University of Mumbai

FitNoQuit

Synopsis of Final Year Project

by

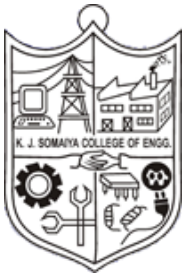
Bhavik Bhatt
Roll No: 1814007

Piyush Chavda
Roll No: 1814010

Bharat Choithani
Roll No: 1814011

Muskaan Nandu
Roll No: 1814020

Guide
Prof. Sonali Patil



Department of Information Technology
K. J. Somaiya College of Engineering, Mumbai-77
(Autonomous College Affiliated to University of Mumbai)

Batch 2021-22

K. J. Somaiya College of Engineering, Mumbai-77

(Autonomous College Affiliated to University of Mumbai)

Certificate

This is to certify that the report entitled **FitNoQuit** is bona fide record of Final Year Project work done by Bhavik Bhatt, Bharat Choithani, Piyush Chavda and Muskaan Nandu in the Sem VII, year 2021 under the guidance of under the guidance of Prof. Sonali Patil of Department of Information Technology during the period of August 2021- November 2021, in partial fulfillment of requirements for the degree of Bachelor of Technology.

Prof Sonali Patil	**Dr Ujwala Bhangale**
_____	_____
Guide	Head of the Department

Date: 5th November 2021

Place: Mumbai-77

K. J. Somaiya College of Engineering, Mumbai-77

(Autonomous College Affiliated to University of Mumbai)

Certificate of Approval of Examiners

We certify that this report entitled **FitNoQuit** is bona fide record of LY Project work done by Bhavik Bhatt, Bharat Choithani, Piyush Chavda and Muskaan Nandu under the guidance of Prof. Nilkamal More.

This project is approved for the award of Bachelors in Technology Degree in Information Technology of University of Mumbai.

_____

Internal Examiner

_____

External Examiner

Date: 5th November 2021

Place: Mumbai-77

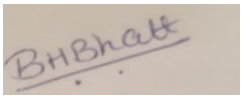
K. J. Somaiya College of Engineering, Mumbai-77

(Autonomous College Affiliated to University of Mumbai)

DECLARATION

We declare that this written report submission represents the work done based on our and / or others' ideas with adequately cited and referenced the original source. We also declare that we have adhered to all principles of intellectual property, academic honesty and integrity as we have not misinterpreted or fabricated or falsified any idea/data/fact/source/original work/ matter in our submission.

We understand that any violation of the above will be cause for disciplinary action by the college and may evoke the penal action from the sources which have not been properly cited or from whom proper permission is not sought.

Bhavik Bhatt  **Signature of the Student** **1814007** **Roll No.**	**Piyush Chavda**  **Signature of the Student** **1814010** **Roll No.**
Bharat Choithani  **Signature of the Student** **1814011** **Roll No.**	**Muskaan Nandu**  **Signature of the Student** **1814020** **Roll No.**

Date: 5th November 2021

Place: Mumbai-77

Dedicated to

Our guide and colleagues.

Contents

1. INTRODUCTION

1.1 Problem Definition

FitNoQuit aims to recommend personalized, customized and preference-based diets to its users that is sustainable and satisfies the end goal (weight loss, weight gain, maintain) of the user. The web application also aims to suggest workout to the users based on their comfort and taking into account the time duration they wish to dedicate for the workout. The system also suggests blogs to its users based on their categorical likings and preferences.

1.2 Motivation

The motivation for doing this project is to help people stay fit and eat healthy. FitNoQuit helps users to achieve their fitness goal and also share their experience with others through the blog community.

1.3 Scope of Project

FitNoQuit has 3 primary features:

- Diet Recommendation

  The system considers user inputs such as height, weight, age, diseases, food preferences, end goal etc to generate a personalized diet using machine learning algorithm. The diet consists of meals and foods that best suits the user and will help him/her to achieve their end goal.

- Workout Recommendation

  The system suggests workouts to user using mathematical computations and compliments with diet recommended to impact the overall calorie management of the user. The user is given options with respect to the time he/she wishes to dedicate for the workout along with workout options they prefer.

- Blog suggestions

  FitNoQuit offers a blog section to its users wherein they can select categories of blogs they wish to read. This section aims at keeping the users abreast with the current fitness trends and updates. With the help of collaborative and/or content-based filtering, users will be recommended blogs based on their interest and likings.

1.4 Functional and Non-Functional Requirements

Functional Requirements

The requirements/features that end-users shall expect and will be incorporated in the system are:

a. **Registration/Sign Up**
   New users will be able to access the system by registering or creating an account with "FitNoQuit".

b. **Login/Sign In**
   Existing users should be able to access the system using their username and password. Premium features will be available to premium registered users.

c. **User Profile**
Users will be able to edit and view their personal information from the User Profile section.

d. **Workout Recommendation**
The user will be recommended with appropriate workout and exercise based on their end goal (weight gain, weight loss maintain weight or manage a health condition)

e. **Diet Recommendation**
Suggest Diet Meals to the users in categories of Breakfast, Lunch and Dinner based on user's food preferences like Veg/Non Veg/Jain/Vegan, health condition like Thyroid, PCOS, Heart Conditions, etc

f. **Blog Suggestions**
The users will be suggested with relevant blogs in the Community Blog section using Content-Based filtering based on the previously liked blogs.

g. **Logout**
The user should be able to safely log out of the system without any loss of information.

Non-Functional Requirements

The basic quality constraints that will be satisfied by the system is as follows:

a. **Authorization:**
The user data will be confidential and only authorized nutritionists will be made available with the data.

b. **Speed:**
The application will respond to user's queries as soon as possible with a minimum waiting time.

c. **Portability:**
The system will be highly portable as it is a web-app and will work efficiently even if the user changes the device.

d. **Compatibility:**
Since the system is a web-app it can be accessed on any devices like laptop, tablet or mobile phones, irrespective of its Operating System.

e. **Security:**
Sensitive information like user passwords will be encrypted by using algorithms like SHA and then stored on the database for verification.

f. **Availability:**
The web-app will be available to all users with an internet-connectivity as we plan to host the website.

g. **Reliability:**
The recommendation system would be reliable as we would try to achieve maximum accuracy and get the recommendation model verified by a certified nutritionist.

1.5 Organization of the Report

Problem Definition

In this, we have explained the problem to be addressed in regards to diet and workout recommendations and how we have come upwith a solution with our project.

Background work

In this, we have explained about all the prerequisites and the background work needed before starting this project.

Motivation

In this, we have explained the main reason and motivation behind us taking this project.

Scope

In this we have explained the extent to which our project is limited. We have explained the limitations that user will face while inputting the form and up to what extend the graph simulation will work.

Future Scope

In this, we have explained all the things we can add in our project in the future to make our project even better.

Functional Requirements

In this, we have explained all the functional requirements.

Non Functional Requirements

In this, we have explained all the non-functional requirements.

Technologies Used

In this, we have explained about all the coding languages, algorithms, frameworks and data structures we have used.

2. Literature Review

Table 2.1 Literature Review

Sr. No	Title	Aim	Conclusion
1	A Personalized Healthy Diet Recommender System	The main aim is to develop a method to provide every user with meals of their choice, while ensuring that the correct proportion of nutrients are present in them. This is done by developing a diet recommendation system which recommends a healthy and appropriate food quantity to users.	The system presented is made up of two parts: the first part provides content based diet recommendation while the second part uses Pearson Correlation Coefficient to compare food nutrients and recommends alternative food items, thus allowing users to make choices. The functionalities were tested by making use of data of patients collected from the OAUTHC Hospital Complex. Thus, a system that considers an individual's daily energy requirement in order to maintain a healthy weight and reduce the risk of chronic diseases has been developed by considering the food preferences of the user.
2	Website on Diet Recommendation Using Machine Learning	The objective of this study is to consider various important aspects of the user's lifestyle while recommending a healthy and nutritious diet for the user and encouraging user to incorporate physical activity in their lifestyle.	The system has 3 phases: Information Collection Phase, Learning Phase and Recommendation Phase. The learning phase makes use of 2 ML Algorithms: K-Means and Random Forest to predict food items based on user inputs. The system will recommend diet in 3 categories:

			Breakfast, Lunch and Dinner. The users can choose from the multiple recommendations and a comparison of calories will be shown based on which user can design a diet plan.
3	A Food Recommender System Considering Nutritional Information and User Preferences	This paper presents a general framework for daily meal plan recommendations with the simultaneous management of nutritional-aware and preference-aware information. The proposal incorporates a pre-filtering stage that uses AHPSort as multi-criteria decision analysis tool for filtering out foods which are not appropriate to the current user characteristics. Furthermore, it incorporates an optimization-based stage for generating a daily meal plan whose goal is the recommendation of food highly preferred by the user, not consumed recently, and satisfying his/her daily nutritional requirements.	The current paper has presented a food recommendation approach focused on generating daily personalized meal plans for the users, according to their nutritional necessities and previous food preferences. It presents a general architecture for food recommendation, composed by an information gathering layer, the user profile dataset, the intelligent system layer, and an end user interface.
4	Website on Diet Recommendation Using Machine Learning	A cloud based food recommendation system, called Diet-Right, for dietary recommendations based on users' pathological reports. The model uses ant colony algorithm to generate optimal food list and recommends suitable foods according to the values of pathological reports	The experimental results show that compared to single node execution, the convergence time of parallel execution on cloud is approximately 12 times lower. Moreover, adequate accuracy is attainable by increasing the number of ants

5	e-Health Monitoring System with Diet and Fitness Recommendation using Machine Learning	Paper proposes a system that aims at improving the health of the patients suffering from various diseases by recommending them healthier diet and exercise plans by analyzing and monitoring health parameters and the values from their latest reports related to the disease. Authors have considered patients suffering from either Diabetes or Blood pressure or Thyroid	The paper deals with health monitoring of disease like Diabetes, Blood Pressure, and Thyroid based on the patient's latest report looking for improvements in every follow-up session and recommending suitable and updated diet and exercise plan in each follow-up session based on the reports and other credentials like height, weight, age, activity level, using the Machine Learning technique i.e. C4.5 decision tree algorithm.

