Critical Design Review V.2.0
12-6-2023

Waste Watcher Built for the Senator George J. Mitchell Center for Sustainability Solutions by:

Sustainable Waste Solutions (Current Team)

- Christian Silva
- Jackson Cyr
- Kevin Bretthauer
- Callen Shaeffer
- Caiden Emerson

Kayak Development Solutions (Past Team)

- Declan Brinn
- Chase Pisone
- Gavin Palazzo
- Finn Jacobs
- Levi Sturtevant

Table of Contents

1. Introduction

Waste Watchers is a food waste tracking application for IOS and Android being developed for the George J. Mitchell Center for Sustainability by Sustainable Waste Solutions in partial fulfillment of a B.S. at the University of Maine.

One of the worlds most challenging issues is the shear amount of food that is wasted each year by humans. The importance of this app is to aid in the reduction of food waste. We realize the importance of this world issue. Given the major economic, social, environmental/climate benifits that come with ending food waste. Our app as a tool that encourages and make the applications users aware of the food the waste has been proven to help reduce a person wasting food in their life. With this app we are aiming to give each individual user the ability to easily track, measure, and reduce their annual food waste. Also providing the user with helpful information to aid in learning more about this global issue, and in turn how the user can either avoid buying excess food or dispose of it in less wasteful ways. This is all in service of Maine's goal to reduce food waste by 50% by 2030.

2. Required Software Project Elements

Backend - Firebase

Google Firebase will be used for several components of the backend of our application including login functionality, database management, etc.

Hosting and Storage: Google's Firebase has a service called Firestore which allows users to securely upload and store documents. Our application will store user information along with data related to their food waste in a NoSQL database through Firestore.

Analytics: Firebase provides user behavior tracking and other insights that will allow us to gain an understanding of how our users interact with the application and tailor it to suit them.

Frontend - React Native

Our application will have a responsive and clear user interface with a focus on usability for all. The front end will allow users to navigate through the application such as account creation, waste tracking, statistics viewing, and leaderboard viewing.

The home page view will present the user with an option to input food waste, as well as some simple metrics such as recent weight and approximate cost of the food wasted within the last week.

There will be a view labeled 'learning' that will show the user helpful information about preventing food waste and how to lessen its impact. First, there will be an interactable map the user can use to locate the nearest food banks, farms, or other locations that food can be donated to. This view will also show the hierarchy of food waste which ranks the ways food can be used, as well as tips for reducing food waste of different categories.

Gamification

A major concern with the creation of our application is that users may not be incentivized to continue to track their food waste continuously, so we are implementing several gamification elements to the application in order to maintain user engagement and encourage consistent food waste tracking.

Friend leaderboard: Users will be able to add friends and view the streaks of their friends in a leaderboard view, encouraging them to continue tracking food waste to beat their friends.

Friend Quests: Similar to language-learning app Duolingo, our application will contain a feature that allows users to invite friends to work together towards a common goal requiring engagement from both users. This type of collaborative challenge increases the likelihood of each individual consistently tracking their food waste.

Badges: In order to keep users feeling like they are working towards something, our application will keep track of information like tracking streaks and award badges to users at certain milestones.

3. Purpose and Risks of Project

The main risks of this project come in two forms, the first being technical problems created by the complexity of backend development introduced when attempting to add database functionality to the application. This is the problem that halted the development of the application last year, and the reason the project was handed to us to continue. To combat this we have elected to move forward with a much simpler backend suite, Google Firebase, which the team has some experience with in the past. The second major risk is that the application will not be capable of attracting enough of a user base to have an effect on food waste production in the

state of Maine. To effectively deal with this we have looked to other apps that attempt to get users to maintain healthy, but sometimes arduous habits and taken design inspiration from them. This has resulted in the idea of a “gamification” of the app, allowing users to gain badges and compete with others to encourage them to track waste daily. Due to the backend of the application being firebase, we outsource most of our risk relating to user data to this 3rd party, where we are not creating our own database and server hosting. Where we do need to ensure the data pipeline is secure and is in compliance with information security standards related to Firebases policys.

4. Objectives of Project

The objective of Waste Watcher is to encourage users to reduce the amount of food they are wasting by increasing the awareness of the amount, type, and location of where the waste occurs. In order to increase awareness the app will display the users statistics and trends on both the amount of food wasted and the food cost to put in perspective the total waste someone or something is generating as well as the total cost of the waste. The users will be encouraged to invite and challenge friends to allow other users to become more aware of their waste habits. The app also aims to gamify the tracking of food waste to captivate users and encourage them to regularly keep track of their food waste. We will take an iterative approach to the development of Waste Watchers and we will use GitHub as a tool for our version control and CI/CD. We will also be using Google Firebase to store and update data.

5. System Level Requirements and Descriptions

To properly demonstrate that we have an understanding of the overall project design and implementation this is shown in our SRS 2.2 revisions and our UIDD 2.1 We also had to make some changes to our overall design and product to ensure the best user experience and fulfill the functionality requirements that are necessary for a successful app. With the revisions of the SRS that we made, it was clear that there were also some changes to the UIDD that needed to be made, so we made those changes. To illustrate the design of our system we created a sitemap *Figure 5.1*, we created a sitemap because our project is an application and we had learned that sitemaps are extremely useful in web/app development, this shows a full breakdown of the application it is a unification of SRS, SDD, and UIDD, allowing us to easily view the full scope of our project. there is a clear system data flow where the app utilizes the database and fulfills each of the document requirements, it is made up of many use cases.

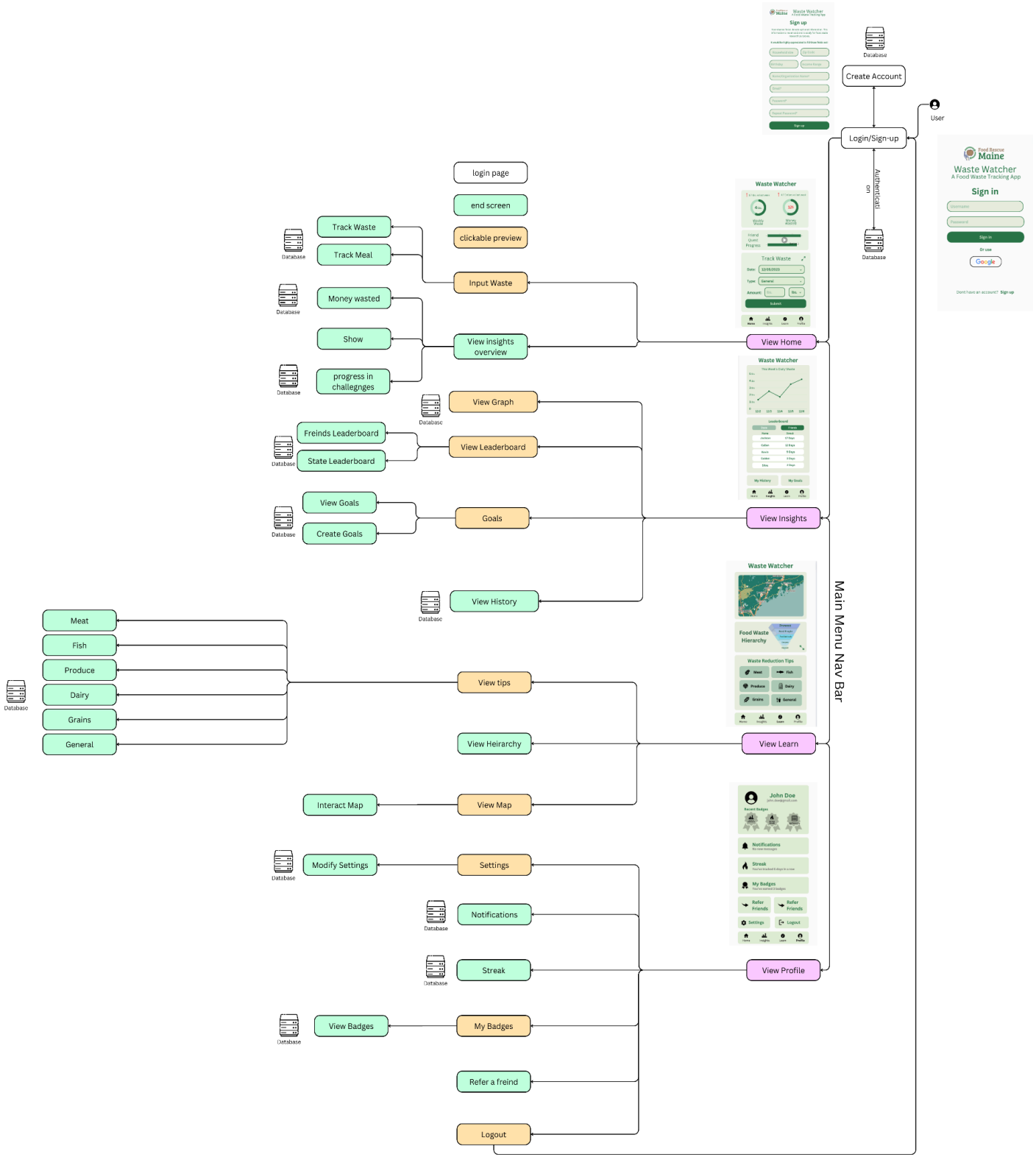


Figure 5.1 Sitemap

6. Completion Criteria

To complete our project, we must deliver a deployed and usable version of the Waste Watcher app on at least one mobile app store. The app must allow users to create accounts and log in through Google Sign-in, allow them to track food waste, and engage with the gamification aspects of our project such as friend challenges, receive badges, etc. This application must securely store all information through Google Firestore.

To be able to call the project completed, several requirements have been drafted into a matrix format, and all of these requirements must be met to deliver the project and call it finalized. These matrices, separated into functional and non-functional requirements are as follows:

Non-Functional Requirements		
Number	**Priority**	**Description**
NFR-01	5	The system shall work cross-platform between IOS and Android devices.
NFR-02	4	The system shall be able to support 5000 concurrent users.
NFR-03	4	The system shall be able to query any user data from a database within 2 seconds 90% of the time.
NFR-04	4	The system shall store user information entered into the UI in the database within 2 seconds 90% of the time.
NFR-05	4	The system shall be able to process touchscreen input from the user within 2 seconds 90% of the time.
NFR-06	5	The system shall adhere to the US Privacy Act of 1974.
NFR-07	5	The system shall implement encryption for storing user locations.
NFR-08	5	The system shall implement encryption for storing demographic information.
NFR-09	3	The system shall generate food waste statistics with data collected within the last week.
NFR-10	5	The system shall generate food waste statistics within 3 seconds of the user navigating to the statistics page of the application 90% of the time.

NFR-11	4	The system shall undergo Non-Functional Requirement (NFR) testing to ensure compliance with all specified NFRs.

		Functional Requirements
Number	**Priority**	**Description**
FR-01	5	The system shall allow the user to enter daily food waste information (the amount by weight or volume, the type of food by category, the cause of waste, in-home or out-of-home, and the meal eaten/BLD, date eaten into UI.)
FR-02	5	The system shall be able to request information about the user such as username, zip code, household size, user gender, age, and income range.
FR-03	4	The system logs the destination of food waste and presents a hierarchy for managing food waste: consumption, followed by donation to individuals, animals, compost, and trash
FR-04	3	The system shall allow the user to add friends and family.
FR05	3	The system shall provide a referral option (link to the app store) to elicit users to invite their friends/family to use the application.
FR-06	5	The system shall include an option to opt out of data collection other than location.
FR-07	3	The system shall use user data to create a “leaderboard” of food waste reducers based on food waste reduction percentage, as well as by participation streak.
FR-08	2	The system shall allow the user to earn progress-based badges to their profile.
FR-09	5	The system shall provide food-saving tips to the user based on their type or cause of waste.
FR-10	4	The system shall have a history feature that displays the users past food waste data in a graphical view.
FR-11	4	The system shall display the total approximate cost of their food waste for a selected time period.
FR-12	4	The system shall store information related to home/away-from-home eating habits (meals eaten away from home per week. And display the difference).
FR-13	5	The system shall provide a section that contains additional food waste resources per interest, and a map of places to donate food.
FR-14	5	The user must be able to create and log into a user account.

