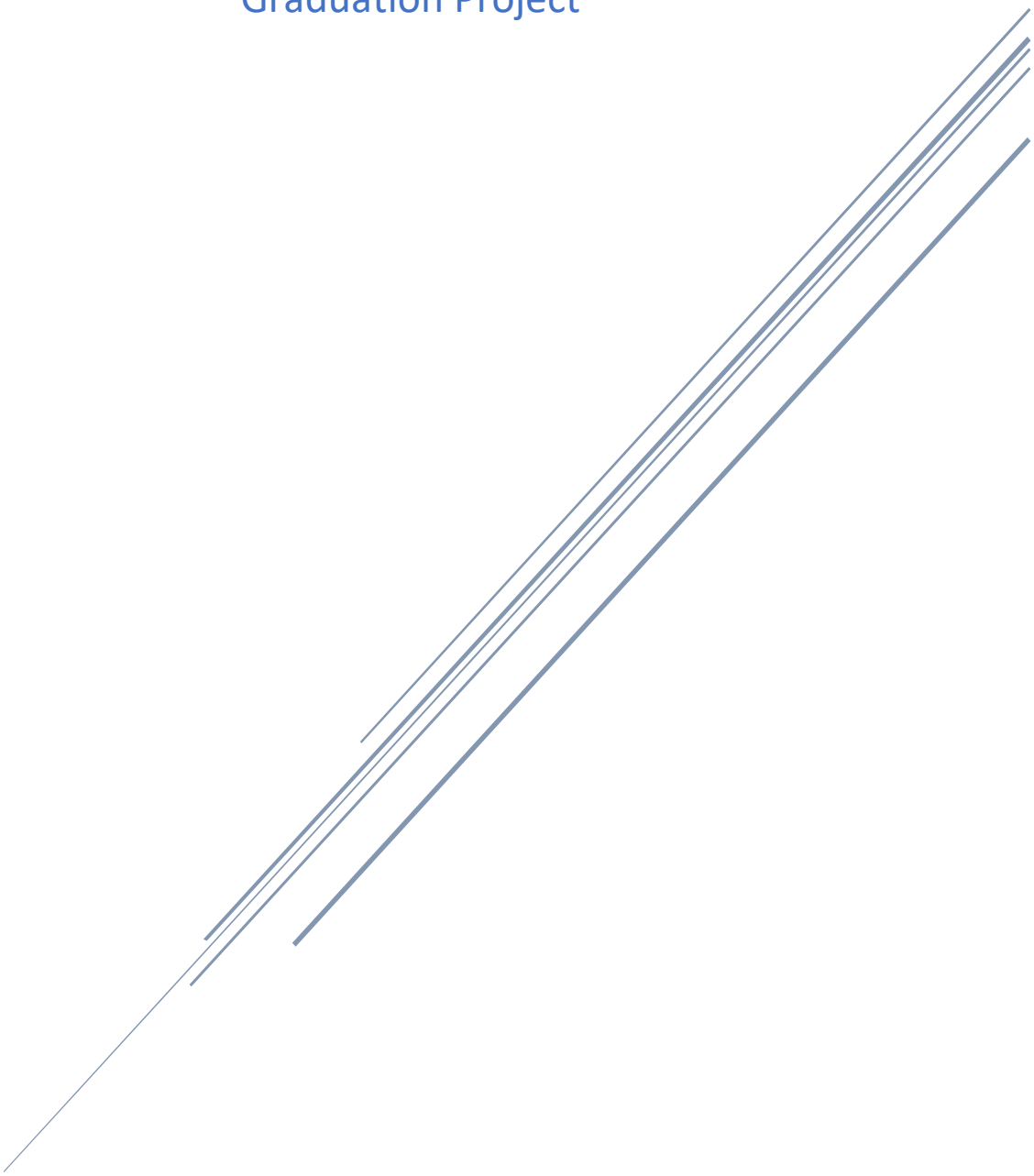
SMARTCAL: INNOVATING MOBILE TECHNOLOGY FOR REAL-TIME CALORIE TRACKING AND DIETARY GUIDANCE

Graduation Project



Project Management Plan Document

Table of Contents

List of Figures

List of Tables

1. Project Overview

Recently, with the aim of living a healthy life in a global sense, healthy eating and regular sports habits have started to be adopted in our country. In this context, it is of great importance for users to eat meals and do sports for their own purposes. Considering that almost everyone now has a smartphone, a mobile application that can be used for this purpose, which is accessible at any time, will make things easier for people. This application, which was developed with this in mind, set out with the aim of meeting the increasing needs of users.

1.1. Project Summary

Within the scope of Calorie Tracker application, it is aimed that all users who want to gain, lose or maintain weight can do this in a healthy way and this mobile application has been developed.

1.1.1. Purpose, scope and objectives

The mobile application, which will be developed within the scope of this project, can be accessed from mobile devices using Android. In this way, we aimed to reach the maximum number of users that can be reached by developing an application that can run on the most widely used mobile operating system in the world.

The dates we choose for ourselves as due dates;

- Release-1 System Demo - May 4, 2021
- Release 1.1 System Demo - May 25, 2021
- Release 1.2 System Demo - June 8, 2021
- Pre-Final System Demo Release 1.3 - June 15, 2021
- Project Representations and Final System Demo - June 29, 2021
- Project Final with All Project Deliverables: July 8, 2021

It will include the following features on an application basis.

- The user will enter their personal characteristics such as gender, age, height, weight, body fat.
- The user will choose the suitable option for himself/herself from the ones of losing weight, gaining weight or maintaining his/her weight. Also, if there is anything different from these options, s/he can add anything with the ‘other’ option.
- The user will enter the food items to the application for calorie calculation.
- The user will also enter sport activities to the application for calorie tracking.

- Then, coach will reach the information of user and prepare the list that includes meals and exercises according to his/her information and purpose.
- This list will be prepared according to calorie calculation but this is presented to user, s/he should use this list by consulting an expert. Because coach will prepare a list regardless of health problems and special conditions.
- The user enters what s/he eats and the sports s/he does and see how many calories s/he needs to burn and take daily according to the calories that the coach says to take and the sport that is supposed to do.

1.1.2. Assumptions and constraints

Project life cycle model that will be implemented in the project process has been determined as Incremental Model because, the we determine the project scope at the beginning and cost and estimates would be modified according to the progress at the future process. Also, there can be elements which will be added with the aim of gain functionality to the application.

In the release of the project, target market is people who try to track calorie consumption by giving some information about their lives. Calorie Tracker application will be developed for all the mobile devices using Android. We will provide our users with software as a service application. Our applications frontend will run on mobile platforms, specifically Android. While on the backend we will store and process the relevant information on the platforms provided by cloud services.

1.1.3. Project deliverables

Main product of this project is the mobile application and the other documents that will be prepared in this process can be seen from the Table 1 below.

Name of Deliverable	Responsible Person
Software Project Management Plan (PMP)	Project Manager
Software Requirement Specifications (SRS)	Technical Manager
Software Detailed Design Document (SDD)	Design Engineer
Software Test Plans (TP)	Design Test and Validation Resp.
Final Report	All Project Members

Table 1 Deliverables and Responsibilities

1.1.4. Schedule and budget summary

Project Milestones	Target Date	Completion %
Publish Project Management Plan (PMP)	Week 3	%15
Requirement Elicitation	Week 4	%25
Publish Requirement Specifications (SRS)	Week 5	%30
Preliminary Design Phase	Week 8	%40
Release-1 System Demo	Week 8	%50
Publish Detailed Design Document (SDD)	Week 11	%60
Detailed Design Stage	Week 12	%70
Pre-final (release-2) system demo	Week 13	%80
Publish Test Plans (TP)	Week 14	%85
Project representation and final system demo	Week 16	%95
Submission of final reports	Week 17	%100

Table 2 Project Schedule

1.2. Evolution of the Plan

All pre-planned or spontaneous changes to be made in the process of calorie tracking that we plan to develop are monitored in accordance with the desired document format and through the control system. All changes to be made will be discussed within the group and all group members will be involved in the decision-making process. If there is an indecision situation, the Project Manager will make the final decision. Minor changes that there is no need to make a common decision beforehand, can be made by the members of the group by taking responsibility. The person who made this change should then announce this to the whole group and show the change he has made.

In addition, in the group, where instant communication is provided with the WhatsApp group, it will be ensured that the group members cannot feel themselves outside of the process with weekly (sometimes two or three times a week) meetings. At the same time, all members will be able to access documents and project files at any time via the created google drive share folder.

2. References

- ISO/IEC/IEEE 16326 (2009) - Systems and Software Engineering – Lifecycle Processes – Project Management

3. Definitions

Application: Mobile application that can be worked at the android devices.

Client: The user that can add their personal info and choose his/her purpose with the aim of taking meal and exercise list from the coach.

Goals: As a goal, the user's target weight value, the body type they want to reach, in many days long they want to reach this body type, and extra client comments such as allergies or eating habits are also received.

Coach: A user has some privileges such as seeing the profile of the users who set a goal from the application.

Mobile Device: This consists of mobile devices running Android mobile system.

Sport Activities: They are the exercises that has calorie information based on the activity and the number or distance.

Meal List: The list consists of the foods chosen by their calorie values.

Exercise List: The list consists of the exercises chosen by their calorie values.

4. Project Context

The context of the project will be explained under six main categories which are process model, process improvement plan, infrastructure plan, methods, tools and techniques, product acceptance plan and project organization.

4.1. Process Model

During the project development phase Incremental process model will be implemented as the Life Cycle model. Project will be split into back-end and front-end modules which will be developed in parallel. There are four milestones and these are;

Milestone-1: In five weeks into the project first main milestone will be reached. By this point, requirements gathering and analysis and specification steps will be completed. Design phase of the project will be semi-completed. Relevant documentation will be prepared and they will be reviewed by the project stakeholders.

Milestone-2: By eight weeks into the project, utilizing feedback received from previous step, design phase will be completely detailed. Basic structures of the back-end and front-end

modules will be implemented and integrated. First release will be presented to project stakeholders.

Milestone-3: By thirteen weeks into the project, feedback from the first release will be evaluated and changes will be implemented where needed. Design phase documentation will be completed. In this phase, incremental implementations of the project will be released in quick successions and implementation and integration phase will be completed.

Milestone-4: By sixteenth week, test plans and functional tests will be completed. Project presentations and system demos will be presented. All the documentation of the project will be completed.

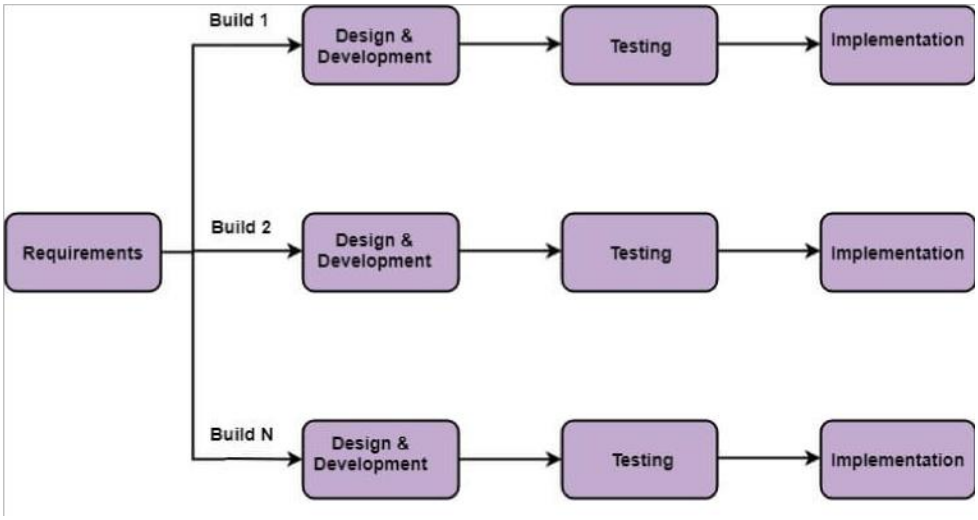


Figure 1 Incremental Model of Project

4.2. Process Improvement Plan

Since the Incremental model is determined as the Life Cycle model of the project, additions and arrangements will be made in the process to make the project (application) better. For these arrangements, we have always decided to create an agenda on this issue in our regular meetings. At the end of each meeting, what kind of additions or arrangements can be made to make the application more functional or more user-friendly, device-friendly will be discussed. At the same time, all kinds of improvements can be made to the project, as well as taking time constraints into consideration, major changes that may disrupt the project flow will be avoided. The impact of any changes planned to be made in a major or minor will be evaluated in detail by considering all factors and will be instantly evaluated after all changes made.