Whats Different in FitNoQuit?

- Our proposed system takes into account both diet and workout recommendations which has not been included in the papers covered in the literature review and existing system like "Fittr".
- Existing systems like "Cure fit" and "Google fit" don't give personalized diet recommendations and our web app aims to address this feature and incorporate it.
- Workout suggested in present systems like "HealthifyMe" do not take into account the user preferences. FitNoQuit will provide workout options to choose from which gives the user a better experience.

FitNoQuit - SPMP

1814007, 1814010, 1814011, 1814020

September 2021

1 INTRODUCTION

1.1 Product Overview

FitNoQuit is a health and fitness web app made for anyone who wants to enter the world of fitness. The main purpose of this product is to help and properly guide beginners. This website recommends a fully customized and personalized diet and workout plans for different individuals (based upon their inputs). This helps them to not only have an effective routine, but also helps them to enjoy thier routines as it is made according to their likings.

1.2 Project Deliverables

Deliverable	Delivery Date
Project details	3rd Sept 2021
Project Presentation	23rd Sept 2021
Scope, Technology and Tools	24th Sept 2021
SRS and SPMP	1st Oct 2021
SDD	8th Oct 2021
STD	15th Oct 2021
Presentation and Demonstration	22nd Oct 2021
Synopsis	5th Nov 2021
Data Collection	20th Jan 2022
Sign up and Login	31th Jan 2022
Handling User Inputs	15th Feb 2022
Diet Recommendation	3rd March 2022
Blog Section	3rd March 2022
Workout Recommendation	31th March 2022
UI/UX	15th April 2022
Testing and Research Paper	29th April 2022

2 PROJECT ORGANIZATION

2.1 Software Process Models

This section describes the project process model and its organizational structure. The project uses an Agile development model that allows the customer to interact with and work with software that works at the end of each iteration and provide feedback on it. This approach allows teams to easily adapt and make adjustments if needed. Agile method, software is built and more extruded in duplication. The development process is organized into a number of activities.

We have divided each module into several Iterations. All changes will be simultaneous (between two to eight weeks). At the end of each iteration, an active product will be sent. Any remaining features that cannot be brought to the first iteration will be taken to the next iteration or to the next iteration, depending on the progress. At the end of the first duplication, the team will submit active software with features that were completed during that period. The Agile approach places great value on team interaction, customer interaction, responsiveness to changes and the delivery of active software.

We can satisfy the customers because after all the Sprint operating features are delivered to them. Customers can be seen with performance that meets expectations. If customers have specific reports or changes in the item

they may be included in the current product release. In an Agile way every-day communication is needed between entrepreneurs and developers. In this way good product care is considered. Changes to needs were welcomed even in recent stages of development.

2.2 Roles and Responsibilities

Roles	Description	Person
Project Manager	Overall planning, risk analysis execution planning, monitoring and closure planning	Bhavik Bhatt
Front-End Developer	To develop and implement a design of the required system and ensuring that all requirements are satisfied.	Muskaan Nandu,
Technical Lead	Leading the development team	Bharat Choithani
Development Team	Implementing the functionalities required in the application through coding.	Bhavik Bhatt Piyush Chavda Bharat Choithani Muskaan Nandu
Database Management	To design and maintain the database and to add, modify, delete, retrieve data from the database efficiently.	Bhavik Bhatt Piyush Chavda Bharat Choithani Muskaan Nandu
UI/UX Designer	To prepare a prototype of the website and UI	Piyush Chavda
Testing team	To check whether the functionalities work as expected and to find out bugs in the system.	Bhavik Bhatt Piyush Chavda Bharat Choithani Muskaan Nandu

2.3 Tools and Techniques

Activity	Tools required	Technique to be used
Documentation /User Manual	Microsoft word, PDF converter, GanttProject, LaTeX	The procedure would be given in steps wherever necessary labeled images and diagrams will be used

UI/UX Designing	Marvel App, Adobe XD	The prototype will be first prepared before coding After the design is confirmed, implementation will be done
Development	VS Code, Sublime text, Atom MySQLdatabase, Github Python, Django, Sckit-learn, Keras	The development teams, would be working on their prototype for development.
Meetings	Microsoft Teams, Google Meet, ZOOM	The agenda of the meeting along with the meeting link would be shared with all the attendees prior to meet
Testing	Mobile, desktop and tablet having different browsers, testing tools	Test cases will be prepared for the testing team to conduct the tests

3 Project Management Plan

3.1 Tasks

3.1.1 Requirement Analysis (Task1)

3.1.1.1 Description : The team is required to elicit and elaborate on the requirements. All essential requirements must be noted down by asking questions and taking surveys from the customer.
3.1.1.2 Deliverables and Milestones : A list of all requirements must be submitted in a brief document along with an elaborated SRS document which clearly explains all the requirements of the software.
3.1.1.3 Resources Needed : Surveys, Research, Feedback meetings with the mentor.
3.1.1.4 Dependencies and Constraints' : The surveys should be appropriate and should not have unnecessary questions that might waste the mentor's time.
3.1.1.5 Risks and Contingencies : If the requirements are not analysed well there is a high chance of an unhappy and unsatisfied mentor. Hence all the requirements must be clearly discussed with the mentor and constant communication should be maintained to ensure mentor satisfaction.

3.1.2 User Module (Task2)

3.1.2.1 Description : This task contains developing the user dashboard. The User should be able to login to the system, enter different inputs like his height, weight, workout time, food preferences etc.and get an appropriate diet and workout plan. The user also gets an option to interact with the blog community that is present in out website, where he can like, comment on and get recommendations on the blogs.
3.1.2.2 Deliverables and Milestones : At the set date of meetings the team should show a demo of the login system, user inputs and previous recommendations respectively. Along with live demo a document with images and description of each vital function must be submitted.
3.1.2.3 Resources Needed : Documents, coding tools, packages.
3.1.2.4 Dependencies and Constraints' : The sub-functionality of each module must be completed on the specified deadline as set in the deliverable dates and gantt chart.

3.1.2.5 Risks and Contingencies : The risk of user being unsatisfied with the work remains, hence back up plans and ideas must be provided to the user when such situations arise.

3.1.3 Machine Learning Module (Task3)

3.1.3.1 Description : This task contains developing the ML model. Proper and appropriate ML models must be trained on rich and diverse datasets. These models must then be fine tuned according to specific needs upto the point of lowest error score.

3.1.3.2 Deliverables and Milestones : At the set date of meetings the team should show a demo of the ML model working according to specific inputs entered by the team and the results must be accepted by the certified dietitian. Along with live demo a document with images and description of each vital function must be submitted.

3.1.3.3 Resources Needed : Documents, coding tools, packages, schit-learn, keras, pandas, numpy, dataset.

3.1.3.4 Dependencies and Constraints' : The sub-functionality of each module must be completed on the specified deadline as set in the deliverable dates and gantt chart. The model must have an accuracy of at least 90

3.1.3.5 Risks and Contingencies The risk of not meeting deadlines can be very daunting. Thus, work must begin on time. If a particular deadline might not be met then the customer must be informed immediately. The risk of the ML model not predicting accurately and not satisfying the user needs can be very daunting.

3.1.4 Testing Software (Task4)

3.1.4.1 Description : The software must be tested well before deployed for public use.

3.1.4.2 Deliverables and Milestones : A document of all errors found while testing must be prepared and submitted. Each error must be logged and must be taken care of.

3.1.4.3 Resources Needed : Documents, images, videos, survey forms.

3.1.4.4 Dependencies and Constraints' : The testing phase might take a little longer as all errors must be resolved and thus sufficient time must be assigned for it.

3.1.4.5 Risks and Contingencies If the software is not tested well there stands a risk of deploying a faulty software, Thus before making it available for public use the software must be well tested.

3.2 Assignments

Mr. Bhavik Bhatt

The roles assigned to Bhavik are: Project Manager, Technical Lead, Database Management and Development Team. Thus, he must take care of the overall planning, risk analysis, developing and ensuring all the customer's requirements have been full-filled.

Mr. Piyush Chavda

The roles assigned to Piyush are: UI/UX Designer, Development Team, Database Management and Testing Team. Thus, he must take care of the overall designing process, contribute to the development and testing team.