Appendix DOC-A - SRS

System Requirements Specification V.2.1
11-1-2023

Waste Watcher Built for the Senator George J. Mitchell Center for Sustainability Solutions by:

Sustainable Waste Solutions (Current Team)

- Christian Silva
- Jackson Cyr
- Kevin Bretthauer
- Callen Shaeffer
- Caiden Emerson

Kayak Development Solutions (Past Team)

- Declan Brinn
- Chase Pisone
- Gavin Palazzo
- Finn Jacobs
- Levi Sturtevant

7. Introduction

The Food Waste Tracking & Measuring software application is a University of Maine Capstone project for Susanne Lee, in partial fulfillment of the Computer Science BS degree for the University of Maine. Susanne Lee is a Faculty Fellow at the Mitchell Center for Sustainability Solutions at the University of Maine. Susanne leads a student/faculty team working towards developing solutions to end food waste in Maine. Due to the massive amounts of food that goes to waste each year the need for our project arose. The goal of our Food Waste Tracking & Measuring app is to make the user aware of how much food they waste and what they can do to minimize food waste in their household. This project is being developed to provide a cost-free product that will assist in the tracking of food waste among schools, businesses, and households in Maine.

"In 2015, a Mitchell Center multidisciplinary team identified eliminating food waste as the single most important issue to ensure a more sustainable waste system in Maine." **- [1]**

7.1 Purpose of This Document

This document lays out the foundation for the construction of the food waste tracking system. This document defines the necessary requirements the product must meet before being considered complete and deployable. This document not only enumerates the product requirements, but also the user interface design, and the final deliverables of the product. The primary audience for this document is our client, Susanne Lee and the Mitchell Center for Sustainability Solutions, and the secondary audience is our peer team in order to gain insight as to how we can improve our system requirements.

7.2 References

kayakuser (2022), food-waste-tracker, kayak-development-capstone
https://github.com/kayakuser/Kayak-Development-Capstone

callenshaeffer (2023), Waste-watcher, Waste-watcher
https://github.com/callenshaeffer/Waste-Watcher

Senator George J. Mitchell Center for Sustainability Solutions at UMaine. (2023, November 28). Senator George J. Mitchell Center for Sustainability Solutions.
https://umaine.edu/mitchellcenter/

Food Rescue MAINE, 4 Nov. 2023, http://umaine.edu/foodrescuemaine/.

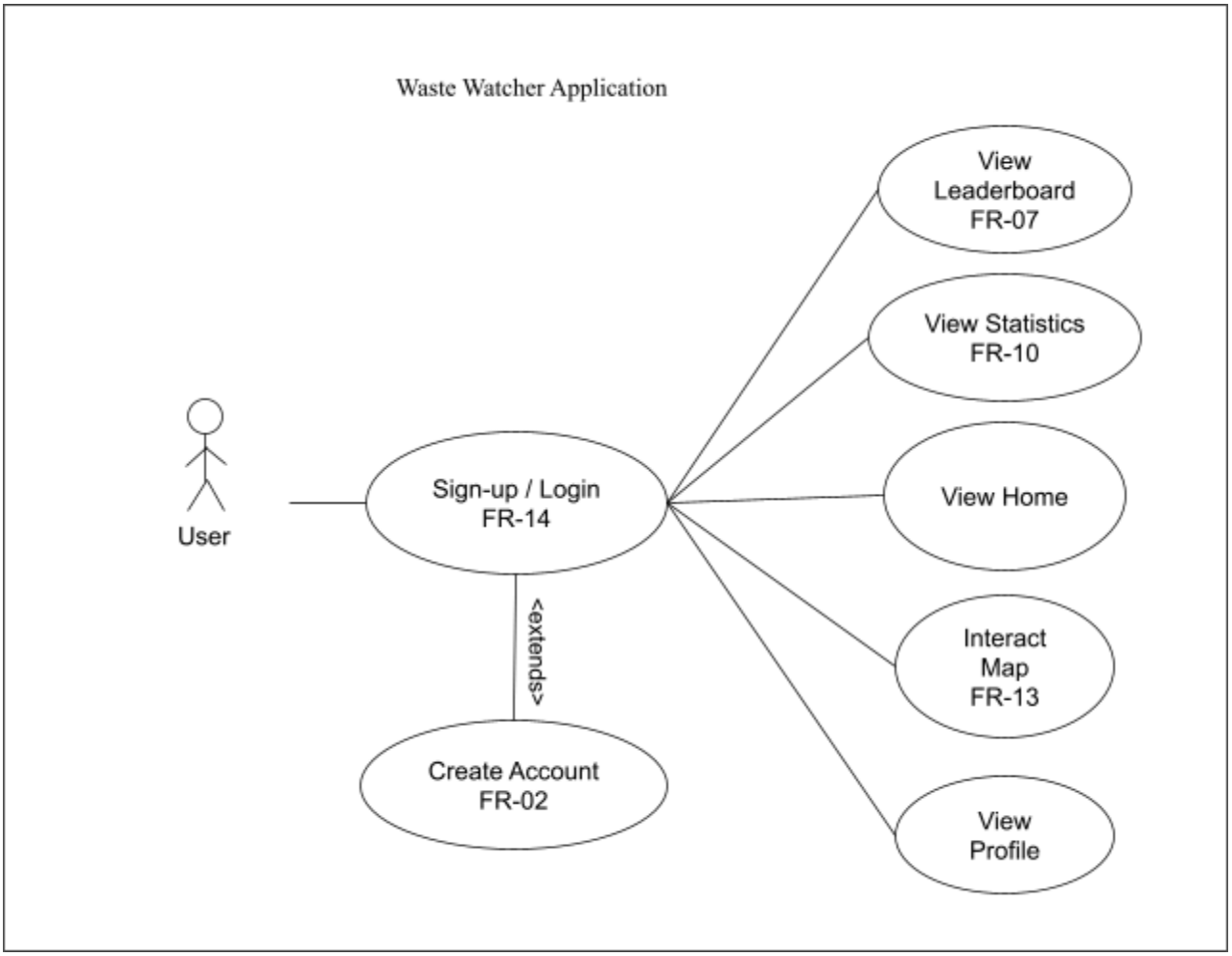
Sustainable Waste Solutions has been provided with the login to a GitHub repository created by the Kayak Development Solutions team who has worked on this project in the past. In addition to the start of a working product, their team also provided us with considerable documentation on the prototype, which has assisted greatly in getting the codebase running and planning our completion of the project. In particular, the SRS has proven useful to give us an idea of what the original goals of the project were, and what still must be accomplished.

7.3 Purpose of the Product

This product is intended to be a cheap and versatile tool which aids in the tracking and research of food waste among schools, businesses, and households. While some tools do exist for this purpose, they are either too expensive or too cumbersome to be useful for the research needs of the Senator George J. Mitchell Center for Sustainability, This is located on the University of Maine campus. Therefore we must design a new tool that can properly address these needs.

7.4 Product Scope

Product Scope Use-Case Diagram: Figure 1.4.1



Description: 1.4.1

This figure displays a top level diagram of how our functional requirements are part of the application as a whole. The user is first prompted to Login or Sign up. There is no option for a user to navigate the app without logging in or signing up. After a user logs in or signs up they are taken to the home screen which can access any of the other major pages within the application.

8. Functional Requirements

This section serves to break down all of our functional requirements into descriptions based on use case diagrams and use cases. This is to define the effect each choice the user makes based on the requirements provided. All possible scenarios need to be accounted, expected, and adjusted for.

8.1 System Functional Requirements

Figure 8.1 Functional Requirements

Functional Requirements		
Number	**Priority**	**Description**
FR-01	5	The system shall allow the user to enter daily food waste information (amount by weight or volume, the type of food by category, the cause of waste, in home or out of home, and the meal eaten/BLD, date eaten into UI.)
FR-02	5	The system shall be able to request information about the user such as, username, zip code, household size, user gender, age and income range.
FR-03	4	The system logs the destination of food waste, and presents a hierarchy for managing food waste: consumption, followed by donation to individuals, animals, compost, and trash.
FR-04	3	The system shall allow the user to add friends and family.
FR05	3	The system shall provide a referral option (link to app store) to elicit users to invite their friends/family to use the application.
FR-06	5	The system shall include an option to opt out of data collection other than location.

FR-07	3	The system shall use user data to create a “leaderboard” of food waste reducers based on food waste reduction percentage, as well as by participation streak.
FR-08	2	The system shall allow the user to earn progress-based badges to their profile.
FR-09	5	The system shall provide food saving tips to the user based on their type or cause of waste.
FR-10	4	The system shall have a history feature that displays the users past food waste data in a graphical view.
FR-11	4	The system shall display the total approximate cost of their food waste for a selected time period.
FR-12	4	The system shall store information related to home/away from home eating habits (meals eaten away from home per week. And display the difference).
FR-13	5	The system shall provide a section that contains additional food waste resources per interest, and a map of places to donate food.
FR-14	5	The user must be able to create and log into a user account.

8.2 Use Case Specifications

Number	FR-01	
Name	Input Daily Food Waste Information	
Summary	Allow for the user to input food waste information such as the amount by weight or volume, the type of food by category, the cause of waste, in home or out of home, date, and the meal eaten/BLD (breakfast, lunch, dinner)	
Priority	5	
Preconditions	The user must be logged in	
Postconditions	The users food waste information is stored to the database	
Primary Actor	User	
Secondary Actors	The food waste database	
Trigger	User presses the button to input their data	
Main Scenario	**Step**	**Action**
	1	User selects the button to input their food waste information
	2	The app displays fields for all of the required information and prompts the user to fill out each of the fields
	3	The app stores the data that the user input into the database

Number	FR-02	
Name	User Data Request	
Summary	The system shall be able to request information about the user such as, username, zip code, household size, user gender, age and income range.	
Priority	5	
Preconditions	The user must be in the process of creating an account	
Postconditions	The information is stored in the database attached to their account	
Primary Actor	User	
Secondary Actors	Database	
Trigger	User presses the create new account button	
Main Scenario	**Step**	**Action**
	1	User is served with data collection UI
	2	User selects submit button
	3	Data is stored in database

Number	FR-03	
Name	Waste Destination Recording	
Summary	The system records the destination of waste and displays Food waste hierarchy, the hierarchy being eaten, then donating to a person, then feeding animals, then compost, then trash.	
Priority	5	
Preconditions	The user is logged in	
Postconditions	The data the user enters is saved in a database attached to their account	
Primary Actor	User	
Secondary Actors	Database	
Trigger	The user selects the track waste button	
Main Scenario	**Step**	**Action**
	1	The user is served with the waste tracker data collection UI
	2	The user fills out the form
	3	The user selects submit
	4	The data the user entered is stored in a database attached to their account

