**Vendi**

***A vending machine locator app***

**Project Part 2: Specification**

Computer Science, University of Nevada Reno

Team 23

Members: Catherine Wedin, Maxwell Synard, Charles Dunn, Joey Paschke

Instructors: Sergiu Dascalu, Devrin Lee, Vinh Le, Zach Estreito

External Advisor: Clayton Synard

November 4th, 2022

## Table of contents

[**2. Introduction**](#_szo00acq2wwp) **3**

[**3. Summary of stakeholders’ interviews**](#_jf8j8v3cq9ts) **4**

[**4. Technical Requirements specification**](#_du5zly9in2wf) **11**

[**5. Use case modeling**](#_jz6sk55mw16t) **13**

[**6. Requirement traceability matrix**](#_68roiaqnunnj) **18**

[**7. Initial snapshots**](#_b2tk06f6iouf) **19**

[**8. Glossary of terms**](#_wosiboajlsg2) **24**

[**9. List of references**](#_dqk81dbi1lhn) **26**

[**10. Contributions of team members**](#_n4ozarnwpka9) **28**

## 2. Introduction

The project’s focus is on the development of an app called Vendi. It will be freely accessible on the Android and iOS app stores. The purpose of the app is to give people a free and easily accessible way to locate vending machines in a general area around them. When the app is opened they’ll be greeted with a clean UI that’ll ask them to log in to their account. Once logged in, the users will be greeted with a map interface that lets them swipe around and look for nearby vending machines. If they select a vending machine, depending on the data currently on the machine, they will be given a list of items that are in the machine as well as potential out-of-stock items within the machine. If the users are within a sufficient distance of a vending machine and the information on the machine is out of date, the user will have the option to update the information with a picture or entering information. This will reward them with points that help incentivize the crowd-sourcing mechanic. On top of this map functionality, the app will also have a popout menu on the side of the screen. On the popout menu, the user will have options to view their favorite machines, filter their machines depending on what snack, drink, or supplies they could potentially be looking for, and finally they’ll have options for feedback and point redemption. Since the previous paper, there haven’t really been any changes. However, throughout further research for the project, we have further narrowed some of the specs of the project such as the looks of the app, the features the app will have, and the technology that will be used to develop the app as well. Initially, when the group wasn’t too certain about app development, we planned on only developing the app for iOS but with the cross-platform compatibility of many popular frameworks, this is no longer a limitation. The idea for the project now is to use the Google framework, Flutter. This framework allows for easy UI and frontend design for apps using the Google programming language Dart. Along with this, our database specialist, Charles, will be using the database engine, SQLite, to help manage all of the data that needs to be efficiently accessed, created, and deleted as required. The only major change that has occurred since the previous paper would be Clayton Synard becoming an official external advisor for the team. He will be helping us with the development of our app over the next year.

## 3. Summary of stakeholders’ interviews

**Interview Questions and Answers**

**Person A:** Joey Paschke, *Team member and UNR student*

**Person B:** Indiana Guerrazzi, *UNR student*

**Person C:** Clayton Synard, *External advisor*

1. **What kind of features would you look for in this kind of app?**

**A.** Joey states that some of the features in this app would consist of an easy-to-use, minimalist, and modern app that is very user-friendly and geared towards college students. A key feature Joey emphasizes is the consistency and accuracy of the items listed available in the machine. He wants it to be updated on a regular basis so that the user is never disappointed if an item is out of stock or no longer available.

**B.** Indiana would like to see vending machine information including locations, products available, product prices, and how often a vending machine is being visited or maintained. He also thinks that a review and reward system would be beneficial.

**C.** Clayton believes that something that would greatly benefit the app would be implementing a popular maps platform, such as google or apple maps, into the app or even creating a way to connect with vending machine owners for more accurate information.

1. **Do you think crowdsourcing would be feasible without an incentive to do so?**

**A.** Joey thinks it will initially be difficult to get enough crowd-sourcing but overtime he feels with the awareness of the app there would be a steady increase in it.

**B.** Indiana thinks that in order to be successful long-term, we will need to implement a point system. He compared the potential reward system to the Starbucks app, where for every dollar you spend you earn 2 points, and you can trade in your points for free drinks, food, add-ons, and more. That being said, he believes that in the beginning, the idea might be exciting enough to kick start the app without a point system straightaway.

**C.** Clayton thinks that, while not everyone would do it, if made as simple as a single click enough people would be willing to do it to the point where it would still be feasible.