Mr. Bharat Choithani

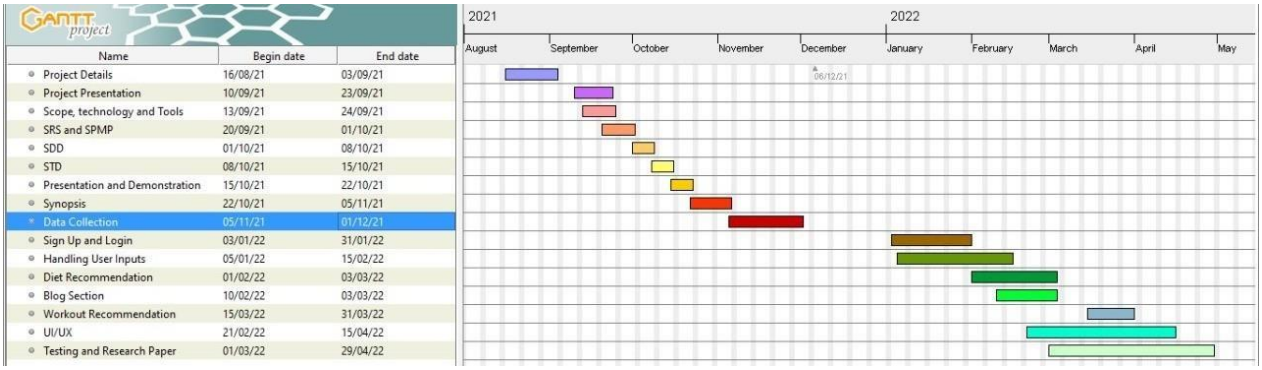
The roles assigned to Bharat are: Technical lead, ML Engineer, Development Team, Database Management and Testing Team. Thus, he must take care of the all the technical aspects including ML and contribute to the development and testing team.

Ms. Muskaan Nandu

The roles assigned to Muskaan are: Front End Engineer, database Management, Developing and Testing Team. Thus, she must take care of the design and maintenance of database, prepare a prototype for the website and test the app to check whether the functionalities are as expected without any bugs in the system.

3.3 Timetable

Represented using a Gantt Chart as shown below.



Name	Begin date	End date
Project Details	16/08/21	03/09/21
Project Presentation	10/09/21	23/09/21
Scope, technology and Tools	13/09/21	24/09/21
SRS and SPMP	20/09/21	01/10/21
SDD	01/10/21	08/10/21
STD	08/10/21	15/10/21
Presentation and Demonstration	15/10/21	22/10/21
Synopsis	22/10/21	05/11/21
Data Collection	05/11/21	01/12/21
Sign Up and Login	03/01/22	31/01/22
Handling User Inputs	05/01/22	15/02/22
Diet Recommendation	01/02/22	03/03/22
Blog Section	10/02/22	03/03/22
Workout Recommendation	15/03/22	31/03/22
UI/UX	21/02/22	15/04/22
Testing and Research Paper	01/03/22	29/04/22

FitNoQuit - Software Requirement Specification

1814007 - Bhavik Bhatt, 1814010 - Piyush Chavda,
1814011 - Bharat Choithani, 1814020 - Muskaan Nandu

26 September 2021

1 INTRODUCTION

1.1 Product Overview

Every human being has a different body with different requirements and thus as the saying goes “one size may not fit all”, a single diet and workout routine would not serve all. To solve this issue, “FitNoQuit” aims to provide its users with personalized diet and workout recommendations. Using machine learning algorithms, we aim to provide our users with a customized diet and workout routine that best suits them. These algorithms take into context a ton of user details like height, weight, age, gender, food preference, medical conditions etc. These details enable the algorithm to formulate a plan that is accurate and easy to follow for the users. The users can also browse through the blog section where in a plethora of information with respect to current diet and workout trends, myths regarding health and so on is available. The user is recommended with related blogs using content-based filtering which aims to recommend users with blogs from a category/author that the users previously showed interest in.
To sum up, “FitNoQuit” is a knowledge-based system that aims to provide diet and workout services to its users that is personalized and most apt to the user.

2 SPECIFIC REQUIREMENTS

2.1 External Interface Requirements

2.1.1 User Interfaces

The screen formats and menu structure should be in such a way that users will find easy to use. The product must be user-friendly and inter-active. The interface must be easy to understand. The user interface includes

- SCREEN FORMATS/ORGANIZATION: The introductory screen will be the user dashboard from where the user can access different modules of the web-app like diet recommendation, workout recommendation, blog section etc.
- WINDOW FORMAT/ORGANIZATION: When the user chooses some other option, then the information pertaining to that choice will be displayed in a new window which ensures multiple windows to be visible on the screen and the users can switch between them.
- DATA FORMAT: The data entered by the users will be alphanumeric.
- END MESSAGES: When there are some exceptions raising error like entering invalid details, then error messages will be displayed prompting the users to re-enter the details.

2.1.2 Hardware Interfaces

Are as follows:

Source of input: Keyboard, Mouse
To accept data from user like height, weight, age, user preferences and other personal and medical information

Destination of Output: Desktop Screen
Display the personalized workout and diet recommendations along with the option of browsing through blogs via the desktop screen.

2.1.3 Software Interfaces

- Languages: Python, HTML, CSS, JavaScript, PHP
- Framework: Django, BootStrap

- Database: SQLite

2.1.4 Communication Protocols

Usage of simple electronic forms which make use of SMTP and HTTPS protocols.

2.2 Software Product Features

2.2.1 Functional Requirements

The requirements/features that end-users shall expect and will be incorporated in the system are:

1. **Registration/Sign Up**

New users will be able to access the system by registering or creating an account with "FitNoQuit".

2. **Login/Sign In**

Existing users should be able to access the system using their username and password. Premium features will be available to premium registered users.

3. **User Profile**

Users will be able to edit and view their personal information from the User Profile section.

4. **Workout Recommendation**

The user will be recommended with appropriate workout and exercise based on their end goal (weight gain, weight loss maintain weight or manage a health condition)

5. **Diet Recommendation**

Suggest Diet Meals to the users in categories of Breakfast, Lunch and Dinner based on user's food preferences like Veg/Non Veg/Jain/Vegan, health condition like Thyroid, PCOS, Heart Conditions, etc

6. **Blog Suggestions**

The users will be suggested with relevant blogs in the Community Blog section using Content-Based filtering based on the previously liked blogs.

7. **Logout**

The user should be able to safely log out of the system without any loss of information.

2.2.2 Non-Functional Requirements

The basic quality constraints that will be satisfied by the system is as follows:

1. **Authorization**

The user data will be confidential and only authorized nutritionists will be made available with the data.

2. **Speed**

The application will respond to user's queries as soon as possible with a minimum waiting time.

3. **Portability**

The system will be highly portable as it is a web-app and will work efficiently even if the user changes the device.

4. **Compatibility**

Since the system is a web-app it can be accessed on any devices like laptop, tablet or mobile phones, irrespective of its Operating System.

5. **Security**

Sensitive information like user passwords will be encrypted by using algorithms like SHA and then stored on the database for verification.

6. **Availability**

The web-app will be available to all users with internet-connectivity as we plan to host the website.

7. **Reliability**

The recommendation system would be reliable as we would try to achieve maximum accuracy and get the recommendation model verified by a certified nutritionist.

2.3 Software System Attributes

2.3.1 Reliability

The factors needed to establish the software expected reliability are

- The user inputs should be valid and within the given range.
- Normal execution of the system without glitches
- Appropriate navigation and restricting access to unauthorized users

2.3.2 Availability

The factors guarantee the software's availability includes proper termination and correct input details. Also the resources used for the project development

are Python Certified which speaks of its high quality standards.

2.3.3 Security

Are as follows:

- It must be ensured that access will be provided to the authorized persons through user ID and password.
- Passwords must be Strong.
- No loss of data for any user must be ensured.
- Checks can be performed at regular internals to ensure data integrity.

2.3.4 Maintainability

The software will be developed by implementing the concept of modularity which in turn reduces the complexity involved in maintaining it. The administrator should have a sound technical knowledge about maintaining the software and further enhancements will be undertaken by the developer

2.3.5 Portability

The web application is portable to any device-mobile or desktop and is adaptable for use on different browsers with different device models and standards.

2.3.6 Performance

FitNoQuit application should be able to respond to the queries submitted by the customer without much delay.
- When a user inputs personal information in order to obtain workout and diet recommendations,the application should return the results in minimum duration.
- The user should be displayed blog recommendations that are aligned with their liking and the loading time for blogs should be less than 5 seconds.

2.4 Database Requirements

The system requires the use of SQLite database to maintain the customer's personal details and their medical information. An entity should be used to denote all the workout plans, diet classes (Veg, Non-Veg, Jain and Vegan) and the number of calories associated with them

FitNoQuit - Software Design Document

BHAVIK BHATT (1814007) PIYUSH CHAVDA (1814010)
BHARAT CHOITHANI(1814011)MUSKAAN NANDU (1814020)

October 2021

1 Introduction

Daily nutrition and food, particularly for those suffering from small or serious ailments, is one of the most important aspects for a good life. According to numerous researches, improper and inadequate daily dietary intake are the primary causes of a variety of health problems and illnesses. In light of the above facts, it is important to maintain a well-balanced diet. However, due to the vast variety of food items, it is difficult for the average individual to keep track of personal food requirements. Therefore, a systematic diet recommendation system is required to recommend the appropriate food considering the user preferences.

1.1 Design Overview

FitNoQuit is a web app which provides users with satisfactory, personalized, customized and enjoyable workout and diet routines. Users just need to input his/her food and activity choices and will get an appropriate plan. This web app also contains a big blog community where users can share their thoughts, give feedbacks, ask questions and answer to other user queries.