Number	FR-04	
Name	Add Friends	
Summary	The system shall allow the user to add friends and family.	
Priority	3	
Preconditions	The user must be logged in	
Postconditions	The system sends a friend request from one user to the target user	
Primary Actor	User	
Secondary Actors	The user database	
Trigger	User presses the button to add a friend	
Main Scenario	**Step**	**Action**
	1	User selects the button to add a friend
	2	The app displays a field and prompts the user to input the username of the user they would like to add
	3	The app sends a friend request to the target user
	4	The target user is able to either accept or decline the friend request
Extensions	**Step**	**Branching Action**
	4a	The target user accepts the friend request,two users become friends
	4b	The target user declines the friend request

Number	FR-05	
Name	Friend referrals	
Summary	The system shall provide a referral option (link to app store) to elicit users to invite their friends/family to use the application.	
Priority	3	
Preconditions	The user must be logged in	
Postconditions	The user shares a referral link via social media	
Primary Actor	User	
Secondary Actors		
Trigger	The user pressed the invite friends button	
Main Scenario	**Step**	
	1	The system prompts the user to select the platform in which they would like to share the referral
	2	The system generates a referral link that leads to the app store
	3	The system sends the link along with a brief message about the app and the user that is referring them

Number	FR-06	
Name	Data-collection opt-out	
Summary	The system shall include an option to opt out of data collection other than location.	
Priority	5	
Preconditions	The user must be in the process of creating an account.	
Postconditions	Data other than location will not be collected for this user.	
Primary Actor	User	
Secondary Actors		
Trigger	User selects opt-out of data collection button after or during sign-up process	
Main Scenario	**Step**	
	1	User selects option to opt-out of data collection

Number	FR-07	
Name	Friend leaderboard	
Summary	The system shall use user data to create a “leaderboard” of food waste reducers based on food waste reduction percentage, as well as by participation streak.	
Priority	3	
Preconditions	Users must have created an account, and access the database where the data is stored for the leaderboard. Users must be logged in.	
Postconditions	User can view leaderboard displaying their stats and those of friends	
Primary Actor	User	
Secondary Actors	Database	
Trigger	User presses button to switch to leaderboard view	
Main Scenario	**Step**	
	1	User presses leaderboard view button
	2	UI is switched to leaderboard view

Number	FR-08	
Name	Profile badges	
Summary	The system shall allow the user to earn progress-based badges to their profile.	
Priority	2	
Preconditions	Users must be logged in and actively using the application to track food waste.	
Postconditions	A badge is added to the users profile to showcase their achievements	
Primary Actor	User	
Secondary Actors	Badge System	
Trigger	The user achieves a specific milestone related to food waste reduction	
Main Scenario	**Step**	
	1	The user engages with the application to track their food waste activities.
	2	The badge system evaluates the user's activities and progress.
	3	The user achieves a specific milestone set by the badge system.
	4	The badge system awards a badge to the user's profile.
	5	The user receives a notification about the new badge and can view it on their profile.
Extensions	**Step**	
	3a	If the user does not achieve the milestone, they continue using the app and the badge system re-evaluates at a later time.
Notes	The badges should serve as motivation for users to continue reducing their food waste and engaging with the app.	

Number	FR-9	
Name	View Food waste tips	
Summary	The system shall provide food saving tips to the user based on their type or cause of waste.	
Priority	5	
Preconditions	The user must have entered food waste data into the system. User must be logged in.	
Postconditions	The user receives personalized food saving tips	
Primary Actor	User	
Secondary Actors	Food saving tips database	
Trigger	The user inputs food waste information into the system.	
Main Scenario	**Step**	
	1	The user inputs food waste data
	2	The system analyzes the entered data
	3	The system retrieves relevant food saving tips from the database
	4	The user receives personalized tips based on the analyzed data
Extensions	**Step**	
	2a	If the system does not have enough data for personalized tips, it provides general food saving advice
Notes	The tips should help the user better manage their food waste	

Number	FR-10	
Name	View Food waste graph	
Summary	The system shall have a history feature that displays the users past food waste data in a graphical view, shown on the Statistics screen.	
Priority	4	
Preconditions	The user must be logged in. The user must have input their food waste information at least three times	
Postconditions	The user gets to view a graphical representation of their recent food waste	
Primary Actor	User	
Secondary Actors	Food waste database	
Trigger	The user selects the food waste history button	
Main Scenario	**Step**	
	1	The database requests the users recent food waste history
	2	The system display the information in a graphical view

Number	FR-11	
Name	Food waste cost counter	
Summary	The system shall display the total approximate cost of their food waste for a selected time period.	
Priority	5	
Preconditions	Complete request from database of food waste cost over a predetermined time period, correct calculation of total cost.	
Postconditions	Total approximate cost is displayed on the track waste page.	
Primary Actor	User	
Secondary Actors	Firebase	
Trigger	Track waste page selection	
Main Scenario	**Step**	
	1	Track waste page is selected
	2	Application sends request to database with timeframe
	3	Data is received
	4	Average is calculated by application
	5	System displays calculation on track waste page
Extensions	**Step**	
	5b	Track waste page is selected
	5c	Application sends request
	5d	No data is present
	5e	System displays “Please input food waste data”
Notes		

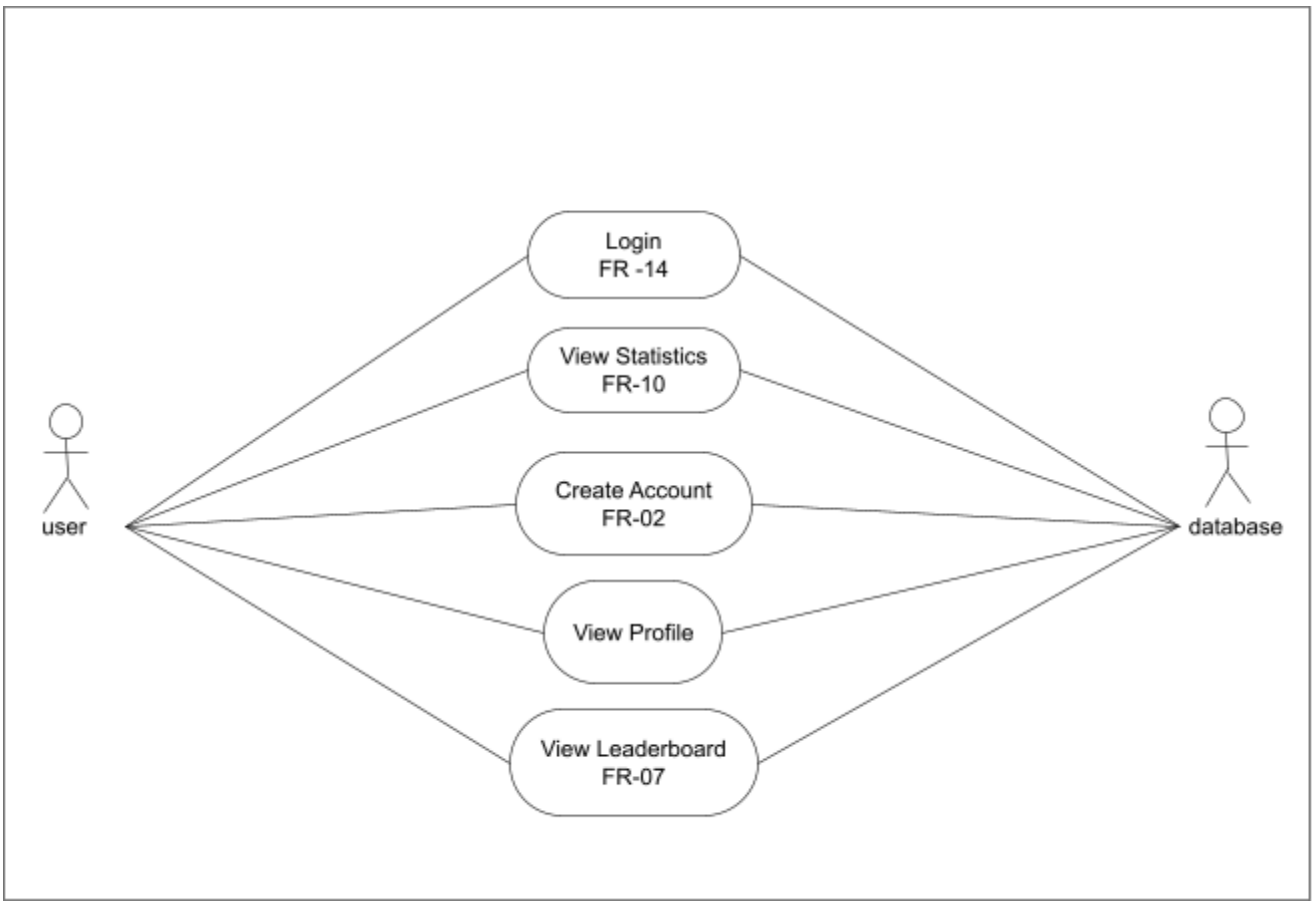
Number	FR-12	
Name	Meal location display	
Summary	The system shall store information related to home/away from home eating habits (meals eaten away from home per week. And displayed the difference).	
Priority	5	
Preconditions	Complete request from database of food waste location over a predetermined time period, correct calculation of total cost.	
Postconditions	Difference in waste in house/out house means are displayed on the track waste page.	
Primary Actor	User	
Secondary Actors	Firebase	
Trigger	Track waste page selection	
Main Scenario	**Step**	
	1	Track waste page is selected
	2	Application sends request to database with timeframe
	3	Data is received
	4	Difference is calculated by application
	5	System displays calculation on track waste page
Extensions	**Step**	
	5b	Track waste page is selected
	5c	Application sends request
	5d	No data is present
	5e	Displays “Please input food waste data”
Notes		

Number	FR-13	
Name	Donation map	
Summary	The system shall provide a section that contains additional food waste resources per interest, a map of places to donate food.	
Priority	5	
Preconditions	Google maps api key is accurate Application request to api is granted	
Postconditions	User has stable internet connection	
Primary Actor	User	
Secondary Actors	google maps api	
Trigger	Welcome page is activated	
Main Scenario	**Step**	
	1	App is opened and welcome page is activated
	2	Request is sent to google maps api
	3	Request is granted
	4	Google maps with relative locations are displayed
Extensions	**Step**	
	5b	App is opened and welcome page is activated
	5c	Request is sent to google maps api
	5d	Request is not successful
	5e	Display google logo
Notes		

Number	FR-14	
Name	Account Handling	
Summary	The user must be able to create and log into a user account.	
Priority	5	
Preconditions	User is not currently logged into an account	
Postconditions	User is logged into an account	
Primary Actor	User	
Secondary Actors	User data collection system	
Trigger	User presses the sign in button	
Main Scenario	**Step**	
	1	User is served with the login button
	2	User enters their account name and password
	3	User account is authenticated and user is logged into the system
Extensions	**Step**	
	2a	User instead selects the create new account button
	2b	User inputs new account information
	2c	New user account is saved into database and user is logged in
	3a	User account is not authenticated and user is served an error message explaining this
Notes		