4.3. Infrastructure Plan

Hardware-OS-Network: System will utilize Amazon Web Services Application Server and DynamoDB platforms. These platforms run on Amazon Web Services cloud data centers and utilize their infrastructure.

Facilities: Project will be developed remotely; each member will work from his own home.

Workstation: Personal computers of individual team members will be utilized as workstations.

4.4. Methods, Tools and Techniques

Because we decided to make an Android application, we are going to use Visual Studio as a Development Environment. Code will be written by using 2 language:

- C# will be used for Backend,
- XAML will be used for Graphical User Interface.

To store data, because NoSQL database is offered, Project's data will be stored on AWS DynamoDB. To communicate with AWS DynamoDB, AWS's Cognito Identity service was chosen. This service will be downloaded as a .NET package to our project.

Google Drive and GitHub platforms are decided to be used as a project management tools. The product will be created for mobile devices which have Android Operating System.

4.5. Product Acceptance Plan

Project stakeholders expect the milestones defined in the process model to be met. At the end of the project there should be a working release that should be presented. In addition, proper documentation of Project Management Plan, Software Requirements Specification, Software Design Document and Test Plans should also be presented. Project should also fulfill requirements specified by the stakeholders at the beginning of the project such as utilizing cloud platforms and implementing a NoSQL database in a meaningful manner.

4.6. Project Organization

In this project, there are four different roles and project members are responsible for different or common tasks.

The first role is the role of project manager, who is responsible for planning, organizing, and directing the completion of specific tasks. Besides, he/she leads the project life cycle and

must be competent in managing the six aspects of this project, i.e., scope, schedule, finance, risk, quality, and resources.

The second role is the role of technical manager and he/she is responsible for providing project planning and management for established initiatives within the project. He/she ensures that the project is completed to specification, within an established time frame and budget.

The third role is the role of design engineer and he/she is responsible for identifying software problems and designing programs to find solutions by designing, modifying, developing, and implementing software programming applications.

The last role is the role of test and validation engineer and he/she is responsible for preparing validation and acceptance criteria and evaluate test results to assure that software attains requirements and parameters.

4.6.1. External Interfaces

The project manager is responsible for analyzing and defining new requirements based on users' additional requests. Therefore, the project manager will have the authority to make decisions that impact the project from a technical and functional perspective and will have the authority to allocate resources to the development team when required.

4.6.2. Internal Interfaces

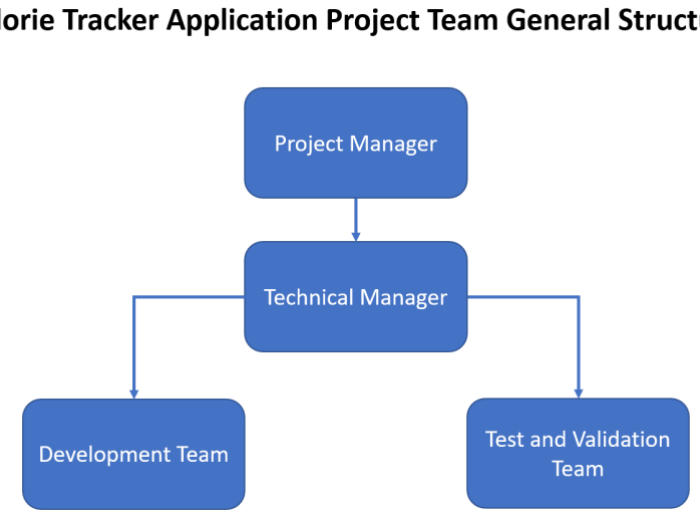


Figure 2 Project Team Structure

4.6.3. Authorities and Responsibilities

These are explained under four main categories.

4.6.3.1. Project Manager

- Leads the technical management and implementation team,
- Plans what work needs to be done, when and by whom it is going to be done,
- Looks at the risks involved in a particular project and managing these risks,
- Coordinates the work done by different people,
- Makes sure the project is running on time and is within the budget,
- Deals with changes to the project as and when necessary,
- Motivates the team of people involved in the project,
- Makes sure the project delivers the expected outcomes and benefits.

4.6.3.2. Technical Manager

- Builds the development, test, and validation team,
- Leads the development, test, and validation team,
- Coordinates the development and testing,
- Analyzes, plans, and develops requirements and standards in reference to scheduled projects,
- Establishes and implements training processes and strategies for all technical personnel,
- Assigns and oversees the daily tasks of technical personnel while ensuring all subordinates are actively working toward established milestones,
- Holds regular technical team meetings to determine progress and address any questions or challenges regarding project,
- Determines and defines clear deliverables, roles, and responsibilities for staff members,
- Makes sure the work is done to the right standard,
- Notifies the project manager of variances from plan and submits recommendations for corrective action.

4.6.3.3. Development Team

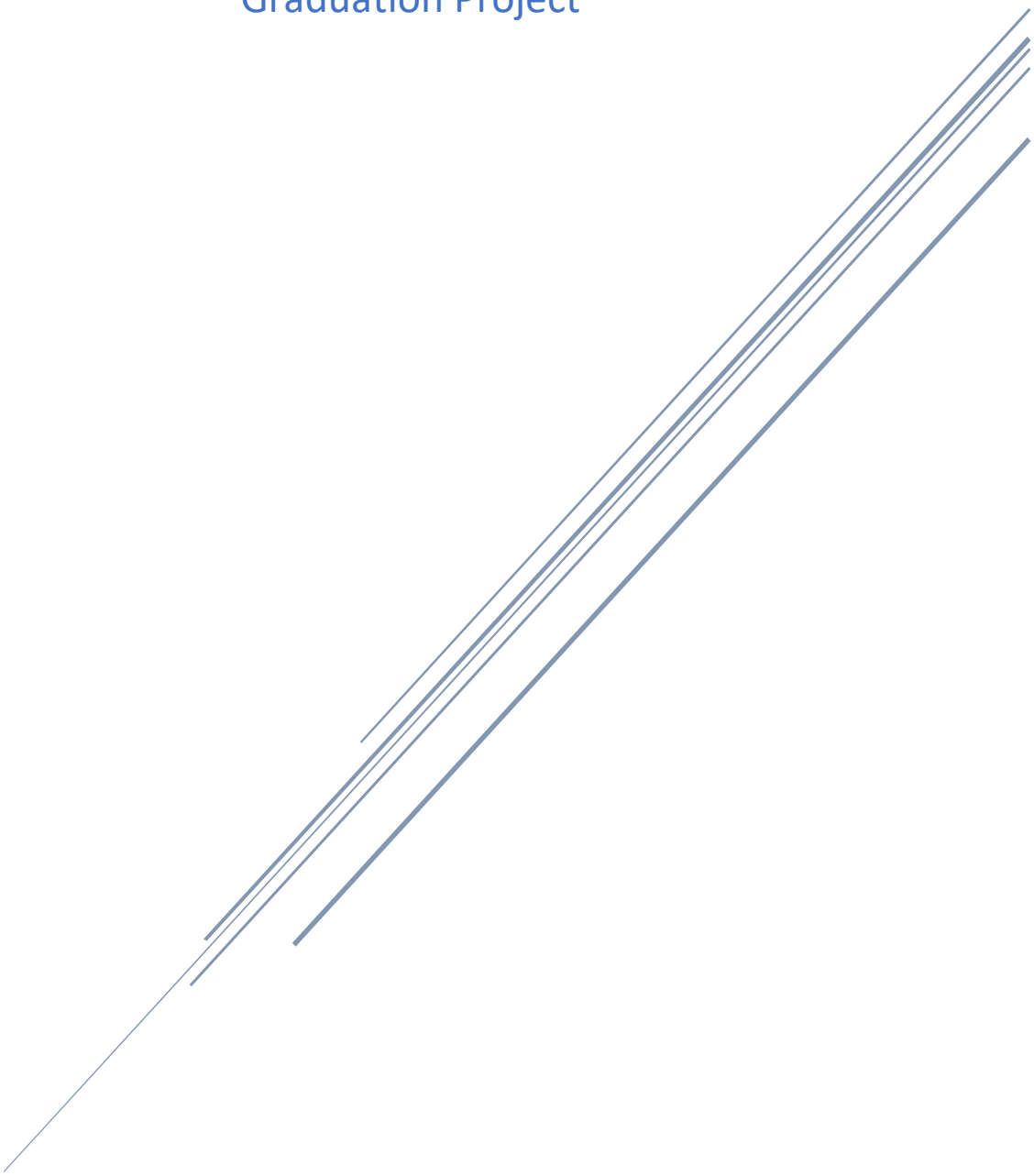
- Made up of a technical manager and programmers,
- Responsible for the functional design and system construction phases of the system functions assigned to the design,
- Documents the system or application in detail,
- Maintains the system or application with updates and fixes as necessary,
- Recommends software upgrades for application.

4.6.3.4. Test and Validation Team

- Responsible for test and validation activities throughout product development cycles,
- Establishes testing requirements for application, based on software specifications,
- Composes reports and adjusts processes that require improvement,
- Establishes performance and quality criteria.

SMARTCAL: INNOVATING MOBILE TECHNOLOGY FOR REAL-TIME CALORIE TRACKING AND DIETARY GUIDANCE

Graduation Project



Software Requirements Specification Document

Table of Content

List of Figures

List of Tables

1. Overview

Recently, with the aim of living a healthy life in a global sense, healthy eating and regular sports habits have started to be adopted in our country. In this context, it is of great importance for users to eat meals and do sports for their own purposes. Considering that almost everyone now has a smartphone, a mobile application that can be used for this purpose, which is accessible at any time, will make things easier for people. This application, which was developed with this in mind, set out with the aim of meeting the increasing needs of users.

You can see the details of the application from the context diagram which can be seen at the figure 1.

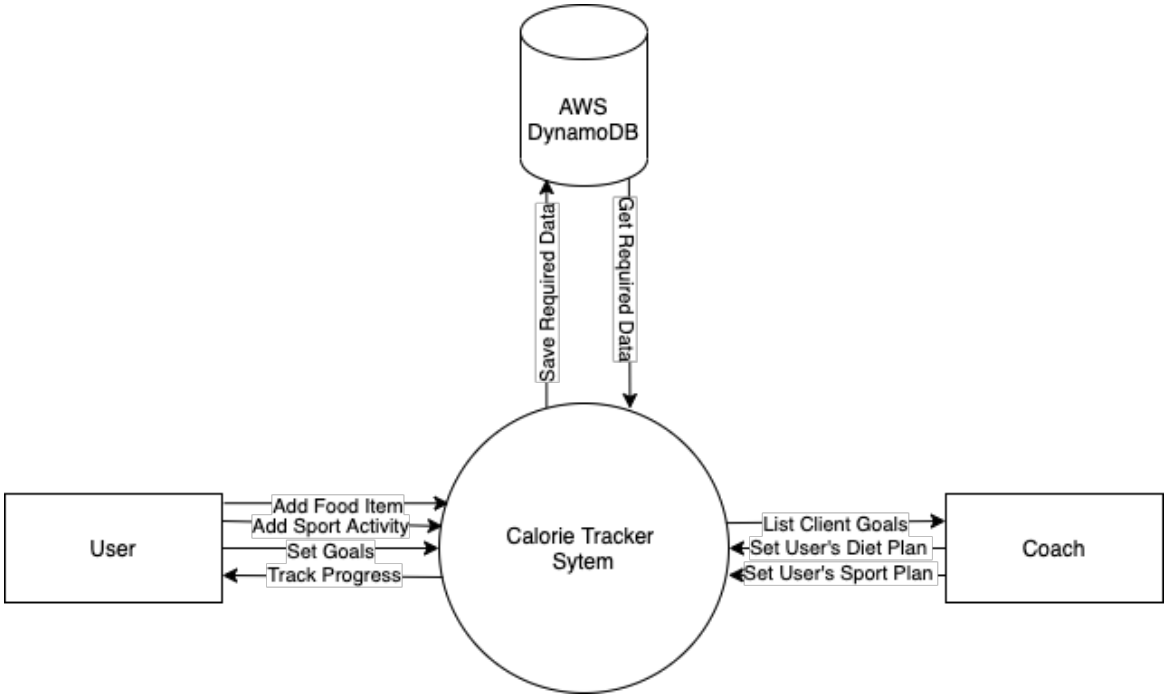


Figure 1: Context diagram

2. Purpose

Within the scope of Calorie Tracker application, it is aimed that all users who want to gain, lose or maintain weight can do this in a healthy way and this mobile application has been developed.

3. Scope

The mobile application, which will be developed within the scope of this project, can be accessed from mobile devices using Android. In this way, we aimed to reach the maximum

number of users that can be reached by developing an application that can run on the most widely used mobile operating system in the world.