1.2 Requirements Traceability Matrix

Component / Requirements	Authentication	Updating Profile	Diet Recommendation	Workout Recommendation	Blogs
User Name	X	X	X	X	X
Password	X				
Database Connectivity	X	X	X	X	X
User choice of foods			X		
User medical conditions			X		
User choice of physical activities				X	
Blog/post details					X

Requirement Traceability Matrix

2 System Architectural Design

2.1 Chosen System Architecture

The system is defined using an Entity Relationship diagram and a data flow diagram. The data flow diagram is attached at the end of the document.
The different entities of this system are as follows:

1. User - The user entity has different attributes like username, age, height, weight, gender, email id, food preferences, health condition, weight goal, derived attribute like BMI and primary key User ID.

2. Diet Plan - It has different attributes like Diet Plan ID which is the primary key and a composite attribute the recommended Diet Meals which is further divided into Breakfast, Lunch, Snacks and Dinner.

3. Workout Plan - It has different attributes like Workout Plan ID which is the primary key and a composite attribute Workout plan which is further divided into three different options which the users can choose.

4. Blogs - The users can read different blogs ehich has the primary key Blog ID and other attributes like date of publication and number of likes.

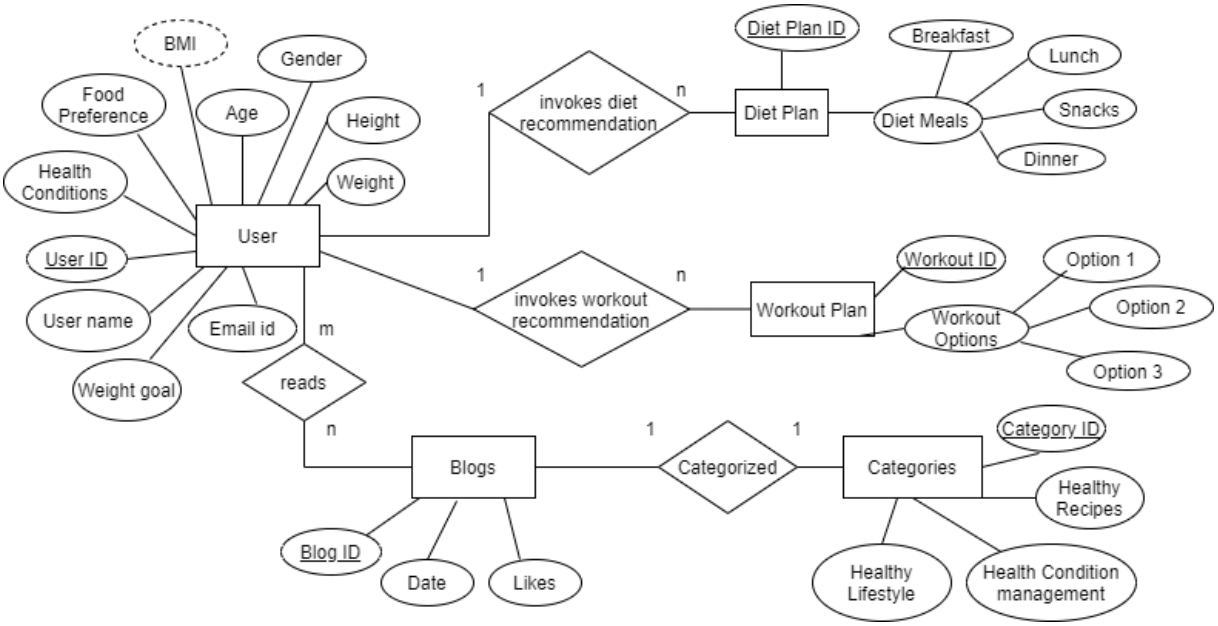
5. Categories - The blogs are further divided into categories like healthy lifestyle, health condition management, healthy recipes, etc. Each category is identified by a primary key labelled Category ID.

2.2 Discussion of Alternative Design

An Alternative Design element that can be used to model the system can be an Component diagram where different components would be the Authentication, Recommendation of Diet, Recommendation of Workout and Blog Suggestions.

2.3 System Interface Description

The system will be able to run on Windows, Linux and Mac OS Platforms. The system will run on a web server using different graphics and an interface which makes it easy for the user to execute various functionalities of the web app.



ER Diagram

3 Detailed Description of Components

3.1 Authentication

Responsibility - Bhavik Bhatt
Constraints - User must sign up or sign in to the system to be able to access the profile and obtain workout and diet recommendations.
Composition - The user must provide details like user name or Email ID and sign in with their password.
Interactions - Client server interaction using the web browser of user
Resources - Database and Validation.

3.2 Recommendation of Diet

Responsibility - Bhavik Bhatt
Constraints - User needs to be registered and signed in
Composition - Enter personal details like height, weight, age, end goal and food preferences.
Interactions - Client server interaction using the web browser of user
Resources - Database, Authentication and Machine Learning model.

3.3 Recommendation of Workout

Responsibility - Muskaan Nandu
Constraints - Only an authorised person can access the profile and obtain workout routine.

Composition - Enter personal details like height, weight, age, end goal and food preferences.
Interactions - Client server interaction using the web browser of user.
Resources - Database, Authentication and Machine Learning model.

3.4 Blog Suggestions

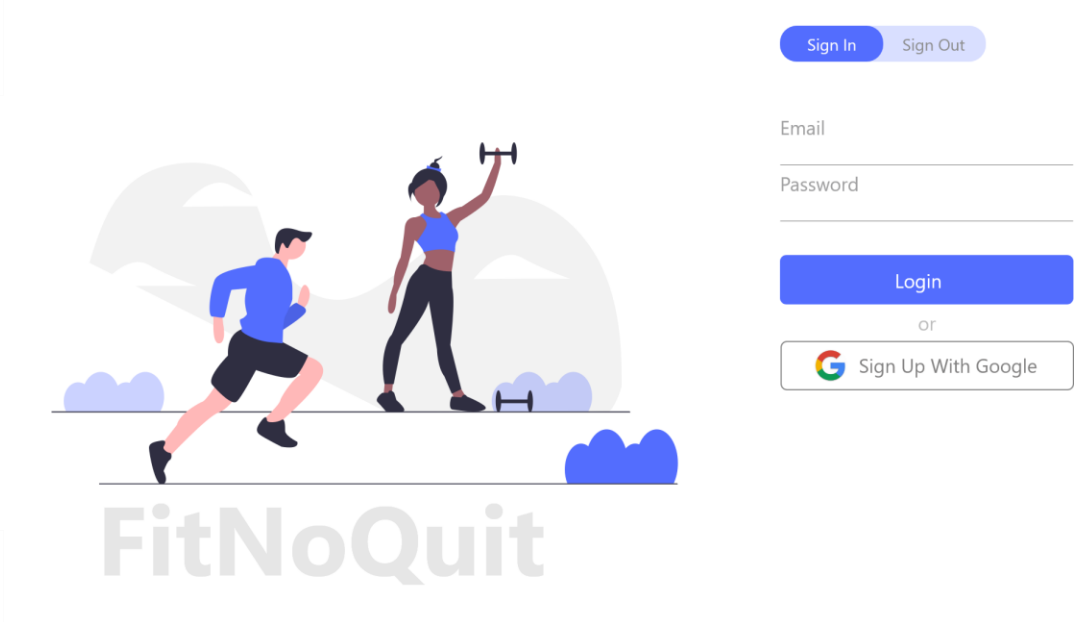
Responsibility - Bhavik Bhatt
Constraints - The registered user must select a category from the available categories of blogs.
Composition - Enter category of interest for blog browsing.
Interactions - Client server interaction using the web browser of user
Resources - Database, Authentication.

4 User Interface Design

4.1 Description of the User Interface

4.1.1 UI Screen for Login Page

4.1.1.1 Screen Images



UI Screen for Login Page

4.1.1.2 Objects and Actions
The Different Objects on this screen are a descriptive image for fitness followed by two text fields which take the user's email and password as input. On clicking Login the user credentials are verified and a valid user is taken to his/her account. If the user is not a

registered user he/she must click on the Sign Up button which takes the user to Sign Up page. The user can also sign in using google if he has registered using google.

