A PROJECT REPORT ON

Personalized Diet and Health planner system

SUBMITTED TO THE SAVITRIBAI PHULE PUNE UNIVERSITY, PUNE
IN THE PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE AWARD OF THE DEGREE

OF

BACHELOR OF ENGINEERING (COMPUTER ENGINEERING)

SUBMITTED BY

Sachin Johnson	Exam No. B150054253
Vishwajeet Ekal	Exam No. B150054281
Pratik Ghodke	Exam No. B150054289
Rajwardhan Sinha	Exam No. B150054436

DEPARTMENT OF COMPUTER ENGINEERING

PUNE INSTITUTE OF COMPUTER TECHNOLOGY

DHANKAWADI, PUNE – 43

SAVITRIBAI PHULE PUNE UNIVERSITY
2020 -2021

Certificate

This is to certify that your project entitles

Personalized Diet and Health planner system

Submitted by

Sachin Johnson	Exam No. B150054253
Vishwajeet Ekal	Exam No. B150054281
Pratik Ghodke	Exam No. B150054289
Rajwardhan Sinha	Exam No. B150054436

is a bonafide student of this institute and the work has been carried out by him/her under the supervision of **Prof. R. A. Kulkarni** and it is approved for the partial fulfillment of the requirement of Savitribai Phule Pune University, for the award of the degree of Bachelor of Engineering (Computer Engineering).

Prof. R. A. Kulkarni
Internal Guide

Prof. M. S. Takalikar
Head
Department of Computer Engineering

Dr. R. N. Sreemathy
Principal
Pune Institute Of Computer Technology

Signature of Internal Examiner

Signature of External Examiner

Place: Pune
Date:

ACKNOWLEDGEMENT

We are immensely obliged to and sincerely thank our BE Project Coordinator Prof. R. A. Kulkarni and Head of Department Prof. M. S. Takalikar for their support in presenting our project report on ***'Personalized Diet and Health planner system'****.*

We also sincerely convey our gratitude towards our mentor Mr. Adwait Sambhare, for his constant support, providing all the help, motivation and encouragement from beginning till the end to make this a grand success.

Sachin Johnson
Vishwajeet Ekal
Pratik Ghodke
Rajwardhan Sinha

ABSTRACT

The global pandemic situation has pushed people around the world to focus on their health more than ever. Countless number of searches have been made on search engines regarding immunity, exercises and food that can be prepared at home for physical wellbeing.However, people stand at a risk of eating food that could be detrimental to their health because of being unaware to their disease history, allergies, irregular schedules and lethargic attitude towards going through the nutritional value of the ingredients they consume as a part of the whole process. We aim to provide an application that addresses all the above concerns under one roof. The system provides a diet plan by taking a nuanced approach towards the user's preferences, environment and strives to predict the timeline that could help the users reach their physical goals. An accurate system that suggests a perfect diet plan by taking into account all the factors could help workers in the healthcare sector to propose a diet plan which might consider issues that might have escaped the attention of a doctor or nutritionist as it works on the information fed by them and also the information and patterns that might go unnoticed due to human error. A user-friendly system can help people take more interest towards the nutritional value of food they eat and follow a disciplined lifestyle which is our aim.

Contents

List of Abbreviations

Abbreviation	Full form
UI	User Interface
API	Application Programming Interface
SDLC	Software Development Life Cycle
UML	Unified Modeling Language

Table 1: Abbrevations

List of Tables

List of Figures

1 INTRODUCTION

1.1 Motivation

Making decision about what and where to eat is a major problem in our everyday lives due to a wide variety of ingredients, culinary styles, ethnicities, cultures, and personal tastes. Choosing right dish at the right time seems to be a very difficult task.
Every user has their own specific nutritional needs and requirements. An appropriate nutrition policy can therefore help improve the user's health and immune system. However, the appropriateness cannot depend upon traditional algorithms or evolutionary algorithms as they do not quickly respond to user preferences and real-time data. The existing devices in market provide diet plans but then do not account for economical costs that arise due to certain preferences and disease history.It can't be recommended to doctors and nutritionists to come up with a proper diet plan that meets all the constraints of their patients. With advances in the field of machine learning, it's possible to build flexible systems that can take into account user's environment and body health to accurately predict diet plans and recipes also providing widely-shared advices on physical exercises that can help accelerate efforts to reach physical goals and raise alerts in case of emergencies.

1.2 Problem Definition

To design and create an appropriate and personalized diet platform that proposes a daily menu using nutritional knowledge for a decision analysis approach based on multiple criteria to screen inappropriate foods. These criterias include BMI, diseases(clinical or heredity), location, dietary preferences, real-time data and feedback.

2 LITERATURE SURVEY

No	Title	Methods	Dataset	Remarks
1	A Food Recommender System Considering Nutritional Information and User Preferences [1]	Probabilistic Approach	W3-Recipe	Generates meal plans by maximizing or minimizing some criteria depending on preferences but is still a work in progress.
2	Realizing an Efficient IoMT-Assisted Patient Diet Recommendation System [2]	Machine Learning Model	Nutrient Database	Provides different recommender evidence for patient diseases which might require different treatment and special care
3	Intelligent Nutrition in Healthcare and Continuous Care [3]	Neural Networks	FDA database	Intelligent planning of the user's meals, based on their clinical conditions
4	Personalized Meal Planning for Diabetic Patients [4]	Multi-Criteria Decision-Making Approach	FDA database	Generates meal plans by maximizing or minimizing some criteria depending on preferences but is still a work in progress.

Table 2: Literature survey on various approaches for diet planning

3 SOFTWARE REQUIREMENTS SPECIFICATION

3.1 Introduction

3.1.1 Project Scope

The goal of this project is to provide a personalised diet-plan as well as explore the possibility of improving existing methods that are used to create a diet-plan. The scope of this project is limited to making a prototype, rather than a production ready system. However, the implementation of a successful prototype may pave the way for a production ready application in the future

3.1.2 User Classes and Characteristics

1. Technical users: Have some knowledge about the product.
2. Non-technical users: Have little to no knowledge about the product.He/She can use the system for creating and analyzing his/her diet plan, having a check on medical history and providing feedback for preferences
3. Admin: This is a user that will have specific controls to the application which will allow them to control certain modules in the application. The admin can change the forum settings to make the users happy.
   Admin will administer the overall control of the website and can override any setting, constraints in any module as he/she wants and help the clients to maintain their diet plan efficiently and guarantee them healthy and nutritious food. However they won't have access to personal data.

3.1.3 Assumptions and Dependencies

1. User manuals are provided to the system in PDF format.
2. Assumption is that the user should have some basic knowledge of computers. We are assuming that the user can navigate through multiple pages of an application.
3. Knows how to read and write data.
4. The medical history of the user being provided is verified and all allergies are accounted for.

3.2 Functional requirements

3.2.1 User interface

a. Description: The UI will enable the user to navigate through multiple functionalities of the application and enter relevant data with the intent of requiring particular data from user.

b. Response sequence:

   i. User submits the data to the UI
   ii. UI relays it to the back-end through API.
   iii. UI displays a modifiable meal plan.

c. Functional requirement:

   i. User submits a profile containing preferences
   ii. System processes the profile and similar profiles
   iii. System relays meal plan relevant to the profile
   iv. The results are displayed on the UI.

3.2.2 System should provide meal plan based on the user profile

a. Description: System should take into account the profile of the user while submitting the meal/diet plan.

b. Response sequence:

   i. User submits a query to the UI through an API.
   ii. UI relays it to the system.
   iii. System responds with summaries relevant to the query.

c. Functional requirements:

   i. User submits a profile through the interface.
   ii. Retriever module searches through the recipes matching the profile.
   iii. It ranks results based on relevance and passes most relevant results to the retriever module.
   iv. The recommender module individually summarizes these results.
   v. The meal plans are relayed back to the UI for display.