Users will enter their personal information through the application and determine their priority targets in line with their purposes. As a target, the user's target weight value, the body type they want to reach, in many days long they want to reach this body type, and extra client comments such as allergies or eating habits are also received.

4. Product Perspective

4.1. System Interfaces

Since our application will run independently on mobile devices with android systems, it will not interact with any internal / external system, so there is no system interface.

4.2. User Interfaces

At the user interface diagram, use case diagram is the main interface for our application as you can see from Figure 1.

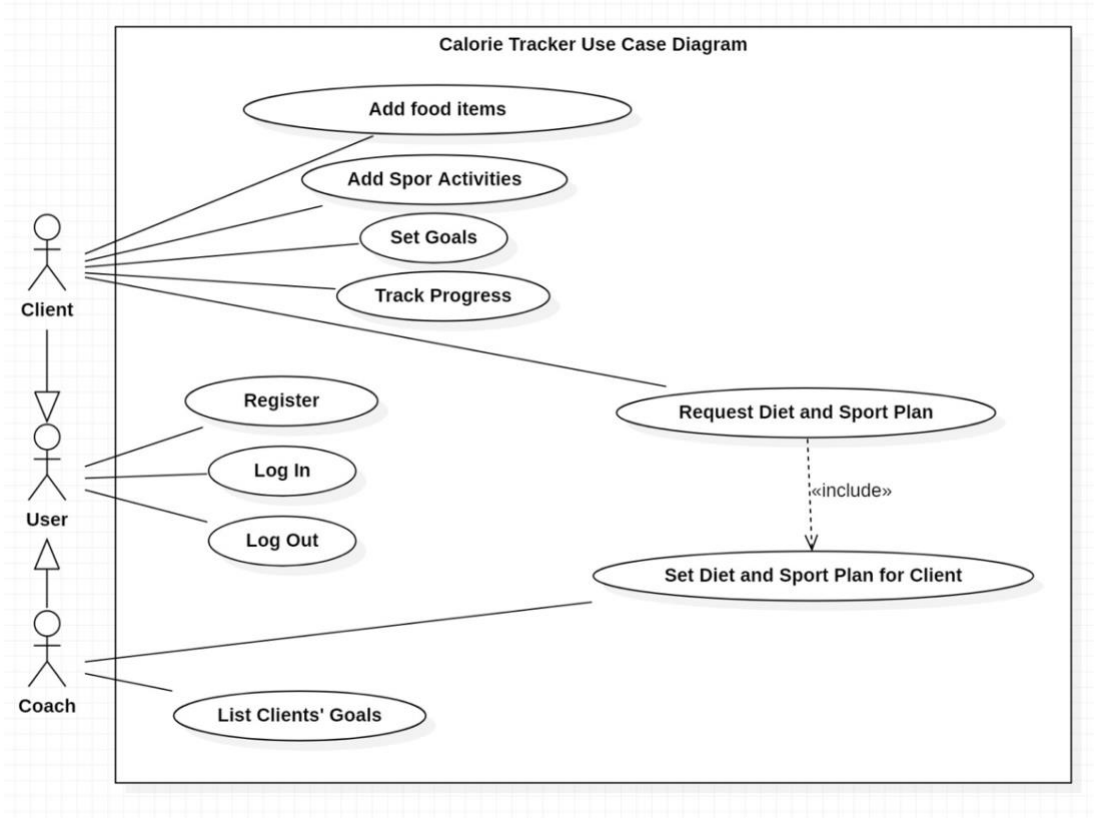


Figure 2: Use Case Diagram for Calorie Tracker Application

There are descriptions for all the interaction and you can see those from the ten tables below.

Use Case ID	1
Use Case Name	Register
Actors	Client, Coach
Descriptions	Every client and coach have to register to the system to use the application.
Pre-conditions	1.The user or coach should not have an account in the system.
Post-conditions	1.There is not any post-condition for this use case.
Normal Flow	1.User enters his/her username. 2.User enters his/her password. 3.User enters profile type. (Client or Coach) 4.Client enters his/her personal characteristics such as gender, age, height, weight and body fat. Coach enters his/her CV and experience.
Alternative Flows	None.
Exceptions	None.

Table 1: Use case description for Register

Use Case ID	2
Use Case Name	Login
Actors	Client, Coach
Descriptions	Every user has to login to the system to use the application.
Pre-conditions	1.The user or coach has to have an account in the system.
Post-conditions	1.Profile page for the user will be shown.
Normal Flow	1.User enters the username. 2.User enters the password. 3.User clicks login button. 4.User profile page will be shown to the user.
Alternative Flows	1.Check password, if it is wrong. 2.User enters username and password again.
Exceptions	None.

Table 2: Use case description for Login

Use Case ID	3
Use Case Name	Logout
Actors	Client, Coach
Descriptions	Every user has to log out of the system when they finish their work.
Pre-conditions	1.The user or coach has to be online in the system.
Post-conditions	1.There is not any post-condition for this use case.
Normal Flow	1. User presses logout button when user is online.
Alternative Flows	None.
Exceptions	None.

Table 3: Use case description for Logout

Use Case ID	4
Use Case Name	Add Food Items
Actors	Client
Descriptions	Client can add food items to the system for calorie intake calculation.
Pre-conditions	The client has been logged into the system.
Post-conditions	None.
Normal Flow	1.Client clicks “add food items” button to add his/her consumptions. 2.Client adds his/her food items by specifying food type and amount.
Alternative Flows	None
Exceptions	None

Table 4: Use case description for Add Food Items

Use Case ID	5
Use Case Name	Add Sport Activities
Actors	Client
Descriptions	Client can add his/her sport activities to the system for calorie tracking.
Pre-conditions	The client has been logged in to the system.
Post-conditions	None.
Normal Flow	1.Client clicks “add sport activities” button to add his/her sport activities. 2.Client adds his/her sport activities to the system.
Alternative Flows	None.
Exceptions	None.

Table 5: Use case description for Add Sport Activities

Use Case ID	6
Use Case Name	Set Goals
Actors	Client
Descriptions	Client can set goal for gaining, losing or maintaining his/her weight.
Pre-conditions	The client has been logged in to the system.
Post-conditions	The client will see his/her updated profile page.
Normal Flow	1.Client clicks “Set Goals” button to add new goals. 2.New page for setting goals will be shown to the client. 3.Client adds or removes his/her goals.
Alternative Flows	None.
Exceptions	None.

Table 6: Use case description for Set Goals

Use Case ID	7
Use Case Name	Track Progress
Actors	Client
Descriptions	Client can track his/her progress and see how close s/he is to the set goals.
Pre-conditions	1.The client has been logged in to the system. 2.Client should have added his/her goals to the system.
Post-conditions	New page for tracking progress will be shown to the user.
Normal Flow	1.Client clicks track progress button. 2.Tracking progress page will be shown to the user.
Alternative Flows	None
Exceptions	None

Table 7: Use case description for Track Progress

Use Case ID	8
Use Case Name	Request Diet and Sport Plan
Actors	Client
Descriptions	Client can request a diet and sport plan to achieve his/her goals.
Pre-conditions	1.The client has been logged in to the system. 2.Client has set goals.
Post-conditions	Diet and sport plan request will be saved for coaches.
Normal Flow	1.The client clicks the “Request Diet and Sport Plan” button.
Alternative Flows	None
Exceptions	None

Table 8: Use case description for Request Diet and Sport Plan

Use Case ID	9
Use Case Name	List Clients’ Goals
Actors	Coach
Descriptions	Coaches can view clients’ goals and their personal attributes to set the appropriate diet and sport plans for them.
Pre-conditions	1.Clients should have registered to the system. 2.Clients should have set their goals. 3.Clients should have requested for diet and sport plan.
Post-conditions	Clients’ goals and their attributes will be shown to the coach.
Normal Flow	1.The coach logs in to the system. 2.The coach navigates to the “Clients’ Requests” page. 3.The coach chooses a client to set a diet and sport plan for him/her. 4. The coach clicks “View Client Profile” button.
Alternative Flows	None.
Exceptions	None.

Table 9: Use case description for View Client Profile

Use Case ID	10
Use Case Name	Set Diet and Sport Plan for Client
Actors	Coach
Descriptions	Coaches set diet and sport plan for clients to help them achieve their goals.
Pre-conditions	The clients' requests have been sent to the coaches.
Post-conditions	New diet and sport plan will be sent to the client by coach.
Normal Flow	1.The coach logs in to the system. 2.The coach navigates to the "Clients' Requests" page. 3.The coach chooses a client to set plan for him/her. 4. The coach views the client profile. 5.The coach sets diet and sport plan for the client and send it to the client.
Alternative Flows	None
Exceptions	None

Table 10: Use case description for Set Diet and Sport Plan for Client

4.3. Hardware Interfaces

AWS will be used as a cloud server in the project.

The Mobile Phone which has Android OS on it will be used as a device. Mobile phone should have at least 30 MB empty storage, internet connection.

Android version should not be older than 7.0.

4.4. Software Interfaces

Because the interaction with other application is not required, there are not any software interfaces to mention.

4.5. Communications Interfaces

AWS's Cognito Identity service packet for .NET will provide communication between application and AWS DynamoDB.

4.6. Memory Constraints

All data will be stored in an AWS DynamoDB database. Because there are many types of tiers, we will start with Free Tier which has 5 GB memory. Depends on number of active users, tier can be updated, and memory constraints can be different.

Also, application should not use more than 30 MB of storage.

4.7. Operations

Because data will be stored on AWS DynamoDB, recovery and backup process can be done by the system with given period. If needed, AWS S3 instance can be created for these operations.

Also, because DynamoDB is NoSQL database, Data Processing functions can be handled easily such as sorting, validation, classification etc.

The application will have user-initiated operations. User can do CRUD operations for his or her profile except Diet or Sport Plans. This column can only be written or changed by the user whose type is coach.

4.8. Site Adaptation Requirements

Because the application will be designed for android mobile phones, there will be no site adaptation requirement for application side. However, there can be some requirements for AWS part. If customers want to add or change any requirement for storing data which can be adding new column or value limitation etc., this information should be documented and added to both database side and application side by developers.

4.9. Interfaces with Services

Cloud service is the backend part in this project and we use Amazon Web Services as cloud service and use it as Software as Service (SAAS).

5. Product Functions

There are three important objectives for this project. First one is that user can enter his or her daily activities without no effort and with the help of coach, user can learn what s/he should do for next day depends on user's desired goals.

The other objective is time saving. User can reach the coach without reservation or going coach's office. There will be no wasting time with using this application.

Last objective is that because user's activities will be saved on the system, user's retrospective analysis, records, achievements, failures etc. can be observed by the coach and user without using external programs. Reaching this information can help the coach to observe user's condition and also the user can be motivated via seeing the process.

6. User Characteristics

We can say that the healthy status of clients will be the limitations. Because we assume that clients have healthy status and they have no health problem. Also, there should be some

limitations for the Coach side. To register as a Coach, the person should provide documents such as the certificate to prove s/he has sufficient information about this job.

7. Limitations

Due to the current pandemic process, as the main limitation, we can show that we have to take the lesson via distance education facilities and develop this application with remote connections with the project group. In addition, the fact that we will use a cloud service for the first time will create some limitations in terms of time management. In addition, although we foresee that we will supply benefit users with the application, the idea that the proposed meal and exercise program will cause some possible problems due to the fact that we will be deprived of some background information such as eating habits or health problems of the users creates limitations for us. The limitations at this stage are limited to these, but this part is open to add content in the process of developing application.

8. Assumptions and Dependencies

It is anticipated that users will approve the necessary permissions in order to comply with some legal regulations. It is also among our assumptions that the opportunities that the cloud service we will use in the context of the application will be sufficient within the scope of free membership. Also, GitHub and Google Drive will provide communication between developers. Finally, it is also assumed that the users will obey the meal and exercise list sent by the coach and share the correct results.

9. Specific Requirements

Specific requirements presented at the table below with ID that is in accordance with the use case descriptions and requirement descriptions.

Requirement ID	Requirement Description
1	The system shall enable user to register to the system.
1.1	The application shall provide coach graphical user interface where they can register by their username, password, profile type.
	The application shall provide coach graphical user interface where they can enter their past experience and add their CV.
1.2	The application shall provide client graphical user interface where they can register by their username, password, profile type.