1. **Would you like to use the finished app without logging in?**

**A.** Ideally, Joey believes that most people would prefer to not have an account because it just extra steps and work and something else that would need to be kept track of , but he believes incorporating key features that will make the app unique and more marketable would require a login system.

**B.** Indiana says that creating an account wouldn’t be a big deal because he feels as though you have to do that for most apps nowadays anyway. He also thinks that if users can use the app without logging in, this would discourage them from leaving photos and reviews about the machines.

**C.** Clayton wants both options to be available. He believes people who often use vending machines when traveling as a quick way to get access to goods and most people wouldn’t want to make an account for something they are only going to use on vacation out of the country.

1. **How many times have you had issues with item availability in vending machines?**

**A.** In Joey’s personal experience, half the time he has utilized the vending machines they have not had what he wanted, and feels like certain machines do not properly portray if they are in stock or not.

**B.** Several. Indiana describes a time where a drink machine on campus has taken his money since it will not tell you if items are out of stock and you cannot physically see if there is product in the machine. He said that it would be helpful to know if others had the same issue with that machine so he wouldn’t have had to waste money.

**C.** Clayton doesn’t really ever have big issues with availability. Most of the time when there happens to be an issue with one item another option just as preferable tends to be open. Though despite all of that his top option is out of stock around 50% of the time.

1. **When searching for a vending machine near you, what kind of filters would you like to use?**

**A.** He wants to incorporate filters based on price and type of item to narrow down the vending machines within a certain area to make it more convenient and accessible to users.

**B.** He would like to see what type of products are inside the machine, along with possibly the brand of products (for instance, if a drink machine has Coke or Pepsi products). He also thinks it would be very neat if one could search for a specific snack (i.e. Doritos) and the app could tell you which machines nearby have them.

**C.** Soda, snacks, electronics, light meals, and different filters for distance such as distance from the user if walked or driven.

1. **What kind of features on top of base functionality (Finding and filtering machines, and crowdsourcing) would you expect or want in this kind of app?**

**A.** On top of base functionality, Joey wants to implement some sort of AI component that would make the app a lot more streamlined. Ideally, this AI implementation would reduce the work needed to be done by the user by scanning pictures sent in by crowd-sourcing and determining what is in the machine.

**B.** Indiana thinks that a point system is much needed. He comes up with the idea that users can have a friend list and possibly a leaderboard among their friends to interact with other users and make it a competition. He also brings up the app Pokemon Go where players can take over certain locations with their team. This could be a way to make locating and reviewing vending machines more enjoyable.

**C.** If possible, Clayton wants the app to have correct currency values for places outside of the U.S, a way to connect with vending machine owners as he previously said in question 1, and would like to see a lot of the crowdsourcing on the backend so there is more incentive to do it.

1. **What platforms would you want the app to be accessible from?**

**A.** With Joey being an iPhone user himself, he wants it to ideally be available in the Apple app store, but also available in the Google Play store to be available for all smartphones.

**B.** Indiana thinks the app should be available on all smartphones.

**C.** Clayton thinks that it should be available to as many mobile operating systems as possible. On top of this, because a lot of laptops are becoming far more portable, a web application was also suggested.

1. **For the scope of this project we are focusing on finding vending machines on the UNR campus. With the focus audience being college students, what do you think are the most important features to have? (Ease of Use, Accessibility, Accuracy, etc.)**

**A.** For college students, Joey wants this app to focus on usability because students in general are very busy. He believes this can be accomplished by making it simple, modern, and easy to use. Additionally ensuring the items listed within the vending machine are as accurate as possible.

**B.** He believes that the main focus should be ease of use since college students do not have the time to be dealing with a confusing app, especially since they are most likely using the app to save time for themselves in the first place.

**C.** Clayton thinks accessibility and ease of use are the most important aspects in an app but for this kind of app an accuracy rating for the machines would also be important.

1. **How many times a week do you use a vending machine?**

**A.** Personally, Joey uses a vending machine twice a week on average.

**B.** Very rarely. Indiana says he only uses them in emergency scenarios.

**C.** Clayton uses vending machines maybe 4-6 times a week.

1. **Do you think having a type of chat/review system would be beneficial for our app? Where users can communicate with each other about what products are in the machines, what they would like to see, etc.?**

**A.** He is open to the idea of potentially implementing a chat system in distant the future for crowd-sourcing purposes, but as of right now he does not believe it is necessary and wants to release the app without that feature and get feedback from the users.

