**** ****

**Graduation Project Report**

**Department of Software Engineering**

**&**

**Computer Science**

**Faculty of Information Technology**

**Al-Ahliyya Amman University**

**Eat-Right**

*A project submitted**in partial fulfillment of the requirements for the degree of**Bachelor in Software Engineering*

**by:**

Hazem Mohammed Mousa Doudin (201910799)

Mustafa Sami Mustafa Yahia (202010784)

Yousef Mohamad Ahmad Al-Tbakhi (202010596)

Hassan Jawdat Khalid Al-Salman (202130158)

**Supervised by:**

Dr. **Abdelrahman Hussein**

November, 2023

Undertaking

This is to declare that the project entitled “Eat Right” is an original work done by undersigned, in partial fulfillment of the requirements for the degree “Bachelor in Software Engineering & Computer Science” at Software Engineering & Computer Science Department, Faculty of Information Technology, Al-Ahliyya Amman University. All the analysis, design and system development have been accomplished by the undersigned. Moreover, this project has not been submitted to any other college or university.

Name and Signature ………Hazem Mohammed Mousa Doudin…… Date: ……16/12/2023……

Name and Signature ………Mustafa Sami Mustafa Yahia……...…… Date: ……16/12/2023……

Name and Signature ………Yousef Mohamad Ahmad Al-Tbakhi…… Date: ……16/12/2023……

Name and Signature ………Hassan Jawdat Khalid Al-Salman……… Date: ……16/12/2023……

Acknowledgement

We would like to express our sincerest appreciation and gratitude to all the professors at our university. Their exceptional dedication, commitment, and hard work in teaching and mentoring have been an inspiration to us all. Their wealth of knowledge, expertise, and guidance in their respective fields have helped us to achieve our academic and professional goals.

Their passion for teaching, innovative approach, and ability to make complex subjects understandable and engaging have profoundly impacted our education. Their willingness to provide guidance, support, and mentorship inside and outside the classroom has been invaluable.

We thank them for the countless hours spent preparing lectures, grading assignments, and providing personalized feedback. Their dedication to the success of their students is truly admirable and has greatly contributed to our personal and professional growth.

We are deeply grateful for the opportunity to learn from such accomplished educators and we feel honored to have had the opportunity to work alongside them. We sincerely appreciate all their efforts and support.

Abstract

"**Eat Right**" is a comprehensive nutrition and fitness tracking website, designed so that system users can move in the right way towards achieving their desired fitness goals and the optimal health regimen that suits each individual system user.

The goal of **Eat -Right** is to help individuals make informed choices about their nutrition and take care of their overall health so that everyone achieves their best fitness and health. Through the training programs that the system provides to the user, he/she will be able to improve his/her overall sports performance and promote a sustainable approach to achieve Balanced healthy life.

Since the system to be developed contains a set of detailed instructions based on the information that the user will provide to the system, the site will, based on the information provided by the user, create a nutritional program that suits each user in particular, which includes the appropriate amounts of proteins, carbohydrates, and fats to achieve health goals.

The system also provides an easy-to-use platform that enables him to choose what he prefers from the meals provided by the system and the foods not provided, it also monitors the physical activities of the system user. The system to be developed also contains the possibility of determining the amount of weight that the user wishes to gain or lose based on his personal desire within a specific period of time.

**Table of Contents**

[Chapter 1: Introduction: 7](#_Toc156908037)

[1.1 Overview: 7](#_Toc156908038)

[1.2 Project Motivation: 8](#_Toc156908039)

[1.3 Review of Related Literature and Systems: 9](#_Toc156908040)

[1.4 Problem Statement (limitation of current systems): 10](#_Toc156908041)

[1.5 Project Objectives: 11](#_Toc156908042)

[1.6 Project Plan: 12](#_Toc156908043)

[Chapter 2: The Development Process: 13](#_Toc156908044)

[2.1 Agile Methodology: 13](#_Toc156908045)

[Chapter 3: Requirements Engineering and Analysis: 15](#_Toc156908046)

[3.1 Stakeholders 15](#_Toc156908047)

[3.2 Functional Requirements: 15](#_Toc156908048)

[**3.2.1 Index Screen Before Login:** 16](#_Toc156908049)

[**3.2.2 Registration Screen:** 16](#_Toc156908050)

[**3.2.3 Login Screen:** 17](#_Toc156908051)

[**3.2.5 Index Screen After Login:** 18](#_Toc156908052)

[**3.2.6 Recipes Screen:** 18](#_Toc156908053)

[**3.2.7 Food Screen:** 18](#_Toc156908054)

[**3.2.9 Contact Us Screen:** 19](#_Toc156908055)

[**3.2.10 Nutrition Tracker Screen:** 19](#_Toc156908056)

[**3.2.11 Admin Screen:** 21](#_Toc156908057)

[3.3 Non-Functional Requirements: 23](#_Toc156908058)

[3.4 Process Requirements: 25](#_Toc156908059)

[3.5 Use Case Diagram: 25](#_Toc156908060)

[3.6 Use Cases Descriptions: 27](#_Toc156908061)

[Chapter 4: Architecture and Design: 32](#_Toc156908062)

[4.1 Software Architecture: 32](#_Toc156908063)

[4.2 Software Design: 33](#_Toc156908064)

[**4.2.1 Sequence Diagrams:** 33](#_Toc156908065)

[**4.2.2 Class Diagram:** 36](#_Toc156908066)

[**4.2.3 ER Diagram:** 37](#_Toc156908067)

[**4.2.4** **Activity Diagram:** 38](#_Toc156908068)

[4.3 User Interface Design: 41](#_Toc156908069)

[Chapter 5: System Implementation: 41](#_Toc156908070)

[5.1 Overview: 41](#_Toc156908071)

[5.2 Tools and Technologies Used: 42](#_Toc156908072)

[5.3 Important Code Fragments: 46](#_Toc156908073)

[Chapter 6: Testing and Validation: 50](#_Toc156908074)

[6.1 Black-Box: 50](#_Toc156908075)

[Chapter 7: Conclusion and Future Work: 51](#_Toc156908076)

[7.1 Conclusion: 51](#_Toc156908077)

[7.2 Future Work: 51](#_Toc156908078)

[References: 52](#_Toc156908079)

List of Figures:

[Figure 1.1 Project Plan 13](file:///C:\Users\Hazem%20Dudin\Desktop\last%20semester\Hazem's%20profile\Graduation%20Project\New%20Microsoft%20Word%20Document.docx#_Toc156906644)

[Figure 2.1 Agile software development cycle. 15](file:///C:\Users\Hazem%20Dudin\Desktop\last%20semester\Hazem's%20profile\Graduation%20Project\New%20Microsoft%20Word%20Document.docx#_Toc156906645)

[Figure 3 42](#_Toc156906646)

[Figure 4 EF-Core (Code First) 44](#_Toc156906647)

[Figure 5 Asp.net Identify: We can see the relationship between all these tables that identity was create. 45](#_Toc156906648)

[Figure 6 Food Diary: 46](file:///C:\Users\Hazem%20Dudin\Desktop\last%20semester\Hazem's%20profile\Graduation%20Project\New%20Microsoft%20Word%20Document.docx#_Toc156906649)

[Figure 7 Food Diary: 47](#_Toc156906650)

[Figure 8 Registration Screen: 47](#_Toc156906651)

[Figure 9 Registration Screen: 48](#_Toc156906652)

[Figure 10 Registration Screen: 48](#_Toc156906653)

[Figure 11 Registration Screen: 49](#_Toc156906654)

List of Tables:

[Table 6 User, Admin login: 27](#_Toc156907877)

[Table 7 User Registration: 28](#_Toc156907878)

[Table 8 User Search for Recipes: 29](#_Toc156907879)

[Table 9 Admin Add Food: 30](#_Toc156907880)

[**Table 10. User Testing:** 49](#_Toc156907881)

List of Abbreviations:

|  |  |
| --- | --- |
| Abbreviation | Stands For |
| MTBF | Mean Time Between Failure |
| MTTR | Mean Time to Repair |

# Chapter 1: Introduction:

## 1.1 Overview:

The purpose of this document is to provide a detailed description of the "**Eat Right**" system that we will implement in our Website. It will explain the purpose and features of the "**Eat Right**" system, the interfaces of the system and what the system will do, and how users will be able to use the system to embracing a more active lifestyle.

"**Eat Right**" is designed to be a user-friendly and wellness application by providing the users with a holistic health and wellness experience. The application focuses on two main aspects: nutrition and fitness. By offering a range of features, "**Eat Right**" aims to assist users in achieving their health goals, making mindful decisions about their diet, and embracing a more active lifestyle.

“**Eat Right**” provides an overview of the system functionality and system interaction with other systems. This chapter also introduces different types of users and their interaction with the system. Further, the chapter also mentions the system constraints and assumptions about the services.

The system requirements specification in detailed terms and a description of the different system interfaces. Different specification techniques are used in order to specify the requirements more precisely for different User’s.

## 1.2 Project Motivation:

The motivation of developing the “**Eat Right**” is that exercise is an important part of maintaining human health, helps humans prevent diseases, helps maintain a proper weight, helps us to perform our daily duties efficiently and energetically, and improves the efficiency of the body.

**Eat-Right** helps you stay on track, monitor your progress, and adjust your routine accordingly.

The key to success is setting reasonable goals, making an achievable plan, and finding activities that you enjoy and that correspond to the nature of each individual person.

As it will be one of the keys to success to reach the best physical activity for the individual through the services it will provide that take care of each person individually, as each user must provide information that is displayed by the system, such as: weight, height, body shape, whether he is thin or fat.... Etcetera.

Then he builds the system on the information explained to him. The famous system is the second appropriate food for the user, and the user must track the specifications that he must acquire or that must be burned based on a mathematical equation that the system performs to find the best and most appropriate level of it for each user.

To fully expand the human scope in our journey towards achieving health and fitness goals, it is necessary to keep pace with your development, so you need a willpower museum of required activities.

Through inclusive activism and nutrition awareness, our platform is a struggle that favors aspirations over tangible results.

Through Eat Right, users can access comprehensive support for their health and fitness goals, suggesting that with encouragement anyone can achieve human well-being.