8.3 Use Case Sub-Diagrams

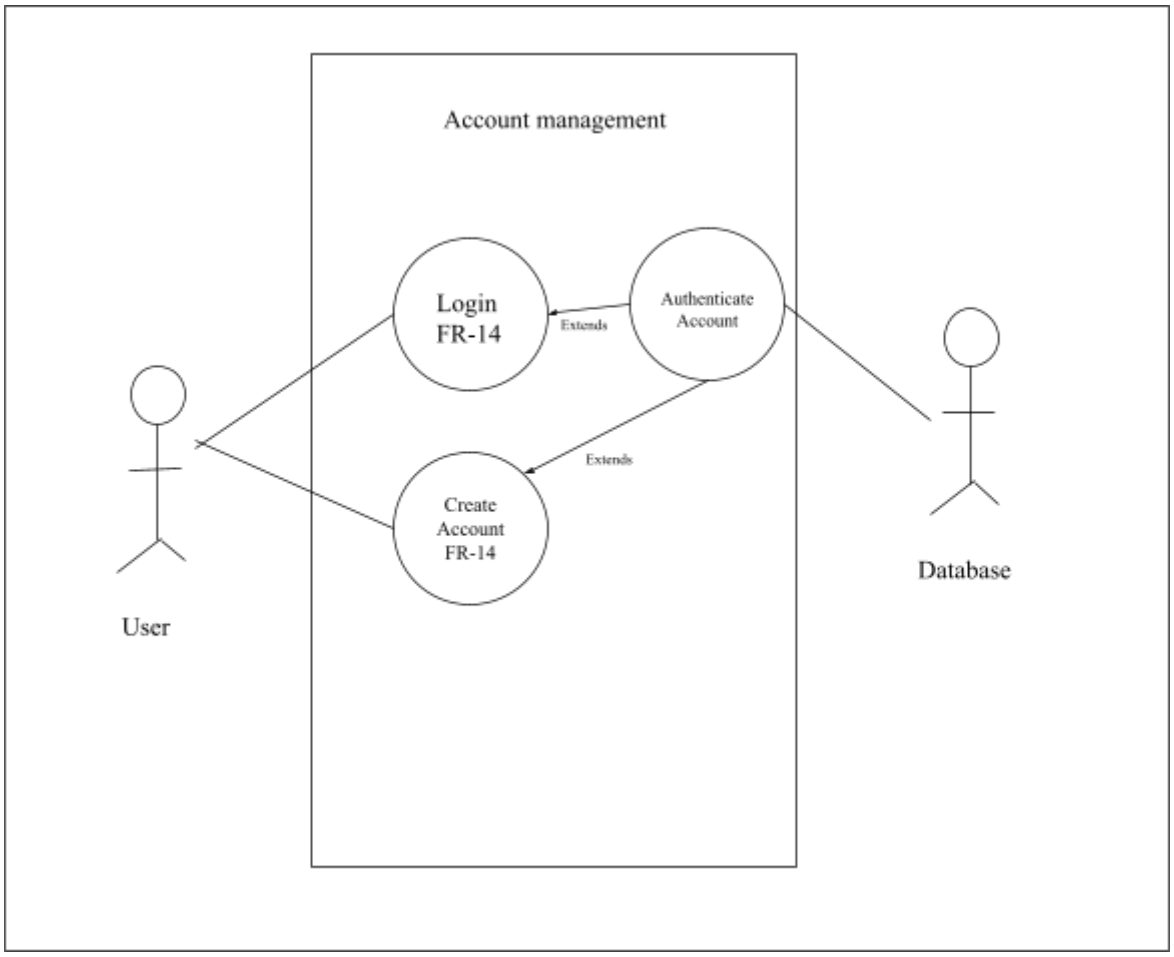
Figure 8.3.1 Data Collection Use Case Diagram



Description: Figure 8.3.1

This figure displays the relationship between the user and the database in terms of the data that the system collects as well as the information the user can access. This data includes the users login information and profile, the food waste and donation statistics, as well as the users personal statistics and leaderboard.

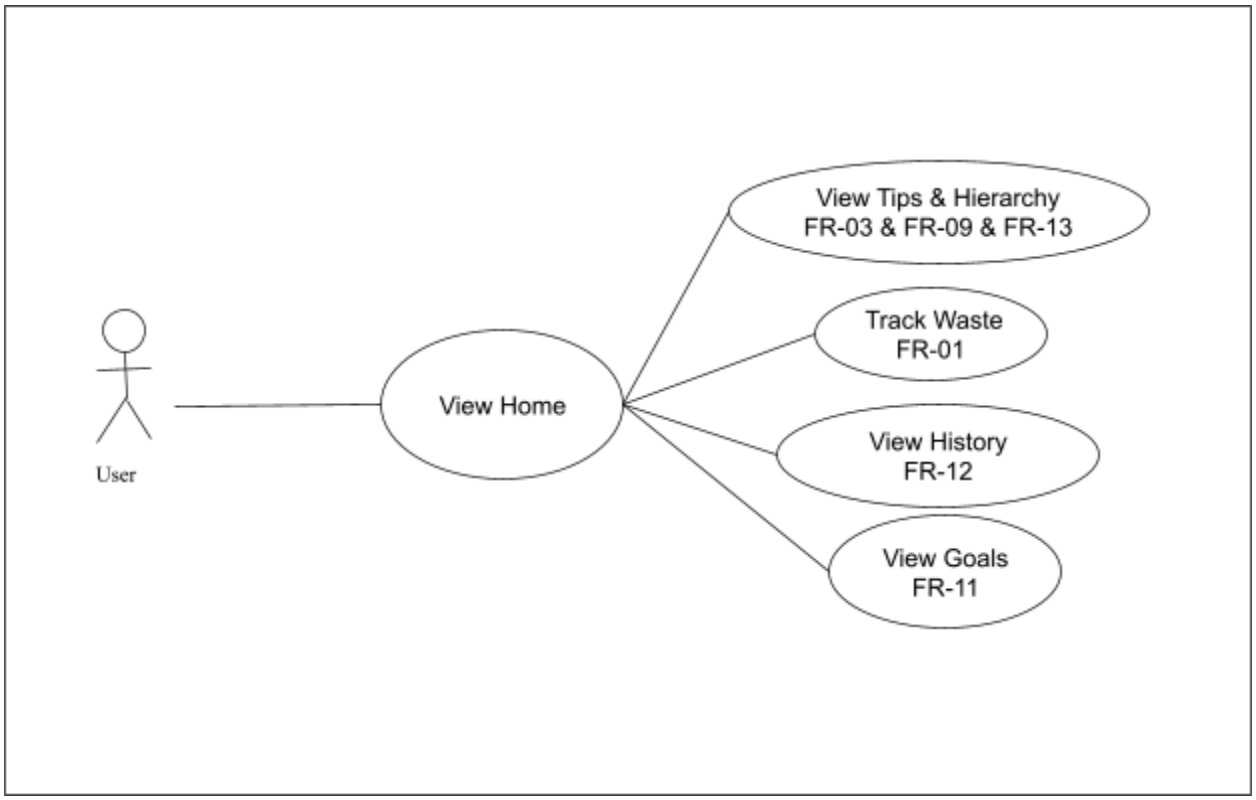
Figure 8.3.2 Account Management Use Case Diagram



Description: Figure 8.3.2

This figure shows the interaction between the user and the database when the user is either logging in or creating an account. The database will check whether the user account exists when the user attempts to login, if it doesn't exist it will prompt the user to instead create one. It will also check if an email is already in use when the user attempts to create an account, if it already exists then it will prompt the user to login instead.

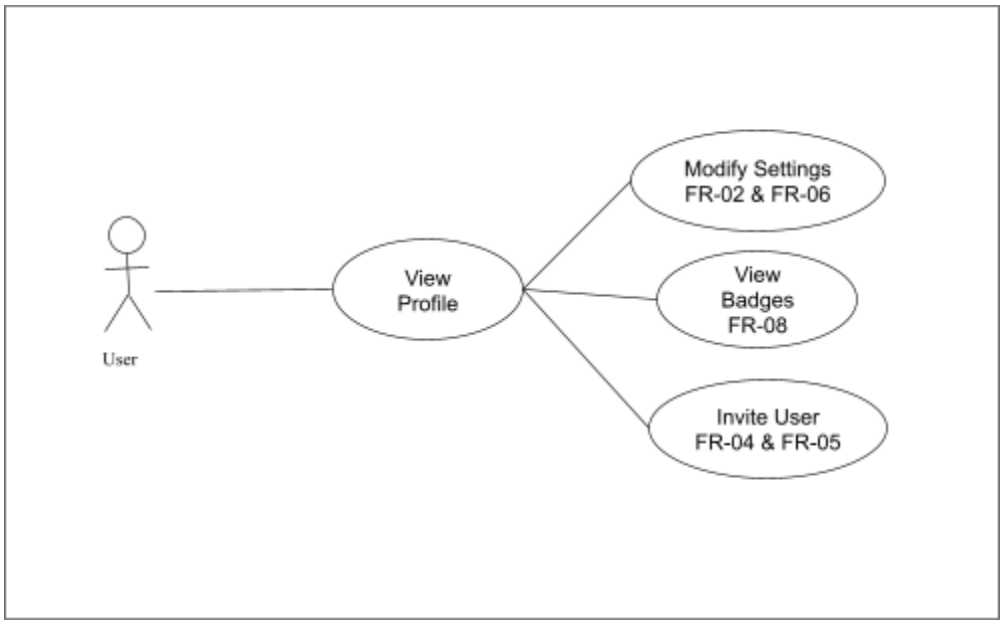
Figure 8.3.3 View Home Use Case Diagram



Description: Figure 8.3.3

This figure displays the different pages that will be available on the applications main page. From the app's home page the user will be able to select to input their food waste, view their waste history, view their goal progress, and view tips based on their recent food waste.

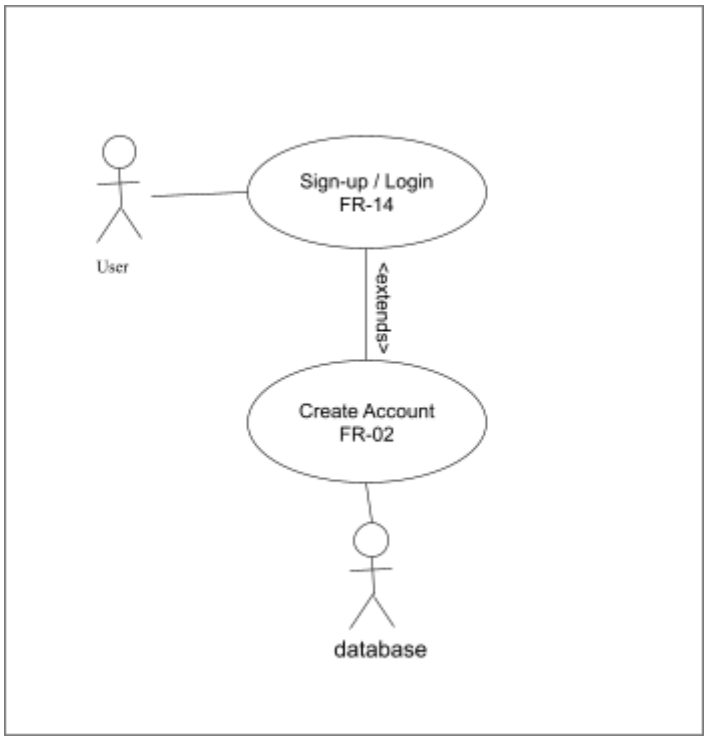
Figure 8.3.4 View Profile Use Case Diagram



Description: Figure 8.3.4

This diagram shows the different pages that will be available from the applications profile menu. From the user profile the user will be able to select to modify the applications settings, view their profiles badges, and invite users to join the app.

Figure 8.3.5 User Data Request Case Diagram



Description: Figure 8.3.5

This diagram displays the data transfer between the application and the database. When a user signs up they are sent to the account creation screen which prompts the user if they want to accept or decline the application gathering about the user, as well as creating a username and password. The data being collected is username, zip code, household size, user gender, age and income range. The login that was created is sent to the database, and any user data that was input is sent to the database as well. After the account creation requirements have been fulfilled the user is sent back to the Login page and ultimately the application. In regards to FR-06 Data-collection opt-out, if the user wants to opt out of data collection the part of the account creation page that accepts input on data collection can be left blank and that would be specified on that page.