**B.** Indiana thinks that being able to communicate with other users would increase interaction inside the app, and would ultimately cause users to spend more time on the app. He thinks that the app might not get much usage because in most cases a person will open it, find the nearest vending machine, and close it all within 15 seconds. Features that are more interactive would increase time spent on the app, at least for some people.

**C.** Clayton says there are a lot of chat systems in other apps so he’s not sure how much a chat system would help the app. As an alternative he thinks implementing the ability to share vending machine locations with friends would be useful.

1. **What kind of rewards would you like to receive given a points system? Would these rewards incentivize you to participate in crowd-sourcing when no data is currently available?**

**A.** Joey is a little unsure, but believes the reward system has potential and can be worth it. He believes free or discounted snacks would be the best approach after a certain amount of picture uploads, but is unsure where the funding for that would come from. Potentially vending machine retailers.

**B.** Indiana would like to see points going towards discounted or free items, or maybe even a Visa gift card that one could cash out. Another idea he came up with was having bonus points for streaks in order to make using the app a routine. For example, if you walk past and take a picture of the same machine 5 days in a row, you might get 5x points.

**C.** Clayton thinks if the app could interact with vending machines it would be cool to implement free or discounted snacks. Without this mechanic the best alternative would be just giving people gift cards for various retailers.

1. **Are there any other kinds of machines you would like to see other than vending machines on our app?**

**A.** Joey would love to see the possible inclusion of claw machines.

**B.** Indiana says it would be nice to locate water bottle refilling stations.

**C.** Coffee vending machines, meal vending machines, self-service rental kiosks, or perhaps arcade machines in popular areas.

## 4. Technical Requirements specification

* Level 1: Features we plan to implement this fall.
* Level 2: Features we plan to have implemented in the spring.
* Level 3: Features that probably won't be implemented but would be useful and nice to have.

| **4.1 Functional Requirements** | | |
| --- | --- | --- |
| **FR1.** | **[1]** | VENDI shall allow users to type information on the screen. |
| **FR2.** | **[1]** | VENDI shall allow users to use the app through the use of a touchscreen. |
| **FR3.** | **[1]** | VENDI shall allow users to manage their accounts. |
| **FR4.** | **[1]** | VENDI shall allow users to log out of their accounts. |
| **FR5.** | **[1]** | VENDI shall allow users to store private data efficiently and securely. |
| **FR6.** | **[1]** | VENDI shall allow a user to log in or create an account. |
| **FR7.** | **[1]** | VENDI shall ask the user to use the camera and location within the app. |
| **FR8.** | **[2]** | VENDI shall allow users to favorite vending machines in the area. |
| **FR9.** | **[2]** | VENDI shall give users detailed directions to vending machines. |
| **FR10.** | **[2]** | VENDI shall allow users to enter the information of a new vending machine in their exact location. |
| **FR11.** | **[2]** | VENDI shall allow users to see the payment method required for their vending machine. |
| **FR12.** | **[2]** | VENDI shall allow users to view vending machine contents and prices. |
| **FR13.** | **[2]** | VENDI shall allow the user to take pictures of a vending machine. |
| **FR14** | **[2]** | VENDI shall allow the user to use a search box to find a specific item. |
| **FR15.** | **[2]** | VENDI shall allow the user to filter their search with specific criteria such as payment method, product type, and more. |
| **FR16.** | **[2]** | VENDI shall allow users to leave a text review with the option to attach a photo of that specific machine. |
| **FR17.** | **[2]** | VENDI shall allow users to use the google maps API to see vending machine locations relative to where the user is. |
| **FR18.** | **[2]** | VENDI will be accessible on Android and IOS platforms. |
| **FR19.** | **[3]** | VENDI shall have AI image analysis to determine vending machine contents better. |
| **FR20.** | **[3]** | VENDI shall allow vendors to create a vendor account that will give them vendor privileges. |
| **FR21.** | **[3]** | VENDI shall allow vendors to post pictures and data about their machines. |

Table 4.1: Functional requirements for the app Vendi, a vending machine location app.

| **4.2 Non-Functional Requirements** | | |
| --- | --- | --- |
| **NFR1.** | **[1]** | VENDI will use SQLite to store data. |
| **NFR2.** | **[1]** | VENDI will be run on IOS. |
| **NFR3.** | **[1]** | VENDI will be implemented using the Dart coding language. |
| **NFR4.** | **[1]** | VENDI will be run on Android. |
| **NFR5.** | **[1]** | VENDI will be implemented using the Flutter framework. |
| **NFR6.** | **[2]** | VENDI will provide extensive information on the vending machines. |
| **NFR7.** | **[2]** | VENDI will have a streamlined and intuitive user interface. |
| **NFR8.** | **[3]** | VENDI will have an option for dark mode. |
| **NFR9.** | **[3]** | VENDI will minimize the resources needed from the device. |

Table 4.2: Functional requirements for the app Vendi, a vending machine location app.