1.3	The application shall provide client graphical user interface where they can enter their personal characteristics such as gender, age, height, weight and body fat.
2	The system shall enable user to login to the system.
2.1	The application shall provide coach graphical user interface where they can login by their username, password, profile type.
2.2	The application shall provide client graphical user interface where they can login by their username, password, profile type.
3	The system shall enable user to log out from the system.
3.1	The application shall provide coach a button which they can log out at any time at graphical user interface.
3.2	The application shall provide client a button which they can log out at any time at graphical user interface.
4.	The system shall enable client to add food items to the system.
4.1	The application shall provide client 'add food items' button to add his/her consumptions at graphical user interface.
4.2	The application shall provide client to select food item from the dropdown list.
4.3	The application shall provide client to select the food from the search result.
4.4	The application shall provide client to enter the amount of food s/he eat.
5	The system shall enable client to add sport activity to the system.
5.1	The application shall provide client 'add sport activities' button to add his/her exercises at graphical user interface.
5.2	The application shall provide client to select exercise from the dropdown list.
5.3	The application shall provide client to select the exercise from the search result.
5.4	The application shall provide client to select the set number, distance or time etc. according to the exercise type.
6.	The system shall enable client to set goal to the system.
6.1	The application shall provide client set goal button to add his/her purpose at the graphical user interface.
6.2	The application shall provide client to enter the target weight value, the body type they want to reach, in many days long they want to reach this body type, and extra comments such as allergies or eating habits at the graphical user interface.

6.4	The application shall provide client to add or update his/her goal by clicking the the button at the graphical user interface.
7	The system shall enable client to track his/her progress.
7.1	The application shall provide client 'Track Progress' button to see calorie of the past meals and exercises at the user profile page.
7.2	The application shall provide client the feedback of the coach at the graphical user interface.
8	The system shall enable client to request diet and sport plan.
8.1	The application shall provide client "Request Diet and Sport Plan" button at the graphical user interface.
9	The system shall enable coach to list clients' goals.
9.1	The application shall provide coach to view Client's Request page at the graphical user interface.
9.2	The application shall provide coach to choose a client to set a diet and sport plan for this client.
9.3	The application shall provide coach to view the profile of client.
10	The system shall enable coach to set diet and sport plan for client.
10.1	The application shall provide coach to view Client's Request page at the graphical user interface.
10.2	The application shall provide coach to choose a client to set a diet and sport plan for this client.
10.3	The application shall provide coach to view the profile of client.
10.4	The application shall provide coach to set diet and sport plan for sending the related client.

Table 11: Specific Requirements

10. External Interfaces

Class diagram can be seen from the Figure 3 below.

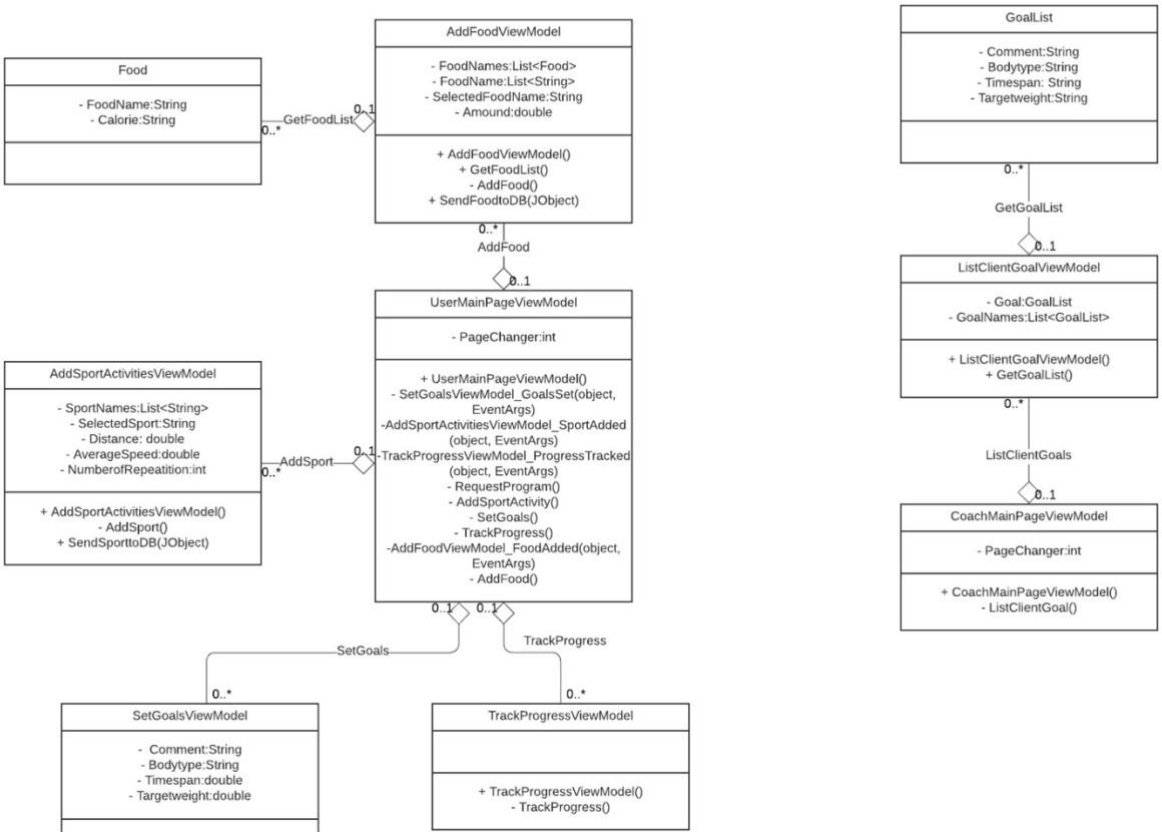


Figure 3: Class Diagram

11. Functions

There are some general calorie calculations which will be used at this application. According to the accepted global standards (regardless of health problems and special conditions), the number of calories that men and women should take according to their daily physical activity and age ranges in order to maintain their current weight is certain. At the same time, the amount of carbohydrate, protein and fat contained in foods are presented in percentages, and it has been scientifically determined that 1 gram of fat is 9 kcal and 1 gram of carbohydrate or protein is 4 kcal.

Likewise, it is assumed that certain exercises cause how many calories a person is consumed according to the set or number. In the app, the coach will suggest a food and exercise list using this information in line with the personal information entered and the purpose. Detailed information on other fundamental actions will be added in the next steps.

12. Software System Attributes

Availability: The system should be up and live all the time and this will be guaranteed by the Amazon Web services.

Security: Amazon Web Services provides its own security. To quote from their own sites; *"Cloud security at AWS is the highest priority. As an AWS customer, you benefit from a data center and network architecture that is built to meet the requirements of the most security-sensitive organizations."* Therefore, cloud security is provided by itself. Also, each user (both clients and coaches) must register and login to the system. The password entered during registration is also saved in the system using the hash function, and therefore, the necessary security is provided by restricting access outside the user.

Reliability, Maintability and Portability parts will be added later.

13. Verification

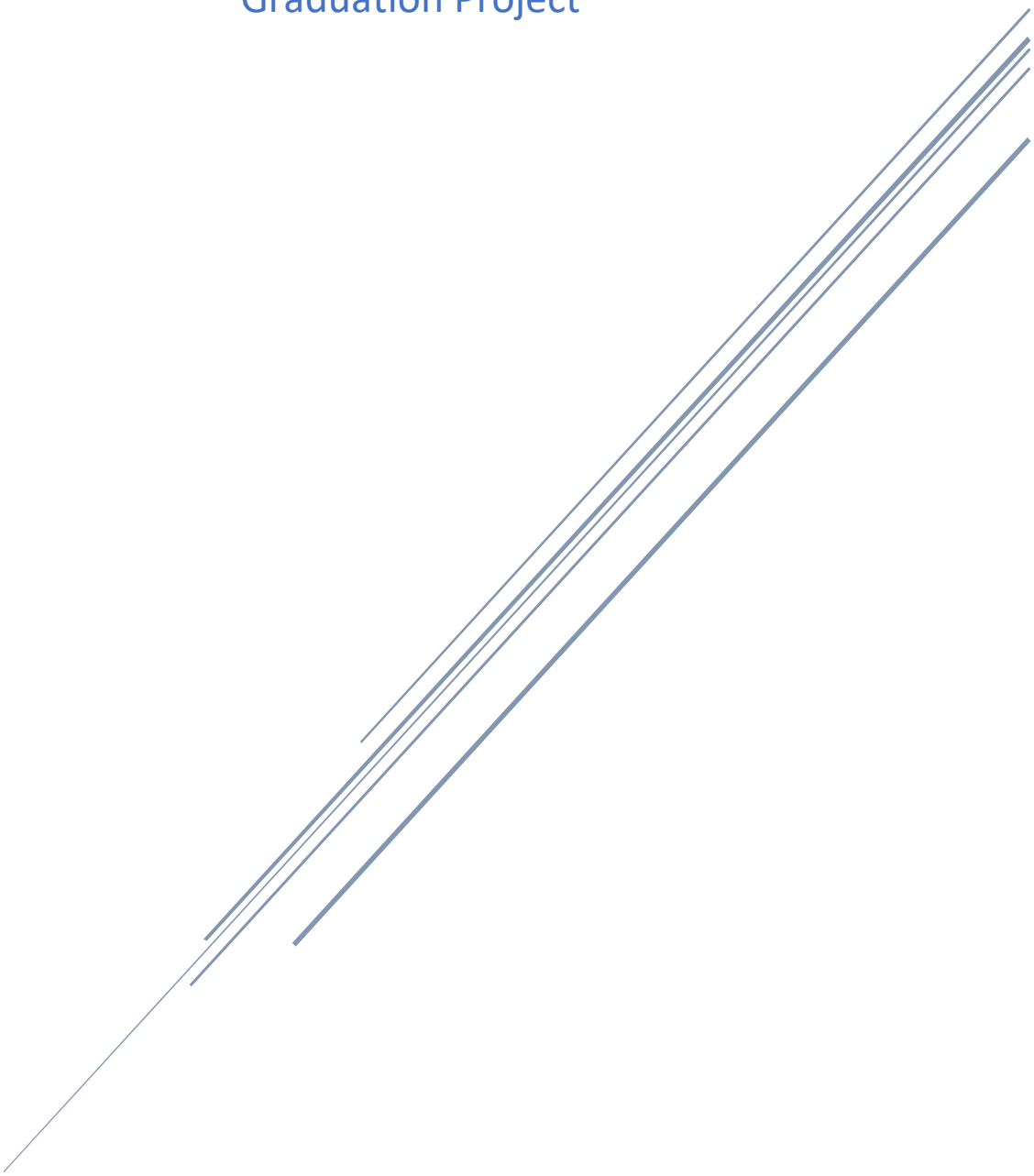
Verification will be with the testing according to the ISO-IEC-IEEE-29119-3-2013 standards at the 12th week.

14. Supporting Information

C# will be used as a software language and the visual studio will be used as an IDE at this project.

SMARTCAL: INNOVATING MOBILE TECHNOLOGY FOR REAL-TIME CALORIE TRACKING AND DIETARY GUIDANCE

Graduation Project



Software Design Descriptions Document

Table of Contents

List of Figures

List of Tables

1. Overview

Recently, with the aim of living a healthy life in a global sense, healthy eating and regular sports habits have started to be adopted in our country. In this context, it is of great importance for users to eat meals and do sports for their own purposes. Considering that almost everyone now has a smartphone, a mobile application that can be used for this purpose, which is accessible at any time, will make things easier for people. This application, which was developed with this in mind, set out with the aim of meeting the increasing needs of users.

A design was designed in line with the needs that emerged during the SRS (software requirements specification) and it was aimed to proceed based on the findings in the SRS document. Specific requirements are the main influence for existing design and design constraints. The SDD software may change or modify the documentation in the life cycle in relation to the previous SRS and the next Test document.

Existing changes that may occur during the design may cause us to change the SRS. In our project, different from SRS, a change was made as follows. In the application, the user entered his/her target weight value, the body type s/he want to reach, in many days long s/he want to reach this body type, and extra client comments such as allergies or eating habits and then requested a diet/exercise list from the coach. Afterwards, he set a goal for himself/herself in the set goals section and followed up on the way to reach this goal. However, instead of considering the set goals part and the diet/exercise program request as separate units, they were both integrated into the set goals because it was predicted that the coach's suggestion would not be very effective without a goal being set.

Our specific requirements were basically to register to the client system, create their own goal from the set goals section, set a goal, and an exercise/meal list would be presented to them and followed up. It can be said that these requirements can be met with the design introduced at this stage, but some minor changes were also made. For example, it is planned to create an exercises table for sports activities. With this table, it was aimed to show the user which sports s/he could do. However, since different fields had to be filled in for different sports types (it was predicted that there would be a standardization problem since different types of activities progressed through different units), this was abandoned and it was decided to keep the fields to be filled on the mobile application instead of pulling the information from the system with the created table.