9. Non-Functional Requirements

Non-functional requirements break down different operational standards that the system shall adhere to. These non-functional requirements are represented as tables detailing the requirements, priority and a brief testing plan for testing the respective NFR. Below is a list of NFRs. This NFR Tables document gives more detail about these NFRs.

Non-Functional Requirements		
Number	**Priority**	**Description**
NFR-01	5	The system shall work cross platform between IOS and android devices.
NFR-02	4	The system shall be able to support 5000 concurrent users.
NFR-03	4	The system shall be able to query any user data from a database within 2 seconds 90% of the time.
NFR-04	4	The system shall store user information entered into the UI in the database within 2 seconds 90% of the time.
NFR-05	4	The system shall be able to process touch screen input from the user within 2 seconds 90% of the time.
NFR-06	5	The system shall adhere to the US Privacy Act of 1974.
NFR-07	5	The system shall implement encryption for storing user locations.
NFR-08	5	The system shall implement encryption for storing demographic information.
NFR-09	3	The system shall generate food waste statistics with data collected within the last week.
NFR-10	5	The system shall generate food waste statistics within 3 seconds of the user navigating to the statistics page of the application 90% of the time.
NFR-11	4	The system shall undergo Non-Functional Requirement (NFR) testing to ensure compliance with all specified NFRs.

10. User Interface

This section serves as reference to our User Interface Design Document in which we detail the aspects of the User Interface that have been designed for this application. See “User Interface Design Document for Waste Watcher”.

11. Deliverables

This section details which deliverables and which various documents will be delivered to the customer. Each deliverable will be given to the client both electronically in pdf format and in a paper format if they request it. The due dates used in this document will be the same as the course syllabus and are subject to change.

- Systems Requirements Specification - **Due Nov. 1st 2023**
- System Design Document - **Due Nov. 15th 2023**
- User Interface Design Document **- Due Nov. 29th 2023**
- Critical Design Review Document Draft - **Due Dec. 4th 2023**
- Critical Design Review Document Final - **Due Dec. 15th 2023**

12. Open Issues

- Deciding who we want to use as a database - **Due Dec. 5th 2023**
- Deciding whether this is a web page based application, or if it will rely on the expo go bundler, a mobile app development platform. - **Due Dec. 5th 2023**
- Deciding the best way to display the food waste donation map. - **Due Dec. 5th 2023**

Appendix A1 – Agreement Between Customer and Contractor

This section describes what Susanne Lee and Waste Management Solutions will be agreeing to upon signing this document.

By signing this document, Susanne Lee and Waste Management Solutions agree upon the basic system requirements for this project. This agreement also verifies the project scope, deliverables, and project purpose.

In the case that there are any changes to the requirements, scope, or other important details regarding this project, the development team will edit this document for future deliverables as necessary, and the client will use Docusign to approve the changes made.

Client Signature:

Name: Susanne Lee	Date: 11/10/23
Signature: 	
Comments:	

Team Signatures:

Name:Kevin Bretthauer	Date:11/01/23
Signature: 	
Comments:	

Name: Callen Shaeffer	Date: 11/01/23
Signature: 	
Comments:	

Name: Christian Silva	Date: 11/01/23
Signature: 	
Comments:	

Name: Caiden Emerson	Date: 11/01/23
Signature: 	
Comments:	

Name: Jackson Cyr	Date: 11/01/23
Signature: Jackson Cyr	
Comments:	

Appendix A2 – Team Review Sign-off

By signing this document, the team members agree that they have reviewed the document and agree on its content and format.

Team Signatures:

Name:Kevin Bretthauer	Date:11/01/23
Signature: 	
Comments:	

Name: Callen Shaeffer	Date: 11/01/23
Signature: 	
Comments:	

Name: Christian Silva	Date: 11/01/23
Signature: 	
Comments:	

Name: Caiden Emerson	Date: 11/01/23
Signature: 	
Comments:	

Name: Jackson Cyr	Date: 11/01/23

Signature: Jackson Cyr
Comments:

Appendix A3 – Document Contributions

- Kevin Bretthauer
  - Contributed to section 2
  - Created 2 use case tables

- Christian Silva
  - Wrote section 1.1-1.4
  - Created Figures 1.1 and 2.4
  - Created 3 Use Case Tables

- Caiden Emerson
  - Contributed to sections 2, 3, 5
  - Created 3 Use Case Tables

- Callen Shaeffer
  - Contributed to Sections 1, 2, 3, 6
  - Created 3 Use Case Tables
  - Created 4 Use Case Sub-diagrams

- Jackson Cyr
  - Created Use Case tables 7 & 8
  - Contributed to sections

Appendix DOC-B - SDD

System Design Document V.2.0
11-6-2023

Waste Watcher Built for the Senator George J. Mitchell Center for Sustainability Solutions by:

Sustainable Waste Solutions (Current Team)

- Christian Silva
- Jackson Cyr
- Kevin Bretthauer
- Callen Shaeffer
- Caiden Emerson

Kayak Development Solutions (Past Team)

- Declan Brinn
- Chase Pisone
- Gavin Palazzo
- Finn Jacobs
- Levi Sturtevant

1Introduction

The Food Waste Tracking & Measuring software application is a University of Maine Capstone project for Susanne Lee, in partial fulfillment of the Computer Science BS degree for the University of Maine. Susanne Lee is a Faculty Fellow at the Mitchell Center for Sustainability Solutions at the University of Maine. Susanne leads a student/faculty team working towards developing solutions to end food waste in Maine. Due to the massive amounts of food that goes to waste each year the need for our project arose. The goal of our Food Waste Tracking & Measuring app is to make the user aware of how much food they waste and what they can do to minimize food waste in their household. This project is being developed to provide a cost-free product that will assist in the tracking of food waste among schools, businesses, and households in Maine.

"In 2015, a Mitchell Center multidisciplinary team identified eliminating food waste as the single most important issue to ensure a more sustainable waste system in Maine." [1]

1.1 Purpose of This Document

The purpose of this document is to provide a technical description of the system that we are designing for the app Waste Watcher. The document will include sections for the system architecture, database design, and the requirements for each component from our SRS. This will act as our reference when we begin developing these systems.

1.2 References

[1] "Home." *Food Rescue MAINE*, 4 Nov. 2023, umaine.edu/foodrescuemaine/.

[2] "Waste Management." *Waste Management, Maine Department of Environmental Protection*, www.maine.gov/dep/waste/index.html. Accessed 15 Nov. 2023.

[3] "Recycling Drop off Locations." *Ecomaine*, www.ecomaine.org/what-can-be-recycled/recycling-drop-off-locations/. Accessed 15 Nov. 2023.

UML Distilled, 2nd edition, Martin Fowler and Kendall Scott, 2000

It should be noted that this is version 2.0 of the Waste Watchers System Design Document, and it is building off of the work done by the Kayak Development team. There have been some significant changes made to the design of the backend data storage system that will be

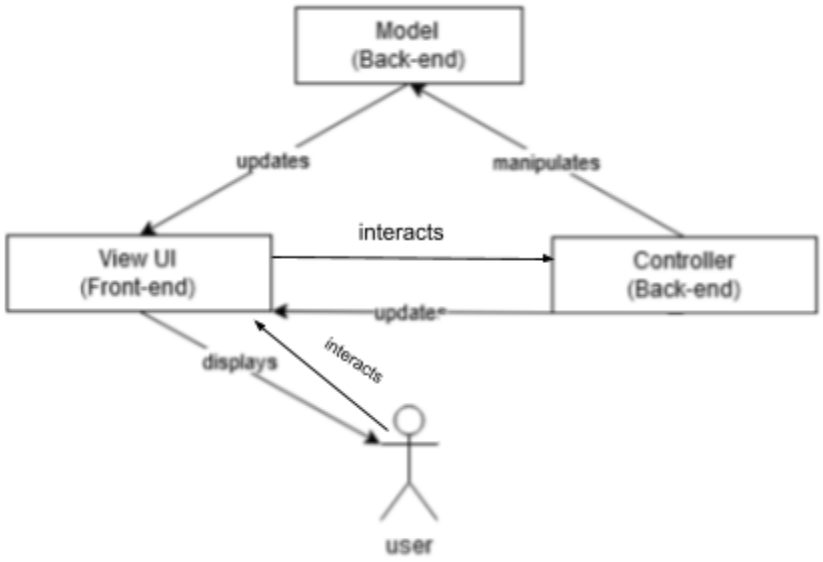
used for the project, as well as the login system implementation details. We have gone through and updated the design for the system to account for these changes, but much of the design remains constant.

2 System Architecture

This section will outline the ways that we will design and build each system within Waste Watcher. This includes the interactions between front-end and back-end components as well as the hardware and software requirements. The requirements will be supplemented by illustrations that will correspond to each of the requirements.

2.1 Architectural Design

Model-View-Controller (MVC) Architecture Diagram: Figure 2.1.1

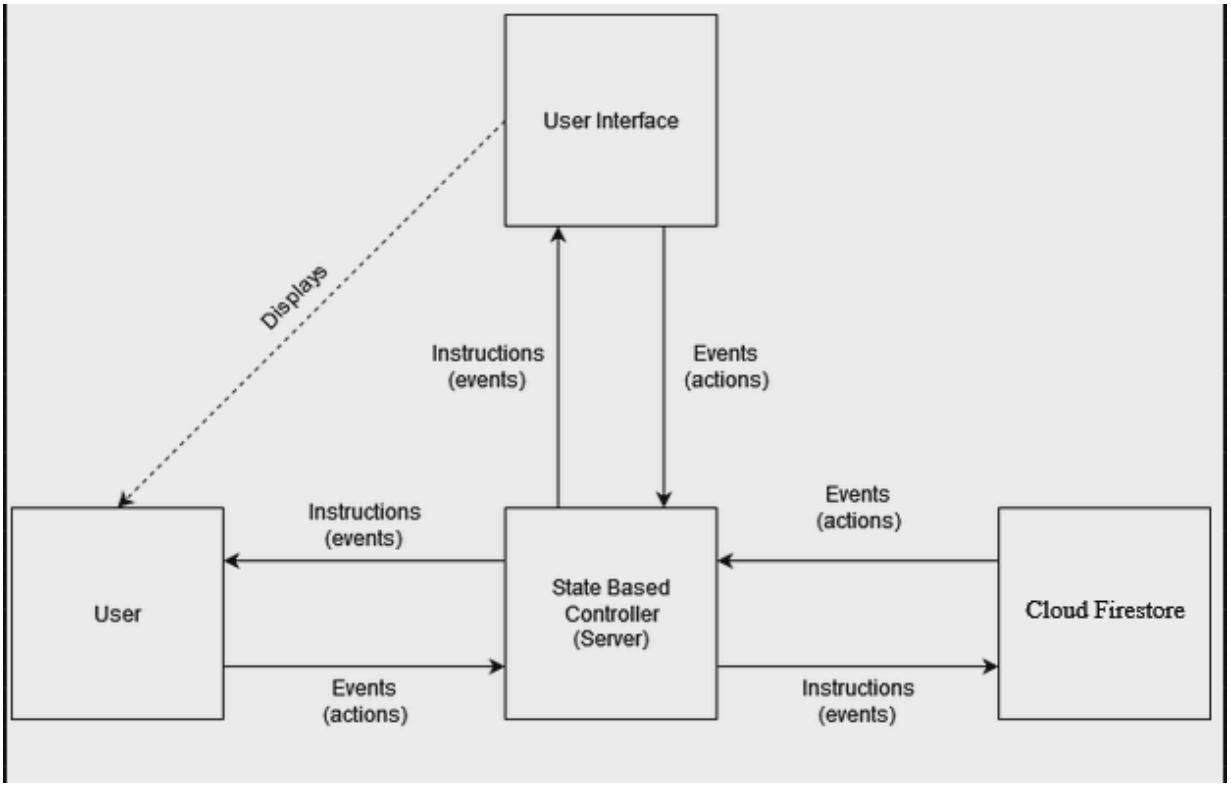


Description: 2.1

The mobile application must be able to be downloaded for both Android and Apple products. To do this the application architecture will consist of two main components, a back-end and a front-end. The Front-end, or the UI, will be created using React Native which is an open-source JavaScript UI software framework for mobile application development. This will allow the application to have a displayed interface that the user can interact with. This includes buttons, pictures, graphs, text boxes for inputs, and various other visuals/interactive elements, which will then give these inputs to the controller to process and provide the product functionality. This back-end is the database/server side. This is where user information and data will be stored and analyzed.