3.2.3 Realtime Feedback

a. Description: Suggested meal plan will always be improved by taking instant feedback and dynamic changes to calorie requirements

b. Response sequence:

   i. User submits a feedback through UI
   ii. UI sends data to the system to modify meal plan
   iii. System responds with new tailored meal plan.

c. Functional requirements:

   i. User receives the meal plan after submitting data.
   ii. User provides feedback through a checklist
   iii. The recommender system accepts and changes meal plan
   iv. The meal plans are relayed back to the UI for display.

3.3 External interface requirements

3.3.1 User Interface

a. Admin Interface: Once admin logs in, it will redirect admin to admin-dashboard.Admin Dashboard will allow admin to manage user activities and manage overall functionality of the website, give warning or delete troublesome users. In this interface admin will have full control over this web application. Various fields available on this screen will be:

   i. Username
   ii. Password
   iii. Number of Users

b. User Registration Interface: Once a user is successfully registered to the platform, he/she will be automatically redirected to the Login Interface and will be asked to fill preferences (if any). This interface will allow users to register for the first time to the Diet recommendation System. Interface will include following fields:

   i. Username
   ii. Password
   iii. Confirm Password
   iv. Contact
   v. Medical History
   vi. Fitness app ID

c. User Login Interface: Once a user is successfully logged into the platform, he/she will be automatically redirected to the personal menu dashboard. This interface will allow already registered users to login to his/her Menu dashboard. This interface will include following fields:

   i. Username
   ii. Password

d. User Dashboard Interface:In this interface users can manage their own portfolio and get insights based on their current portfolio. This interface will have these major tabs:

   i. User menu(standard recommendations)
   ii. Preferences
   iii. Profile
   iv. Feedback

e. User Portfolio : This interface will allow the user to see their personal diet chart, nutrients provided in the diet. Each meal will take care of medical history and allergies .

f. Preferences : The user may provide his/her preferences and menu changes accordingly.

g. Profile : This interface will allow the user to see the details of Profile and update any details like name, password, contact,medical condition, etc.

h. Recommendations : This interface will display the recommended recipes and the amount of nutrients they have.

3.3.2 Hardware Interfaces

Users can use any screen size above 300px width, i.e. the web interface will be responsive based on user screen sizes. Apart from this, users will need any interactive device like a mouse, touch pen for clicking on the menu options and keyboard (either physical or virtual) to enter the details for login/register.

3.3.3 Software Interfaces

. User : Web Browser, Javascript-Enabled

. Web Server : Flask Inbuilt Server

. Database Server : SQLite3 -¿ MySQL if needed for deployment

. Backend Development : Python3, Flask (Framework), Tensorflow/Keras/Pytorch - ML

. Frontend Development : HTML, CSS, JS, Flask-templates

3.3.4 Communication Interfaces

. Users will interact with the platform using HTTP/HTTPS protocol.

. TCP/IP Network stack will be used for communication.

3.3.5 Operating Environment

The model and application back end services are expected to run on a cloud server running a Linux OS, with a simple file storage system to store extra incoming data. The prototype will focus on a web based UI for the users to interact with the system.

3.4 Non-functional requirements

3.4.1 Performance Requirements

The system should have a high performance rate when executing user's input and should be able to provide feedback or response within a short time span usually 50 seconds for advanced credentials and 20 to 25 seconds for less detailed profiles.

3.4.2 Safety Requirements

Error should be considerably minimized and an appropriate error message that guides the user to recover from an error should be provided. Validation of user's input is highly essential. Also the standard time taken to recover from an error should be 15 to 20 seconds.

3.4.3 Security Requirements

The subsystem should provide a high level of security and integrity of the data held by the system, only the end-user should gain access to their own credentials and admin with necessary credentials be able to get a bird's eye view of the application

3.4.4 Software Quality Attributes

..

1. Flexibility: The system can go through and get meal plans with varying number of meals-per-day.
2. Availability: The system is platform independent and thus available to all users. his system should always be available for access at 24 hours, 7 days a week
3. Interoperability: Any system can interact with our system through the API.
4. Usability : The interface should use terms and concepts, which are drawn from the experience of the people who will make the most of the system.
5. Efficiency : The system must provide easy and fast access without consuming more cost.
6. Reliability : Users should never be surprised by the behavior of the system. It should be easy to see all different options which are being offered in as few clicks as possible.
7. Transparency : Recommendations made should be as transparent as possible. It will be handled by providing the reports of different ML algorithms based on which recommendations are made.
8. Security : System provides authentication mechanism without which no user can get into the system. Sharing of credentials will not be considered in the security aspect of the product.
9. Maintainability : Software will be very well documented for easy maintenance of the system. Also in the occurrence of any major system malfunctioning, the system should be available in 1 to 2 working days.

3.4.5 Business Rules

A business rule is anything that captures and implements business policies and practices. A rule can enforce business policy, make a decision, or infer new data from existing data. This includes the rules and regulations that the System users should abide by. This includes the cost of the project and the discount offers provided. The users should avoid illegal rules and protocols. Neither admin nor member should cross the rules and regulations.

3.5 System requirements

The user interface for the system will be in the form of a website and can be accessed through any device with a browser and an active internet connection.

3.5.1 Software Requirements

This application can be accessed by a user through a machine having any web browser. The client devices must preferably have browsers like IE9 or above, Mozilla Firefox (version 60.02 quantum) or Opera 54.0 or chrome (version 68.0.3) or safari installed in their OS

3.6 System Implementation Plan

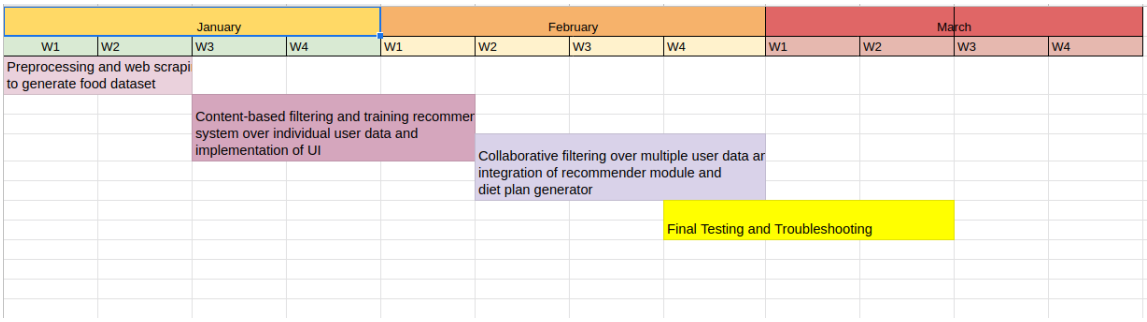

Figure 1: System Implementation Plan (Jan to Mar)