## 5. Use case modeling

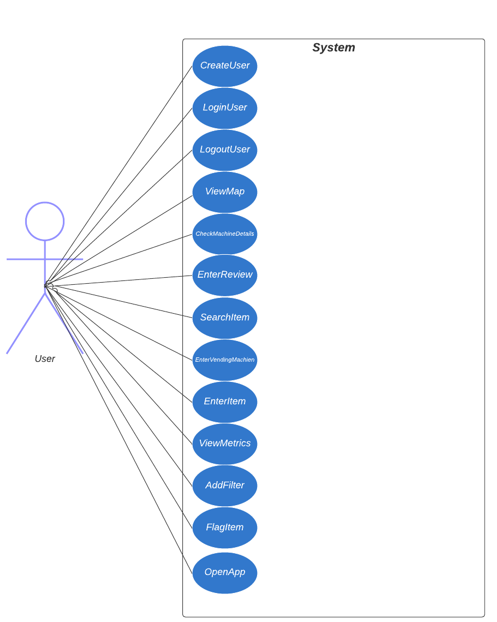


Figure 5.1: Use case diagram

| **ID** | **Use Case** | **Description** |
| --- | --- | --- |
| **UC01** | **CreateUser** | Users will have the option to create an account to utilize all the features within the app and store data and reviews. |
| **UC02** | **LoginUser** | This inquires the server for the validity of the user’s credentials and will accept or deny the user to log in and access their stored data. |
| **UC03** | **LogoutUser** | This feature will log out the user and return to the home screen. |
| **UC04** | **ViewMap** | Users will be able to view the map of vending machines within the area using the Google map API. |
| **UC05** | **CheckMachineDetails** | Users will be able to select a vending machine on the map and check its details. These details will consist of user reviews, machine ratings, the items the vending machine will consist of, the price, the exact location inside or outside a building and floor, and the availability of the given item. |
| **UC06** | **EnterReview** | This feature will allow users to enter a review for the vending machine. The user can highlight some of the items within it, the quality of the vending machine, if it accepts cash, card, or coins, and anything else they feel is important. |
| **UC07** | **SearchItem** | This key feature will allow the user to enter a specific item or type of item and it will highlight all the vending machines this item or type of item are available in. |
| **UC08** | **EnterVendingMachine** | This feature will allow the user to enter a vending machine and the exact location of it. |
| **UC09** | **EnterItem** | This feature allows the user to enter data for current items or pre-existing items in a vending machine. This would be part of the open-source crowdsourcing data component. |
| **UC10** | **ViewMetrics** | This will allow vending machine retailers to view metrics on their machines on the back end. |
| **UC11** | **AddFilter** | This will allow users to filter their search to make the type of item they are searching for more concise. |
| **UC12** | **FlagItem** | This will allow the user to flag an item if it is not accurately depicted in the machine or is overstocked. The user would indeed verify by submitting a photo which will then be pending for review. |
| **UC13** | **OpenApp** | Users will be able to interact with the app on their given platform. Once selected, the app will boot up and create an environment that allows the user to log in. |

Table 5.2: Use case descriptions

| **Use Case: CreateUser** | |
| --- | --- |
| **Use Case ID** | UC01 |
| **Actor(s)** | User |
| **Preconditions** | The user has downloaded and opened the app on their phone. |
| **Flow of Events** | 1. User will select “Create Account” button and will be directed to account creation section 2. The user will be prompted for the following fields:    1. Name    2. Username    3. Password    4. Password Confirmation    5. Email 3. Upon submission, the data will be sent to an email for verification 4. If an error has occurred, user will be prompted to fix the highlighted field(s) |
| **Postconditions** | If no error has occurred, the account will be created and stored and the user will be prompted to login. |

Table 5.3.1: Template for “Create User “ use case.

| **Use Case: LoginUser** | |
| --- | --- |
| **Use Case ID** | UC02 |
| **Actor(s)** | User |
| **Preconditions** | The user has opened the application on their phone. |
| **Flow of Events** | 1. User will be prompted to enter the following credentials:    1. Username    2. Password 2. Data is sent to the server for verification    1. If invalid user will be prompted to re-enter credentials    2. If valid, user will be logged in |
| **Postconditions** | User will be redirected to the home screen. |