## 1.3 Review of Related Literature and Systems:

**MyFitnessPal: “**MyFitnessPal” is a renowned fitness and nutrition tracking app. It allows users to log meals and exercises while accurately tracking calorie intake. With an extensive food database, it makes it easy for users to achieve their health goals [[1].](https://www.myfitnesspal.com/)

**Lose It: “**Lose It” is an advanced fitness and nutrition tracking app designed to help users achieve weight loss goals. Users can easily log meals using the barcode scanner and receive personalized guidance along with motivational features [[2].](https://www.loseit.com/)

**Fat Secret: “**Fat Secret” provides calorie tracking and allows users to log foods while engaging in an active community for sharing experiences. It also offers comprehensive nutritional information for foods [[3].](https://www.fatsecret.com/)

## 1.4 Problem Statement (limitation of current systems):

The current fitness and nutrition tracking applications, “**MyFitnessPal**”, “**Lose It**”, and “**Fat Secret**”, exhibit notable limitations that hinder their effectiveness in providing a comprehensive user experience.

**MyFitnessPal** struggles with delivering highly personalized nutritional guidance, especially in tailoring recommendations based on individual activity levels and specific weight management goals.

**Lose It** faces challenges in offering detailed insights into micronutrient intake and lacks features that foster a sense of community and peer support.

**Fat Secret**, while robust in calorie tracking, encounters usability issues and could benefit from a more intuitive interface.

In response to these limitations, the **Eat-Right** Website emerges as a unique solution that distinguishes itself through innovative features.

**Eat-Right** focuses in offering highly personalized nutritional guidance by incorporating advanced algorithms that consider individual activity levels, weight management goals, and dietary preferences.

Unlike its counterparts, **Eat-Right** stands out by providing comprehensive tracking of essential vitamins and minerals, facilitating a more nuanced understanding of nutritional intake.

Moreover, **Eat-Right** boasts a dynamic and motivating user interface Through these distinguishing features.

**Eat-Right** seeks to redefine the landscape of fitness and nutrition tracking applications, addressing the shortcomings of existing systems and providing users with a more tailored and motivating health and wellness experience.

## 1.5 Project Objectives:

The eat right system aims to achieve the health and physical goals of individuals, as it is based on providing a set of detailed instructions based on the information that the user will provide to the system. The site will, based on the information provided, create programs for physical, sports and nutritional activities that suit each individual.

The site also contains a group of services that meet the needs of users, including:

The site provides an easy-to-use interface for individuals, as the user creates his own personal account with ease by providing some basic information required by the site, such as first name, Email, password, and phone number.

The system will ask the user to specify the daily activity level, commensurate with the user’s nature and circumstances, by displaying 4 options that determine daily activity levels, including: activities for people of inactive determination, activities for people of moderate determination, or activities for people of high determination

* (Not Very Active, Lightly Active, Active, Very Active).

Users are asked to set their health goal, whether it is to gain weight, maintain it, or lose weight. The system also provides the user to specify the amount of weight he wants to gain or lose weekly.

Calculating calories based on the information provided by the user to the website, as the system creates calorie results that suit the user based on calculating a specific mathematical equation to reach the best result that suits the user. The system also creates nutrition facts that include the appropriate amounts of proteins, carbohydrates, and fats to achieve the user's health goals.

Through the training programs that the system provides to the user, he will be able to improve his overall athletic performance

## 1.6 Project Plan:

**Project Initiation Phase (13 Day’s):**

1. Define the project objectives, **scope**, and **motivation**
2. Identify key stakeholders, **including** User, and Admin.
3. Understand **stakeholders** needs and expectations
4. **Develop** the project plan, including the **schedule** and **resources required**.

**Research and Requirements Gathering Phase (23 Day’s):**

1. Conduct a comprehensive review of current sites and systems.
2. Analyze the current Building website process and identify areas for improvement.
3. Define functional and non-functional requirements for the web app.
4. Document and prioritize the requirements.
5. Create a use case diagram to visualize the interactions between actors and the system.
6. Obtain approval on requirements and use case diagram.

**Prototype Design Phase (11 Day’s):**

1. Create a conceptual class diagram to represent the high-level structure of the web app.
2. Create Design an Entity-Relationship (**ER**) diagram to visualize the database structure
3. Create an activity diagram to represent the flow of activities within the web app.

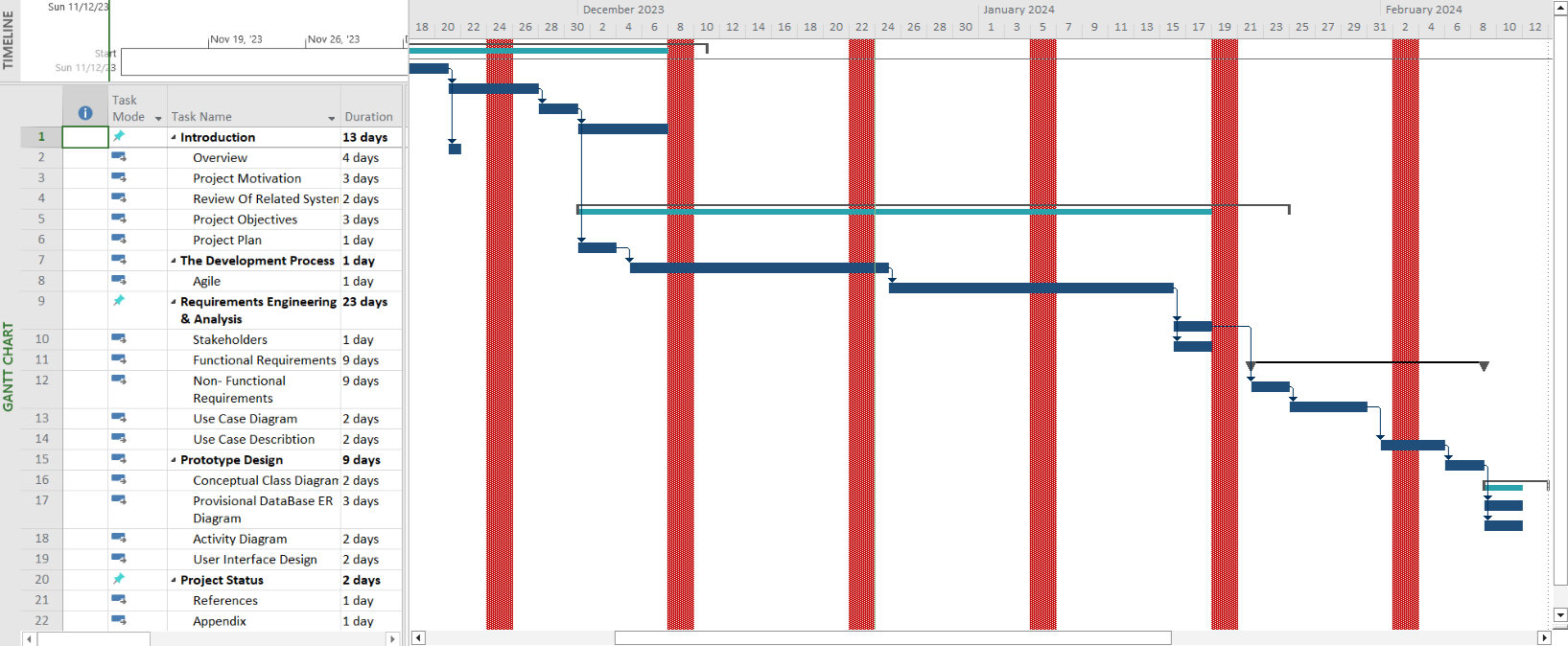
Create Utilize Figma to create the user interface (**UI**) design for the web app

Figure 1.1 Project Plan

# Chapter 2: The Development Process:

## 2.1 Agile Methodology:

This methodology was used because it enables the program team to work collaboratively on the development of the system and provides them with greater freedom of work because this methodology allows them to move between the stages of development processes, whether it is a previous or next stage, and it does not require clearly defined goals and processes at the beginning of development, unlike the waterfall methodology. It enables developers to adapt better, make the required changes, quickly detect problems, lead to faster fixes and better project control, and allow designers to work on models that use their strengths. The objectives, scope and motives of the project were determined, and then the target stakeholders, students and advisors, were identified. And also, to reach the most prominent and important functional requirements, and non-functional requirements that meet the expectations of stakeholders and commensurate with the capabilities of the system to be developed.

* Project Initiation and Planning:
* In this section, we define project feasibility, identify the project, and identify its scope, and discuss it with our supervisor to make sure that we choose a good idea to work on it.
* defining the key stakeholders, such as instructors, students, and admins.
* Create a project vision statement to outline the purpose and goals of the system.
* Conduct initial meetings with the team to gather stakeholders’ input and expectations.
* Project Analysis:
* In this phase, we will understand system needs and requirements, then we structure these requirements; we mean describing system activities and processes. Also, we make a full feasibility study about the whole project and then define the system's functionality.
* Requirements Phase:
* Identify the functional and non-functional requirements for the FIT Advisor web app.
* Break down the requirements into user stories that describe specific features or functionalities.
* Collaborate with stakeholders to refine and clarify the requirements, ensuring a shared understanding.
* Project Design: Project Design:
* The main phase activities are to identify database relations, transfer these relations to ER-Diagram that show all overall system relations.
* Use Figma to design and prototype interfaces.
* Review the user’s requirements gathered during the initiation and requirement phases.
* Use simple shapes, lines to represent various UI elements such as buttons, input fields, and menus.
* Define color schemes, typography, and other visual styles.
* Enhance the design by adding visual details, such as realistic content, images, and icons.
* Utilize Figma's prototyping capabilities to create interactive and clickable prototypes.
* Link different screens together and set up interactive elements like buttons or menus to simulate user interactions.

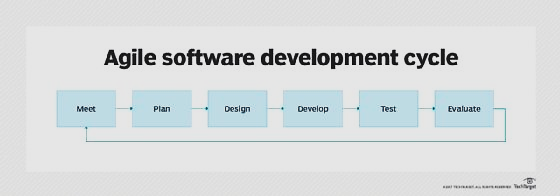


Figure 2.1 Agile software development cycle.

# Chapter 3: Requirements Engineering and Analysis:

## 3.1 Stakeholders

|  |  |
| --- | --- |
| Term | Definition |
| The User | Someone using the website services. |
| The admin | Someone who manages the website users and services. |

**Table 1.1 Definitions.**

## 3.2 Functional Requirements:

A functional requirement is a specific feature or capability that a system or product must have in order to fulfill its intended purpose or meet a particular need. Functional requirements are typically expressed in terms of inputs, processes, and outputs, and they may include specific performance characteristics or design constraints. They are a key part of the software development process, as they are used to define the functionality that the system must provide and to guide the design, development, and testing of the system.