4 SYSTEM DESIGN

4.1 System Architecture

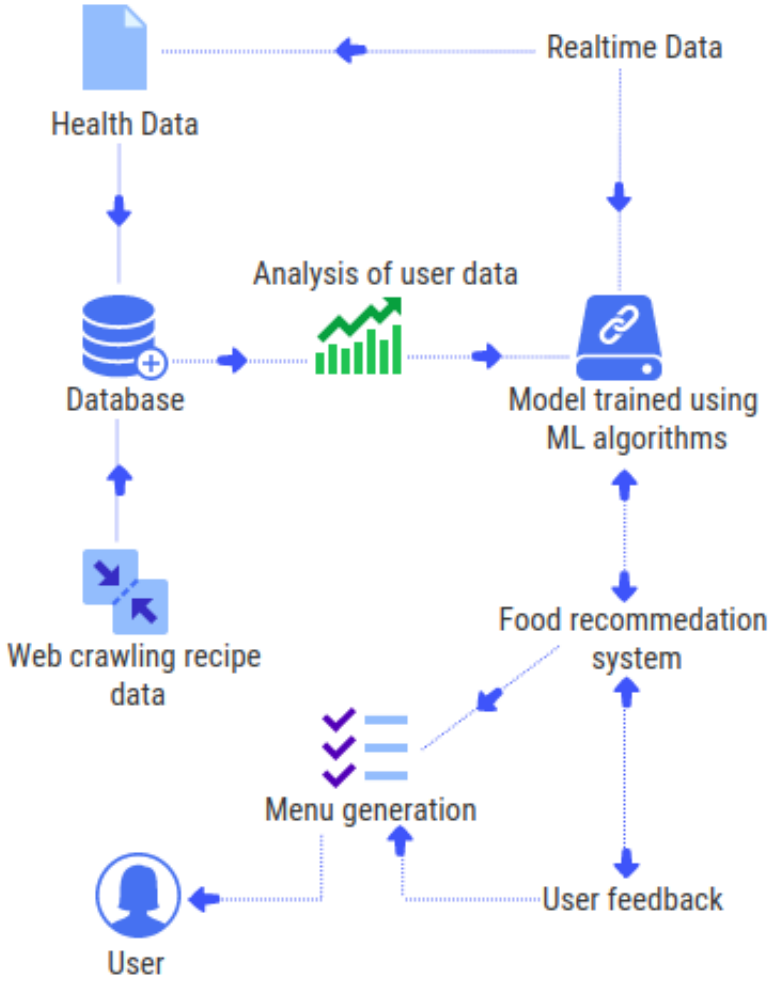

Figure 2: System Architecture Diagram

4.2 Data Flow Diagram

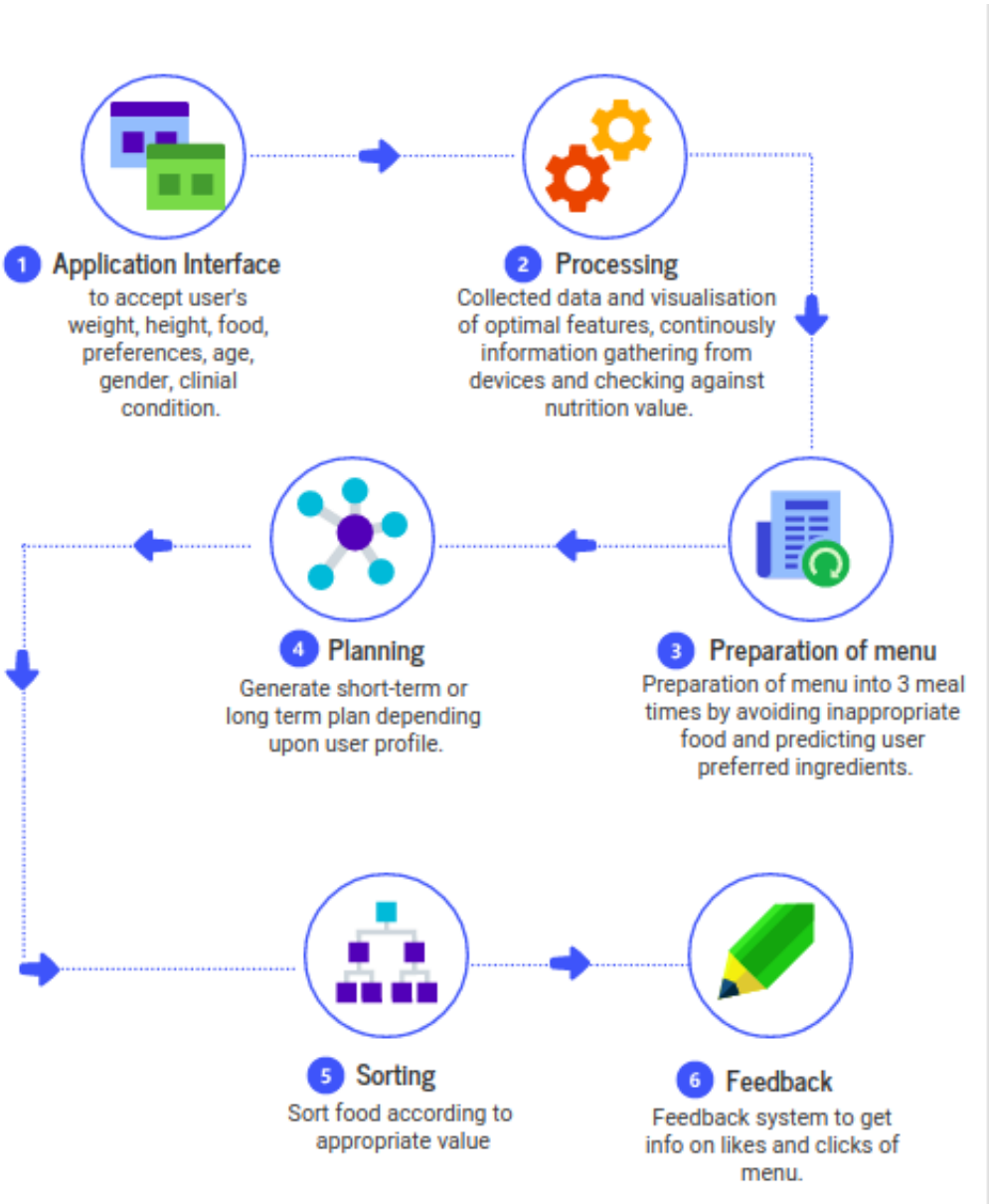

Figure 3: Data Flow Diagram

4.3 UML Diagrams

4.3.1 Class Diagram

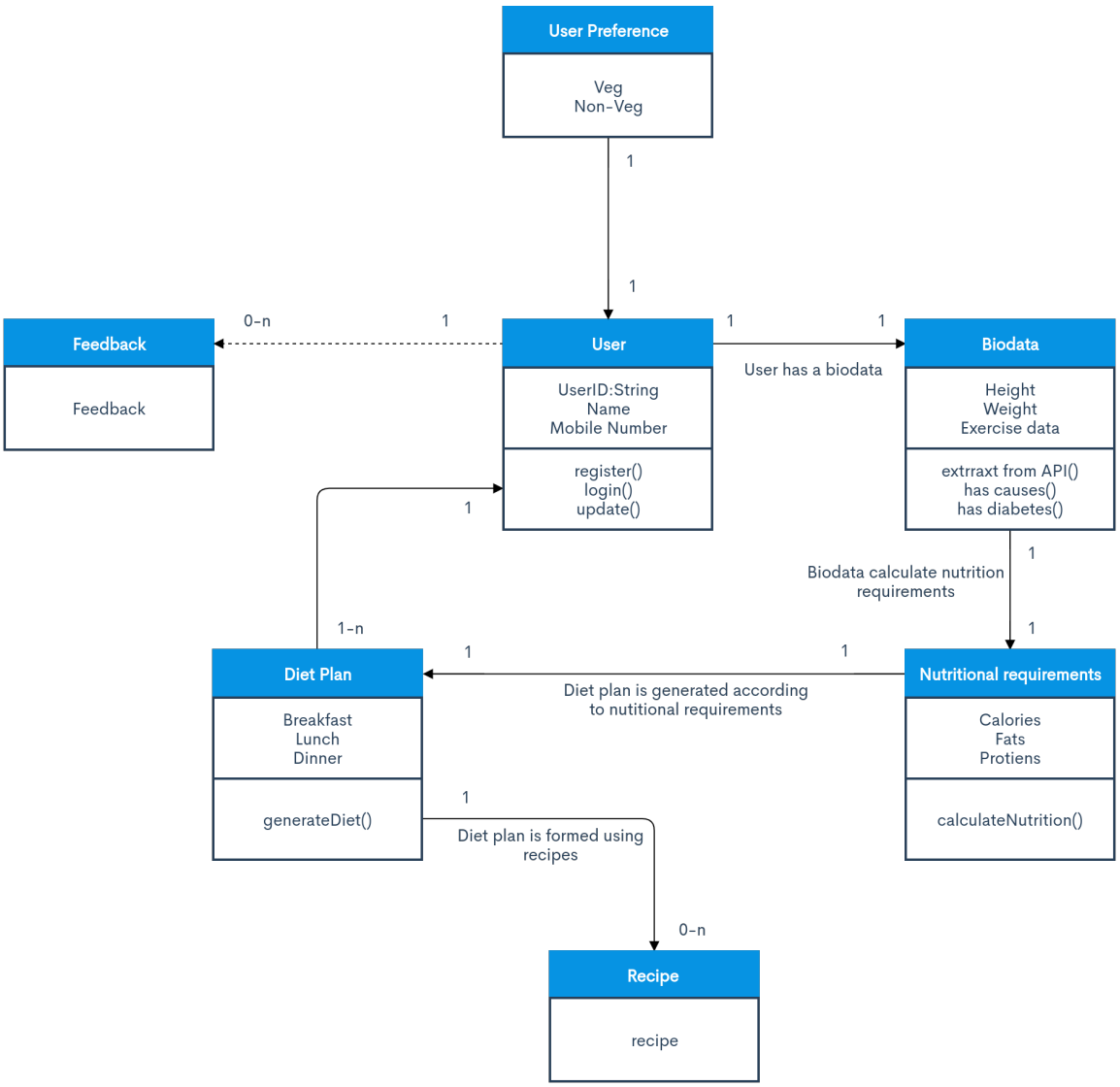

Figure 4: Class Diagram

4.3.2 Activity Diagram

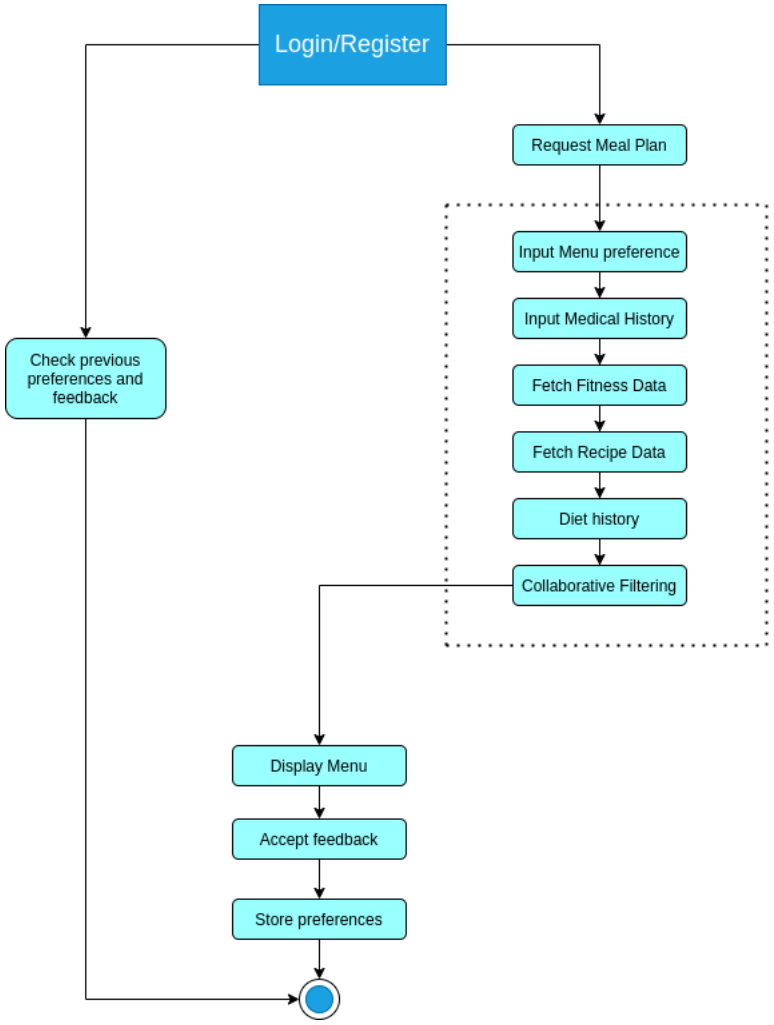

Figure 5: Activity Diagram

4.3.3 Use Case Diagram

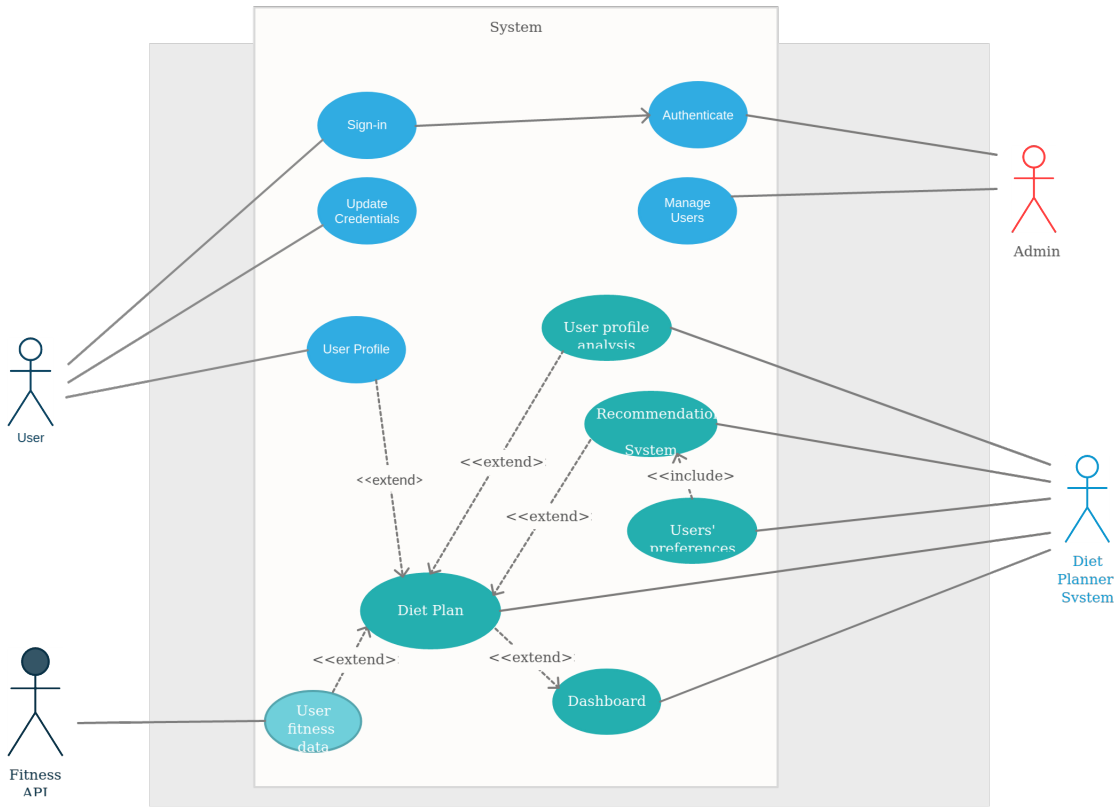

Figure 6: Use Case Diagram

5 OTHER SPECIFICATIONS

5.1 Advantages

a. The system makes the process of generating diet plans more accurate

b. It provides an interactive experience to modifying a plan.

c. Users are only presented with information relevant to their profile

d. Once a meal plan is generated the user is free to provide feedback and get an updated meal plan or integrate health trackers and medical history

5.2 Limitations

a. User can't add fields to profile

b. User should provide accurate info

c. The system is limited only to providing meal plans and displaying vitals using third-party APIs.