Table 5.3.2: Template for “Login User “ use case.

| **Use Case: ViewMap** | |
| --- | --- |
| **Use Case ID** | UC04 |
| **Actor(s)** | User |
| **Preconditions** | The user has logged into the app |
| **Flow of Events** | 1. On the sidebar, user will select “View Map” 2. If location is not already turned on, app will request the user to turn on their location 3. Vending machines will appear within a given radius |
| **Postconditions** | User can select a vending machine of their choice and see what items it consists of. |

Table 5.3.3: Template for “ViewMap “ use case.

| **Use Case: CheckMachineDetails** | |
| --- | --- |
| **Use Case ID** | UC06 |
| **Actor(s)** | User |
| **Preconditions** | The user is logged into the app and has selected the map section on the sidebar |
| **Flow of Events** | 1. The user will select a vending machine that appears within the given radius 2. The machine will list the follow details    1. Exact location (Building, floor, etc)    2. Vending machine image (optional)    3. Item names       1. Item price       2. Is item in stock (yes or no)    4. Reviews    5. Vending machine rating |
| **Postconditions** | The user can read the description and determine if the vending machine meets their requirements/preferences. |

Table 5.3.4: Template for “CheckMachineDeatils “ use case.

## 6. Requirement traceability matrix

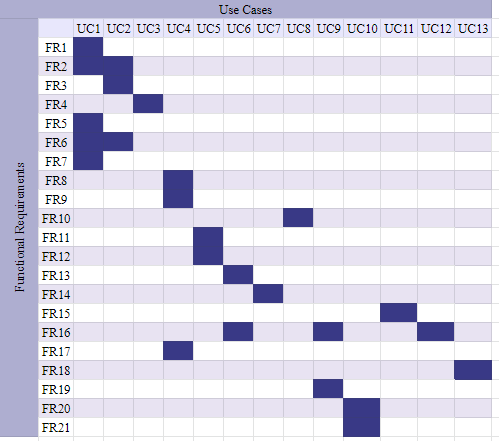
****

Figure 6.1: Requirement Traceability Matrix displaying the relationships between use cases

and functional requirements.