### **3.2.1 Index Screen Before Login:**

3.2.1.1 The **“Eat Right”** Website Shall Provide the User with the ability to click on

“**Register**” Button for go to the registration Page.

3.2.1.2 The **“Eat Right”** WebsiteShall Provide the User with the ability to click on

“**Login**” Button for go to the login Page.

3.2.1.3 The **“Eat Right”** WebsiteShall Provide the User with the ability to view (Home,

Recipe, Food, Contact Us).

3.2.1.4 The **“Eat Right”** Website Shall Provide the User with the ability to click on

“**Discover Now**” button for go to the Nutrition Tracker Page.

3.2.1.5 The **“Eat Right”** WebsiteShall Provide the User with the ability to display “Our-

service” that contain the following Pages:

* (Recipes, Food, Nutrition Tracker).

### **3.2.2 Registration Screen:**

3.2.2.1 The **“Eat Right”** WebsiteShall provide the user with the ability to enter his/her full

Name in the "**Full Name**" input field.

3.2.2.2 The **“Eat Right”** Website Shall provide the user with the ability to enter his/her birthdate

in the "**Birthdate**" input field using the **MM/DD/YYYY** format.

3.2.2.3 The **“Eat Right”** Website Shall provide the user with the ability to select his/her gender

from the provided options: Male, Female.

3.2.2.4 The **“Eat Right”** Website Shall provide the user with the ability to choose his/her goal

from the provided list of options in the "**Choose your Goal**" dropdown menu.

3.2.2.5 The **“Eat Right”** Website Shall provide the user with the ability to select his/her activity

level from the provided options in the "**Choose your activity level**" dropdown menu.

3.2.2.6 The **“Eat Right”** Website Shall provide the user with the ability to enter his/her height in

the "**Height**" input field using any valid unit of measurement.

3.2.2.7 The **“Eat Right”** Website Shall provide the user with the ability to enter his/her current

weight in the "**Weight**" input field using any valid unit of measurement.

3.2.2.8 The **“Eat Right”** Website Shall provide the user with the ability to enter his/her email

Address in the "**Email**" input field.

3.2.2.9 The **“Eat Right”** Website Shall provide the user with the ability to enter a password

in the "**Password**" input field.

3.2.2.10 The **“Eat Right”** Website Shall provide the user with the ability to confirm his/her

password in the "**Confirm Password**" input field.

3.2.2.11 The **“Eat Right”** Website Shall provide the user with the ability to submit the

registration form by clicking on the "**Register Now**" button.

3.2.2.12 The **“Eat Right”** Website Shall provide the user with the ability to show error

messages

         - if the user enters invalid input, such as an invalid birthdate format, an incomplete form,

or a mismatched password.

### **3.2.3 Login Screen:**

3.2.3.1 The **“Eat Right”** WebsiteShall provide users with the ability to log into their accounts

using their email and password.

3.2.3.2 The **“Eat Right”** Website Shall provide the User with the ability to Remember the User

through a “Remember Me” option to keep the User logged in for future visits.

3.2.3.3 The **“Eat Right”** Website Shall provide the user with the ability to access a “Forgot

your password?” message. (A link to allow users to reset their password).

3.2.3.4 The **“Eat Right”** Website Shall provide the user with the ability to access "Don't have

account? sign up" link to allow new users to create an account.

**3.2.4 Users and Admin Profile Screen:**

3.2.4.1 The "Eat Right" Website Shall provide the user/admin with the ability to modify

his/her profile information.

3.2.4.1.1 The "Eat Right" Website Shall provide the user/admin with the ability to update

his/her username, phone number, first name, and birthdate.

3.2.4.2 The "Eat Right" Website Shall provide the user/admin with the ability to update his/her

Password.

3.2.4.3 The "Eat Right" Website shall provide the user/admin with the ability to include a

"Personal data" section where the user can delete his/her personal data and account.

3.2.4.3.1 The "Eat Right" Website shall provide the user/admin with the ability to a

"DELETE" button for the user to confirm the deletion of their personal data and

account.

3.2.4.3.2 The "Eat Right" Website shall provide the user/admin with the ability to require

users to enter His/her password before confirming the deletion of their personal data

and account.

### **3.2.5 Index Screen After Login:**

3.2.5.1 The **“Eat Right”** WebsiteShall Provide the User with the ability to view (Home,

Recipe, Food, Nutrition Tracker, Contact Us).

### **3.2.6 Recipes Screen:**

3.2.6.1 The **“Eat Right”** WebsiteShall Provide the User with the ability to search for recipes

by entering keywords or recipe names.

3.2.6.2 after the user select the recipe that he/she prefer the "**Eat Right”** Website shall provide

the user with the ability to show details information including (the recipe name, a brief

description, and an image) about the selected recipe.

3.2.6.3 The **“Eat Right”** WebsiteShall Provide the User with the ability to search for Another

recipe that he/she prefer.

3.2.6.4 The **“Eat Right”** Website Shall Provide the User with the ability to Show the Nutrition

facts about the selected recipe.

### **3.2.7 Food Screen:**

3.2.7.1 The **“Eat Right”** WebsiteShall Provide the User with the ability to search for foods

by entering keywords or food name.

3.2.7.2 The "**Eat Right”** Website Shall provide the user with the ability to Show the name

and calorie count for each food item.

3.2.7.3 The "**Eat Right”** Website Shall provide the user with the ability to Show Nutrition

information by clicking on “Nutrition info” button.

3.2.7.4 The "**Eat Right”** Website Shall provide the user with the ability to Show Nutrition

facts and an image for the selected food.

3.2.7.5 The **"Eat Right”** Website Shall provide the user with the ability to Show and Search

for foods from the same Category.

**3.2.8 Workout Screen:**

3.28.1 The “**Eat Right**” Website Shall provide the user with the ability to view the Workout that Listed in the Website.

### **3.2.9 Contact Us Screen:**

3.2.9.1 The "**Eat Right”** Website Shall provide the user with the ability to send message to us

by entering his/her first name and last name and e-mail address.

3.2.9.2 The "**Eat Right”** Website Shall provide the user with the ability to show our social

media and phone number and our address.

### **3.2.10 Nutrition Tracker Screen:**

**3.2.10.1 My Home Screen:**

3.2.10.1.1 The "**Eat Right**" Website Shall provide the user with the ability to display his/her

daily summary, including the total calories consumed, calories from proteins,

calories from carbohydrates, and calories from fats.

3.2.10.1.2 The “**Eat Right**” website Shall provide the user with the ability to add food to

his/her daily summary.

3.2.10.1.3 The " **Eat Right** " Website Shall provide the user with the ability to calculate and

display his/her difference between his/her goal and actual caloric intake.

3.2.10.1.4 The " **Eat Right** " website Shall provide the user with the ability to Show his /her

height, weight, target weight, gender, goals, activity level based on his /her last

goal.

3.2.10.1.5 The " **Eat Right** " website Shall provide the user with the ability to show his /her

daily energy needs from calories and protein and carbs and fats that are calculated

based on his/her gender.

**3.2.10.2 Food Diary Screen:**

3.2.10.2.1 The "**Eat Right**" Website Shall provide the user with the ability to add food items

For each meal category (breakfast, lunch, dinner and snacks).

3.2.10.2.2 The " **Eat Right** " Website Shall provide the user with the ability to search for

foodstuffs and add them to the corresponding meal category with the right quantity

and measurement unit.

3.2.10.2.3 The " **Eat Right** " Website Shall provide the user with the ability to display total

calories, carbohydrates, fat and protein for all foodstuffs added to the food diary.

3.2.10.2.4 The " **Eat Right** " Website Shall provide the user with the ability to view and

calculate the "total daily target" that must be configurable by the user and the values

of "net" must be calculated by subtracting total calories.

3.2.10.2.5 The " **Eat Right** " Website Shall provide the user with the ability to remove food

diary items from the diary table if the item is wrongly selected.

**3.2.10.3 Food Database Screen:**

3.2.10.3.1 The **“Eat Right”** WebsiteShall Provide the User with the ability to search for foods

by entering keywords or food name.

3.2.10.3.2 The "**Eat Right”** Website Shall provide the user with the ability to display the name

and calorie count for each food item.

3.2.10.3.3 The "**Eat Right”** Website Shall provide the user with the ability to Show Nutrition

information by clicking on “Nutrition info” button.

3.2.10.3.4 The "**Eat Right”** Website Shall provide the user with the ability to Show Nutrition

facts and an image for the selected food.

3.2.10.3.5 The " **Eat Right** " Website Shall provide the user with the ability to Show The food

diary entry page record the quantity, food item, weight, and meal time for each

entry.

3.2.10.3.6 The " **Eat Right** " Website Shall provide the user with the ability to add a food item

to the diary. This function should include a field for the user to input the name of

the food item and an option to specify the quantity and weight.

**3.2.10.4 Report Screen:**

3.2.10.4.1 The " **Eat Right** " Website Shall provide the user with the ability to track and

monitor nutritional intake of individuals.

3.2.10.4.2 The " **Eat Right** " Website Shall provide the user with the ability to set a goal

weight and a target the date to achieve that weight.

3.2.10.4.3 The " **Eat Right** " Website Shall provide the user with the ability to Show the

required caloric surplus per day to gain 1 kg per week, taking into account the user's

height, weight, age, gender, and activity level.

3.2.10.4.4 The " **Eat Right** " Website Shall provide the user with the ability to track his/her

progress towards his/her goal, including the number of calories consumed and

remaining for the day.

3.2.10.4.5 The " **Eat Right** " Website Shall provide the user with the ability to view his/her

nutritional tracking history and see his/her progress over time.

3.2.10.4.6 The “**Eat Right**” Website Shall provide the user with the ability to see all his/her

history in Eat right based on his/her Selected date.

**3.2.10.5 Change Goal Screen:**

3.2.10.5.1 The “**Eat Right**” website shall provide the user with the ability to enter his/her

Current weight to change his/her goals.

3.2.10.5.2 The “**Eat Right**” website shall provide the user with the ability to enter his/her

Current height to change his/her goals.

3.2.10.5.3 The “**Eat Right**” website shall provide the user with the ability to enter his/her

Current Target Weight to change his/her goals.

3.2.10.5.4 The “**Eat Right**” website shall provide the user with the ability to enter his/her

Current Activity-Level to change his/her goals.