5.3 Applications

a. This pipeline can be adapted to fit a generalised model for any other form of planning system in the health domain.

b. The system can be integrated with a fitness application

6 Project Plan

6.1 Project Resources

6.1.1 Human Resources

- Team Strength: 4
- Technical Expertise: Web Development, Artificial Intelligence/Machine Learning, Networking
- Communication: Every 2-3 weeks, the team met virtually, followed by timely reviews with the internal guide and external interviewers.

6.1.2 Hardware and Software Resoures

- Datasets from websites like food.com, tarladalal.com for scraping and preparing datasets for Indian cuisine
- Hardware: 8GB RAM, 1TB HDD, Intel processor i-5
- Software: Python, Flask, CSS, HTML, Bootstrap, Machine Learning Libraries, Google Collaboratory

6.2 Project Estimates

6.2.1 Reconciled Estimates

- Inconsistent datasets pertaining to recipes due to missing nutritional information
- Low number of Indian Cuisines in Huge datasets, particularly Vegetarian Indian cuisine.
- Large size of dataset proved to be a hassle for testing the models due to unavailability of sufficient storage space

6.3 Risk Management

6.3.1 Risk Identification

- Redundant meal generation after content-based filtering
- Low number of users for testing Collaborative Filterng Model
- Consequent negative feedback leading to lack of suggestions for personalised meal plan
- Unavailability of datasets for Indian users

6.3.2 Risk Analysis

- Lack of high number of users led to the initial cold-start problem which delayed the testing of the model for similar suggestions.
- Constantly generating meal plan for a narrow user-profile resulted into repeated suggestions after 2-3 iterations
- Constant negative feedback resulted into lesser suggestions and more repeated suggestions for the user
- The variety in Indian cuisines weren't accounted for even in major food datasets which actually led to suggestion of cuisines similar in nutritional properties but remote in terms of location.

6.3.3 Risk Mitigation and Management

- Datasets were prepared manually to include Indian Cuisines and clean the data in a system-friendly manner.
- Availability of Indian Recipes Dataset, with all the necessary information.
- Current dataset has created by web scraping an open source website, www.tarladalal.com
- Meal plan algorithm is run again after a sufficient number of generations by the user leading to a new randomised meal set within user profile constraints.
- Constant negative feedback leads to the user getting a fresh meal-plan and the similarity engine is run on the new meal-plan.
- Availability of User Rating Dataset: Current approach is to create this dataset with the help of initial users.

6.4 Project Schedule

6.4.1 Project Task Set

- Decide scope and feasibility
- Paper review and domain research
- Setting up basic UI for testing workflow
- Building database
- Making Web-application and evaluating its performance

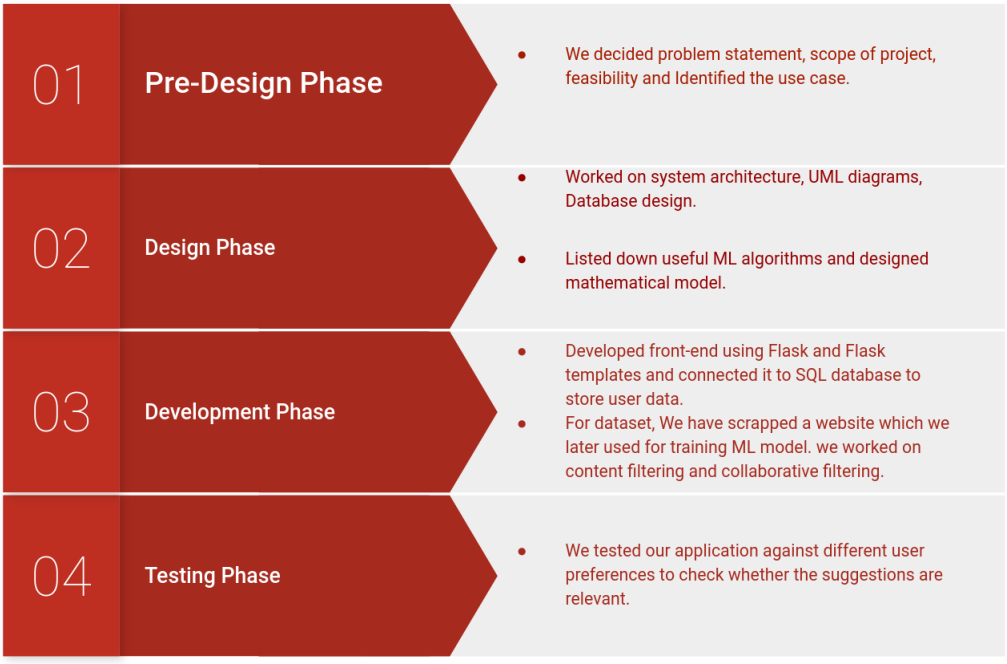

Figure 7: Task Set with description

6.4.2 Timeline

Figure 8: Timeline

7 Project Implementation

7.1 Overview of Project Modules

The project has 3 core modules that include sign-in authentication and user profile generation, meal-plan generation and feedback loop. The modules have been built using the Flask web framework.

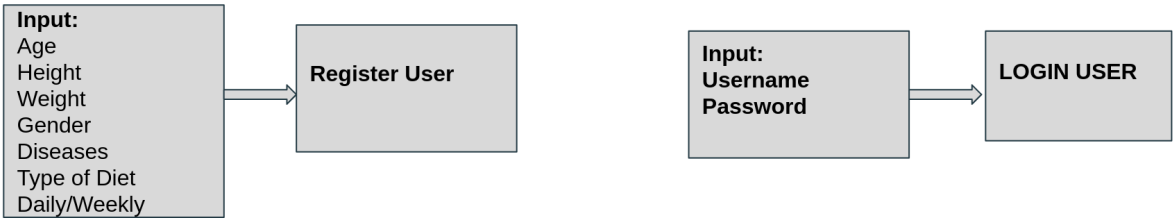

Figure 9: Sign-in module and Profile generation

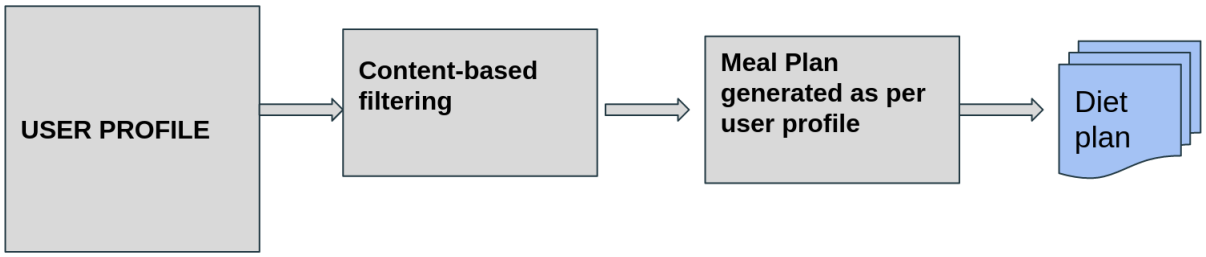

Figure 10: Meal Plan module

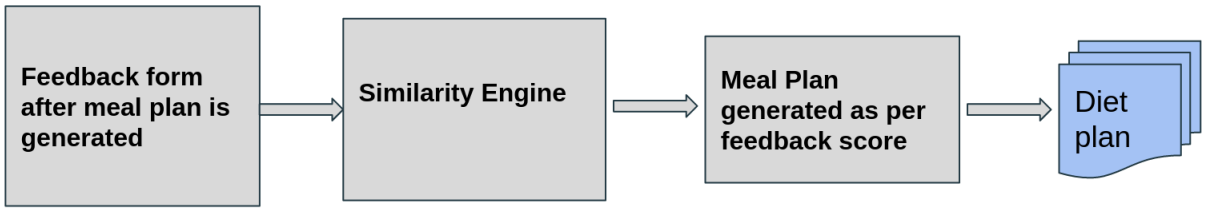

Figure 11: Feedback Loop

7.2 Tools and Technologies Used

- Python libraries used: scipy, sklearn and pandas
- Flask: micro web framework written in Python
- HTML, CSS, JavaScript- Building the frontend of the website.
- Platform used: Sublime, VScode

7.3 Algorithm Details

7.3.1 Content Based Filtering

It's a machine learning methodology that makes decisions based on feature similarity. This approach is frequently used to market or recommend products to people based on information gathered about them.
In this strategy, user interests are compared to product attributes. The product we select has the most features in common with the user's interests. There are two ways to do this-

- We can ask user to choose whatever they identify most from the given list of features.
- We can keep track of the things that the user has chosen in the past and use them. The main concept is to suggest goods that are like those you previously enjoyed.