## 7. Initial snapshots

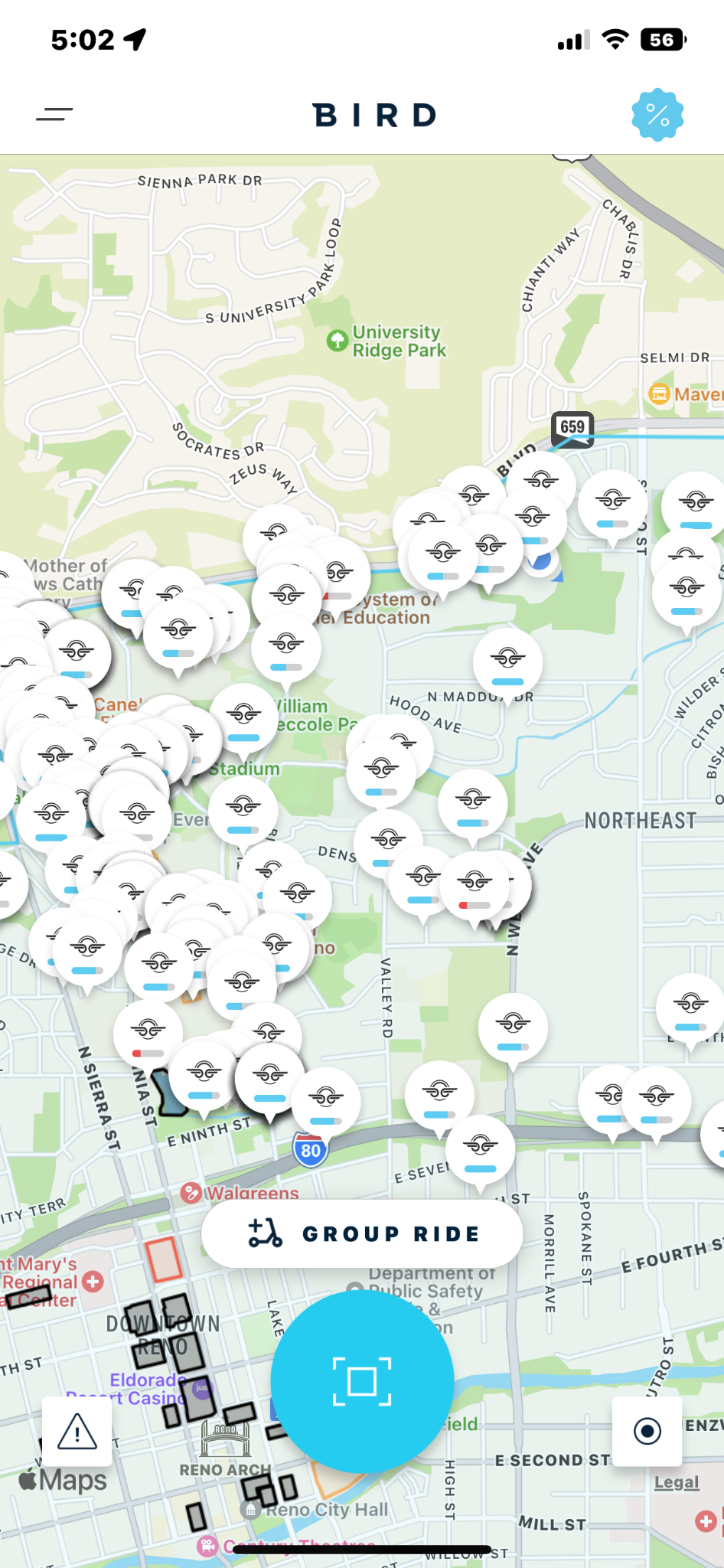


Figure 7.1:Like the Bird Scooter app shown above, Vendi will use the google maps API to show the locations of vending machines in the area. It will also have icons that correlate to the type of machine to increase user usability.

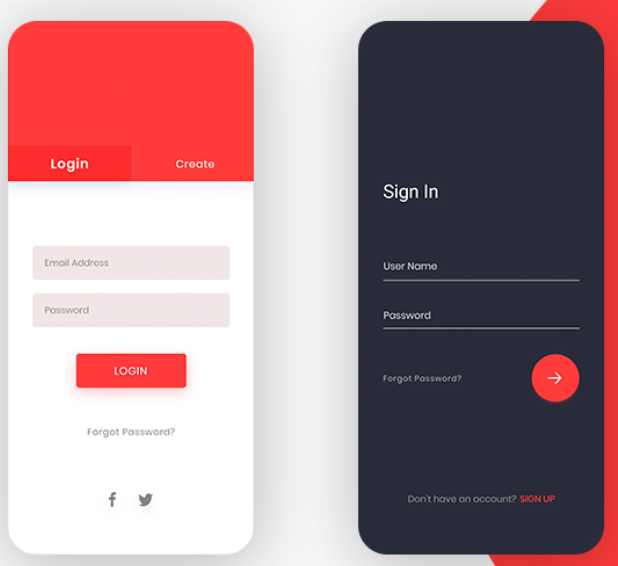
****

Figure 7.2: This is the intended look of the login page of our app. It is both simple and effective and will include an option to sign in or create an account as a retailer or a general user. Depending on what account the user signs into, either user or retailer, it will give certain permissions.