1.1. Scope

The mobile application, which will be developed within the scope of this project, can be accessed from mobile devices using Android. In this way, we aimed to reach the maximum number of users that can be reached by developing an application that can run on the most widely used mobile operating system in the world.

Users will enter their personal information through the application and determine their priority targets in line with their purposes. As a target, the user's target weight value, the body type they want to reach, in many days long they want to reach this body type, and extra client comments such as allergies or eating habits are also received.

1.2. Purpose

Within the scope of Calorie Tracker application, it is aimed that all users who want to gain, lose or maintain weight can do this in a healthy way and this mobile application has been developed.

1.3. Intended Audience

- **Acquirers** use this document to ensure that their design specifications are satisfied.
- This document is used by **project managers** to regulate the dynamics of development operations in accordance with the plans.
- This document is used by **quality assurance professionals** to ensure that development activities are carried out in accordance with the quality plans.
- This document is used by **configuration managers** to audit development efforts for various configurations.
- **Software designers** compare the criteria in this paper to real-world practices.
- This document is used by **programmers** to develop the system in accordance with the design requirements and opinions.
- This document is used by **testers** to create test plans and documentation. This document is used by maintainers to keep the system up to date in response to changing demands after the system has been delivered.
- This document is used by **SE 560 Lecturers** to ensure that the design views are explained in accordance with the IEEE 1016:2009 Standard.

2. Definitions

- **C#:** C# is a general-purpose, modern and object-oriented programming language pronounced as “C sharp”. It was developed by Microsoft led by Anders Hejlsberg and his team within the .Net initiative and was approved by the European Computer Manufacturers Association (ECMA) and International Standards Organization (ISO). C# is among the languages for Common Language Infrastructure and the current version of C# is version 7.2
- **.Net Framework:** .Net applications are multi-platform applications and framework can be used from languages like C++, C#, Visual Basic, COBOL etc. It is designed in a manner so that other languages can use it.
- **Visual Studio:** Microsoft Visual Studio is an integrated development environment (IDE) from Microsoft. It is used to develop computer programs, as well as websites, web apps, web services and mobile apps. Visual Studio uses Microsoft software development platforms such as Windows API, Windows Forms, Windows Presentation Foundation, Windows Store and Microsoft Silverlight. It can produce both native code and managed code.
- **Android Application:** An Android app is a software application running on the Android platform. Because the Android platform is built for mobile devices, a typical Android app is designed for a smartphone or a tablet PC running on the Android OS
- **UML:** The Unified Modeling Language (UML) is a general-purpose, developmental, modeling language in the field of software engineering that is intended to provide a standard way to visualize the design of a system.
- **StarUML:** StarUML is built as a modular and open tool. It provides frameworks for extending the functionality of the tool. It is designed to allow access to all functions of the model/meta-model and tool through COM Automation, and it provides extension of menu and option items.
- **Cloud Computing:** Cloud computing is the delivery of computing services—including servers, storage, databases, networking, software, analytics, and intelligence—over the Internet (“the cloud”) to offer faster innovation, flexible resources, and economies of scale.

- **Amazon Web Services:** Amazon Web Services offers a broad set of global cloud-based products including compute, storage, databases, analytics, networking, mobile, developer tools, management tools, IoT, security and enterprise applications. These services help organizations move faster, lower IT costs, and scale.
- **DynamoDB:** DynamoDB is an Amazon Web Services database system that supports data structures and key-valued cloud services. It allows users the benefit of auto-scaling, in-memory caching, backup and restore options for all their internet-scale applications using DynamoDB.

3. Design Viewpoints

3.1. Introduction

This clause defines several design viewpoints for use in SDDs. It illustrates the realization of these design viewpoints in terms of design language selections, relates design concerns with viewpoints, and establishes language (notation and method) neutral names for these viewpoints. At the below, there are different design viewpoints.

3.2. Context Viewpoint

The Context viewpoint describes services provided by a design subject, relationships and dependencies in the software development environment. The Context viewpoint is a way of representing the system as a black box, users, stakeholders and their interaction are represented. Context diagram is shown below:

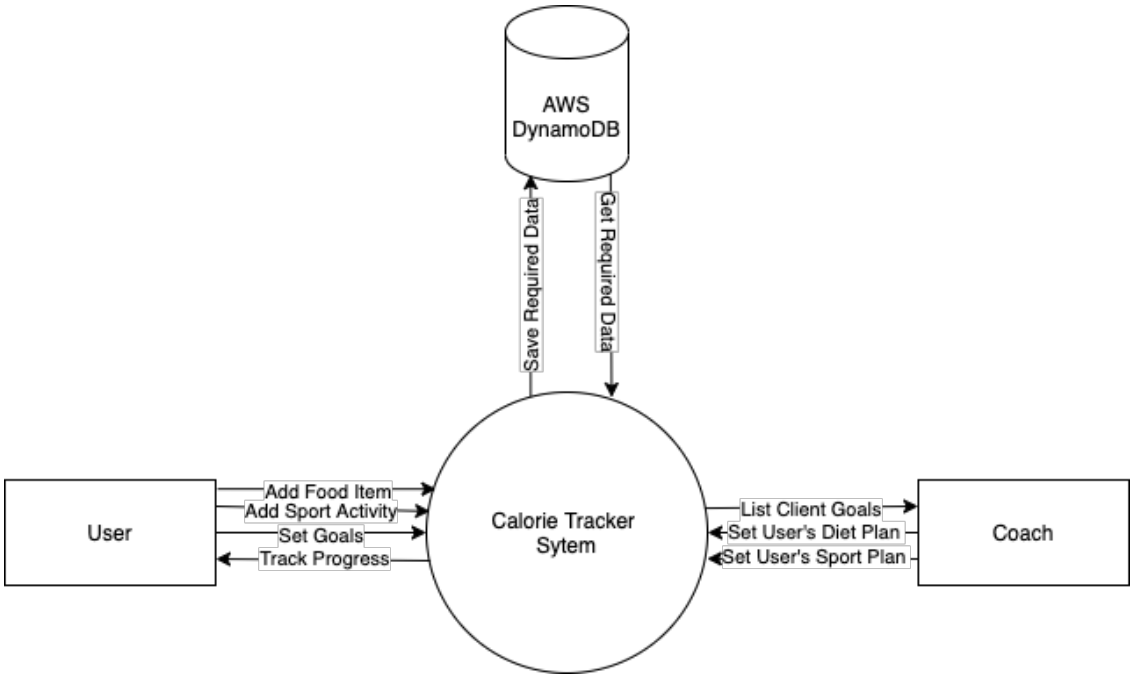


Figure 1: Context diagram

3.2.1. Use Cases

Use case diagram is the main design element which can be seen from Figure 2.

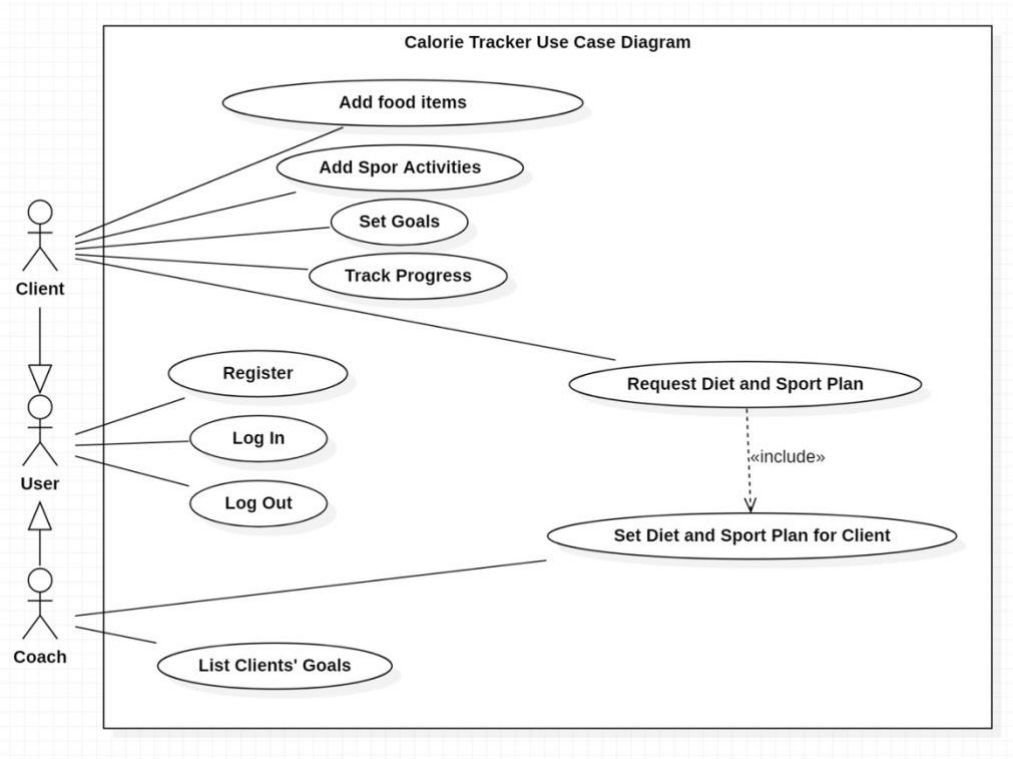


Figure 2: Use Case Diagram for Calorie Tracker Application

3.2.2. Use Case Descriptions

There are descriptions for all the interaction and you can see those from the ten tables below.

Use Case ID	**1**
Use Case Name	Register
Actors	Client, Coach
Descriptions	Every client and coach have to register to the system to use the application.
Pre-conditions	1.The user or coach should not have an account in the system.
Post-conditions	1.There is not any post-condition for this use case.
Normal Flow	1.User enters his/her username. 2.User enters his/her password. 3.User enters profile type. (Client or Coach) 4.Client enters his/her personal characteristics such as gender, age, height, weight and body fat. Coach enters his/her CV and experience.
Alternative Flows	None.

Exceptions	None.

Table 1: Use case description for Register

Use Case ID	**2**
Use Case Name	Login
Actors	Client, Coach
Descriptions	Every user has to login to the system to use the application.
Pre-conditions	1.The user or coach has to have an account in the system.
Post-conditions	1.Profile page for the user will be shown.
Normal Flow	1.User enters the username. 2.User enters the password. 3.User clicks login button. 4.User profile page will be shown to the user.
Alternative Flows	1.Check password, if it is wrong. 2.User enters username and password again.
Exceptions	None.

Table 2: Use case description for Login

Use Case ID	**3**
Use Case Name	Logout
Actors	Client, Coach
Descriptions	Every user has to log out of the system when they finish their work.
Pre-conditions	1.The user or coach has to be online in the system.
Post-conditions	1.There is not any post-condition for this use case.
Normal Flow	1. User presses logout button when user is online.
Alternative Flows	None.
Exceptions	None.

Table 3: Use case description for Logout

Use Case ID	**4**
Use Case Name	Add Food Items
Actors	Client
Descriptions	Client can add food items to the system for calorie intake calculation.
Pre-conditions	The client has been logged into the system.
Post-conditions	None.
Normal Flow	1.Client clicks “add food items” button to add his/her consumptions. 2.Client adds his/her food items by specifying food type and amount.
Alternative Flows	None

Exceptions	None

Table 4: Use case description for Add Food Items

Use Case ID	**5**
Use Case Name	Add Sport Activities
Actors	Client
Descriptions	Client can add his/her sport activities to the system for calorie tracking.
Pre-conditions	The client has been logged in to the system.
Post-conditions	None.
Normal Flow	1.Client clicks “add sport activities” button to add his/her sport activities. 2.Client adds his/her sport activities to the system.
Alternative Flows	None.
Exceptions	None.

Table 5: Use case description for Add Sport Activities

Use Case ID	**6**
Use Case Name	Set Goals
Actors	Client
Descriptions	Client can set goal for gaining, losing or maintaining his/her weight.
Pre-conditions	The client has been logged in to the system.
Post-conditions	The client will see his/her updated profile page.
Normal Flow	1.Client clicks “Set Goals” button to add new goals. 2.New page for setting goals will be shown to the client. 3.Client adds or removes his/her goals.
Alternative Flows	None.
Exceptions	None.

Table 6: Use case description for Set Goals

Use Case ID	**7**
Use Case Name	Track Progress
Actors	Client
Descriptions	Client can track his/her progress and see how close s/he is to the set goals.
Pre-conditions	1.The client has been logged in to the system. 2.Client should have added his/her goals to the system.
Post-conditions	New page for tracking progress will be shown to the user.
Normal Flow	1.Client clicks track progress button. 2.Tracking progress page will be shown to the user.