7.3.2 Collaborative Filtering

The features of the items in the same embedding space are not required for collaborative. On its own, it creates embedding for both users and items. Both users and items are embedded in the same embedding space.

Other users' reactions are considered when giving a recommendation. User preferences are taken into account, as well as the things that people with similar behavior and preferences are interested in.

User feedback is a very important part of this technique, there two methods to obtain the user feedback-

- Implicit
- Explicit

In our project, we explicitly take user feedback into account. As an example, we have a distinct page for user comments, feedbacks, and ratings. Techniques which we used to find similarity amongst users was KNN

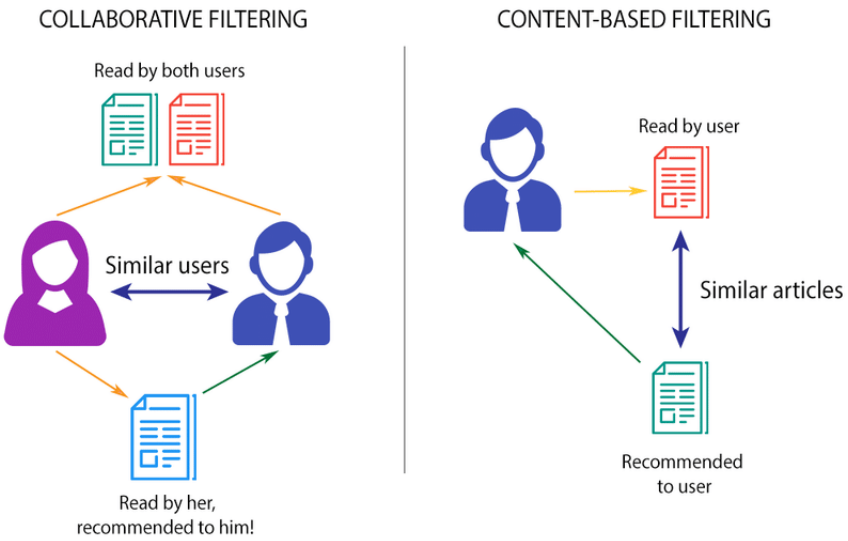

Figure 12: Content Vs Collaborative

- It's a non-parametric (it does not make any strong assumptions about the form of mapping function) and non-probabilistic supervised learning algorithm that classifies data rather than producing membership probabilities.
- It uses distance metrics rather than probabilistic ones. We employed a few similarity measures to calculate the similarities.

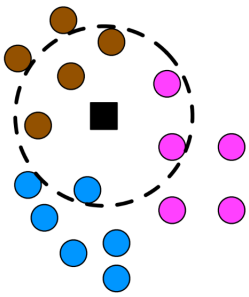

Figure 13: KNN

7.3.3 Cosine Similarity

This distance measure is mostly used to determine how similar two vectors are. It determines whether two vectors are pointing in the same direction by measuring the cosine of the angle between them. In text analysis, it's frequently used to determine document similarity. When combined with KNN, this distance gives us a new viewpoint on a business challenge and allows us to uncover hidden information in the data that the above two distance matrices could not reveal. It is also used in text analytics to compare two papers based on how many times a certain collection of words appears in each.

$$\cos\theta = \frac{\vec{a} \cdot \vec{b}}{\|\vec{a}\| \cdot \|\vec{b}\|}$$

Figure 14: Formula

7.3.4 Pearson Similarity

The most widely used correlation coefficient. It's a metric for how closely two sets of data are related. It only works if the following conditions are met:

- Both variables must be interval or ratio measures
- The relationship between the two variables must be linear.
- Both variables are normally distributed.

  Sample size should be large to use this. Value ranges from -1 to +1.
- 0 indicates no linear correlation
- +1 indicates perfect positive linear correlation
- -1 indicates perfect negative correlation

- **A popular similarity measure in user-based CF: Pearson correlation**
  - a, b : users
  - $r_{a,p}$: rating of user a for item p
  - P : set of items, rated both by a and b
  - Possible similarity values between −1 and 1

$$sim(a,b) = \frac{\sum_{p \in P}(r_{a,p} - \bar{r}_a)(r_{b,p} - \bar{r}_b)}{\sqrt{\sum_{p \in P}(r_{a,p} - \bar{r}_a)^2}\sqrt{\sum_{p \in P}(r_{b,p} - \bar{r}_b)^2}}$$

Figure 15: Formula

7.3.5 MinKowski Distance

It's a metric for vector spaces with real values. Minkowski distance can only be calculated in a normed vector space, which is a space where distances can be represented as a vector with a length that cannot be negative.
There are a few conditions that the distance metric must satisfy:

- Non-negativity: d(x, y) ¿= 0
- Identity: d(x, y) = 0 if and only if x == y
- Symmetry: d(x, y) = d(y, x)
- Triangle Inequality: d(x, y) + d(y, z) ¿= d(x, z)

$$\left(\sum_{i=1}^{n} |x_i - y_i|^p\right)^{1/p}$$

Figure 16: Formula

This Minkowski distance formula is in a generalised form, and we can alter it to acquire other distance metrics.
The p value in the formula can be changed to get various distances, such as

- p = 1, when p is set to 1 we get Manhattan distance
- p = 2, when p is set to 2 we get Euclidean distance
- p = infinity, when p is set to infinity we get Chebychev distance

7.4 Dataset

The application uses data scraped from tarladalal.com website. It has following features:

- Recipe ID
- Name : Name of the recipes
- Calories : Calorie contents
- Proteins : percentage of proteins
- Carbohydrates : percentage of Carbohydrates
- Fiber : percentage of Fiber
- Fats : percentage of Fats
- Cholestrol : Cholestrol contents
- Sodium : Sodium contents
- Ingredients : required ingredients and quantity
- Steps : Stepwise procedure to cook food
- Time : Total time required to cook food
- soup : Related tags

8 Software Testing

8.1 Methodology of Testing

- Unit Testing- All individual units are tested.
- Integration Testing– Testing the project after integrating every unit.
- Software Performance Testing– The model gives expected output or not.
- Functional Testing- This checks various functions of the application.
- User Acceptance Test– testing performed by end user to accept the final prototype of the system.