4.1.2 UI Screen for Sign Up page

4.1.2.1 Screen Images

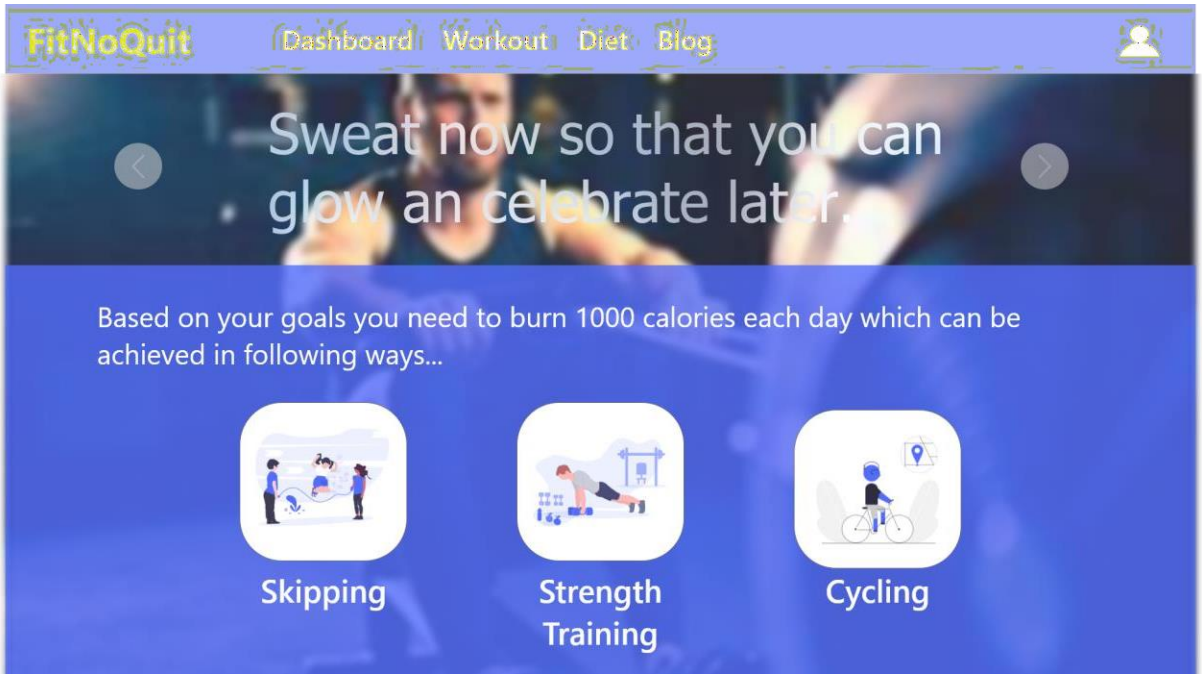
UI Screen for Sign Up screen

4.1.2.2 Objects and Actions

The Different Objects on this screen are a descriptive image for fitness along with five text fields which take the user's name, age, gender, email and password as input. On clicking "Create New Account" the user credentials are validated and a valid user account is registered. The user can also sign up using google account.

4.1.3 UI Screen for Workout Recommendation

4.1.3.1 Screen Images



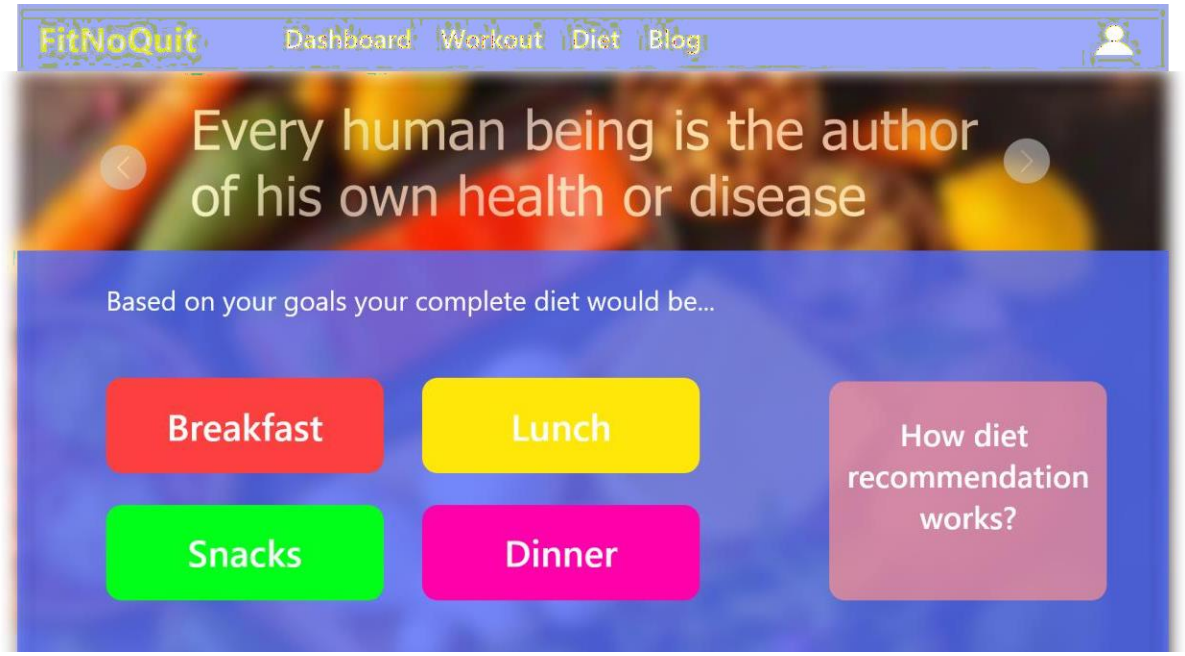
UI Screen for Workout Recommendation Page

4.1.3.2 Objects and Actions

The Different Objects on this screen are a descriptive image for fitness along with different workout options from which the user can select the one that he prefers.

4.1.4 UI Screen for Diet Recommendation

4.1.4.1 Screen Images



UI Screen for Diet Recommendation

4.1.4.2 Objects and Actions

The Different Objects on this screen are a descriptive image for healthy food along with different meals of the day from which the user can select the one that he wants to view the diet plan for.

4.1.5 UI Screen for Blog Suggestions

4.1.5.1 Screen Images

UI Screen for Blog Suggestions

4.1.5.2 Objects and Actions

The screen consists of different categories of blogs from which the user can click on a blog topic and browse through the information consisted in that blog.

5 System Architecture

USE CASE 1	**Sign up**
Description	The system creates a new account for a customer via the Sign-up form/page
Used by	New users
Pre-Conditions	We know user personal details.
Successful End Condition	User creates a new account.
Failed End Condition	User is not able to create a new account.
Actors	Users, Database
Trigger	User wants to use the web app.
Normal Flow	ACTIONS 1. The user selects signup Button from the Navigation Bar 2. The user fills the signup form with all the necessary details like email and password 3. The user submits the form. 4. User information is validated in the system. 5. If the information is valid, then the user is registered in the system. The user is then redirected to the login page.
Alternative Flow	1. Sign up with Google
Extensions	BRANCHING ACTIONS 1. User files invalid details like password and confirm password fields don't match, digits in name, invalid email id, etc. 1. 1. User is not able to sign up.
Priority	High
Frequency of Use	High

Usecase 1

USE CASE 2	**Login**
Description	The system logs into an existing account.
Used by	Registered users
Pre-Conditions	We know user personal details.
Successful End Condition	User logs into his/her account.
Failed End Condition	User is not able to log into his/her account.
Actors	Users, Database
Trigger	User attempts to sign in
Normal Flow	ACTIONS 1. The user selects login Button from the Navigation Bar 2. The user fills the login form with all the necessary details like email and password. 3. The user submits the form. 4. User information is validated in the system. 5. If the information is valid, then the user is logged into the system. The user is then redirected to the homepage.
Alternative Flow	1. Sign in with Google
Extensions	BRANCHING ACTIONS 1. User files invalid details like wrong credentials, invalid email id, etc. 1.1. User is not able to login.
Priority	High
Frequency of Use	High

Usecase 2

USE CASE 3	**Update Profile**
Description	The system updates the user details in the system
Used by	Users
Pre-Conditions	The user should exist in the system and should be logged in
Successful End Condition	User is able to change his/her personal details.
Failed End Condition	User is notable to change his/her personal details.
Actors	Users, Database
Trigger	User wants to change his/her personal details.
Normal Flow	ACTIONS 1. The user logs in to the system. 2. The user selects edit account details from Navigation bar. 3. All details except the email of the user is editable. 4. The user makes the desired changes. 5. The new form is submitted to the system for validation. 6. The system validates the user information. If the information is valid the user details are updated.
Alternative Flow	-
Extensions	BRANCHING ACTIONS 1. User files invalid details like invalid name, wrong image, etc. 1.1. User is not able to update his/her profile.
Priority	High
Frequency of Use	High

Usecase 3

USE CASE 4	**Diet Recommendation**
Description	User fills in his/her choices of food and gets a customized diet plan.
Used by	Users
Pre-Conditions	We know user personal details and food choices.
Successful End Condition	User gets a satisfactory diet plan.
Failed End Condition	User gets a unsatisfactory diet plan.
Actors	Users, Diet Recommendation Model, Database
Trigger	User confirms submission for a healthy and sustainable diet routine.
Normal Flow	ACTIONS 1. User fills in his/her choice of food items and medical conditions (if any). 2. System uses the diet recommendation model to recommend a diet plan. 3. User can make changes in the plan. 4. User finalizes the plan.
Alternative Flow	-
Extensions	BRANCHING ACTIONS 1. User does not like the diet plan. 1.1 User discards the diet plan.
Priority	High
Frequency of Use	High

Usecase 4

USE CASE 5	**Workout Recommendation**
Description	User fills in his/her choices of physical activities and gets a customized workout plan.
Used by	Users
Pre-Conditions	We know user personal details and physical activity choices.
Successful End Condition	User gets a satisfactory workout plan.
Failed End Condition	User gets a unsatisfactory workout plan.
Actors	Users, Workout Recommendation Model, Database
Trigger	User confirms submission for an effective workout routine.
Normal Flow	ACTIONS 1. User fills in his/her choice of physical activities and time span. 2. System uses the workout recommendation model to recommend a workout plan. 3. User can make changes in the plan. 4. User finalizes the plan.
Alternative Flow	-
Extensions	BRANCHING ACTIONS 1. User does not like the workout plan. 1.1 User discards the workout plan.
Priority	High
Frequency of Use	High