Alternative Flows	None
Exceptions	None

Table 7: Use case description for Track Progress

Use Case ID	**8**
Use Case Name	Request Diet and Sport Plan
Actors	Client
Descriptions	Client can request a diet and sport plan to achieve his/her goals.
Pre-conditions	1.The client has been logged in to the system. 2.Client has set goals.
Post-conditions	Diet and sport plan request will be saved for coaches.
Normal Flow	1.The client clicks the "Request Diet and Sport Plan" button.
Alternative Flows	None
Exceptions	None

Table 8: Use case description for Request Diet and Sport Plan

Use Case ID	**9**
Use Case Name	List Clients' Goals
Actors	Coach
Descriptions	Coaches can view clients' goals and their personal attributes to set the appropriate diet and sport plans for them.
Pre-conditions	1.Clients should have registered to the system. 2.Clients should have set their goals. 3.Clients should have requested for diet and sport plan.
Post-conditions	Clients' goals and their attributes will be shown to the coach.
Normal Flow	1.The coach logs in to the system. 2.The coach navigates to the "Clients' Requests" page. 3.The coach chooses a client to set a diet and sport plan for him/her. 4. The coach clicks "View Client Profile" button.
Alternative Flows	None.
Exceptions	None.

Table 9: Use case description for View Client Profile

a	**10**
Use Case Name	Set Diet and Sport Plan for Client
Actors	Coach
Descriptions	Coaches set diet and sport plan for clients to help them achieve their goals.
Pre-conditions	The clients' requests have been sent to the coaches.
Post-conditions	New diet and sport plan will be sent to the client by coach.
Normal Flow	1.The coach logs in to the system. 2.The coach navigates to the "Clients' Requests" page. 3.The coach chooses a client to set plan for him/her. 4. The coach views the client profile.

	5.The coach sets diet and sport plan for the client and send it to the client.
Alternative Flows	None
Exceptions	None

Table 10: Use case description for Set Diet and Sport Plan for Client

3.3. Composition Viewpoint

With the component diagram, which you can see in figure 3 below, you can get rid of the system complexity and look at all the components and their relations at a simple level. With the lollipop interface, different operations carried out by the mobile application can be examined.

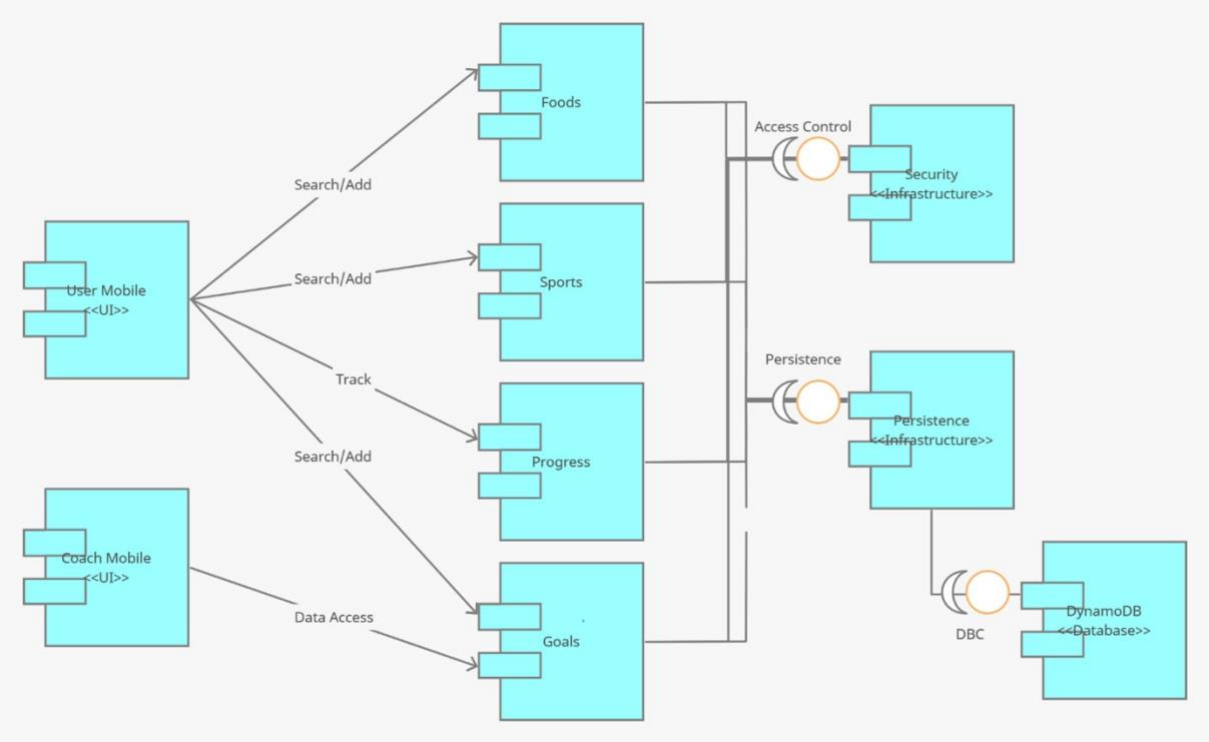


Figure 3 Component Diagram

3.4. Logical Viewpoint

With the class diagram, which you can see on figure 4 created using UML, the classes created in the application and the relationship between them can be discussed. The classes at this diagram are;

Food Class: At the food class, there is GetFoodList() and by that function, client can get the food list and if s/he add a food, it is send to the database. Food names are string type.

User Main Page Class: At this class there are different functions and all of these are presented to user to do things at this page. For example, by the AddSportActivitiesViewModel_SportAdded user can add the sport activity which s/he did and

also, they can track their progress via TrackProgressViewModel_ProgressTracked. There are three main types at this class and these are adding sport activities, setting goals of their own and track their progress.

Coach Class: At this class, there are the functions of the coaches and these are ListCliebtGoalViewModel which supplies the getting goal list and seeing them and GoalList that contains the comment of the coaches/users, body type of the user, the time span since the goal was formed and the target weight.

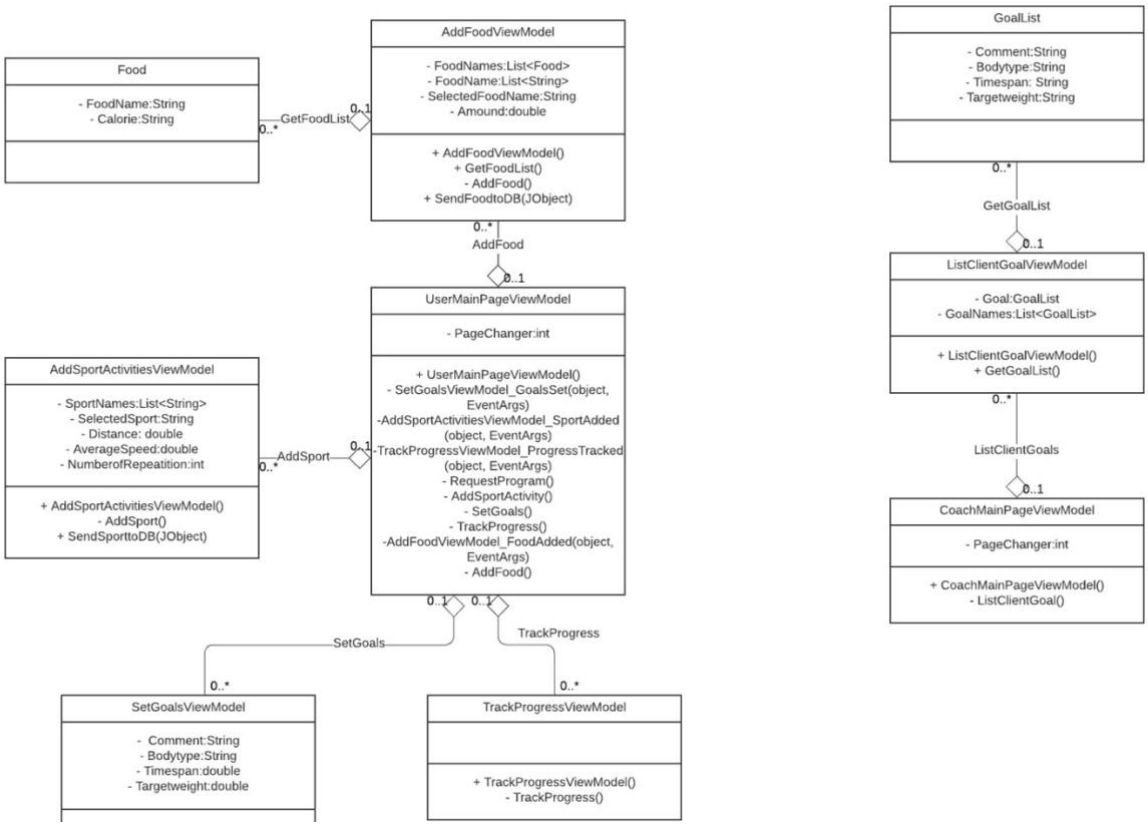


Figure 4: Class Diagram

3.5. Dependency Viewpoint

There are some dependencies of components at our application. We can classify these dependencies as devices, regulations, prerequisites and also system dependencies. For a device, of course o mobile devices which as Android OS on it is needed. Because the application can work on the Android OS. As a regulation and prerequisites, we can consider that the coach should have a certificate about it. Also, as a prerequisites user can have a good health status. Because, if the user has some diseases related to their bodies, the exercises and also foods can be harmful for them. Finally, there are some dependencies for proper running of system. Amazon Web Services play an important role for that project because it is a cloud service we use. So, we store all the backend parts at this cloud service so it is an essential part of the project.

Also, the DynamoDB is the database in the tables stored is the persistence and security part of the project.

3.6. Information Viewpoint

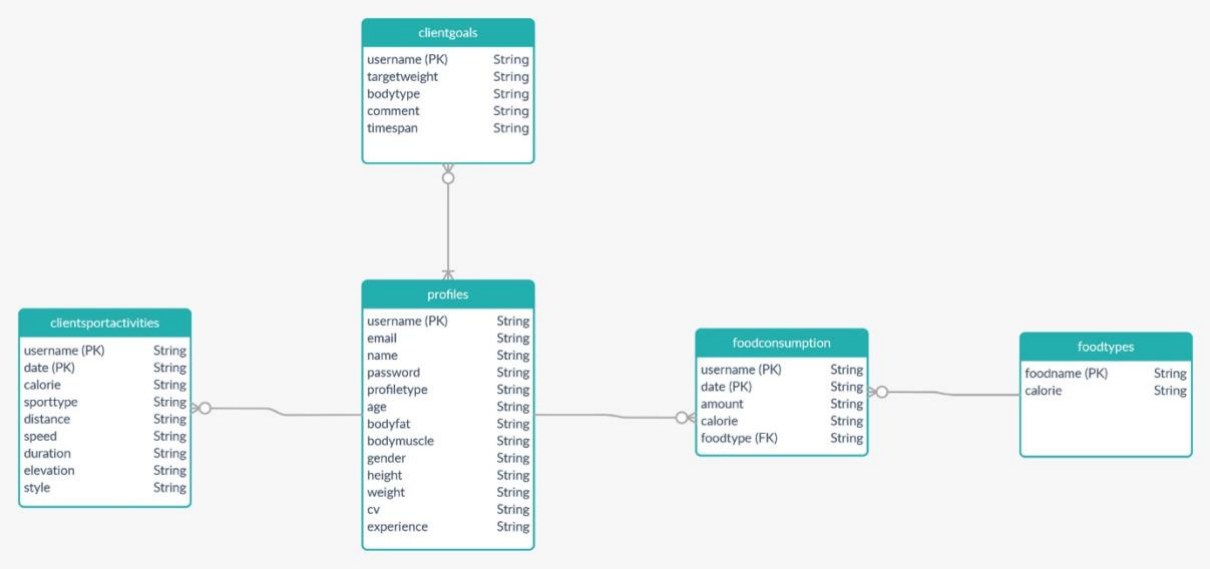


Figure 5 Enhanced Entity Relationship Diagram

Profiles table stores both the clients and coaches of our application. Username(PK), email, name, password, profiletype are the common attributes. Age, bodyfat, bodymuscle, gender, height, weight is specific to clients. CV and experience are specific to Coaches.

Clientgoals table stores the targetweight and bodytype that a user wants to reach in a specified timespan. User can specify his goals and specific wishes as comment. Coaches can access these goals and make suggestions and modifications to the clients.

Clientsportactivities table stores which types of sports a client has done on a specific date and how much. This then used to calculate calorie burned on that date and stored in this table. Differing attributes are used depending on the sporttype.

- Cycling=> Distance, speed and elevation
- Running=> Distance, speed
- Swimming=> Duration, style (freestyle, backstroke...)