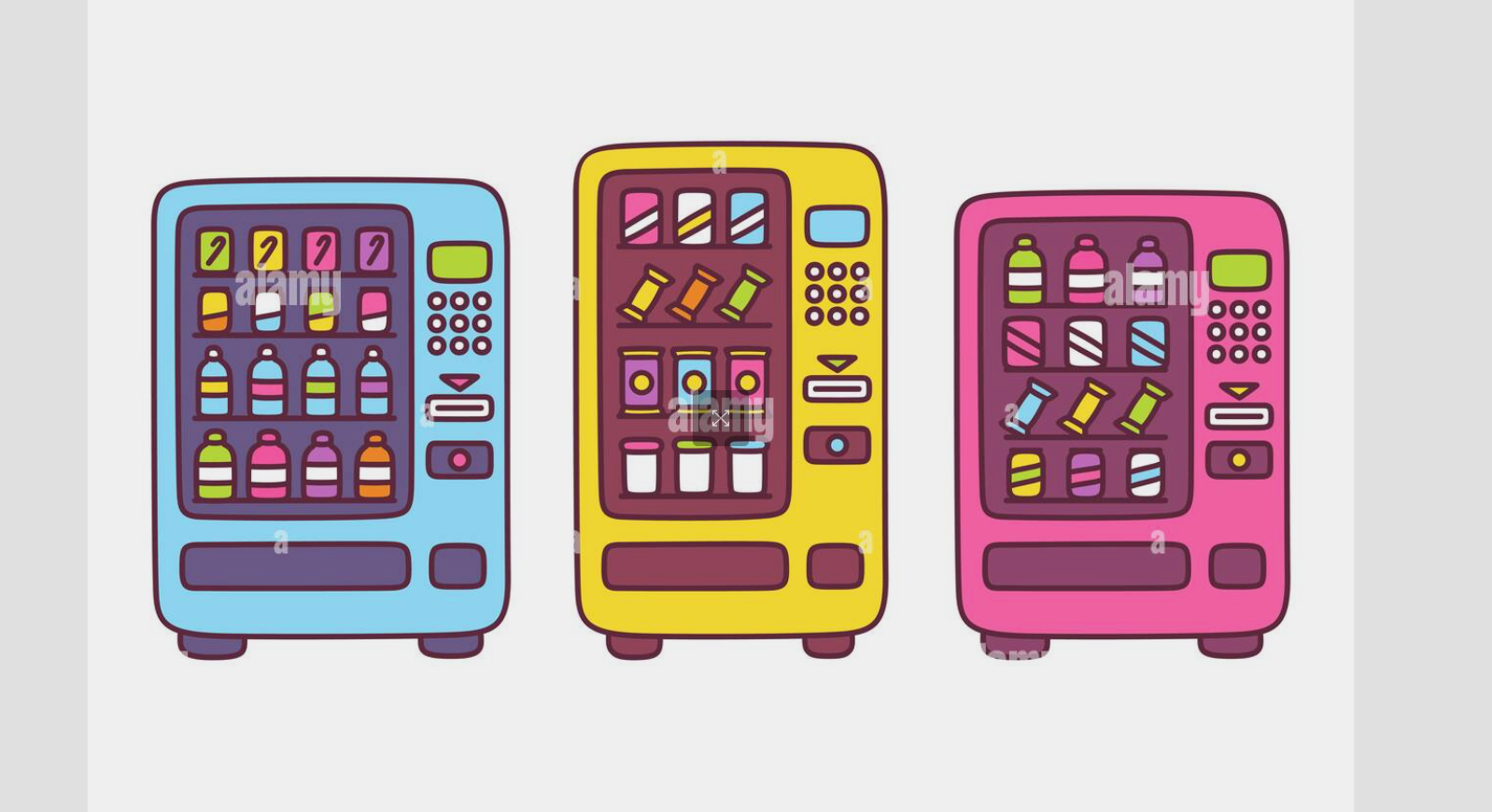


Figure 7.3: Shown above are a few drawings that we found on the internet that we have bought for full rights. We plan to make this the logo for our app. Additionally, we plan to have these drawings become the icons that will show up on the google maps API.

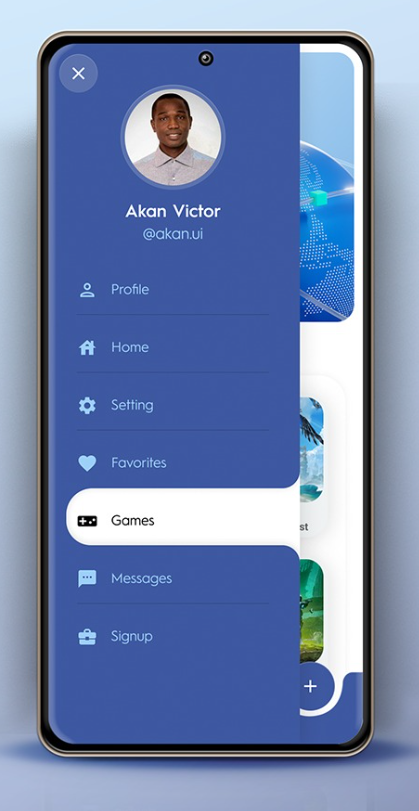


Figure 7.4: As shown above, we plan to have a popout menu on the left side of the screen to show various options depending on whether the user is a retailer or a consumer. Additionally, this will be where the user can change their profile settings.



Figure 7.5: This blueprint above represents how we would like the vending machine contents to be displayed. As soon as the user clicks on the vending machine icon, they will be greeted with this screen detailing what is in the machine and how long ago the picture was taken. Additionally, we plan to have the prices of items listed here as well.