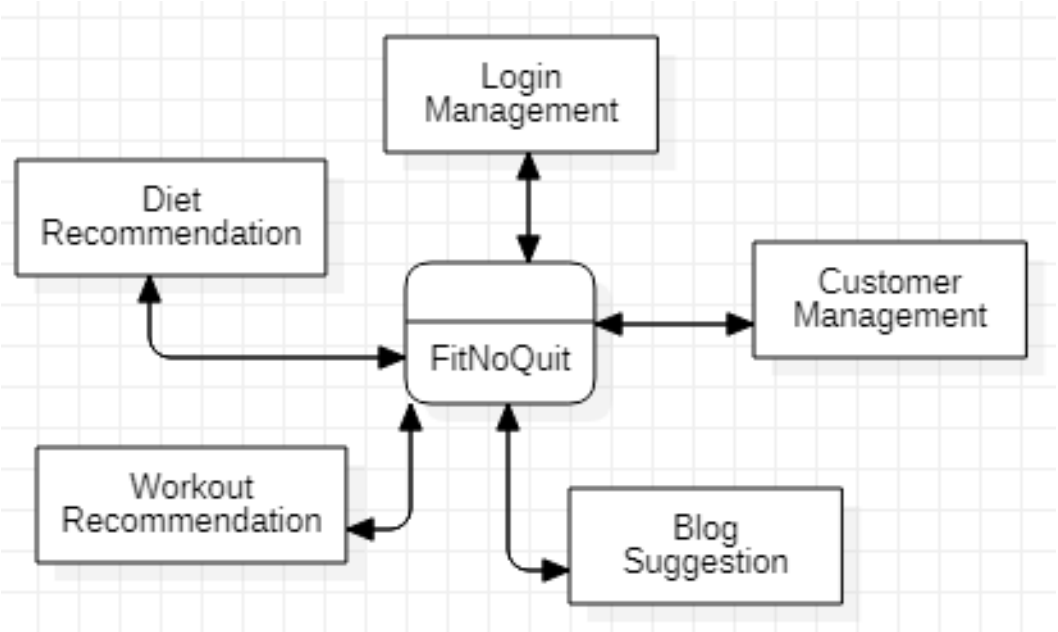
Usecase 5

USE CASE 6	Blog Community
Description	User interacts with other users, upload posts and gets recommended other users' posts.
Used by	Users
Pre-Conditions	We know user personal and post details.
Successful End Condition	User is successfully able to interact in the community and post in it.
Failed End Condition	User is not able to interact in the community and post in it.
Actors	User, other users, database
Trigger	User wants to browse through blogs and read about trending topics.
Normal Flow	ACTIONS 1. User enters the blog page. 2. User selects the category he/she is interested in 3. User enters the home page where blogs are displayed. 4. User gets recommended with similar posts.
Alternative Flow	-
Extensions	BRANCHING ACTIONS 1. User likes multiple blog categories based on interest 1.1 User does not get blogs suggested based on interest
Priority	High
Frequency of Use	High