8.2 Scope of Testing

- Sign-in and authentication should be successful
- Meal plan displayed should adhere to user-profile
- User should be able to provide feedback
- Dashboard should display all vitals of the user
- Meal plan shouldn't be repetitive

No	Test Case	Input	Expected	Actual	Result
1	Register	User Details	Registration Successful	Registration Successful	Success
2	Register	Same User	Registration Unsuccessful	Registration Unsuccessful	Success
3	Login	Registered mail	Login successful	Login successful	Success
4	Meal Plan	Generate meal	Personalised-Meals displayed	Personalised-Meals displayed	Success
5	Feedback	Meal ratings	Updated Meal Plan	Updated Meal Plan	Success

Table 3: Test Case Results

8.3 Test Objectives

- All functional requirements are satisified
- System should meet all quality standards of a recommendation application.
- No security lapses in case of sensitive user information
- All constraints are met before deployment
- User Acceptance Test– testing performed by end user to accept the final prototype of the system.

9 Results

9.1 Interface Screenshots

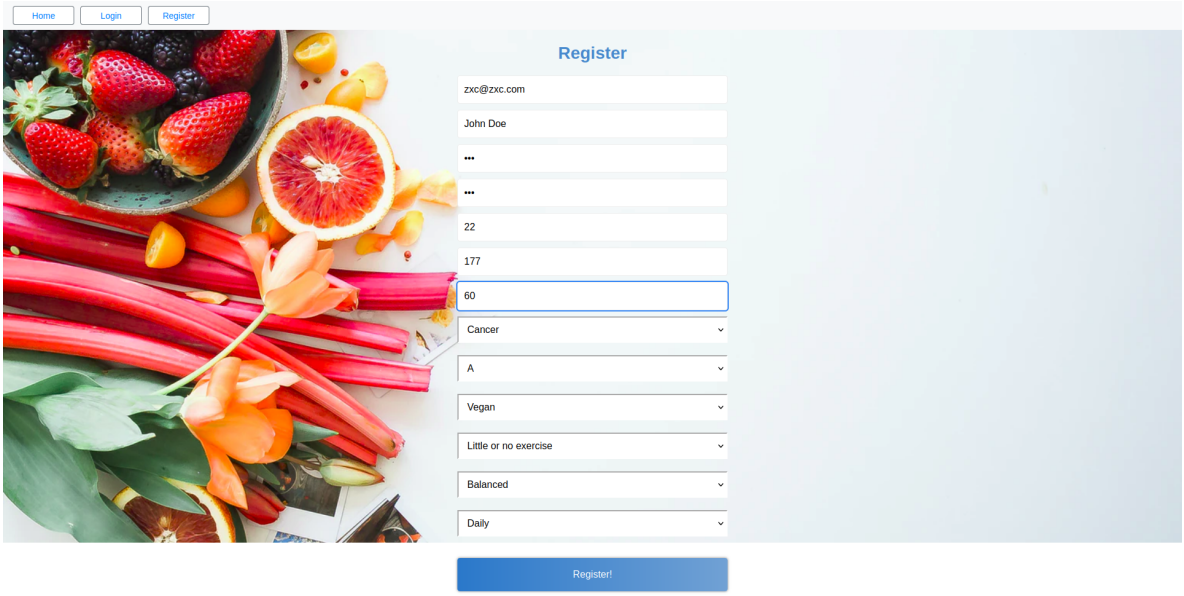

Figure 17: Registration Form

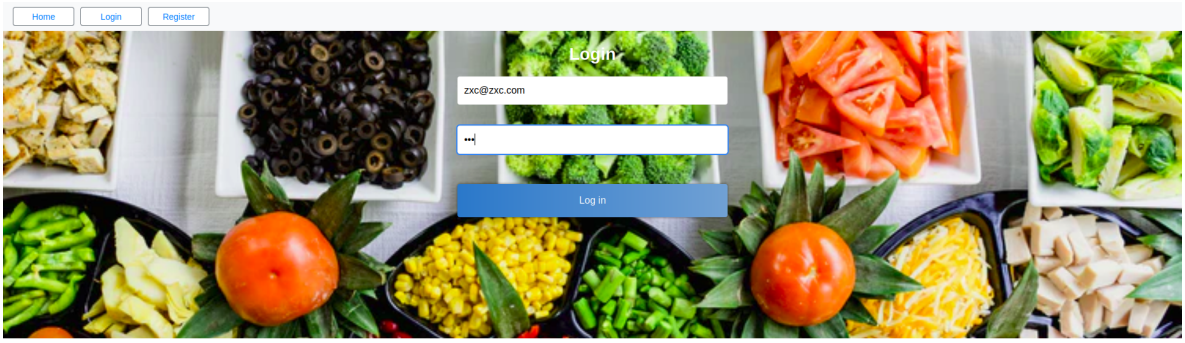

Figure 18: Login

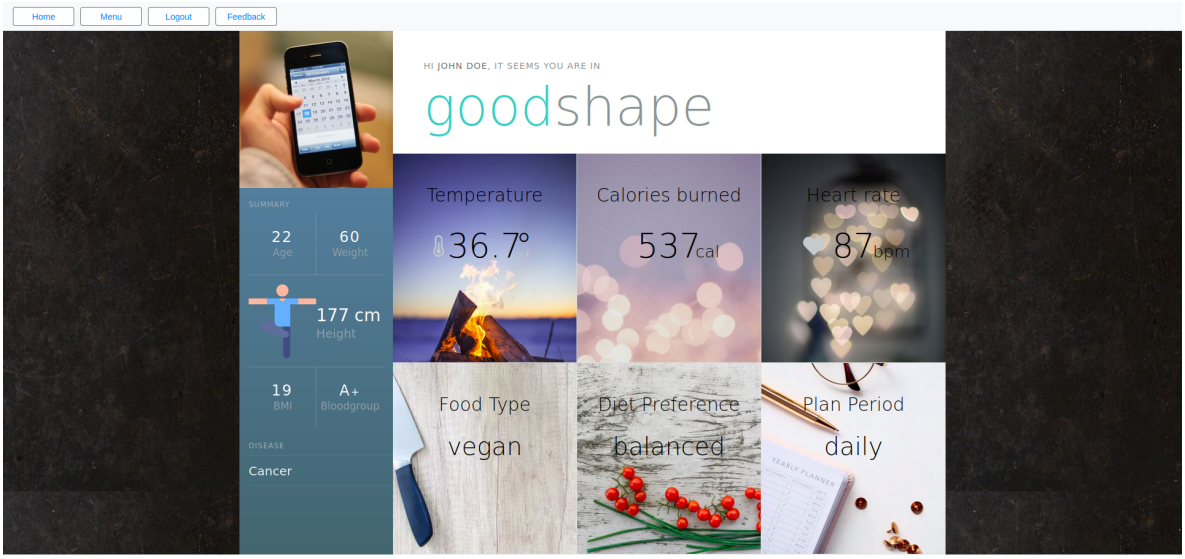

Figure 19: Dashboard

Figure 20: Generated Meal Plan

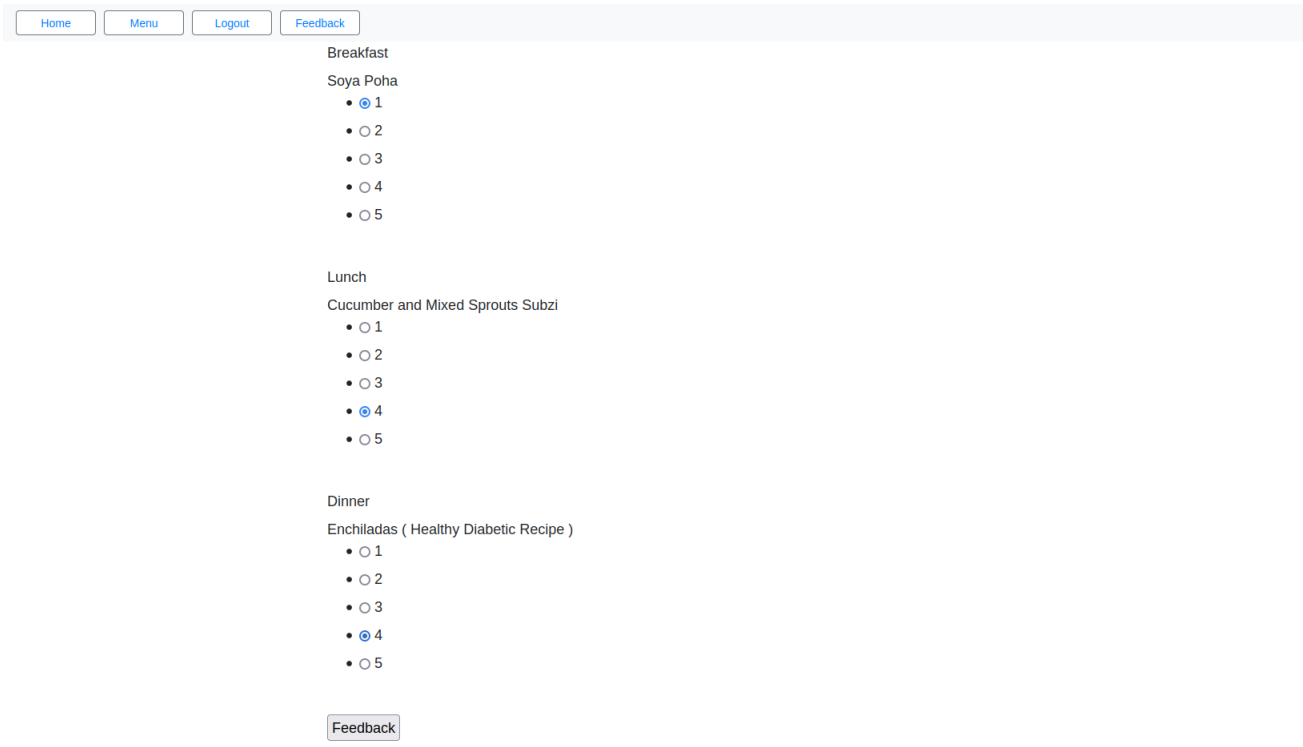

Figure 21: Feedback for User Rating

10 CONCLUSION AND FUTURE WORK

10.1 Conclusion

In this report we have mentioned all the requirements and design constraints of the system. The multiple design challenges posed due to constraints on data gathering and real-time systems have been addressed. Multiple constraints such as lack of availability of concentrated dataset of multiple cuisines and the cold-start problem have been clarified upon. This has helped us chart a course of action for the implementation of this application.

10.2 Future Scope

We aim to push the limits of this application further by making it the go-to application for both fitness and nutrition by aiming for a seamless connection between multiple IoT devices such as home systems and health monitors to provide accurate results that can help draw insights on the user and raise alarms in case of emergencies. We intend to further work on this application to make it a handy mobile/palm application and improve the accuracy of results for a particular demographic and at the same time recommend diet and provide fitness charts by saving the constraints provided by the user on both economic and medical fronts.

Appendix-A

Feasibility Assessment

Complexity Study:

- For a machine learning solution, the complexity of the problem will go hand in hand with the complexity of your solution.
- Response complexity is a calculation to help understand the complexity of the problem given labeled training data. Formula:
- RC= (n — 1) $\sigma 2$ (m — 1)

n ->number of inputs given to the model
$\sigma 2$ ->variance of the output in the training
set m - >number of possible outputs

Mathematical Model:

- Pearson correlation: Measuring user based nearest neighbor collaborative.

a,b : Users
$r_{a,p}$: Rating of user a for item p
p: set of items rated by both a and b

$$sim(a,b) = \frac{\sum_{p \epsilon P} (r_{\mathrm{a,p}} - r_{\mathrm{a}})(r_{\mathrm{b,p}} - r_{\mathrm{b}})}{\sqrt{\sum_{p \epsilon P} (r_{\mathrm{a,p}} - r_{\mathrm{a}})^2} \sqrt{\sum_{p \epsilon P} (r_{\mathrm{b,p}} - r_{\mathrm{b}})^2}}$$