To accomplish this, a Firestore database management system will be employed. All of these features will allow the mobile app to be downloadable cross-platform from both Android and mobile apps.

High-Level Event Driven Component Interaction Diagram: Figure 2.1.2



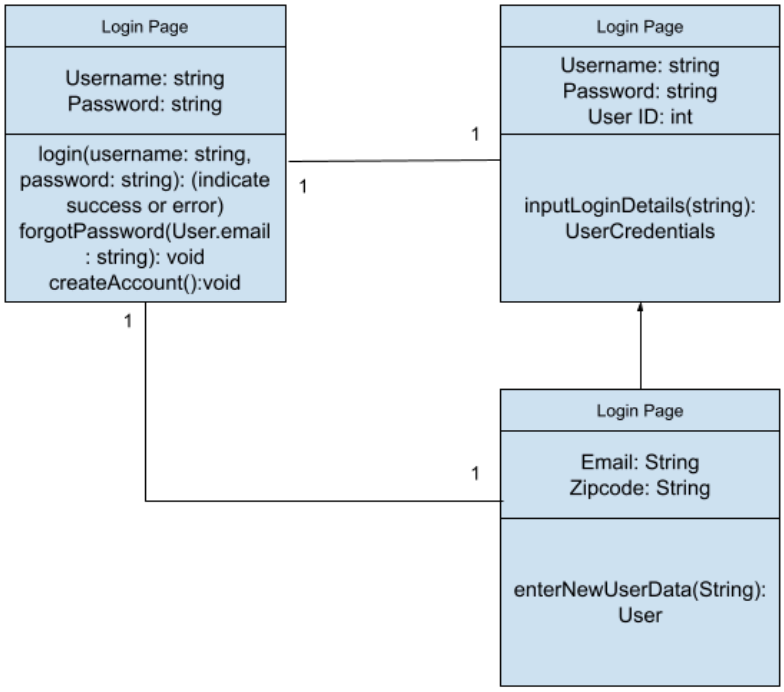
Description: 2.1.2

This is an alternate view of diagram 2.1 which illustrates the base interaction between each of the main systems for the application. This diagram shows the flow of the technology involved in creating the architecture of the program as well as an overview of the interactions between one another. Overall, there are three components: the User, the UI, and the Database all of which take in instructions and respond to events. There is a Controller(Server) as well which takes in events from the components and responds with instructions. Events are only generated by components, an example of which would be the UI taking in an input, from which the event is then sent to the Controller. From here, the Controller receives and interprets the information it was given and formulates a set of instructions in response to the event. The Controller then sends those instructions to the specific components that need to be updated via the given event. Continuing from the previous example, the Server would take in the UI-generated event and

send an instruction to the Firestore database component telling it to store the input given from the UI. The database stores events from the controller and sends back an instruction for the server to execute. The server would also send a similar instruction back to the UI telling it to update its display accordingly which is then shown to the user.

2.2 Decomposition Description

Login Functionality Class Diagram: Figure 2.2.1

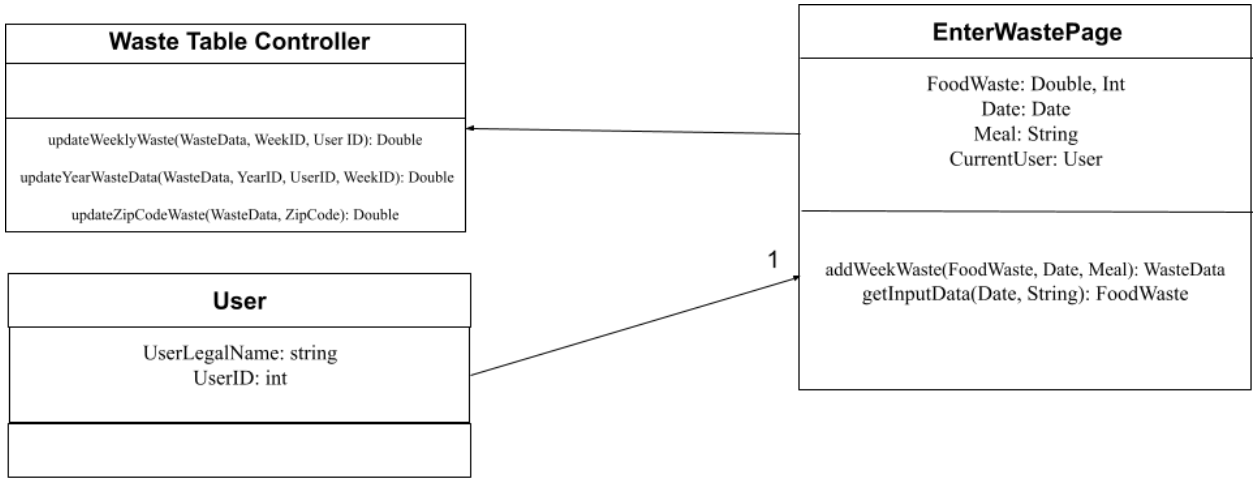


Description: 2.2.1

The separate pages of the application will be object-oriented as each individual page will have its own specific functionalities/classes associated with it. Most pages will inherit class data identified in the login page above with the exception of the input food waste page. Thus, a detailed class diagram illustrating user account login and creation has been provided. Each box represents a class in which it contains its associated variables with their correlated data types (int, string, double). Each class also contains the functions/methods that it will be involved with. Any parameters passed to the function are found within the parenthesis and the functions outputs are found after the colon character. Outputs can be any associated data type based on the function's intended purpose. For example, in the New User class, the *"enterNewUserData(String):User"* method would take in a string for Username, Password, and Email and return a new instance of the User class. links without arrows between classes imply that there is a one-to-one relationship between the two. Essentially what this means is

any singular instance of a class will have exactly one instance of a connected class associated with it. So, in the case of the Login class, there is only one User that can be associated with it.

Waste Entry Diagram: Figure 2.2.2



Description: 2.2.2

This class diagram illustrates the food waste input functionality. The user interacts with the application, entering food waste for each meal on a specified day into the UI. The waste table controller class then takes this data and updates the weekly database table and yearly database table, adding the new data entered by the user to the respective totals. This data will be used to model food waste statistics at a later point in app development.

3 Persistent Data Design

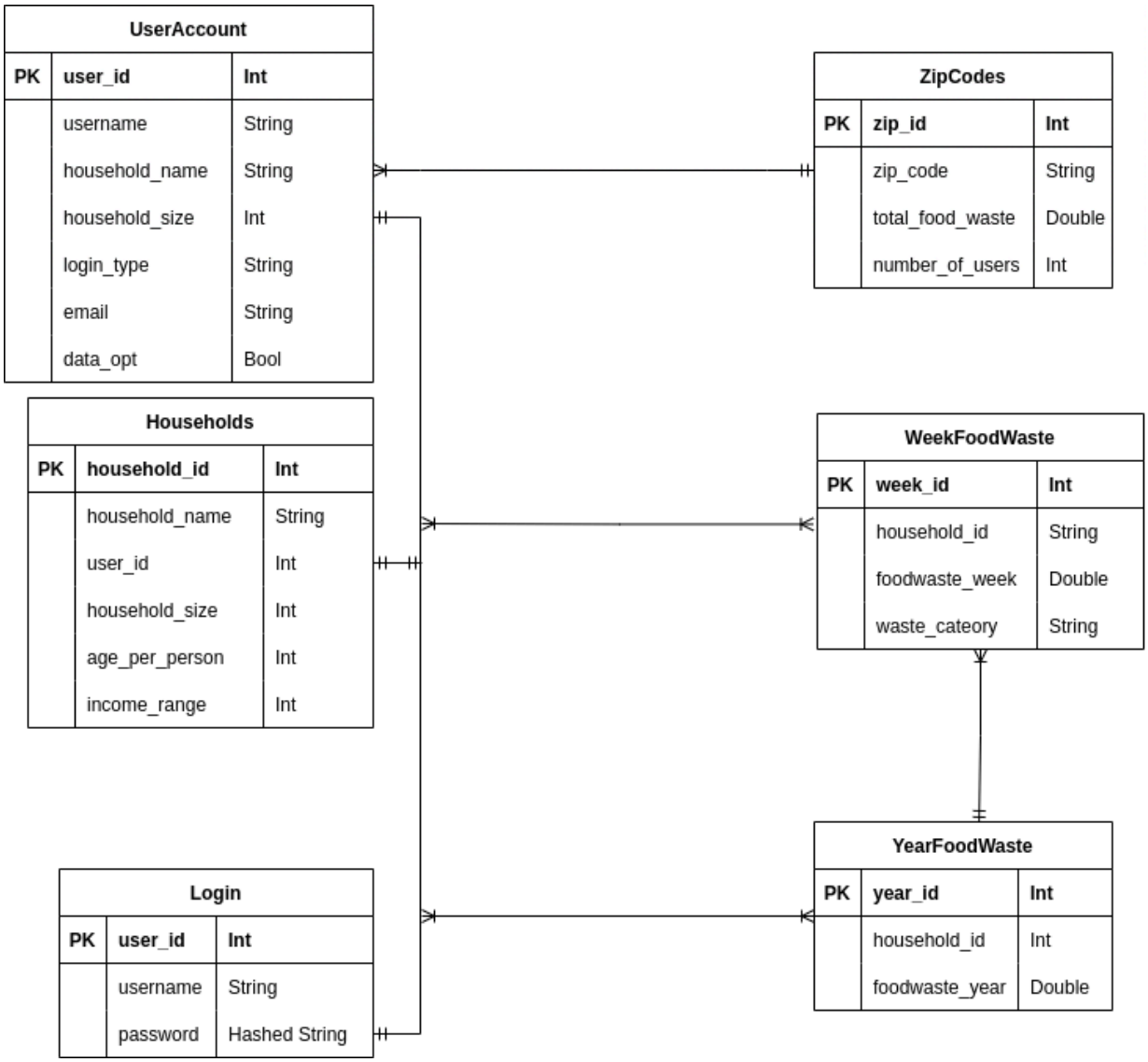
This section will serve as our reference for the system database and will include information such as the file structure and database type. Section 3.1 will illustrate the system database and its interactions.

3.1 Database Descriptions

Each of the tables in the diagram below represents a collection (effectively a JSON format) within our Firestore database. The UserAccount table will store information given to

the UI when a user registers for an account on the application. The Household table stores more granular household information, including the age of each person in the household and the income range of the household. As stated above, information that is user/household specific is split into multiple tables, linked with the user_id data in order to improve the security of the database.The ZipCode table will compile user waste data input by users to provide a breakdown of food waste data within a specific geographic area.