## 8. Glossary of terms

**Integrated Development Environmen**t - It allows the user to write, build, run, and debug code all from a single environment.

**Database Engine** - A piece of software or an API that can be integrated with your code that allows the user to manipulate data in a database.

**SQLite** - A cross-platform database engine that is often embedded into software to help manage data efficiently and often on large scale. It allows for all the features of SQL to be implemented into a code base.

**Smartphone** - A more powerful variation of the mobile phone, allowing people to perform more extensive computing tasks including, but not limited to, gaming, taking HQ videos and pictures, and geographic positioning.

**Vending Machines** - A machine that automates the dispensing of various items. Examples include snacks, drinks, supplies, lottery tickets, and full-sized meals.

**Framework** - A programming tool that provides specific solutions to the general software problems that the framework aims to solve. It allows developers to speed up development by reusing this general solution and building upon it.

**Software Development Kit** - A broad group of tools that can be used to fast-track the development of software. This ranges from hardware that can interface with specific code to frameworks to simple libraries.

**Dart** - An object-oriented programming language created by Google. It is often used for web and mobile app development and can also be used to compile JavaScript code.

**Flutter** - An open-source framework developed by Google. The framework is used to develop cross-platform applications from a single codebase and is paired with the programming language Dart.

**Google Maps Software Development Kit** - An open-sourced google maps SDK by Google. This development kit will allow us to create the map needed to show vending machines near the user's location.

**iOS** - A mobile operating system developed by Apple. It is often used in iPhone smartphones. It allows the user to use apps that come with the OS or apps that are installed through the OS’s AppStore.

**Android** - The term is often used synonymously with the name of the smartphone. It is an operating system based on the Linux kernel that is often used in mobile smart devices. It has its own Appstore called the play store.

**Crowdsourcing** - A specific process of gathering data. This process involves gathering data from the input of groups of people.

**Mobile Apps** - Also known as an app, it is a piece of software designed to run on mobile devices, most commonly smart devices.

**Vendi** - The mobile app in question. An app that allows the user to locate, filter, and participate in crowdsourcing for vending machines.

**User Interface** - The layer between machine code and human input. It allows users to interact with a piece of software without directly interacting with the code it is running on.

**App Security** - Implementing security measures within an application to prevent any attempts from an unauthorized user to access, modify, and/or hijack data.

**User Account** - A body of data comprised of a username/email, password, and other required information. Accessing an account gives you access to data that the account would normally have access to.

**Git** - An open-source solution to source control. It is used to track and manage changes to files on local and remote repositories. It is often used by programmers to work on large projects.

**GitHub** - An online hosting service for source control. It hosts remote repositories and is typically used for hosting open-source software solutions.

## 9. List of references

**Problem Domain Book:**

Katz, Michael, et al. *Flutter Apprentice (Third Edition): Learn to Build Cross-Platform Apps*. 3rd ed., Razeware LLC, 2022.

The book describes the different ways one can use the Google framework, Flutter, to develop Cross-platform apps. Cross-platform being iOS, Android, and web applications. It goes over UI development, using SQLite to manage data in the app, developing using states, and deploying the app to the AppStore. It holds a heavy emphasis on the cross-platform compatibility of apps being developed.

Kreibich, J. A. (2010). *Using SQLite*. O'Reilly.,

This book gives an overview on the SQLite database engine that is designed to be embedded directly into a database application. This book covers the SQLite C programming API, SQL language, and relational database design. This reference is important in understanding the SQLite language application and implementation.

*Howe, J. (2009). Crowdsourcing: Why the power of the crowd is driving the future of business. Three Rivers Press.*

This book gives an overview of what crowdsourcing is, key concepts of it, and the significance it has on businesses. It also gives insight on various working models with crowdsourcing implementation as well as how to leverage these ideologies to accomplish many different tasks and feats.

## 10. Contributions of team members

**Catherine Wedin: 5.5 hours**

Worked on functional requirements, interview questions, a summary of Indiana Guerrazzi’s interview, requirement traceability matrix, and table of contents.

**Joey Paschke: 5 hours**

Worked on idea snapshots, functional and non-functional requirements, helped other team members with their tasks, and answered interview questions.

**Charles Dunn: 5 hours**

Worked on the user cases section, user case model, user case detailed templates, interviewed and summarized Joey Paschke’s responses, and the lists of resources.

**Maxwell Synard: 6 hours**

Worked on introduction, created most of the interview questions, wrote the interview for Clayton Synard, found and summarized the problem domain book, and created the glossary of terms.