- Making Prediction for unseen items:

$$pred(a,b) = \overline{r_{\mathrm{a}}} + \frac{\sum_{b \epsilon N} sim(a,b) * (r_{\mathrm{b,p}} - \overline{r_{\mathrm{b}}})}{\sum_{b \epsilon N} sim(a,b)}$$

Feasibility

- Diet planner recommends an appropriate and personalized diet chart using nutritional knowledge.
- It uses a decision analysis approach based on multiple criteria to screen inappropriate foods.
- These criterias include BMI, diseases(clinical or heredity), location, dietary preferences, real-time data and feedback.
- To improve usability and decrease complexity, we are going to create a simple user interface to ensure that everybody can use the application easily.
- Since our application will work efficient than any human, the diet plan will be flawless.
- Nutritionists can use the application as a helping hand in order to double check, thus increasing reliability.
- To improve the model, we will take feedback from the users about the accuracy of the recommended charts and utilize the same in fine-tuning the model further.

IDEA Matrix:

I	D	E	A
Increase healthy lifestyle habits	Detects for diseases and allergic food	Evolve periodically	Assists nutritionist
Improve reliability	Delivers preferred menu Enhancement		Accelerate healthy growth
Improve usability	Decreases confusion over what to eat	Eliminate allergic food confusion	Attention to detail

Table 4: IDEA Matrix

Appendix-C

Plagiarism Report

Curiginal

Document Information

Analyzed document	BE_Project_Report_Diet_Planner_Group_63.pdf (D108558694)
Submitted	6/11/2021 2:34:00 AM
Submitted by	Rekha Kulkarni
Submitter email	rakulkarni@pict.edu
Similarity	6%
Analysis address	rakulkarni.pict@analysis.urkund.com

Sources included in the report

W	URL: https://core.ac.uk/download/pdf/55305291.pdf Fetched: 5/9/2021 7:40:17 PM	2
W	URL: https://www.static-contents.youth4work.com/y4w/Documents/Portfolio/1a269a70-6cb0-4c5d-bd5b-4549a22d2d10.pdf Fetched: 12/11/2019 7:18:09 AM	1
W	URL: https://dsrajnor.files.wordpress.com/2018/04/project-work-book-2017-18.pdf Fetched: 6/1/2021 10:21:17 AM	1
W	URL: https://www.semanticscholar.org/paper/Realizing-an-Efficient-IoMT-Assisted-Patient-Diet-Iwendi-Khan/5670ce06f2437c966b526e1dcccb8483c74621e8 Fetched: 1/16/2021 11:58:33 AM	1
W	URL: https://www.researchgate.net/publication/338727172_Realizing_an_efficient_IoMT-assisted_Patient_Diet_Recommendation_System_through_Machine_Learning_Model Fetched: 1/16/2021 11:58:35 AM	2

References

[1] Luis Martínez Raciel Yera Toledo, Ahmad A. Alzahrani. A food recommender system considering nutritional information and user preferences. In *2019 IEEE International Conference on E-health Networking, Application Services (HealthCom)*, 2019.

[2] Joseph Henry Anajemba Ali Kashif Bashir Fazal Noor Celestine Iwendi, Suleman Khan. Realizing an efficient iomt-assisted patient diet recommendation system through machine learning model. In *EEE Access, vol. 8, pp. 28462-28474, 2020, doi: 10.1109/ACCESS.2020.2968537*, 2020.

[3] António Abelha José Machado Rui Miranda, Diana Ferreira. Intelligent nutrition in healthcare and continuous care. In *2019 International Conference in Engineering Applications (ICEA), Sao Miguel, Portugal, 2019, pp. 1-6, doi: 10.1109/CEAP.2019.8883496.*, 2019.

[4] Juan Li Shadi Alian Maryam Sadat, Amiri Tehrani Zadeh. Personalized meal planning for diabetic patients using a multi-criteria decision- making approach. In *2019*.