Firestore Database Description: Figure 3.1.1



The WeekFoodWaste table will be a compilation of all food waste submitted by users each week. Entries will be added to the table on a weekly basis, and the total food waste will be updated once users start inputting data. Meal-specific food waste data (i.e. how much food a household wastes at dinner on a specific day) will be collected and stored locally on the user's device, and deleted once it is added to the broader database table. Lastly, the YearFoodWaste table takes data from the WeekFoodWaste table and will create entries on a yearly basis, tabulating total waste figures when weekly data is collected. All entries in these tables will be given a respective identifier which will be used to link data within different tables.

3.2 File Descriptions

No files are required for this system.

4 Requirements Matrix

The following table outlines the relationship between each of our use cases and the functions/methods that will be used to perform them.

Use Case No.	Name	Function/Method
FR-01	Input food waste	TrackWaste()
FR-02	Create an account	CreateAccount()
FR-03	Display food waste hierarchy	ShowFoodChart()
FR-04	Add friend	AddFriend()
FR-05	Send referral message	SendReferral()
FR-06	Opt out of data collection	OptFlag()
FR-07	Leaderboard management	Leaderboard()
FR-08	Badge system, and achievements	Achievements()
FR-09	Tips for users	Tips()
FR-10	Food waste history tracking	WasteGraph()

Use Case No.	Name	Function/Method
FR-01	Input food waste	TrackWaste()
FR-02	Create an account	CreateAccount()
FR-03	Display food waste hierarchy	ShowFoodChart()
FR-04	Add friend	AddFriend()
FR-05	Send referral message	SendReferral()
FR-06	Opt out of data collection	OptFlag()
FR-11	Track cost of wasted food	Cost()
FR-12	Meal location history tracker	TrackMealLocations()
FR-13	Display donation map	DonationMap()
FR-14	Account management	ManageAccount()

Appendix B1 – Agreement Between Customer and Contractor

By signing this document, Susanne Lee and Waste Management Solutions agree upon the basic system design for this project. These requirements are susceptible to change based on if new more important requirements come up or if things become obsolete. In the future, if these changes need to be made, they will be discussed with our customer first for clarification and then updated here once an agreement has been made. Changes to this document should be known by whoever it is relevant to.

Client Signature:

Name: Susanne Lee	Date: 11/10/23
Signature: 	
Comments:	

Team Signatures:

Name: Kevin Bretthauer	Date:11/06/23
Signature: 	
Comments:	

Name: Callen Shaeffer	Date: 11/06/23
Signature: 	
Comments:	

Name: Christian Silva	Date: 11/06/23
Signature: 	
Comments:	

Name: Caiden Emerson	Date: 11/06/23
Signature: 	
Comments:	

Name: Jackson Cyr	Date: 11/06/23
Signature: Jackson Cyr	
Comments:	

Appendix B2 – Team Review Sign-off

By signing below members have indicated that they agree to the information mentioned above in this document. You agree that this document may see changes in the future.

Team Signatures:

Name: Kevin Bretthauer	Date:11/06/23
Signature: 	
Comments:	

Name: Callen Shaeffer	Date: 11/06/23
Signature: 	
Comments:	

Name: Christian Silva	Date: 11/06/23
Signature: 	
Comments:	

Name: Caiden Emerson	Date: 11/06/23
Signature: 	
Comments:	

Name: Jackson Cyr	Date: 11/06/23
Signature: Jackson Cyr	

Appendix B3 – Document Contributions

Christian Silva -
- Did 1.2 and 2.2.1
- Did appendices

Caiden Emerson
- Contributed to sections 1 and 3

Jackson Cyr
- Contributed to section 3.1
- Contributed to section 4

Callen Shaffer
- Contributed to section 1
- Contributed to section 2.2

Kevin Bretthauer
- Contributed to section 2
- Helped with functionality diagram

Appendix DOC-C - UIDD

User Interface Design Document V2.1
11-25-2023

Waste Watcher Built for the Senator George J. Mitchell Center for Sustainability Solutions by Sustainable Waste Solutions and Kayak Development Solutions

Sustainable Waste Solutions (Current Team)

- Christian Silva
- Jackson Cyr
- Kevin Bretthauer
- Callen Shaeffer
- Caiden Emerson

Kayak Development Solutions (Past Team)

- Declan Brinn
- Chase Pisone
- Gavin Palazzo
- Finn Jacobs
- Levi Sturtevant

1. Introduction

The Food Waste Tracking & Measuring software application is a University of Maine Capstone project for the Senator George J. Mitchell Center for Sustainability through Susanne Lee, in partial fulfillment of the Computer Science BS degree for the University of Maine. Susanne leads a student/faculty team working towards developing solutions to end food waste in Maine. Due to the substantial amounts of food wasted each year, the need for our project arose. The goal of our Food Waste Tracking & Measuring app is to make the user aware of how much food they waste and what they can do to minimize food waste in their household. This project is being developed to provide a cost-free product that will assist in the tracking of food waste among schools, businesses, and households in Maine.

"In 2015, a Mitchell Center multidisciplinary team identified eliminating food waste as the single most important issue to ensure a more sustainable waste system in Maine." **[1]**

1.1 Purpose of This Document

This document lays out the User Interface Design for our Food Waste Tracking & Measuring software application. This document defines the requirements we must adhere to. The requirements are relevant to user interface standards that allow the application to be accessible and usable by all users and eliminate bias. The product must meet these requirements before being considered complete and deployable. The primary audience for this document is our client and development team, they are directly using this document to design the UI. The secondary audience is our peer team, to better improve the UI plan before development.

1.2 References

[1] "Home." Food Rescue MAINE, 4 Nov. 2023, umaine.edu/foodrescuemaine/.
[2] "Waste Watcher Capstone Project University of Maine 2022-2023." https://github.com/finnjacobs99/Waste-Watcher
[3] "Waste Watcher Capstone Project University of Maine 2022-2023."https://github.com/callenshaeffer/Waste-Watcher

2. User Interface Standards

This section will outline the design standards we will use to maintain consistency throughout our user interface. This section will also describe the accessibility and inclusive design elements that our application will contain.

The layout of our application will include several screens for each of the main components of the user interface, including a food tracking page, account information, statistics, map, etc. The user will navigate between these pages via a bar on the bottom of the screen which contains a small icon for each screen. This navigation bar remains consistent across all screens, highlighting the selected screen for user clarity. Each screen is thoughtfully structured, and divided into blocks that house different functionalities. Clear and evenly spaced buttons, following a grid pattern for even numbers and a list for odd numbers, provide a consistent and intuitive user interface. Each screen shall contain headers describing the content of the page, which are consistent throughout each screen. Error messages will appear as clear and actionable warnings next to the relevant user interface elements, providing users with guidance on how to resolve the issues that prompted the error message.

2.1 Accessibility and Inclusive Design Elements

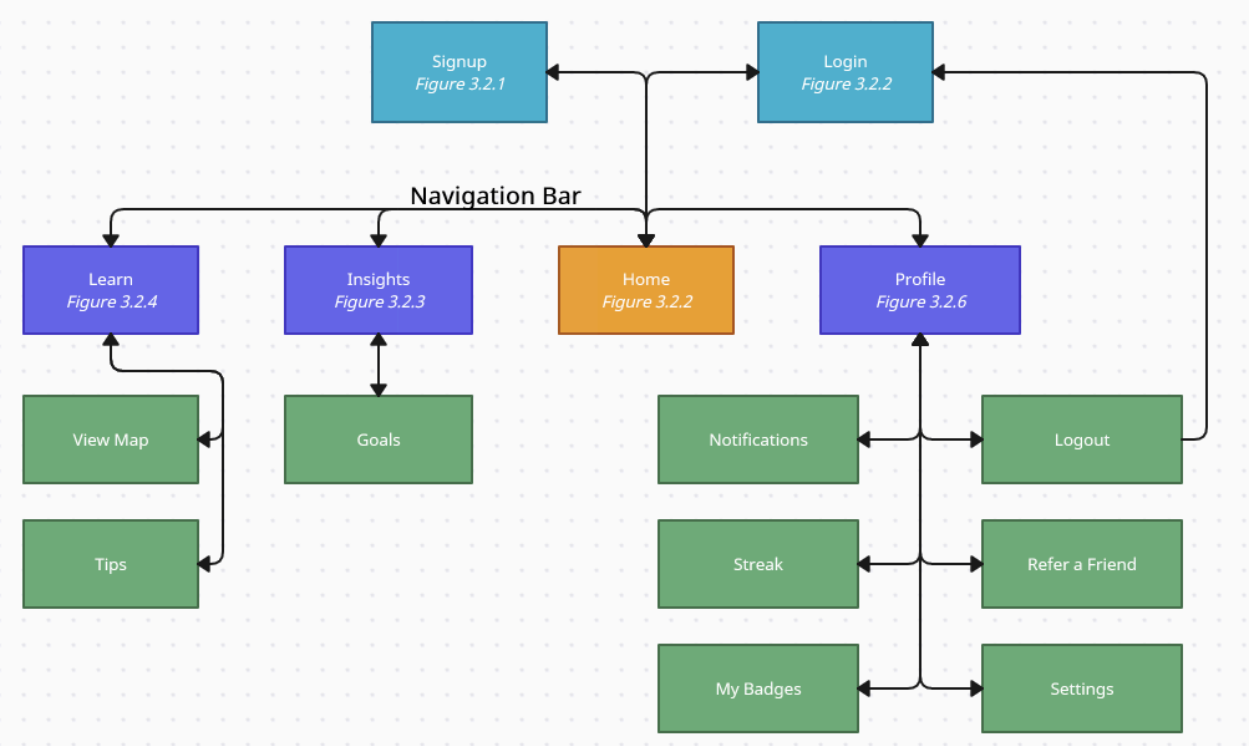
Our application will be designed to be accessible to the largest number of users possible, following accessible design principles to achieve this goal. To cater to visually impaired users, high-contrast colors will be used to enhance the visibility of user actions, as well as large text to be as legible as possible. To ensure users don't have trouble interacting with the user interface, the buttons will be large enough for motor-impaired users to be able to interact with and have enough spacing between them to minimize accidental inputs. Accessibility is a priority for our application, and we are actively looking for ways to make our application usable for the largest number of users possible.

3. User Interface Walkthrough

This section will provide visualizations of the UI and the interactions between the user and the UI elements. 3.1 provides a diagram with all important UI screens. 3.2 provides a walkthrough of all major system screens, shown with screenshots from each major screen area. Each screenshot is given a figure number and is referred to within the walkthrough. Each button and possible ways to navigate throughout the application's user interface will be clearly explained.

3.1 Navigation Diagram

UI Navigation Diagram: Figure 3.1.1



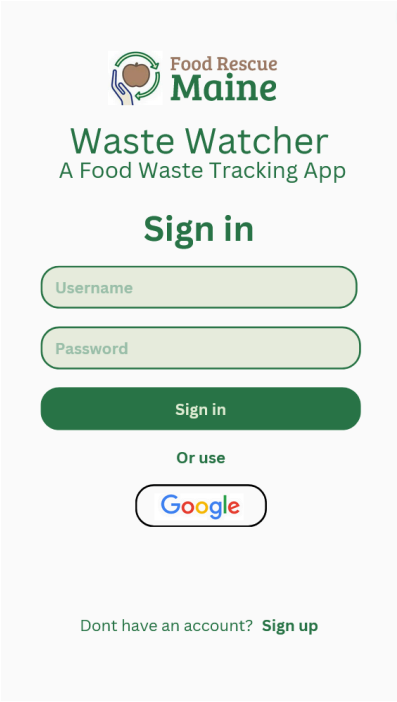
Description: 3.1.1

This diagram displays the general navigation for the waste management application. The user will first be prompted to either ” Login” with an existing account or to “Sign Up” with a new email. The user is then presented with the “Home” screen which will be used to display the user's waste history, goal progress, and current tracking streak, as well as the portal to track waste information. The user can navigate to another screen using the four buttons at the bottom of the screen. The “Insights” page is where users can see a line graph of their weekly waste, a leaderboard showing their streak among friends, their waste history, and where they can set their personal goals. The “Learn” screen will show a map of where users can donate food, it displays the food waste hierarchy, encouraging less destructive ways to get rid of food that would be wasted, and tips on each food type that we track. Finally, the “Profile “ page displays information about the user's account. This information includes their name, recent badges they have earned, the notifications portal, the user's current streak, the badges portal, the ability for users to refer a friend, and the settings and logout buttons.