3.2.10.5.5 The “**Eat Right**” website shall provide the user with the ability to enter his/her new

goal from “Choose your goal” to change his/her goals.

### **3.2.11 Admin Screen:**

3.2.11.1 The "**Eat Right**" Website Shall provide the admin with the ability to a secure login

functionality to access the site using his/her credentials (adminl@gmail.com in this

case).

**3.2.11.2 Food Category Management:**

3.2.11.2.1 The " **Eat Right** " Website Shall provide the admin with the ability to view all

existing food categories.

3.2.11.2.2 The "**Eat Right**" Website Shall provide the admin with the ability to add, edit, or

delete categories as needed.

**3.2.11.3 Recipe Management:**

3.2.11.3.1 The "**Eat Right**" Website Shall provide the admin with the ability to add, edit,

Delete, Display information about Recipes within each food category.

* + This should include fields for recipe name, ingredients, cooking instructions, and any relevant images.

3.2.11.3.2 The "**Eat Right**" Website Shall provide the admin with the ability to search for a

specific recipe by entering the recipe name in the search bar.

**3.2.11.4 Food Management:**

3.2.11.4.1 The "**Eat Right**" Website Shall provide the admin with the ability to add, edit,

Delete, Display information about Foods within each food category.

* + This should include fields for food name, ingredients, food category, nutrition facts, and any relevant images.

3.2.11.4.2 The "**Eat Right**" Website Shall provide the admin with the ability to search for a

specific food by entering the food name in the search bar.

**3.2.11.5 Goal Management:**

3.2.11.5.1 The "**Eat Right**" Website Shall provide the admin with the ability to display a list

of predefined goals related to weight management.

3.2.11.5.2 The "**Eat Right**" Website Shall provide the admin with the ability to search for a

specific goal by entering the goal name in the search bar.

3.2.11.5.3 The "**Eat Right**" Website Shall provide the admin with the ability to display the

goal name and its corresponding value for each goal option.

3.2.11.5.4 The "**Eat Right**" Website Shall provide the admin with the ability to edit an

existing goal by selecting the "Edit" option next to the goal they wish to modify.

3.2.11.5.5 The "**Eat Right**" Website Shall provide the admin with the ability to delete an

existing goal by selecting the "Delete" option next to the goal they wish to remove.

**3.2.11.5.6** The "**Eat Right**" Website Shall provide the admin with the ability to create a new

goal by selecting the "Create new Goal”.

3.2.11.5.6.1 The "**Eat Right** "Website shall provide the admin with the ability to select a goal

option value for each goal.

3.2.11.5.6.2 The "**Eat Right** "Website shall provide the admin with the ability to feedback

when a goal is successfully edited or deleted.

**3.2.11.6 Meal Management:**

3.2.11.2.1 The " **Eat Right** " Website Shall provide the admin with the ability to view all

existing Meals.

3.2.11.2.2 The "**Eat Right**" Website Shall provide the admin with the ability to add, edit, or

delete Meals as needed.

## 3.3 Non-Functional Requirements:

Non-functional requirements are a type of requirement that defines the quality or characteristics of a system or product, but do not specify specific functionality.

These requirements are typically used to describe how well the system performs in areas such as performance, security, scalability, usability, and maintainability.

They are also used to define constraints on the system, such as maximum response time, throughput, or availability.

Non-functional requirements are important to consider during the design and development of a system, as they can have a significant impact on the overall quality and usability of the system.

**3.3.1 Availability:**

3.3.1.1 The system shall be at least 95 percent available on weekdays between 6:00 A.M. and

midnight, and at least 99 percent available on weekdays between 3:00 P.M and 5:00

P.M.

3.3.1.2 Main functions, such as user logins and data retrieval, will keep response times within

acceptable limits, even during peak usage periods.

**3.3.2 Integrity:**

3.3.2.1 The system shall be protected against the unauthorized addition, deletion, or

modification of data.

**3.3.3 Performance:**

3.3.3.1 Webpages shall fully display in an average of 3 seconds or less over a 30

megabits/second Internet connection.

3.3.3.2 The system designed to handle a growing user base without a significant drop in

performance.

**3.3.4 Reliability:**

3.3.4.1 Mean Time Between Failure (MTBF) once a year or, 5000 hours of operation.

**3.3.5 Security:**

3.3.5.1The system must use encryption protocols (HTTPS) to secure data transfer between users

and the server.

**3.3.6 Usability:**

3.3.6.1The user interface should be intuitive and easy to use, requiring minimal training for

new users.

**3.3.7 Efficiency:**

**3.3.7.1 Response Time Efficiency:** The System should aim for quick response times when users

interact with the platform Whether it's retrieving personalized meal plans or displaying

nutritional information, efficient response times contribute to a positive user experience.

**3.3.7.2 Data Processing Efficiency:** The System should handle user data efficiently,

minimizing processing time for tasks such as data input, retrieval, and analysis. This is

crucial for providing real-time feedback and recommendations.

**3.3.7.3 Algorithmic Efficiency:** The system's algorithms, especially those related to

recommending suitable food options and calculating nutritional requirements, should be

optimized for speed and accuracy to ensure efficient processing of user data.

**3.3.8 Maintainability:**

3.3.8.1 In case of a failure that leads to a system outage, the Mean time to repair (**MTTR**) should

not take more than 30 minutes.

**3.3.9 Scalability:**

3.3.9.1 The website shall be able to handle a page-view growth rate of 35 percent per quarter

for at least two years without user perceptible performance degradation.

**3.3.10 Reusability:**

3.3.10.1 At least 30 percent of the website architecture shall be reused from the approved

reference architectures.

## 3.4 Process Requirements:

3.4.1 The system must use a specified database management system.

## 3.5 Use Case Diagram:

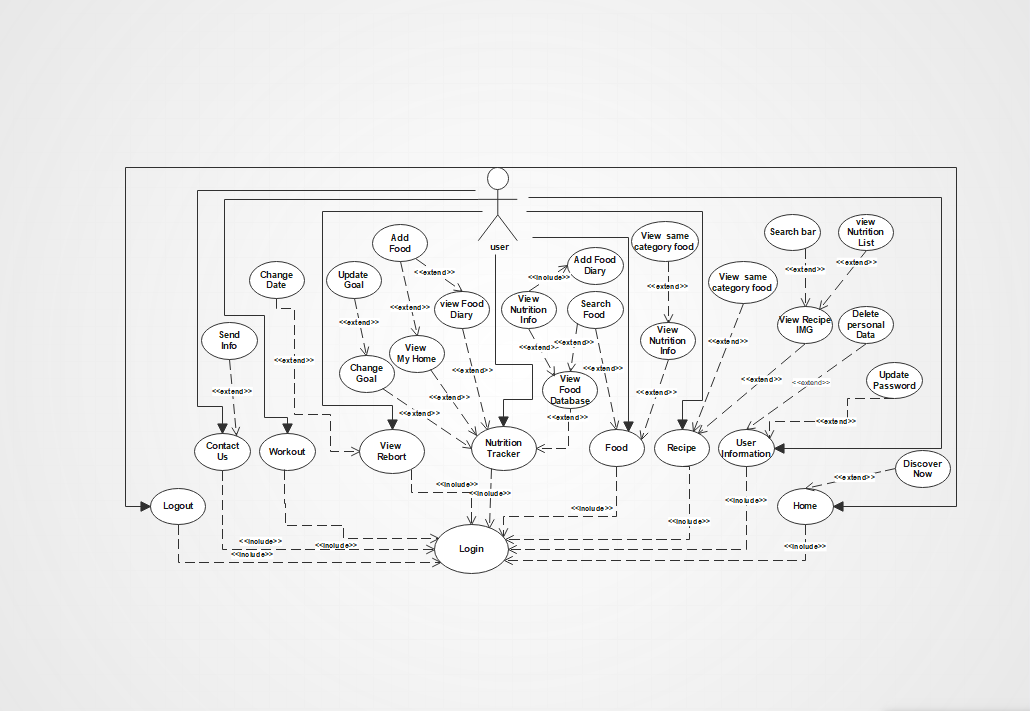
Use case diagrams model the functionality of a system using actors and use cases.

Use cases are a set of actions, services, and functions that the system needs to perform, it demonstrates the different ways that a user might interact with a system.

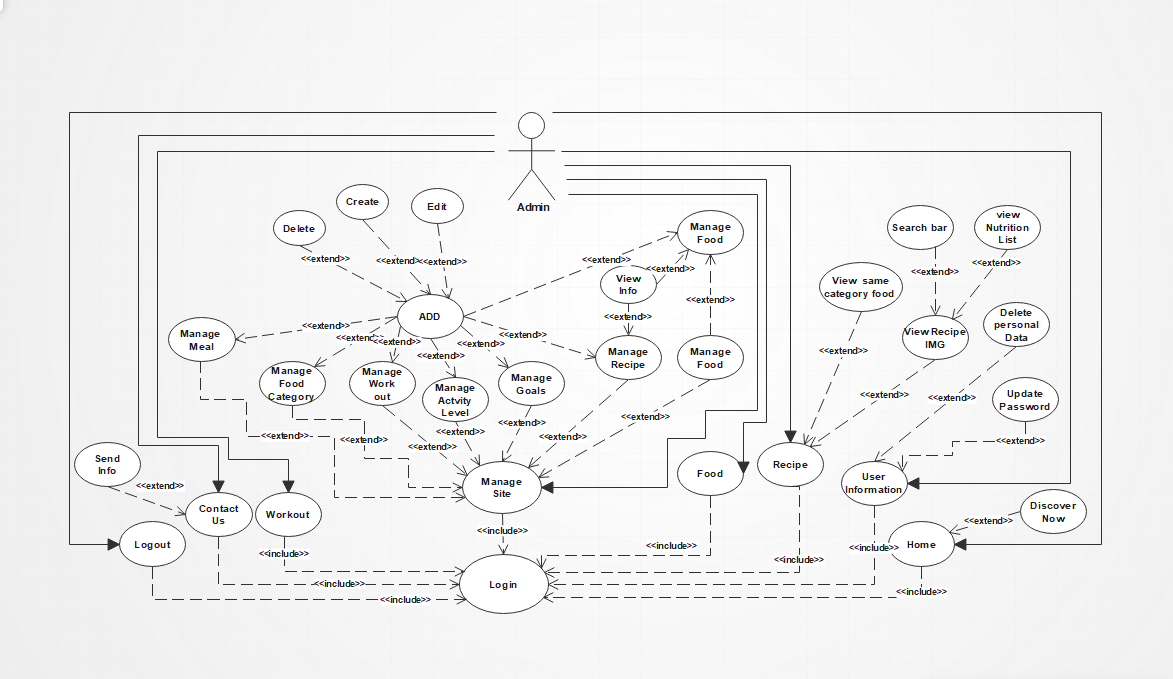
During the requirements analysis process, we found two actors for the proposed system namely the user and the admin, each of them has a set of services that the proposed system should provide.