Foodconsumption table stores which user consumed which food and how much on a specific date. This information is used to calculate calorie taken on that date and stored in this table. Foodtypes are types of food that are stored in the database and their unit calorie values.

3.7. Patterns Use Viewpoint

Model View Controller (MVC) is a software design technique for creating computer user interfaces. It divides a given application into three interconnected parts to separate internal information representations from how information is presented to and accepted by the user.

- *Model:* The pattern's lowest level, which is in charge of data maintenance. That part is the food and sports activities in our application. Because the main data is the foods and exercises at the application. Also, the goals which is set and control by clients can be considered at that type.
- *View:* View is in charge of displaying all or a portion of the data to the user. View part is the interfaces of our application. With the login/register, client profile, list foods/exercise etc. pages are all at the view part of the application.
- *Controller:* Software code that governs how the Model and View interact. For the controlling the goals and desires of the client can be used. By the controlling system all the food, exercises and also user information data and the interfaces are combined and controlled by the system.

3.8. Interface Viewpoint

Deployment diagram of the application are mainly depending on three part and it can be seen from the Figure 6 below.

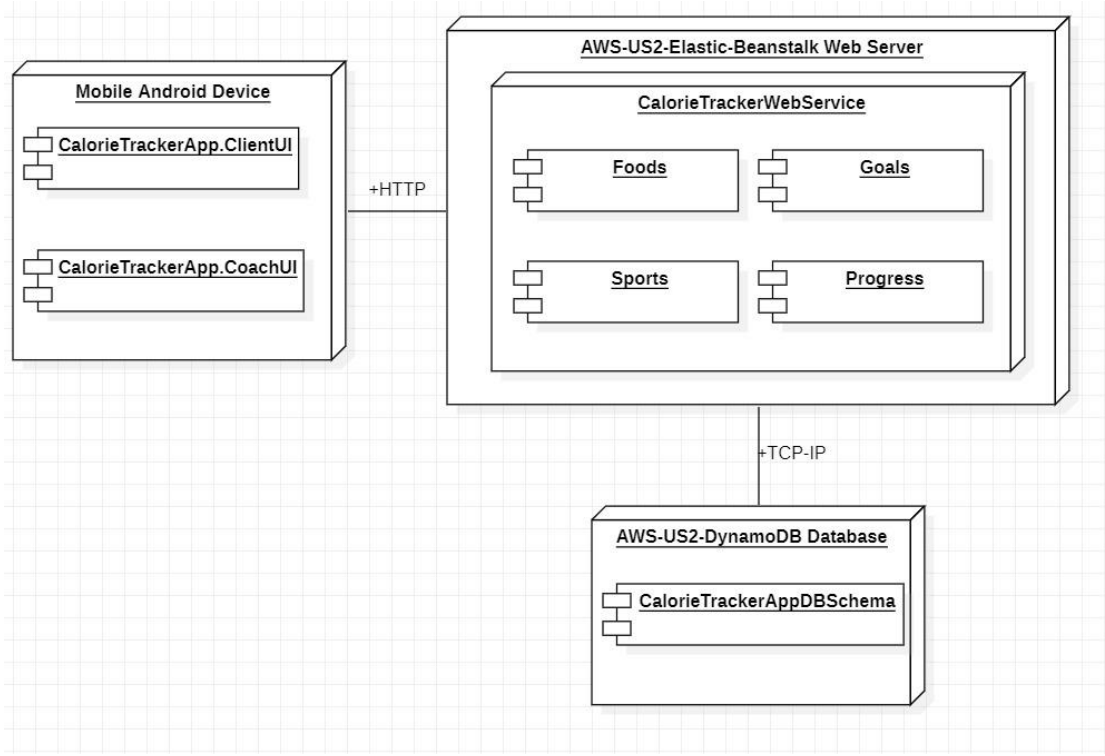


Figure 6 Deployment Diagram

Finally, the sequence diagram which have three parts as user, calories and coach can also be seen from and Figure 7.

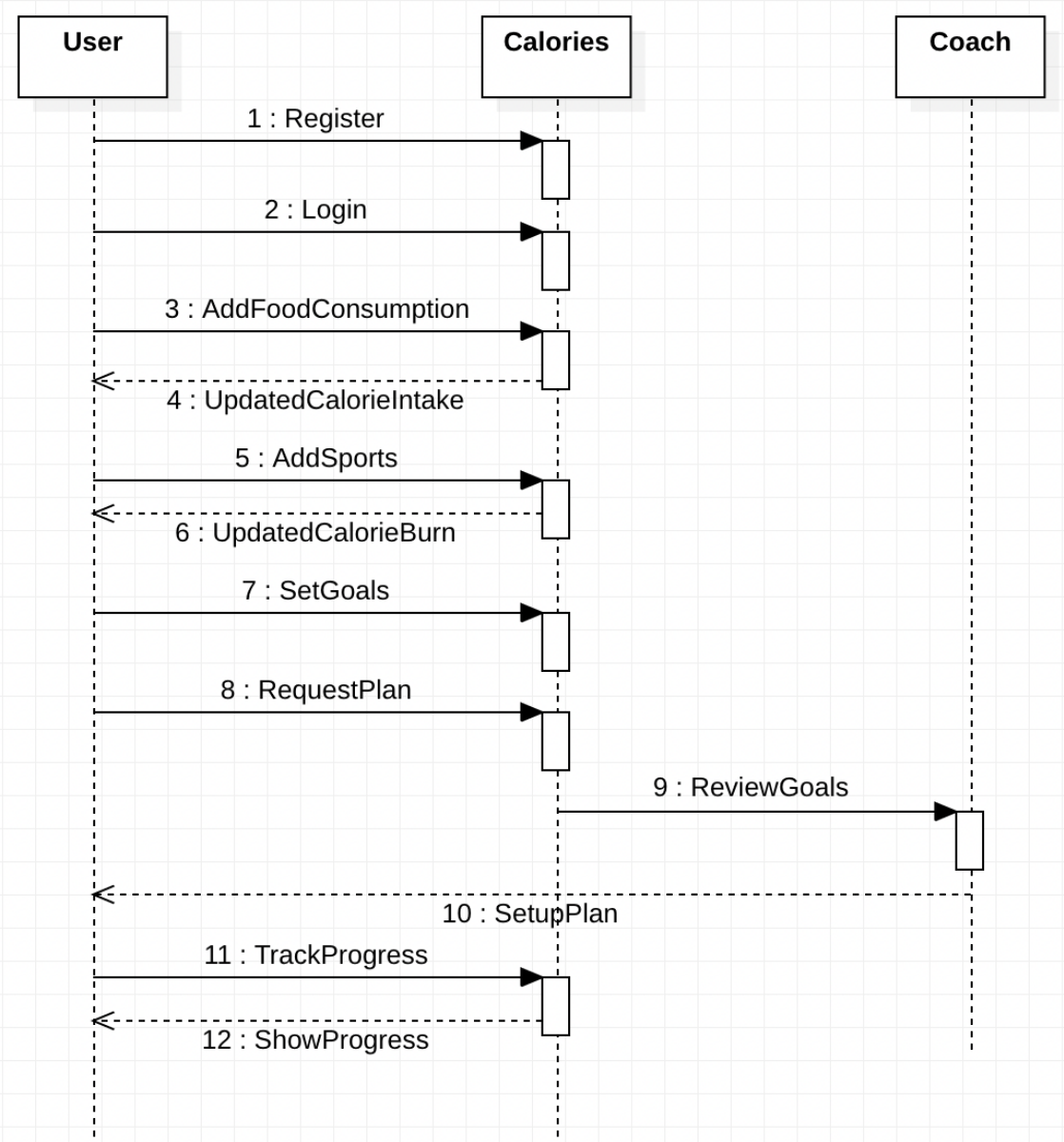
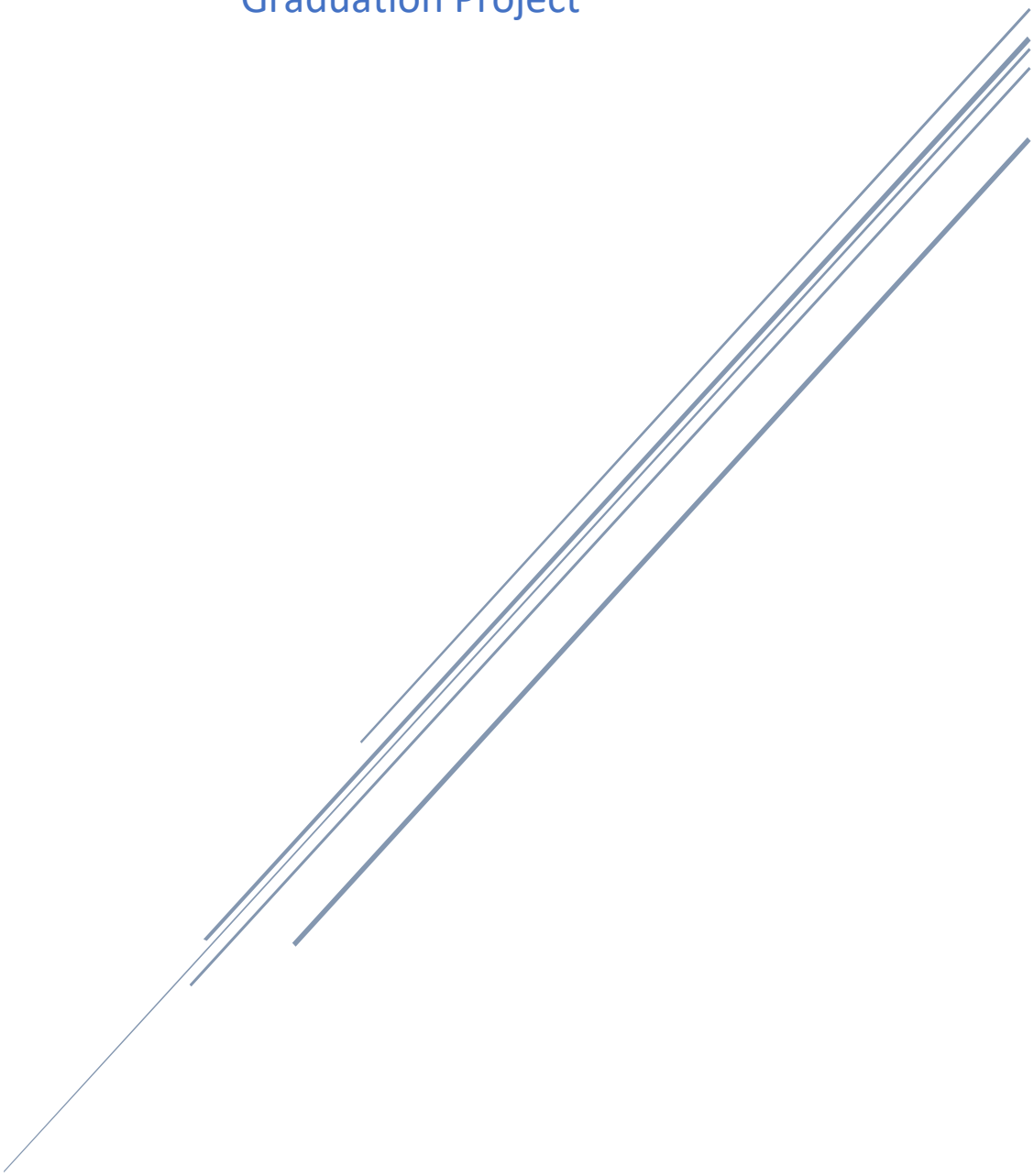


Figure 7 Sequence Diagram

SMARTCAL: INNOVATING MOBILE TECHNOLOGY FOR REAL-TIME CALORIE TRACKING AND DIETARY GUIDANCE

Graduation Project



Software Test Plan Document

Table of Content

List of Figures

List of Tables

1. Overview

Recently, with the aim of living a healthy life in a global sense, healthy eating and regular sports habits have started to be adopted in our country. In this context, it is of great importance for users to eat meals and do sports for their own purposes. Considering that almost everyone now has a smartphone, a mobile application that can be used for this purpose, which is accessible at any time, will make things easier for people. This application, which was developed with this in mind, set out with the aim of meeting the increasing needs of users.

This is the software test plan document of Calorie Tracker Application, which is written based on IEEE 29119-3 standard for Software Project Test Plans.

1.1. Scope

The mobile application, which will be developed within the scope of this project, can be accessed from mobile devices using Android. In this way, we aimed to reach the maximum number of users that can be reached by developing an application that can run on the most widely used mobile operating system in the world.