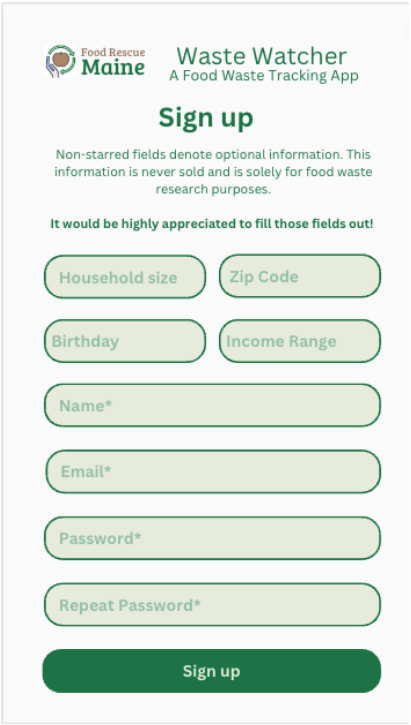
3.2 System Screens Walkthrough

Login Screen: Figure 3.2.1

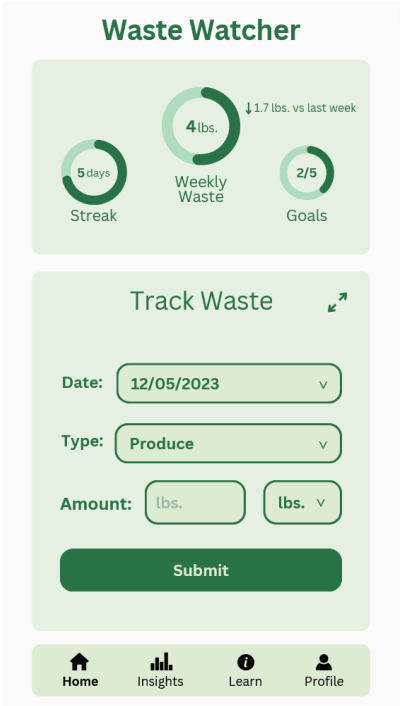
Sign-up Screen: Figure 3.2.2



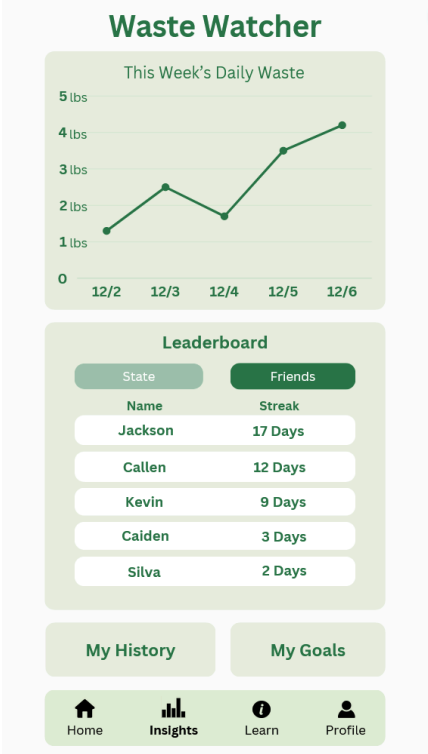
Home Screen: Figure 3.2.3



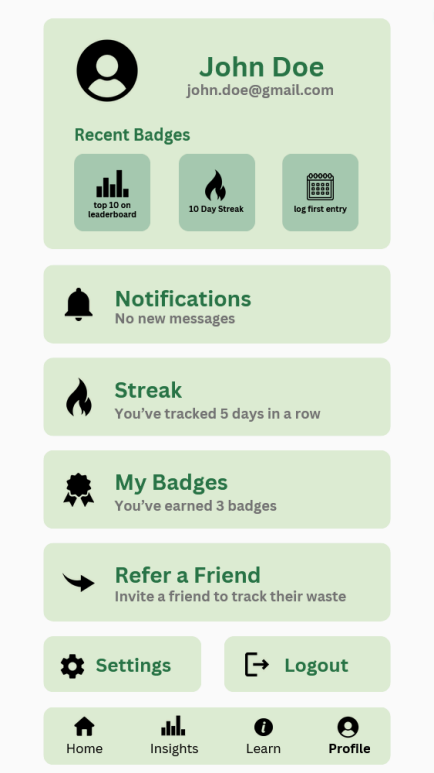
Insights Screen: Figure 3.2.4



Learn Screen: Figure 3.2.5



Profile Screen: Figure 3.2.6



Upon the user opening the application they are taken to the Login screen (*Figure 3.2.1*). From here the user may select the sign-up button, which will navigate them straight to the "Sign Up" screen (*Figure 3.2.2)*. After the user signs in or signs up via Google single sign-on or their email; they are presented with the "Home" Screen (*Figure 3.2.3*). There is a navigation bar located at the bottom of the screen, which contains buttons that allow the user to navigate through each of the main application screens.

Main Menu Button 1: shown as a silhouette of a house, the home screen (*Figure 3.2.3)* is the main landing page after the user enters the application and is logged in. This page displays information and allows for the tracking of waste.

Main Menu Button 2: shown as a silhouette of a histogram, the Insights screen (*Figure 3.2.4)* shows information about the user's current weekly tracking, the leaderboard, and their goals.

Main Menu Button 3: shown as a black circle with a lowercase i in the center as negative space, the Learn page (*Figure 3.2.5*) shows a GPS map with information about donation centers on it. There are also options to receive information about reducing food waste.

Main Menu Button 4: shown as a person's silhouette (*Figure 3.2.6*) and shows information about the user's account. This is also where the user can find the settings menu and log out of their account.

All Main Menu Buttons can be clicked on to navigate to all of the four major system screens. All buttons that take the user to a screen that is not a major system screen will have a return button in the top left corner of the screen allowing the user to go back to their previous application state.

4. Data Validation

4.1 Overview

This section on Data Validation delves into the strategies and methodologies implemented within our system to maintain data quality within our system. These mechanisms not only enhance the reliability of the data but also improve user experience by reducing errors and simplifying the correction process. This section aims to articulate how our interface will remain user-friendly and efficient.

4.2 Data Items

Name	**Description**	**Type**	**Limits/ Constraints**	**Allowable Formats**
Username	Unique identifier for user login	string	Approved by Google SSO standards	Alphanumeric, no spaces
Password	Secure key for user authentication	string	Approved by Google SSO standards	Alphanumeric with at least one uppercase, one lowercase, and one special character
email	User's email address for communication	string	Standard email length	Must be a valid email format (e.g., user@example.com)
age	User's age	integer	0-150	Whole numbers only
location	User's current living place	string	2-100 characters	Alphanumeric, can include spaces and common punctuation
Waste Type	Type of waste user wants to manage	string	1-50 characters	Alphanumeric, can include spaces
Waste amount	Quantitative	string	1-10 characters	Numeric and/or

	measurement of waste			text describing volume/weight (e.g., "5 kg", "10 L")
Waste Goal Description	Description of user's waste management goal	string	10-500 characters	Alphanumeric, can include spaces and common punctuation
Waste Goal requirements	Specific requirements to achieve the waste goal	string	10-200 characters	Alphanumeric, can include spaces and common punctuation

Appendix C1 – Agreement Between Customer and Contractor

By signing this document, Susanne Lee and Waste Management Solutions agree upon the basic user interface design for this project. This design is susceptible to change based on if new more important requirements come up or if things become obsolete. In the future, if these changes need to be made, they will be discussed with our customer first for clarification and then updated here once an agreement has been made. Changes to this document should be known by whoever it is relevant to.

Client Signature:

Name: Susanne Lee	Date: 11/29/23
Signature: 	
Comments:	

Team Signatures:

Name: Kevin Bretthauer	Date: 12/5/2023
Signature: 	
Comments:	

Name: Callen Shaeffer	Date: 12/5/23
Signature: 	
Comments:	

Name: Christian Silva	Date: 12/5/23
Signature: 	
Comments:	

Name: Caiden Emerson	Date: 12/5/23
Signature: 	
Comments:	

Name: Jackson Cyr	Date: 12/5/2023
Signature: Jackson Cyr	
Comments:	

Appendix C2 – Team Review Sign-off

By signing below members have indicated that they agree to the information mentioned above in this document. You agree that this document may see changes in the future.

Team Signatures:

Name: Kevin Bretthauer	Date: 12/5/23
Signature: 	
Comments:	

Name: Callen Shaeffer	Date: 12/5/23
Signature: 	
Comments:	

Name: Christian Silva	Date: 12/5/23
Signature: 	
Comments:	

Name: Caiden Emerson	Date: 12/5/23
Signature: 	
Comments:	

Name: Jackson Cyr	Date: 12/5/2023
Signature: Jackson Cyr	
Comments:	

Appendix C3 – Document Contributions

Identify how each member contributed to the creation of this document. Include what sections each member worked on and an estimate of the percentage of work they contributed. Remember that each team member must contribute to the writing (including diagrams) for each document produced.

Jackson (20%) - Section 2

Callen (20%) - Section 1 & 3.2 UI Mockups and descriptions

Caiden (20%) - Section 3.1 Navigation Diagram

Kevin (20%) - Section 4.1 & Welcome Page Mockup

Silva (20%) - Section 3.1 description

Appendix CRD-A – Team Review Sign-off

By signing this document, Susanne Lee and Waste Management Solutions agree upon the basic user interface design for this project. This design is susceptible to change based on if new more important requirements come up or if things become obsolete. In the future, if these changes need to be made, they will be discussed with our customer first for clarification and then updated here once an agreement has been made. Changes to this document should be known by whoever it is relevant to.

Client Signature:

Name: Susanne Lee	Date: 12/14/23
Signature: 	
Comments: Nice work - look forward to working together to develop the final end product!	

By signing below members have indicated that they agree to the information mentioned above in this document. You agree that this document may see changes in the future.

Team Signatures:

Name: Kevin Bretthauer	Date: 12/11/23
Signature: 	
Comments:	

Name: Callen Shaeffer	Date: 12/11/23
Signature: 	
Comments:	

Name: Christian Silva	Date: 12/11/23
Signature: 	
Comments:	

Name: Caiden Emerson	Date: 12/11/23
Signature:	
Comments:	

Name: Jackson Cyr	Date: 12/11/23
Signature: Jackson Cyr	
Comments:	

Appendix CRD-B – Document Contributions

Identify how each member contributed to the creation of this document. Include what sections each member worked on and an estimate of the percentage of work they contributed. Remember that each team member must contribute to the writing (including diagrams) for each document produced.

Jackson (20%) - Sections 2, 6

Callen (20%) - Sections 1, 3, 5, 5.1

Caiden (20%) - Sections 4, formatting, organization and clarity, apendicies

Kevin (20%) - Sections 7, Professional Writing style, formatting

Silva (20%) - Sections 1, 4