However, the following figures represent the use case diagrams for the proposed system.

**3.5.1 User Use Case:**

****

**3.5.2 Admin Use Case:**



## 3.6 Use Cases Descriptions:

Every use case should be described using use case template that should contain a brief description, flow of events - basic flows and alternative flows, pre-conditions and post-conditions.

The use cases descriptions of the proposed system are represented in the following structural tables.

Table 6 User, Admin login:

|  |  |
| --- | --- |
| ID: | UC-2.1 |
| Title: | Login |
| Description: | It allows for Users to access to the website by entering correct email and password |
| Primary Actor: | User, Admin |
| Preconditions: | User should be registered to the Website |
| Post conditions: | User, Admin login successfully |
| Main success scenario: | 1. User/admin selects "Login" from the index screen. 2. System displays the login page. 3. User/admin fills email and password 4. User/admin clicks "Login" button. 5. Dashboard screen display to the user/admin. |
| Extensions: | 3a email and password incorrect  3a1. Error message appears email and password don’t match  4b Validate admin or user  4b1. If the User is confirmed, display user dashboard page.  4b2.If the Admin is confirmed, display admin dashboard page. |

Table 7 User Registration:

|  |  |
| --- | --- |
| ID: | UC-3.1 |
| Title: | Register |
| Description: | It allows the user to register account by entering his/her information (full name, gender, password, confirm password, Choose Goal, Activity Level, email, birthdate, Height, Weight and Target Weight) |
| Primary Actor: | User |
| Preconditions: | none |
| Post conditions: | User registered successfully |
| Main success scenario: | 1. User select "Register" from the index screen.  2. User fill out the information (full name, gender, password, confirm password, Choose Goal, Activity Level, email, birthdate, Height, Weight and Target Weight).  3.Click on Register Now button. |
| Extensions: | 2a. Validate password and confirm password.   * 2a1. Error message appear if the passwords don’t match * 2a2. Error message appear If the password didn’t contain at least 1 upper letter,1 lower letter, 1 character as {+, -, \*, /, @} |

Table 8 User Search for Recipes:

|  |  |
| --- | --- |
| ID | UC-2 |
| title | Search Recipes |
| Description | Find recipes based on search criteria |
| Primary Actor | User |
| Precondition | Recipes & Food screen is displayed |
| Post Condition | Recipes matching search criteria are displayed |
| Main Success Scenario | 1. User enters search for Recipes item in search box    2. User selects "Search Recipes" button    3. System displays list of recipes matching search criteria |
| Extension | * If there is no recipes match search criteria, system displays message indicating no results found |

Table 9 Admin Add Food:

|  |  |
| --- | --- |
| ID | UC 3 |
| title | Add Food. |
| Description | The Admin adds new Food. |
| Primary Actor | Admin |
| Precondition | The Admin is logged and on admin screen. |
| Post Condition | The recipe is added to the system and can be viewed by all Users. |
| Main Success Scenario | 1. Admin Entered to “Admin Site”. 2. Admin Selects Recipes Page. 3. Admin Clicks on “Create New Recipe” Button 4. The Admin enters the recipe details and clicks the “Add Recipe " button. |
| Extension | 2.a - If the added Recipe is already added to the data base the website shouldn’t added the new one to our data base  2.a2 - if no recipes match the search criteria, the system will display a message indicating this |

# Chapter 4: Architecture and Design:

## 4.1 Software Architecture:

The software architecture we used is based on **MVC** (Model-View-Controller) which is a pattern in software design commonly used to implement user interfaces, data, and controlling logic. It emphasizes a separation between the software's business logic and display.

This "separation of concerns" provides for a better division of labor and improved maintenance.

The three parts of the MVC software-design pattern can be described as follows:

1. Model: Manages data and business logic.
2. View: Handles layout and display.
3. Controller: Routes commands to the model and view parts.

In the early days of the Web, **MVC** architecture was mostly implemented on the server-side, with the client requesting updates via forms or links, and receiving updated views back to display in the browser.

However, these days, more of the logic is pushed to the client with the advent of client-side data stores, and XMLHttpRequest allowing partial page updates as required.

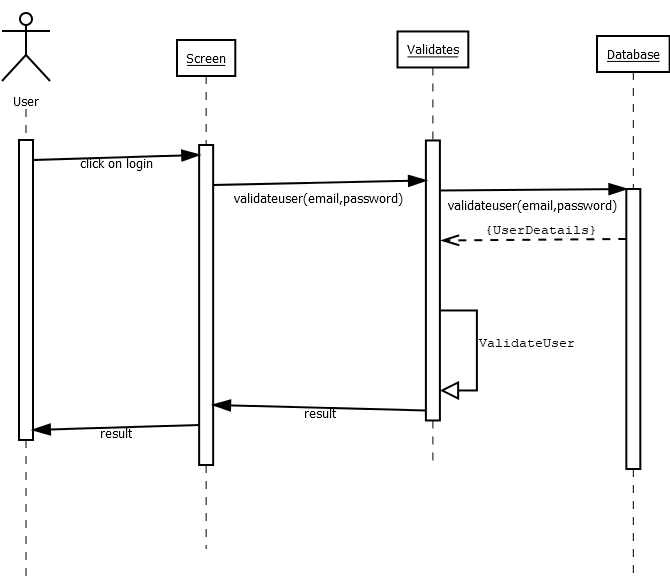
## 4.2 Software Design:

### **4.2.1 Sequence Diagrams:**

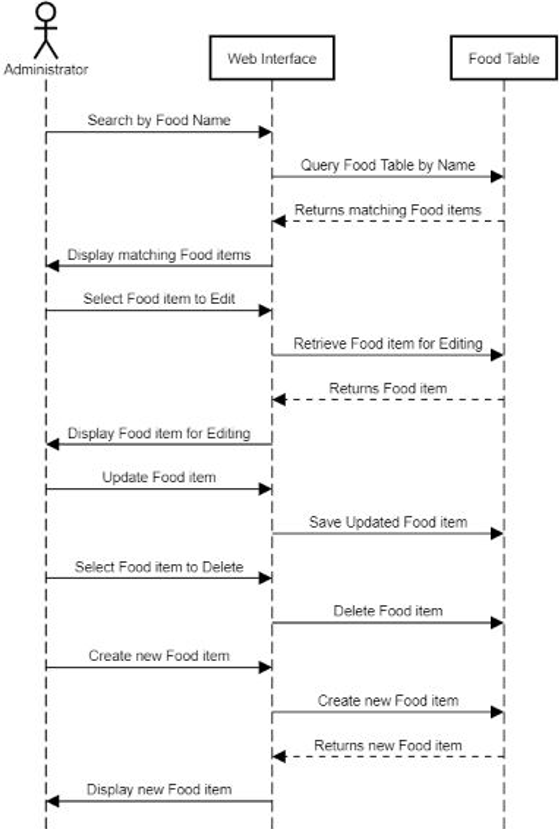
Sequence diagrams are used to model the interactions between objects in a system and the interactions between the objects themselves. Sequence diagram describes an interaction by focusing on the sequence of messages that are exchanged, along with their corresponding occurrence specifications on the lifelines.

Below are the most important sequence diagrams in **“Eat Right”** Website that describe the interaction between the system and objects to perform:

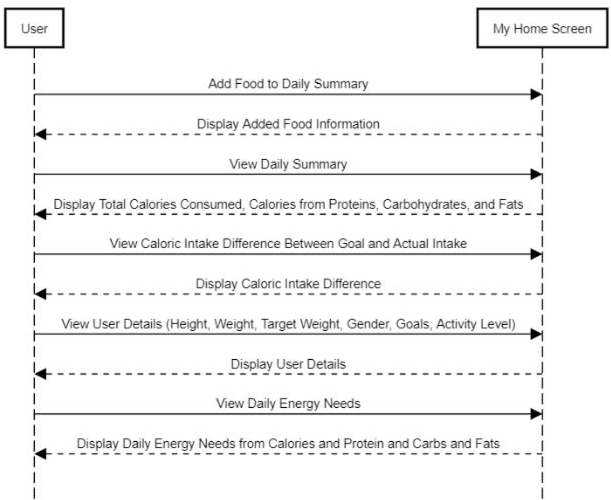
**1. Login.**



**2. Food Management.**

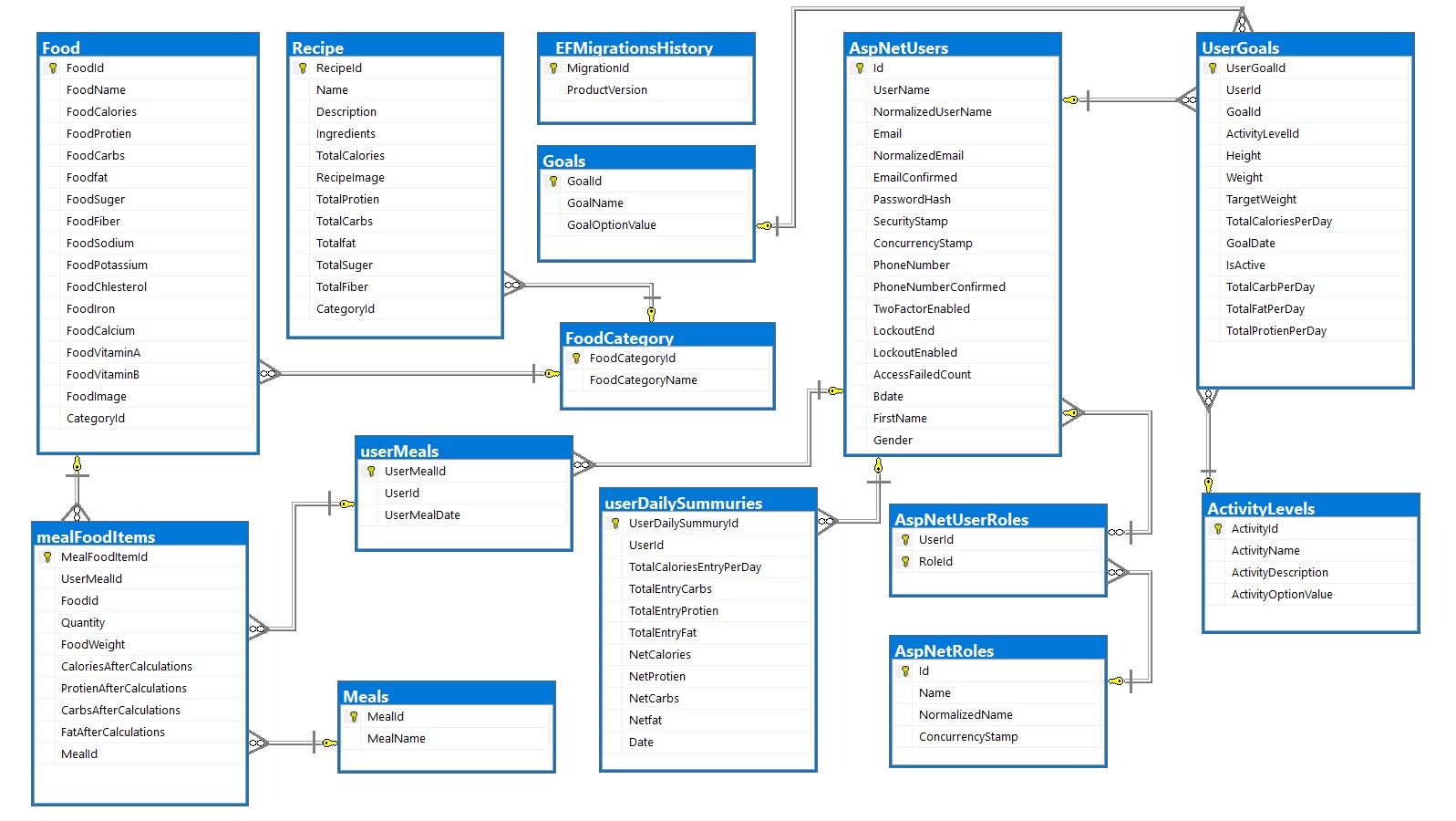


**3.My Home Tracker:**



### **4.2.2 Class Diagram:**

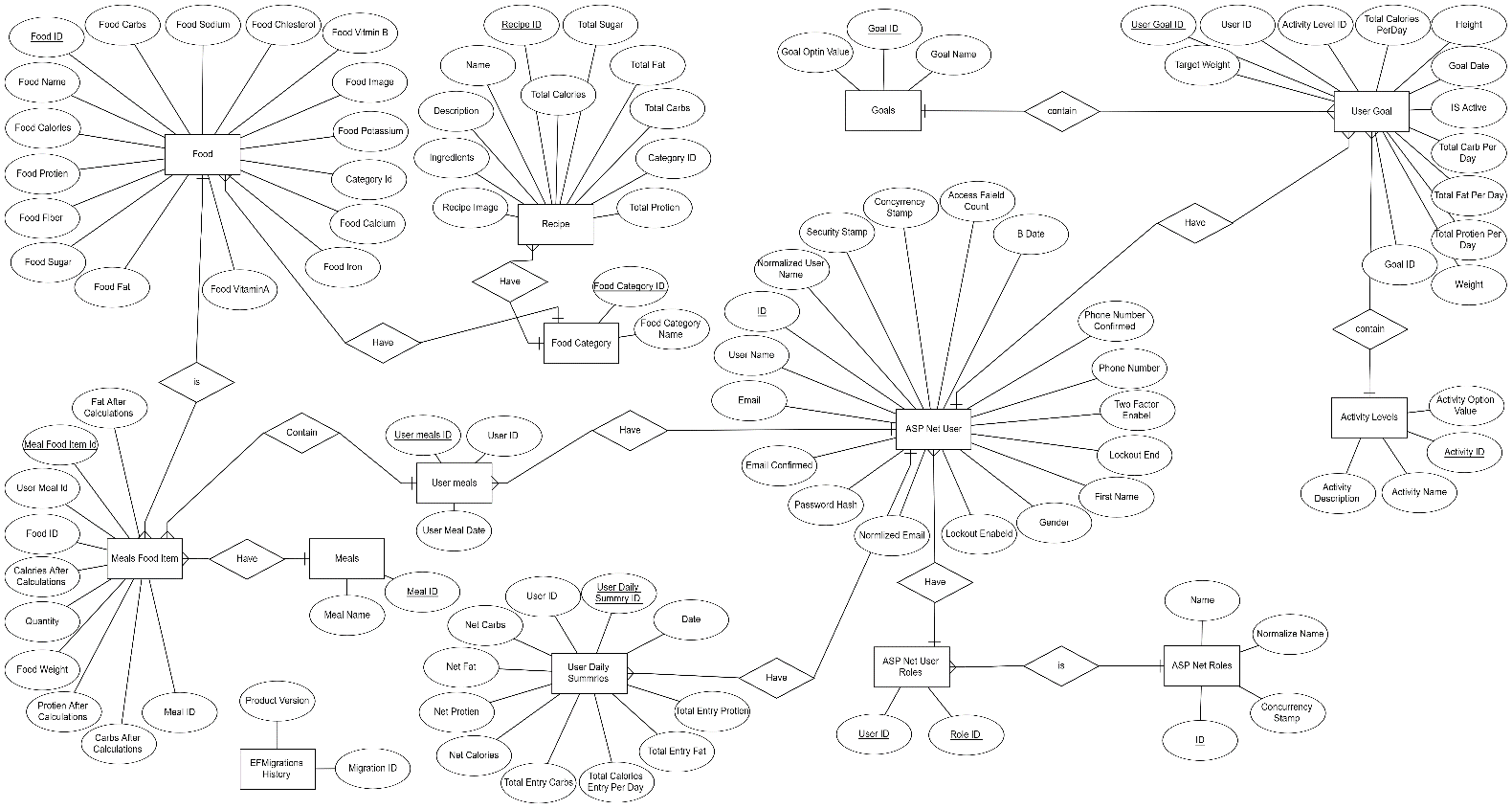
In software engineering, a class diagram is a type of static structure diagram that describes the structure of a system by showing the system's classes, their attributes, operations (or methods), and the relationships among objects.



### **4.2.3 ER Diagram:**

An Entity Relationship (ER) Diagram is a type of flowchart that illustrates how “entities” such as people, objects or concepts relate to each other within a system. ER Diagrams are most often used to design or debug relational databases in the fields of software engineering, business information systems, education and research.

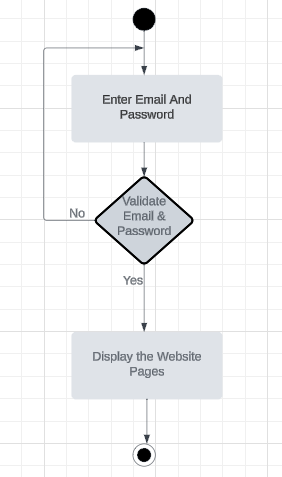
Also known as ERDs or ER Models, they use a defined set of symbols such as rectangles, diamonds, ovals and connecting lines to depict the interconnectedness of entities, relationships and their attributes. They mirror grammatical structure, with entities as nouns and relationships as verbs.



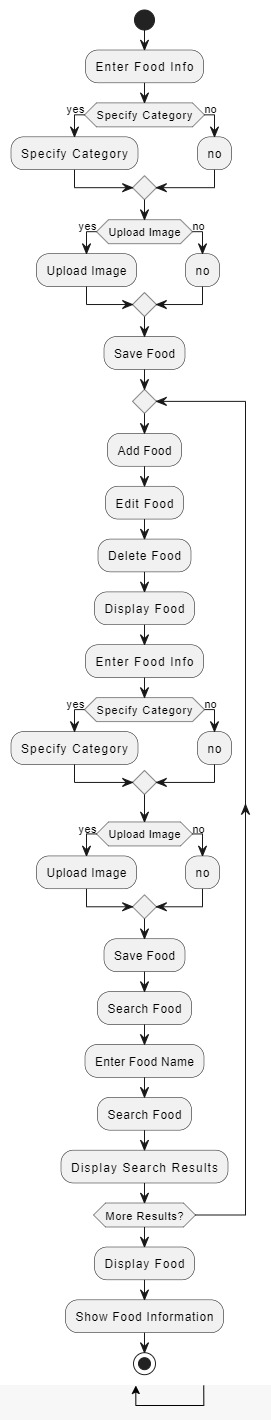
### **Activity Diagram:**

Activity Diagrams describe how activities are coordinated to provide a service which can be at different levels of abstraction. Typically, an event needs to be achieved by some operations, particularly where the operation is intended to achieve a number of different things that require coordination, or how the events in a single use case relate to one another, in particular, use cases where activities may overlap and require coordination. It is also suitable for modeling how a collection of use cases coordinates to represent business workflows.

**1. Login.**



**2. Food Management.**



**3. My Home Tracker Management.**



## 4.3 User Interface Design:

# Chapter 5: System Implementation:

## 5.1 Overview:

The system deployment, installation, and transition into an operational system typically involves several major tasks and components. **These include:**

1. **Planning and preparation:** This involves identifying the resources needed for the implementation, such as hardware, software, facilities, materials, and personnel, and creating a detailed plan for the deployment and installation of the system.
2. **Installation and configuration:** This involve physically installing the system, including any hardware and software components, and configuring the system to meet the specific requirements of the organization.
3. **Testing and validation:** This involves testing the system to ensure that it is functioning correctly and meets the requirements of the organization.
4. **Training and documentation:** This involves training users on how to use the system and creating documentation that explains how to operate and maintain the system.
5. **Go-live and support:** This involves transitioning the system into an operational state, providing ongoing support and maintenance, and monitoring the system to ensure that it continues to meet the organization's needs.

The resources required to support the implementation effort will vary depending on the specific system being deployed and the organization's needs. Factors such as the size of the organization, the complexity of the system, and the number of users will all impact the resources required.

It is important to have site-specific implementation requirements in place to ensure that the system is deployed and installed correctly.

## 5.2 Tools and Technologies Used:

**Front-End:**

1.Html.

2.Css.

3.JavaScript.

**Back-End:**

1.Asp.new MVC.

**Another tool**

1.Identify.

2.EF-Core (Code First).