Usecase 6

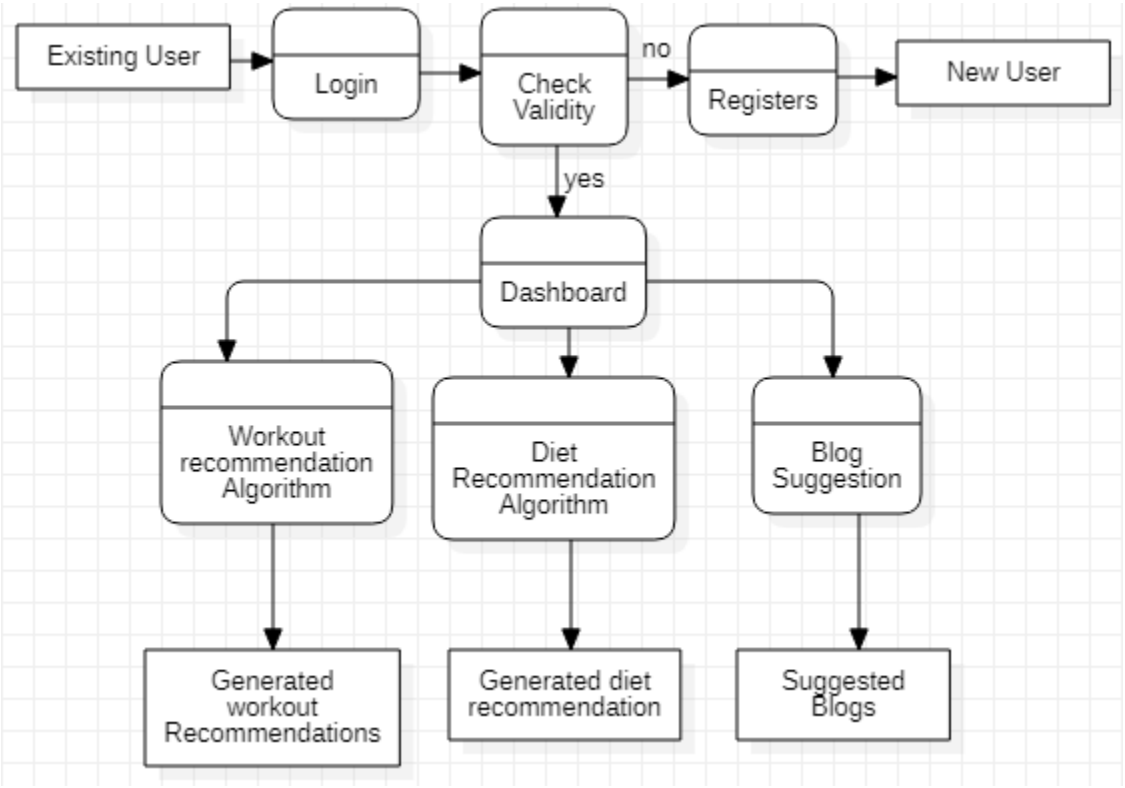
6 Data flow Specifications

6.1 Context Flow Diagram



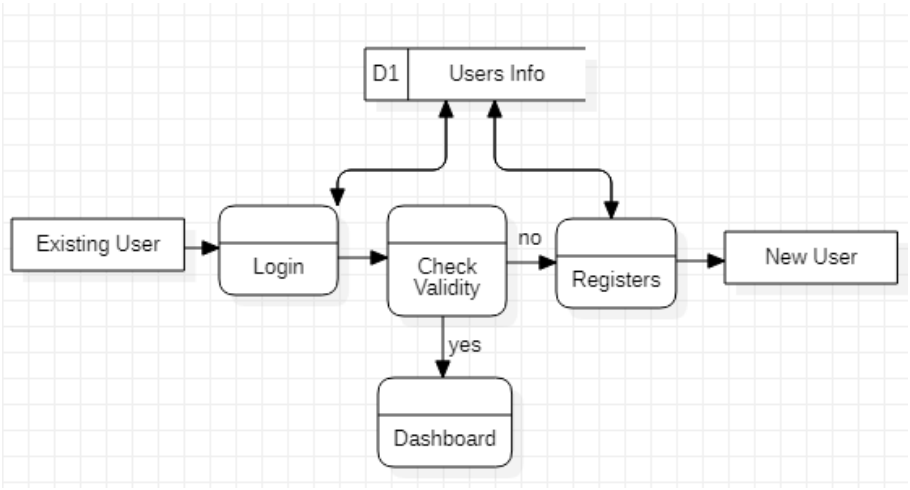
Context Flow Diagram for FitNoQuit

6.2 Data Flow Diagram Level 0

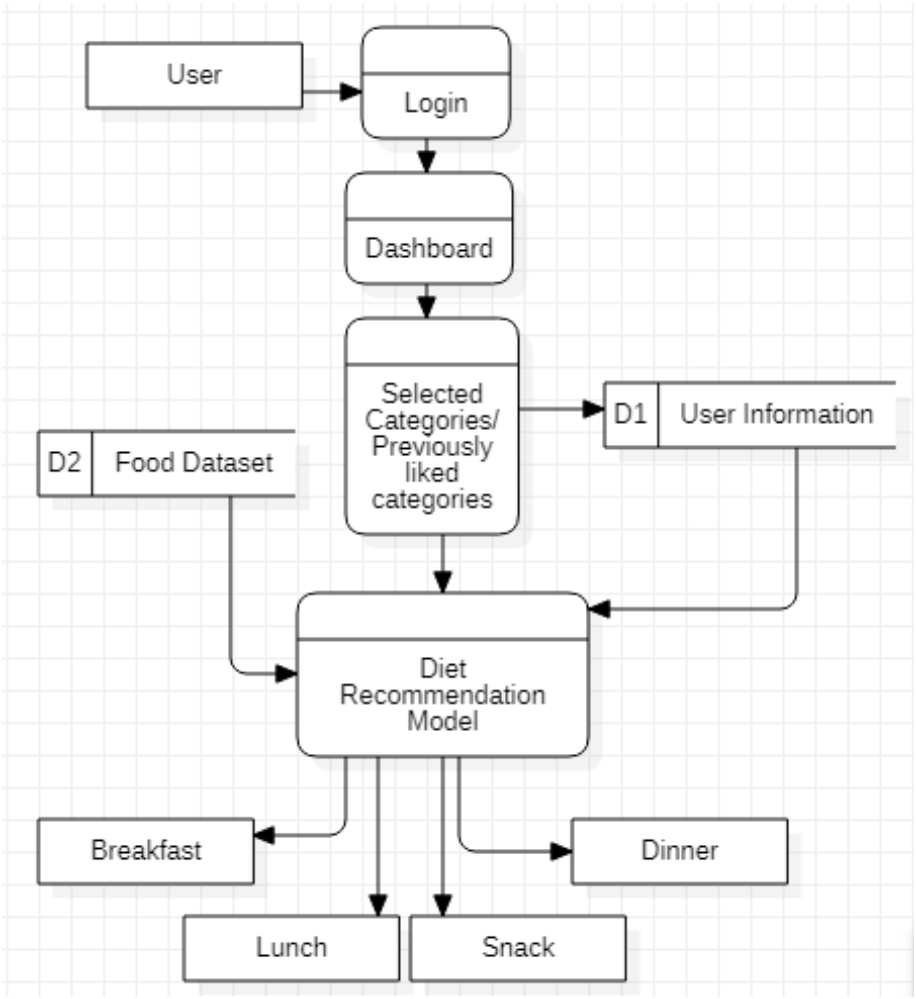


Level-0 diagram for FitNoQuit

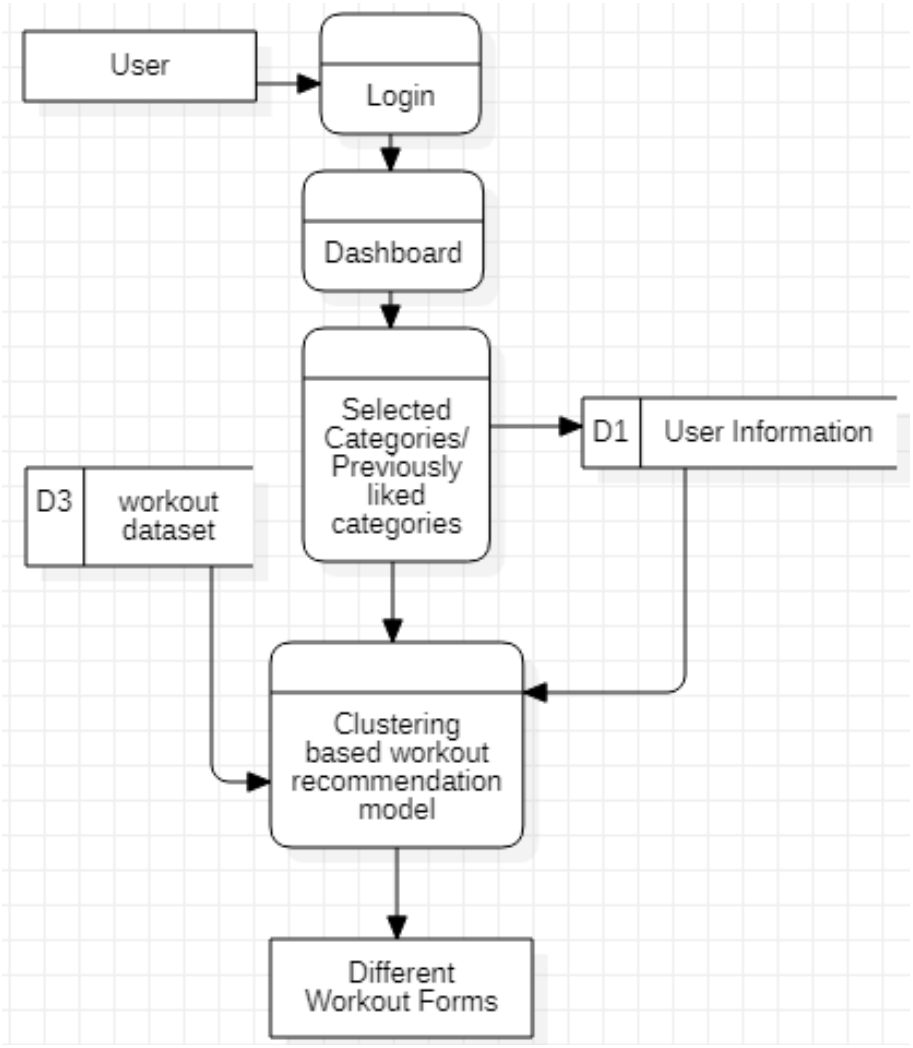
6.3 Data Flow Diagram Level 1



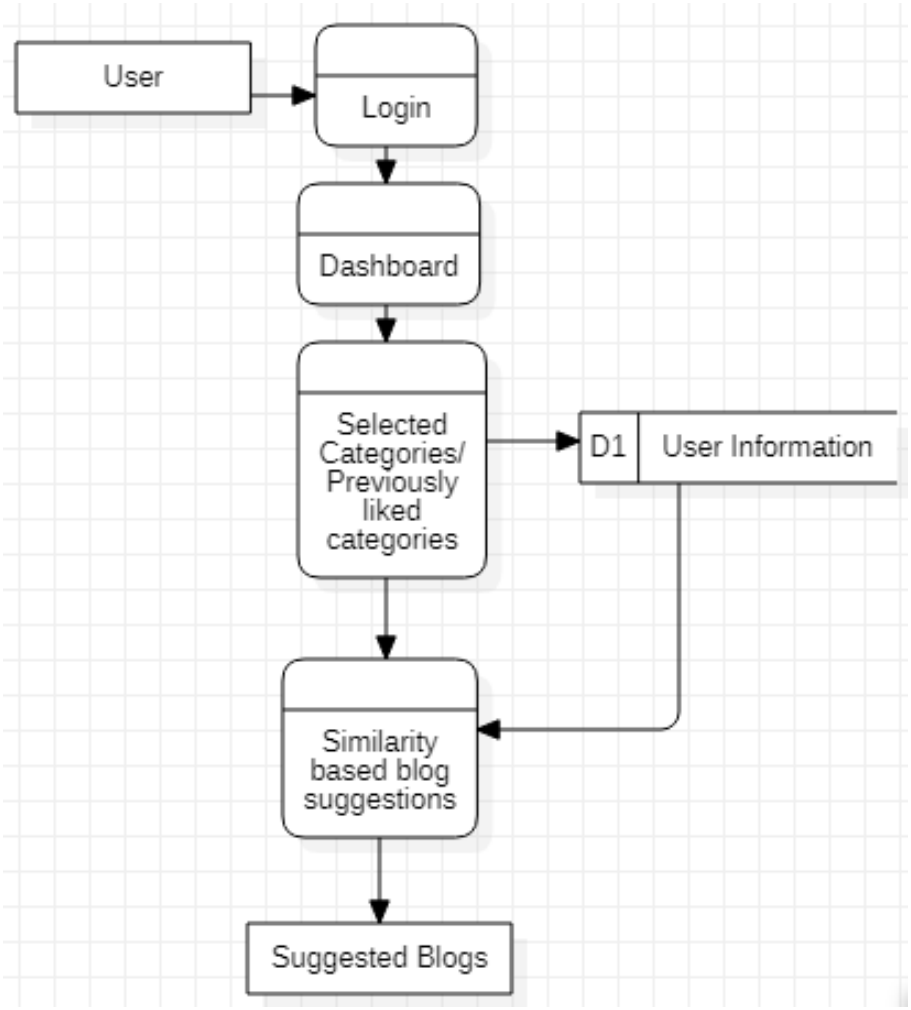
Login Process (Level-1 diagram for FitNoQuit)



Diet recommendation (Level-1 diagram for FitNoQuit)



Workout Recommendation (Level-1 diagram for FitNoQuit)



Blog Suggestion (Level-1 diagram for FitNoQuit)

FitNoQuit - Software Test Document

BHAVIK BHATT (1814007) PIYUSH CHAVDA (1814010)
BHARAT CHOITHANI (1814011) MUSKAAN NANDU (1814020)

October 2021

1 Introduction

1.1 System Overview

Along with development, testing is also a major development cycle which is very important for to achieve the fruition of the Project. Testing helps us to identify the loop holes in the development and avoid future crisis. Testing also helps us to identify the potential failure of system and unwanted access to data. Exhaustive testing is conducted to ensure the system works with accuracy and reliability. This is done to make sure all the bugs are detected before the system is available to end user.

1.2 Test Approach

1.2.1 UNIT TESTING

Unit testing is a type of software testing where individual units or components of a software are tested. The purpose is to validate that each unit of the software code performs as expected. Unit Testing is done during the development (coding phase) of an application by the developers. Unit Tests isolate a section of code and verify its correctness. A unit may be an individual function, method, procedure, module, or object. There are many benefits for this unit testing:

- The unit testing facilitates change in the code.
- It allows testing to be done in a bottom up fashion.

At the same time, unit testing has some disadvantages such as, it might not identify each and every error in the system.

1.2.2 ACCEPTANCE TESTING

User will be involved in this phase of testing to analyse the acceptability and usability of the system. We will use black box testing to ensure the system is suitable to all user with/without technical understanding. This also helps to identify bugs that might have been missed by the previous tests.

1.2.3 SYSTEM TESTING

System consists of all the components that makeup the system to function. This testing is done to ensure that all the components work together with each other and the system as a whole. It should work as expected.

2 Test Plan

The main scope of the test plan for the Airline Reservation System are as follows:

- To identify the features of the system that will be tested.
- To identify and define all the activities necessary to prepare for and conduct the testing process on the Recommendation System
- To define the pass/fail criteria for each item that will be tested
- To identify the deliverables of the testing phase.
- To define any suspension criteria and resumption techniques
- To discuss the testing techniques being used to test the Recommendation System.

