

- The user has accessed the User Manual and gained insights into how to use the software effectively.

Test Case # 4

Test Case Name: Upload Meal

Page

4 of 10

System: Cookie-Click

Subsystem: Model

Designed By: Aryan Garg

Design Date: 17/11/2023

Executed By: Samarth Paliwal

Execution Date: 17/11/2023

Short Description: Upload Image of Meal in Browser

### Pre-Conditions

- User Logged in
- User has Image stored on device
- User is on Home Page

Step	Action	Expected System Response	Pass/Fail	Comment
1	Click Upload Image Button	Upload Dialog Appear	Pass	
2	Upload from Storage	Get Image from Storage	Pass	
3	Click Submit Button	Process Image	Pass	Works

### Post-Conditions

- Image is sent to be processed

Test Case # 5

Test Case Name: Capture Meal

Page

5 of 10

System: Cookie-Click

Subsystem: Model

Designed By: Aryan Garg

Design Date: 17/11/2023

Executed By: Samarth Paliwal

Execution Date: 17/11/2023

Short Description: Capture an Image of Food from the Browser

### Pre-Conditions

- User Logged in
- User has camera on Device
- User has Camera permission enabled
- User is on Home Page

Step	Action	Expected System Response	Pass/Fail	Comment
1	Click Capture Image	Image is captured	Pass	
2	Click Submit Button	Process Image	Pass	

### Post-Conditions

- Image is sent to be processed

Test Case # 6

Test Case Name: Model Response

Page

6 of 10

System: Cookie-Click

Subsystem: Model

Designed By: Aryan Garg

Design Date: 17/11/2023

Executed By: Samarth

Execution Date: 17/11/2023

Paliwal

Short Description: Upload Image to server and then get output

### Pre-Conditions

- User Logged in
- User has uploaded Image for Processing

Step	Action	Expected System Response	Pass/Fail	Comment
1	Image is Processed	Convert to Base64 Image	Pass	
2	Sent to Model	API Call	Pass	
3	Return Result	Get 3 options or cancel	Pass	
4	User Select Button	Option is chosen from 3	Pass	
5	Click Save Button	Option is saved to Backend	Pass	Works

### Post-Conditions

- New Food Entry is entered in the Database

Test Case # 7

Test Case Name: Cancel Selection

Page  
7 of 10

System: Cookie-Click

Subsystem: Model

Designed By: Aryan Garg

Design Date: 17/11/2023

Executed By: Samarth  
Paliwal

Execution Date: 17/11/2023

Short Description: Remove selection after image is sent to server

### Pre-Conditions

- User Logged in
- User has uploaded Image for Processing

Step	Action	Expected System Response	Pass/Fail	Comment
1	Return Result	Get 3 options or cancel	Pass	
2	User Select Button	Cancel Option is Selected	Pass	
3	Cancel Button	Decision is Discarded	Pass	Works

### Post-Conditions

- Option is Discarded

Test Case # 8

Test Case Name: Past

Page

Nutrients Intake

8 of 10

System: Calorie Click

Subsystem: Statistics

Designed By: Samarth Paliwal

Design Date: 17 Nov 2023

Executed By: Aryan Garg

Execution Date: 17 Nov 2023

Short Description: Reviewing past nutrient intake information.

### Pre-Conditions:

- The user is successfully logged into the Calorie Click.

Step	Action	Expected System Response	Pass/Fail	Comment
1.	Click on 'Past Log' button	System displays the Past Log page, fetching past food intake from Database.	Pass	Works

### Post-Conditions

- The user can view a summary of their past nutrient information of present date, week and month based on the food intake.

Test Case # 9

Test Case Name: Check Food

Page

9 of 10

System: Cookie-Click

Subsystem: Website

Designed By: Aryan Garg

Design Date: 17/11/2023

Executed By: Samarth Paliwal

Execution Date: 17/11/2023

Short Description: Get the name of Meals eaten in the day/Week

### Pre-Conditions

- User Logged in

Step	Action	Expected System Response	Pass/Fail	Comment
1	Click on the Past Log Button	User is taken to Past Logs	Pass	
2	Scroll to Past Meal Information	Meal Information with Names	Fail	Partial Working

### Post-Conditions

- User is taken to the home page



Test Case #10

Test Case Name: Logout

Page

10 of 10

System: Calorie Click

Subsystem: Authenticate

Designed By: Samarth Paliwal

Design Date: 17 Nov 2023

Executed By: Aryan Garg

Execution Date: 17 Nov 2023

Short Description: "Logout" button successfully logs out the user

### Pre-Conditions:

- The user is successfully logged out into the Calorie Click.

Step	Action	Expected System Response	Pass/Fail	Comment
1.	Click on the "Logout" button.	Redux store set the currentUser to NULL.	Pass	Works

### Post-Conditions

- The user is logged out and redirected to the login page.