Users will enter their personal information through the application and determine their priority targets in line with their purposes. These targets that is presented to users in three categories as losing weight, maintaining their current weight and gaining weight in a healthy way. In addition, in order to meet a possible different request, 'the other' option will be presented to the user and they will be able to specify their own special requests.

1.2. Purpose

Within the scope of Calorie Tracker application, it is aimed that all users who want to gain, lose or maintain weight can do this in a healthy way and this mobile application has been developed.

1.3. Definitions

- **C#:** C# is a general-purpose, modern and object-oriented programming language pronounced as "C sharp". It was developed by Microsoft led by Anders Hejlsberg and his team within the .Net initiative and was approved by the European Computer Manufacturers Association (ECMA) and International Standards Organization (ISO). C# is among the languages for Common Language Infrastructure and the current version of C# is version 7.2

- **.Net Framework:** .Net applications are multi-platform applications and framework can be used from languages like C++, C#, Visual Basic, COBOL etc. It is designed in a manner so that other languages can use it.
- **Visual Studio:** Microsoft Visual Studio is an integrated development environment (IDE) from Microsoft. It is used to develop computer programs, as well as websites, web apps, web services and mobile apps. Visual Studio uses Microsoft software development platforms such as Windows API, Windows Forms, Windows Presentation Foundation, Windows Store and Microsoft Silverlight. It can produce both native code and managed code.
- **Android Application:** An Android app is a software application running on the Android platform. Because the Android platform is built for mobile devices, a typical Android app is designed for a smartphone or a tablet PC running on the Android OS
- **UML:** The Unified Modeling Language (UML) is a general-purpose, developmental, modeling language in the field of software engineering that is intended to provide a standard way to visualize the design of a system.
- **StarUML:** StarUML is built as a modular and open tool. It provides frameworks for extending the functionality of the tool. It is designed to allow access to all functions of the model/meta-model and tool through COM Automation, and it provides extension of menu and option items.
- **Cloud Computing:** Cloud computing is the delivery of computing services—including servers, storage, databases, networking, software, analytics, and intelligence—over the Internet ("the cloud") to offer faster innovation, flexible resources, and economies of scale.
- **Amazon Web Services:** Amazon Web Services offers a broad set of global cloud-based products including compute, storage, databases, analytics, networking, mobile, developer tools, management tools, IoT, security and enterprise applications. These services help organizations move faster, lower IT costs, and scale.
- **DynamoDB:** DynamoDB is an Amazon Web Services database system that supports data structures and key-valued cloud services. It allows users the benefit of auto-scaling, in-memory caching, backup and restore options for all their internet-scale applications using DynamoDB.

2. Test Strategy

2.1. Test Objectives

The objective of the test is to verify that the functionality of Calorie Tracker Application works according to the specifications and given requirements.

The result of the test will provide;

- A production-ready mobile application;
- A set of stable test scripts that can be reused for functional and user acceptance test execution.

2.2. Test Policy

- Testing will be focused on meeting the business objectives, cost efficiency, and quality.
- There will be common, consistent procedures for all teams supporting testing activities.
- Testing processes will be well defined, yet flexible, with the ability to change as needed.
- There will be entrance and exit criteria.
- Testing will be divided into distinct phases, each with clearly defined objectives and goals.

2.3. Test Assumptions

- Production like data required and be available in the system prior to start of Functional Testing
- Performance testing is not considered for this estimation.
- The Test Team assumes all necessary inputs required during Test design and execution will be supported by development team appropriately.
- Test case design activities will be performed by tester.

2.3.1. Functional Testing

During Functional testing, testing team will use preloaded data which is available on the cloud at the time of execution.

2.3.2. User Acceptance Testing

User acceptance test execution will be performed by the end user under the supervision of developer team.

3. Test Case Design

3.1. Test Case Traceability

At that project each test case will be mapped to the Use Case Diagram of the Calorie Tracker Mobile Application. The use case of this application is presented in the following diagram;

3.1.1. Use Case Diagram

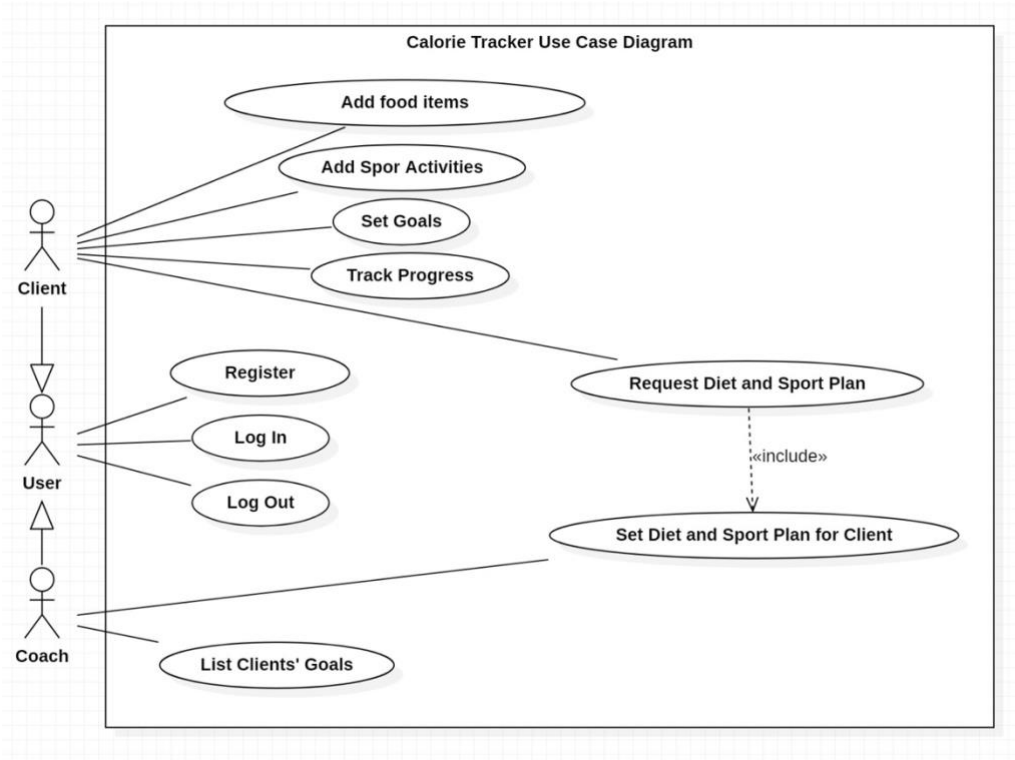


Figure 1 Use Case Diagram for Calorie Tracker Application

3.2. Test Case Specification

Each test case will be associated to specific use case and you can see the test case specification of each use case actions at the below tables.

3.2.1. Test Cases for Calorie Tracker Application

Test Case ID 1	Register User - 1
Objective	To test the registration of a new user to the system
Preconditions	• Client/Coach should not be previously registered and not logged in • Client/Coach should click Register button
Inputs	User should fill all the required fields about his profile
Expected Results	Calorie Tracker should create User with given credentials User should be redirected to Login Page

Table 1 Use Case: Register User & Test Case ID 1

Test Case ID 2	Register User - 2
Objective	To test a failed registration of a new user
Preconditions	• Client/Coach should not be previously registered • Client/Coach should be navigated to Register Page
Inputs	User should not fill all the required fields about his profile
Expected Results	Calorie Tracker should not create User with given credentials User should be shown an error message to fill all required fields

Table 2 Use Case: Register User & Test Case ID 2

Test Case ID 3	Login - 1
Objective	To test the login of an existing user into the system
Preconditions	• Client/Coach should not be logged in • Login page should be open
Inputs	User should not fill username and password fields correctly
Expected Results	User should be shown an error message stating incorrect entry

Table 3 Use Case: Login User & Test Case ID 3

Test Case ID 4	Login - 2
Objective	To test the login of an existing user into the system
Preconditions	• Client/Coach should not be logged in • Login page should be open
Inputs	User should fill username and password fields correctly
Expected Results	User should successfully enter the system and connect to his profile

Table 4 Use Case: Login User & Test Case ID 4

Test Case ID 5	Logout
Objective	To test the logout of a logged in user
Preconditions	• Client/Coach should be logged in
Inputs	Logout button should be clicked
Expected Results	User should be logged out of the system and should be redirected to login page

Table 5 Use Case: Logout User & Test Case ID 5

Test Case ID 6	Add Food Consumption
Objective	To test the addition of consumed food to user history
Preconditions	• User should be logged in • Food Consumption page should be open • Food types should be listed in a drop-down list
Inputs	User should enter food type and portion size of the meal he ate for the day
Expected Results	Consumed food and portion size should be saved to database and calorie taken value should be updated for the day

Table 6 Use Case: Add Food Consumption & Test Case ID 6

Test Case ID 7	Add Sport Activity - 1
Objective	To test the listing and customizing sport activity fields
Preconditions	• User should be logged in • Sport Activity page should be open • Sport types should be listed in a drop-down list
Inputs	User should select sport activity
Expected Results	Based on the selected sport activity, relevant measurement fields should be shown

Table 7 Use Case: Add Sport Activity & Test Case ID 7

Test Case ID 8	Add Sport Activity - 2
Objective	To test the adding of sport activity
Preconditions	• User should be logged in • Sport Activity page should be open • Sport types should be listed in a drop-down list • Sport activity should be selected
Inputs	User should fill measurement fields
Expected Results	Sport activity should be added to the database and calorie burned should be updated for the day

Table 8 Use Case: Add Sport Activity & Test Case ID 8

Test Case ID 9	Set Client Goal
Objective	To test the addition of client goals
Preconditions	• User should be logged in • Client Goals page should be open
Inputs	User should fill body type, target weight, timespan or any other additional comments
Expected Results	User's goal should be set and these goals should be available for review of coaches

Table 9 Use Case: Set Client Goal & Test Case ID 9

Test Case ID 10	List Client Goal
Objective	To test the listing of client goals
Preconditions	• Coach should be logged in • List Goals page should be open • Usernames of users with goals should be listed
Inputs	Coach should click username to get his/her goals that is set by themselves according to their aims
Expected Results	User's goal should be shown to the coach

Table 10 Use Case: List Client Goal & Test Case ID 10

Test Case ID 11	Set Client Plan
Objective	To test the setting of client's plan by coach
Preconditions	• Coach should be logged in • Set Plan page should be open • Coach should select Client's username
Inputs	Coach should fill calorie intake, calorie burn and plan fields
Expected Results	User's client plan should be set and be ready for the user

Table 11 Use Case: Set Client Plan & Test Case ID 11

Test Case ID 12	Get Client Plan
Objective	To test the viewing of personalized plans set by coach
Preconditions	• User should be logged in • Plans page should be open
Inputs	User should click get plan
Expected Results	Plan set by coach should be shown to the user

Table 12 Use Case: Get Client Plan & Test Case ID 12

Test Case ID 13	Track Progress
Objective	To test the tracking progress of the user
Preconditions	• User should be logged in
Inputs	User should open Track Progress page
Expected Results	The calorie taken and burned daily is shown to the user.

Table 13 Use Case: Track Progress & Test Case ID 13

3.2.2. Test Results for Each Use Case Table

Use Case ID	Test Case ID	Expected Result	Test Result	Status
Register User	1	Redirect user to login page	User is redirected to login page successfully	PASS
	2	Show error message as filling all required fields	Error message displayed to user	PASS
Login User	3	Show error message as incorrect entry	Error message displayed to user	PASS
	4	Redirect user to profile page	User is redirected to profile page successfully	PASS
Logout User	5	User logs out the system and redirects to login page	User is logged out from the system successfully	PASS

Add Food Consumption	6	Food and portion size of food saved to database and calorie taken value is updated for the day	Client opened the daily calorie intake screen and verified calorie intake has increased as much as the consumed food calorie	PASS
Add Sport Activity	7	Relevant measurements fields are shown based on the sport activity	Different measurements field is shown to the user with the changing sport activity type	PASS
	8	Sport activity saved to database and burned calorie is updated	Client opened the daily sport activity screen and verified calorie burned has increased as much as the exercise done in a day	PASS
Set Client Goal	9	Client is set his/her goal for reviewing of coaches	Client can set his/her goals by entering body type, target weight, timespan and also comments	PASS
List Client Goal	10	Coach can see the goal of the client	Coach can view the goal of clients by clicking the usernames	PASS
Set Client Plan	11	Coach set the plan for client	Coach can set the plan for client by entering the calorie intake, calorie burn	PASS
Get Client Plan	12	Plan set by coach is shown to the client	Client can see the plan prepared by coach with the clicking in his/her profile page	PASS
Track Progress	13	The calorie taken and burned can be seen	Client can see the taken and burned of his/her by clicking track progress in his/her profile page	PASS

Table 14 Test Results of Use Cases