2.1 Features to be tested

This section of the test plan lists all the items of the Airline Reservation System project that will be tested:

1. Sign up - This feature is for users who are new to the website and have never used it before. They will have to fill in their personal details and register in the system.
2. Login - This feature is for users who want to use the functions of the website. They need enter their credentials in the website so they can use their account which contains all the information about them and their diet and workout plans.
3. Profile - Every user has a different profile that is filled up with his/her details, history, plans, routines etc. Each profile is unique for every individual and should be accurately working in this web app.
4. Diet Recommendation - This feature recommends a personalized and customized diet plan for an individual and is one of the most important features of this web app.
5. Workout Recommendation - This feature recommends a personalized and customized workout plan for an individual and is one of the most important features of this web app.
6. Blog Community - This a community where users can read trending topics in food and workout industry. They can like blogs written by health experts and similar blogs will be suggested to user in the future as well.

2.2 Features not to be tested

This section of the test plan lists all the items of the Recommendation System that will be NOT be tested:

1. Database Connectivity - Database is one of the most important parts of a

web application. Thus, it is mandatory to check if this is working properly or not. We need to check if all the inputs from the users are properly stored in the database or not. We also need to check if the products from the database are displayed to the users or not.
2. Payment Functionality - This features allows the user to select among the different payment methods offered to him/her and get a plan according to his/her needs.

2.3 Testing Tools and Environment

The time allotted for testing phase is 30 days. The testing phase will be executed by all 4 members of the team: Bhavik Bhatt, Muskaan Nandu, Piyush Chavda and Bharat Choithani. The testing process will carried out with the help of below listed tools:

- Selenium online tool
- TestingWhiz tool
- Ranorex environment

3 Test Cases

3.1 Sign Up

3.1.1 Purpose

Registers the user in the software's database and allows him/her to use the features provided by the software

3.1.2 Inputs

Incorrect Input:
Wrong format entered in the input fields for the registration page
Correct Input:
The correct input would be a valid e-mail id and valid email format of the user and a valid strong format password.

3.1.3 Expected Outputs Pass/Fail criteria

Incorrect:
An appropriate message should be generated to the user saying that he has entered the wrong format in the specific input field.
Correct:
An appropriate message should be generated to the user saying that he has entered the correct format in the specific input field and redirect to the software dashboard.

3.1.4 Test Procedure

The testing would be carried out on FitNoQuit while registering on the system as a new user to the system.

3.2 Sign In

3.2.1 Purpose

Logs in the user to the software where the user can access the features provided by the software.

3.2.2 Inputs

Incorrect Input:
Username: which is the email-id in the case of the Recommendation System FitNoQuit.
Password: with respect to the valid username.
Correct Input:
The correct input would be a valid e-mail id of the user and a correct password associated with the email-id which he uses to log in.

3.2.3 Expected Outputs Pass/Fail criteria

The user should be directed to the dashboard of the software after he/she logs into the system.
On invalid login attempts, the user should remain on the login page or can go to sign up page.

3.2.4 Test Procedure

The testing would be carried out on the FitNoQuit while logging into the system as a customer or a normal user of the system.

3.3 User Profile

3.3.1 Purpose

Every user has a different body type and requirements. The user profile takes into account all the preferences of the user and provides recommendation based on it.

3.3.2 Inputs

Incorrect Input:
Wrong format entered in the input fields on the user profile page.
Correct Input:

The correct input would be valid height, weight, food preferences, health conditions etc

3.3.3 Expected Outputs Pass/Fail criteria

Incorrect:
An appropriate message should be generated to the user saying that the inputted value is invalid for the given field.
Correct:
An appropriate message should be generated to the user saying that he has entered the correct data in the fields.

3.3.4 Test Procedure

The testing would be carried out on FitNoQuit while entering different combinations of data in the input fields.

3.4 Diet Recommendation

3.4.1 Purpose

Recommends a personalized and customized diet plan for an individual based on his/her preferences and medical conditions.

3.4.2 Inputs

Incorrect Input:
Invalid data entered on user information/profile page Correct Input:
Valid data entered on user information/profile page

3.4.3 Expected Outputs Pass/Fail criteria

Incorrect:
An appropriate message should be generated to the user saying that the entered details are invalid and no diet should be recommended.
Correct:
A personalized diet should be recommended to the user satisfying all his preferences and medical conditions.

3.4.4 Test Procedure

The testing would be carried out on FitNoQuit by entering a combination of health conditions and food preferences.

3.5 Workout Recommendation

3.5.1 Purpose

Recommends a personalized and customized workout plan for an individual based on his end goal and considering user's physical disability if any.

3.5.2 Inputs

Incorrect Input:
Invalid/Improbable end goal or extreme physical disability. Correct Input:
Valid end goal and inputs entered by the user

3.5.3 Expected Outputs Pass/Fail criteria

Incorrect:
An appropriate message should be generated to the user saying that the inputted data is invalid.
Correct:
Different options of workouts need to be displayed to the user from which the user can follow a workout that best suits him/her.

3.5.4 Test Procedure

The testing would be carried out on FitNoQuit entering a combination of inputs and validating the workout suggested based on it.

3.6 Blog Community

3.6.1 Purpose

Users can read trending topics in food and workout industry. They can like blogs written by health experts and similar blogs will be suggested to user in the future as well.

3.6.2 Inputs

Incorrect Input:
None Correct Input:
Liking of a blog.

3.6.3 Expected Outputs Pass/Fail criteria

Incorrect:
None.
Correct:

Blogs on trending topics and topics previously liked by the user should be suggested.

3.6.4 Test Procedure

The testing would be carried out on FitNoQuit by viewing different blogs.

4 Test Results

Test Number	Test Case	Test Description	Expected Outcome	Actual Outcome	Test Result
FNQ-T01	Login Button	An email and password are entered by the user	The system logs in the user and redirects him/her to the home page		
FNQ-T02	Register Button	User enters his/her details	The system logs in the user and redirects him/her to the home page		
FNQ-T03	Update profile	User changes his/her details	The system updates user details successfully and displays it.		
FNQ-T04	Diet Recommendation	User inputs his/her choice of foods and medical conditions if any.	The system recommends a personalized and satisfactory diet plan		
FNQ-T05	Workout Recommendation	User inputs his/her choice of physical activities and his/her goals.	The system recommends a personalized and satisfactory workout plan		
FNQ-T06	Blog Community	User likes a blog or views blogs that are on trending topics	User is shown blogs similar to the ones he/she previously liked along with the new trending blogs.		

4. CONCLUSIONS AND LEARNINGS

4.1 Conclusions

- Hence all in all, we plan to develop and implement a web application that is made for all kinds of users and helps them achieve their fitness goals by providing a personalized, sustainable and flexible workout and diet plans.
- We read different technical papers to understand how machine learning can be used for recommendation systems and how it can be applied in our project.
- We aim to incorporate features that lack in the existing systems by providing personalized diet and workout options that is completely customizable and satisfy the user's needs.
- For the workout recommendation system, we plan to use mathematical computations and/or machine learning algorithms like K-means clustering algorithm on the exercise dataset available on Kaggle, to find 3 different exercise options to burn certain number of calories.
- We plan to create a diet recommendation system that recommends users with meals for an entire day divided in 4 categories: breakfast, lunch, snacks and dinner.
- We also plan to provide users with a blog section where users are suggested with blogs relate to the categories they show interest in.
- Hence, FitNoQuit is a one stop destination to track your health. It is an all-in-one solution that provides customized diet plans and suitable workout options to enable users to achieve their end goal.

References

[1] F. Rehman, O. Khalid, N. u. Haq, A. u. R. Khan, K. Bilal and S. A. Madani, "Diet-Right: A Smart Food Recommendation System," KSII Transactions on Internet and Information Systems, vol. 11, no. 6, pp. 2910-2925, 2017. DOI: 10.3837/tiis.2017.06.006.

[2] Kardam, Shubham Singh, et al. "Website on Diet Recommendation Using Machine Learning." In International Research Journal of Engineering and Technology (IRJET) Volume: 08 Issue: 04, April 2021.

[3] Ojokoh, B., and A. Babalola. "A Personalized Healthy Diet Recommender System." Organization for Women in Science for the Developing World (OWSD) (2016): 388-393.

[4] R. Yera Toledo, A. A. Alzahrani and L. Martínez, "A Food Recommender System Considering Nutritional Information and User Preferences," in IEEE Access, vol. 7, pp. 96695-96711, 2019, doi: 10.1109/ACCESS.2019.2929413.

[5] D. Mogaveera, V. Mathur and S. Waghela, "e-Health Monitoring System with Diet and Fitness Recommendation using Machine Learning," 2021 6th International Conference on Inventive Computation Technologies (ICICT), 2021, pp. 694-700, doi: 10.1109/ICICT50816.2021.9358605.

Acknowledgements

In this tough time, when the world is suffering from a worldwide pandemic, we would like to thank the almighty to keep us occupied with projects, like this one and help us learn and grow without many distractions. We are sure that this project will help us to explore the field of recommendation systems using machine learning along with different aspects of software development like software documentations, front-end development, dataset management, etc.

After the supreme power, we would like to thank our parents, for giving us this opportunity of learning. Next, we would like to thank each other for immense support throughout the process of the project and helping each other grow.

We would like to give our special thanks to our project guide Prof. Sonali Patil, for guiding us with this project and giving us an opportunity to learn, communicate and explore throughout the project. We would also like to thank Dt. Dhwani Parekh, a qualified nutritionist to assist our team by suggesting changes in the dataset and validate the recommendations made by our model "FitNoQuit".

Lastly, we would like to thank the project committee of our college, K. J. Somaiya College of Engineering for providing us a chance to work on this project.