EF-Core (Code First):

Entity Framework introduced the Code-First approach with Entity Framework 4.1. Code-First is mainly useful in [Domain Driven Design](http://msdn.microsoft.com/en-us/magazine/dd419654.aspx). In the Code-First approach, you focus on the domain of your application and start creating classes for your domain entity rather than design your database first and then create the classes which match your database design. The following figure illustrates the code-first approach.



Figure 3 EF-Core (Code First).

As you can see in the above figure, EF API will create the database based on your domain classes and configuration. This means you need to start coding first in C# or VB.NET and then EF will create the database from your code.

To do code first approach there are a few steps will do:

1. Adding Migrations (add-migration):

After defining your domain classes in the Code-First approach, you use the add-migration command to create a migration. A migration is essentially a snapshot of the changes to your domain model. It captures the differences between your current model and the desired state.



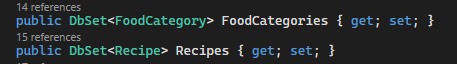
1. Updating the Database (update-database):

Once a migration is created, the update-database command is used to apply those changes to the actual database. It's like executing the steps outlined in the migration to synchronize your database schema with your application's domain model.



1. Using DB Set in DB Context:

In your DB Context class, DB Set represents a set of entities that can be queried and saved to the database. Each DB Set corresponds to a table in the database, acting as a reportable representation of your domain entities



1. Final result will be Two tables in data base:

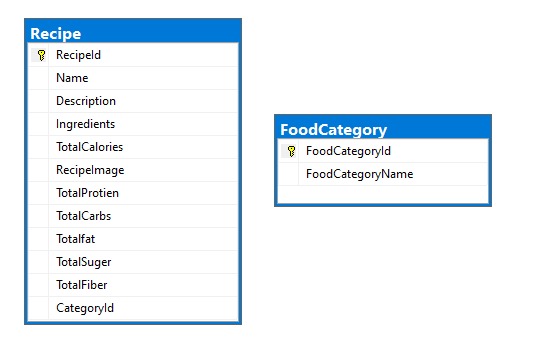


Figure 4 EF-Core (Code First)

Asp.net Identity:

ASP.NET Identity is a membership system provided by Microsoft for building authentication and authorization functionality into ASP.NET applications. It is a robust framework that simplifies user management, authentication, and authorization processes

Key Features and Capabilities:

1. User Management: ASP.NET Identity facilitates the creation, modification, and deletion of user accounts. It includes features for managing user profiles, passwords, and account recovery.
2. Authentication: It supports various authentication methods, including username/password, social logins (OAuth), and multi-factor authentication, providing flexibility for developers and end-users.
3. Authorization: The framework allows developers to define and enforce roles and claims, enabling fine-grained access control within the application.
4. Extensibility: ASP.NET Identity is highly extensible, allowing developers to customize user data, authentication mechanisms, and storage providers to meet specific application requirements.
5. Integration with ASP.NET Frameworks: Seamlessly integrates with ASP.NET Core and ASP.NET MVC, making it a natural choice for building secure web applications on the Microsoft technology stack.

Benefits:

1. Security: ASP.NET Identity provides secure password hashing, protection against common attacks like Cross-Site Scripting (XSS), and supports industry-standard security practices.
2. Flexibility: Developers can choose from various authentication methods and customize user-related functionalities according to the application's needs.
3. Consistent APIs: Offers a consistent programming interface across different ASP.NET frameworks, making it easier for developers to work with user authentication and authorization in different types of applications.
4. Storage Agnosticism: ASP.NET Identity supports various storage providers, allowing developers to choose between using a SQL Server database, Azure Table Storage, or other data stores.
5. Integration with External Logins: Supports easy integration with external identity providers (such as Google, Facebook, Twitter) for a seamless and secure login experience.

Some of tables that adp.net Identity will create:

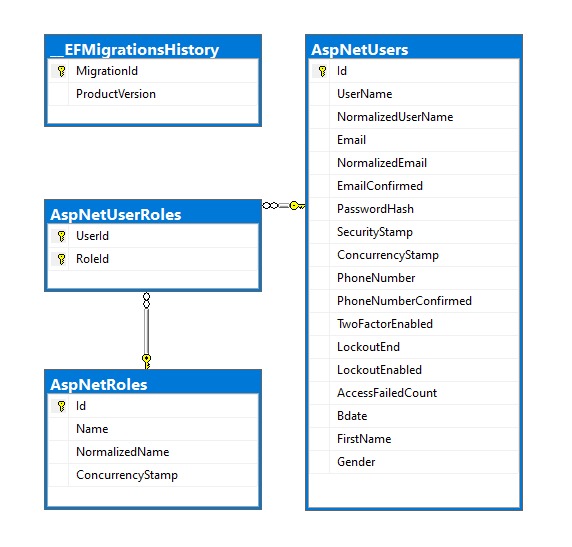


Figure 5 Asp.net Identify: We can see the relationship between all these tables that identity was create.

## 5.3 Important Code Fragments:

**5.3.1 Food Diary Screen:**

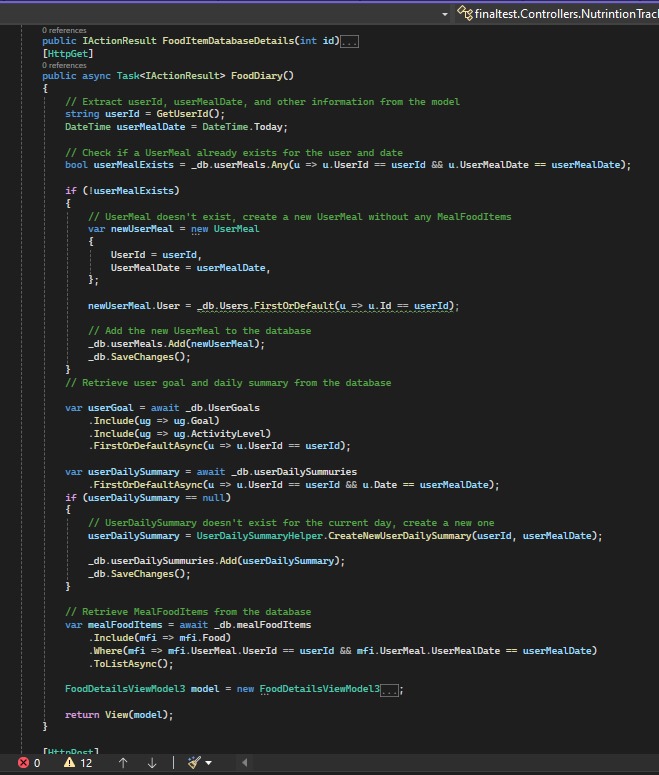
****

Figure 6 Food Diary:

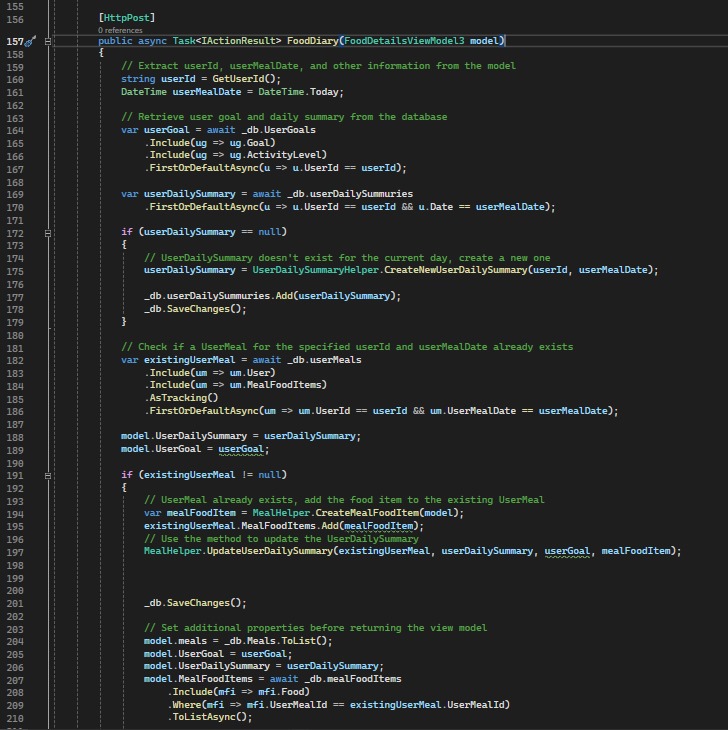


Figure 5.5 Food Diary:

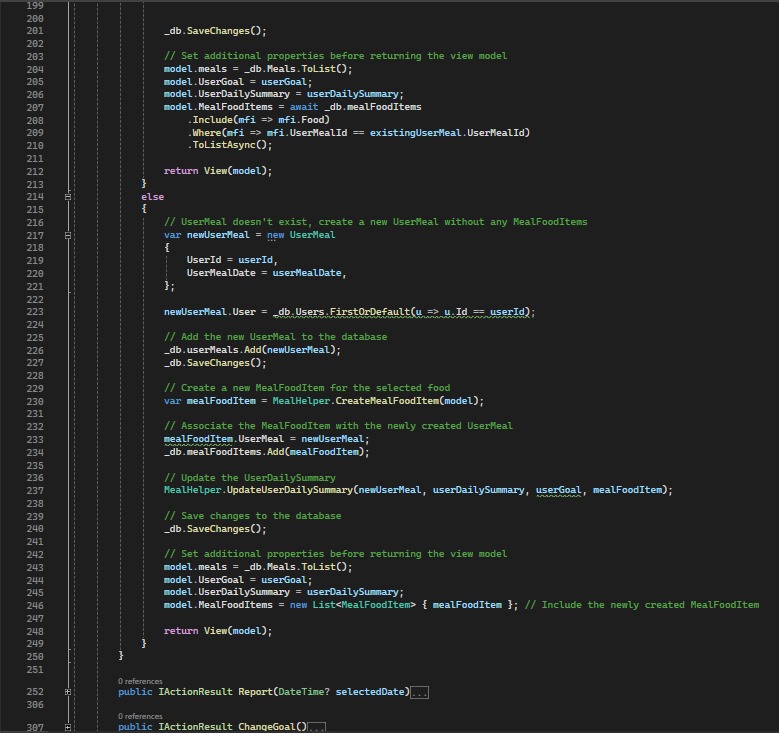


Figure 7 Food Diary:

**5.3.1 Registration Screen:**

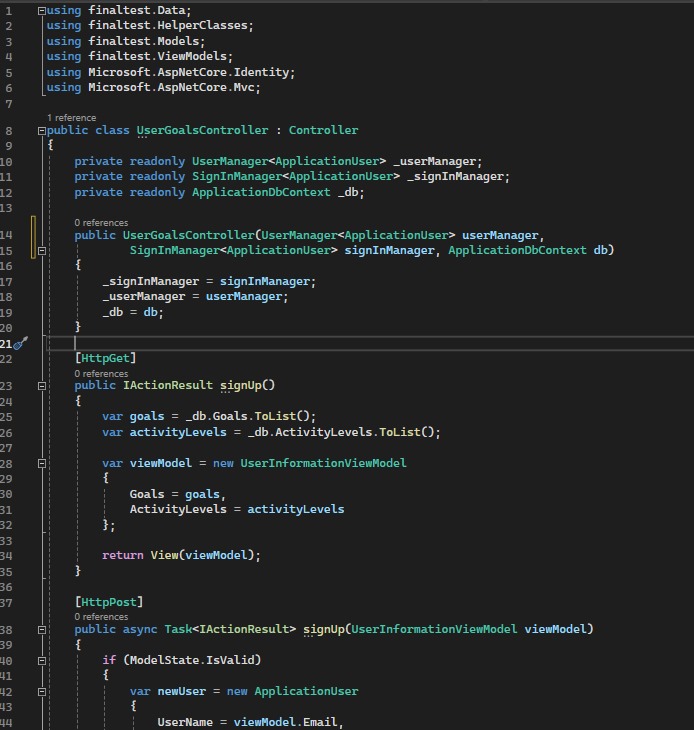


Figure 8 Registration Screen:

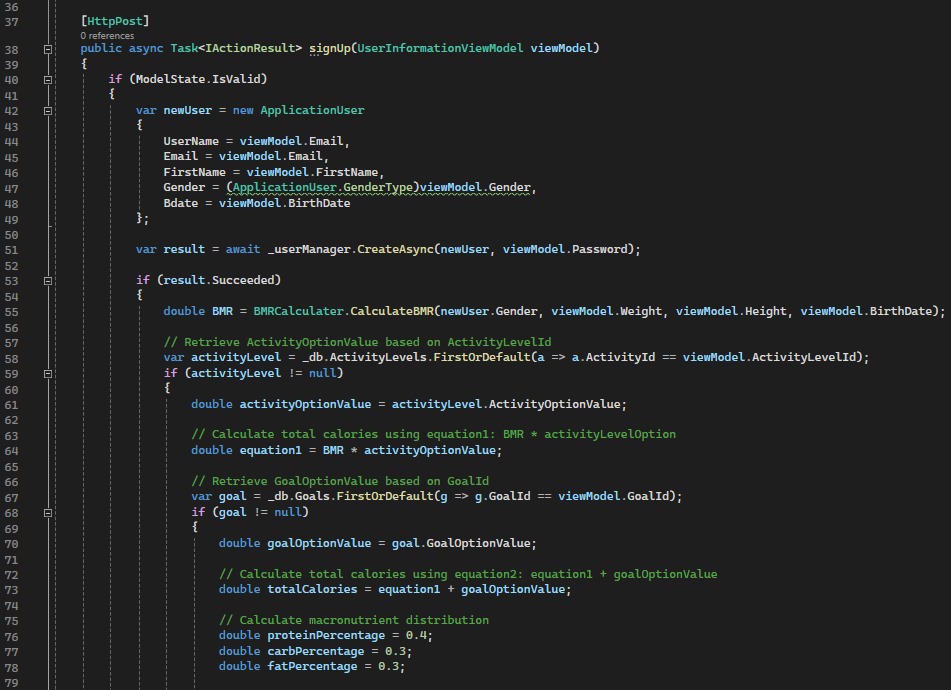


Figure 9 Registration Screen:

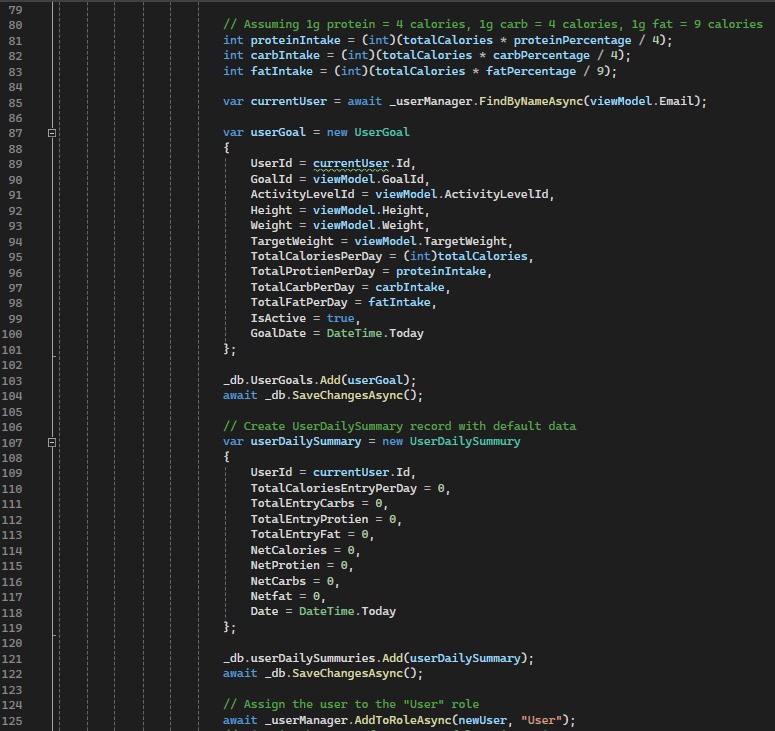


Figure 10 Registration Screen:

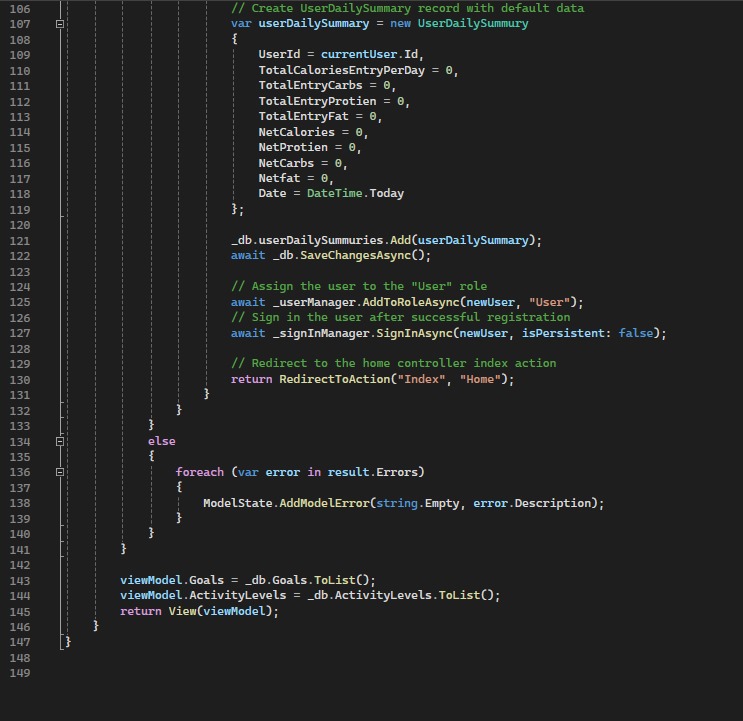


Figure 11 Registration Screen:

# Chapter 6: Testing and Validation:

This section describes the scope, approach, resources and schedule of intended test activities.  
It identifies amongst others test items, the features to be tested, test coverage, the test environment, and the test design techniques.

## 6.1 Black-Box:

Black-box testing is a method of software testing that examines the functionality of The Website without peering into its internal structures or workings.

**Table 10. Black Box Test Table:**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Test Case ID** | **Test Scenario** | **Test Steps** | **Test Data** | **Expected Results** | **Actual Results** | **Pass/ Fail** |
| 01 | Check user Login with valid Data | 1. Enter email. 2. Enter Password. 3. Click login. | email = Test@gmail.com  Password = Test@123 | User should Login into the website | As Expected. | Pass |
| 02 | Check user login with invalid data | 1.Enter email.  2. Enter password.  3.Click login. | email = Test@gmail.com  Password =  test@123 | User shouldn’t Login into the website | As Expected. | Pass |
| 03 | Check user target weight is in range | 1.Enter Height.  2.Enter Target weight= | height = 180  Target weight = 90 | System shall Show validation and reject the value | As Expected. | Pass |
| 04 | Check user target weight is in range | 1.Enter Height.  2.Enter Target weight= | height = 180  Target weight = 70 | System shall Show accept the value | As Expected. | Pass |
| 05 | User change  His/her goal  With data = current data | 1.select exactly the same data that equal exactly the current data | Current weight =70  Current goal= lose 1 kg etc.  new weight =70  new goal= lose 1kg | System shall return user to change goal page with message “There is no changes” | As Expected. | Pass |
| 06 | Gust user want to access nutrition tracker page | Click on nutrition tracker in navigation bar |  | System shall take gust to login page to login or register | As  Expected | Pass |

# Chapter 7: Conclusion and Future Work:

## 7.1 Conclusion:

The "Eat Right" website is designed to help users maintain a healthy lifestyle by providing them with access to nutritious recipes, food information, and a nutrition tracker. The website offers a user-friendly interface with features such as recipe search, nutrition facts display, and food diary management. The registration and login screens ensure user data privacy and security. The contact us screen allows users to reach out for further assistance or inquiries. The website aims to provide personalized services by considering user information such as weight, height, body shape, and activity level.

To implement this as a web application we used ASP.NET as the Technology. ASP.NET has several advantages such as enhanced performance, scalability, built-in security and simplicity.

## 7.2 Future Work:

The future of this project is already discussed, and these are the new features that we plan to implement going ahead with project.

As for future work, the project team can consider implementing additional features such as The integrating in a feature to track physical activities and exercises would provide users with a more comprehensive platform to monitor their health and fitness goals.

Secondly, incorporating artificial intelligence or machine learning algorithms to provide personalized meal and exercise plans based on user data and preferences can be beneficial.

Thirdly, adding a community feature where users can share their progress, recipes, and tips would foster a supportive environment and motivate users to stick to their health goals. Lastly, integrating the website with wearable devices such as smartwatches and fitness trackers would enable automatic data syncing and provide more accurate tracking.

# References:

1-BMR formula for men and women: