



GNG 2101

**Design Project User and Product Manual**

**Project Deliverable I**

Submitted by:

**Group 1.2 - Personal Safety Project**

Mohammed Riyad El Azhari, 300145946

Zachary Ebanks, 300013189

Ayman Fakri, 300120735

Payton Gomes, 8854858

April 10, 2022

University of Ottawa

# Table of Contents

---

Introduction	<b>1</b>
Overview	<b>1</b>
Conventions	4
Cautions & Warnings	5
Getting started	<b>5</b>
Configuration Considerations	6
User Access Considerations	6
Accessing/setting-up the System	7
System Organization & Navigation	7
Navigation	8
Exiting the System	9
Using the System	<b>9</b>
Start Page	9
Troubleshooting & Support	<b>13</b>
Error Messages or Behaviors	13
Special Considerations	14
Maintenance	14
Support	14
Product Documentation	<b>15</b>
Subsystem 1 of Prototype	16
BOM (Bill of Materials)	16
Equipment list	16
Instructions	17
Testing & Validation	17
Conclusions and Recommendations for Future Work	<b>21</b>
Bibliography	<b>22</b>
APPENDIX I: Design Files	<b>23</b>
APPENDIX II: Other Appendices	<b>24</b>

## List of Figures

---

Figure 1: Login Page	3
Figure 2: Sign Up Page	3
Figure 3: Profile Page	3
Figure 4: Changing Font Size	3
Figure 5: App States and API Communication	4
Figure 6: Front and Back End Workflow	4
Figure 7: Sign Up and Login Pages	6
Figure 8: Sign Up Page	10
Figure 9: Home Page	11
Figure 10: Profile Page	13
Figure 11: Login with Correct Fields	17
Figure 12: Login with Incorrect Password	17
Figure 13: Login with Incorrect Email	17
Figure 14: Sign Up with Incorrect Fields	18
Figure 15: Sign Up Correct Name	18
Figure 16: Sign Up Email and Name Correct	18
Figure 17: Sign Up all Fields Correct Except Password	19
Figure 18: Sign Up Password Less Than 5 Characters	19
Figure 19: Sign Up all Fields Correct	19
Figure 20: Check In Page	20
Figure 21: Check In Completed	20
Figure 22: Check In Notification Received	20
Figure 23: Check In Notification Received 2	21
Testing & Validation	18
Conclusions and Recommendations for Future Work	22
Bibliography	23
APPENDIX I: Design Files	24
APPENDIX II: Other Appendices	25
APPENDIX II: Other Appendices	25

## List of Tables

---

Table 1. Acronyms	v
Table 2. Glossary	v
Table 3. Referenced Documents	23

# List of Acronyms and Glossary

---

Table 1. Acronyms

Acronym	Definition
---------	------------

Table 2. Glossary

Term	Acronym	Definition
------	---------	------------

# **1 Introduction**

This User and Product Manual provides the information for any user of the application to properly use all of its included features. The application was built to allow the users to check in daily and alert their loved ones if a check in is not completed. This allows their loved ones to know to check in on the user should they receive an emergency message from the application. The goal is to remove stress from the user and responsibility from their loved ones to ensure they are checking in on them enough.

The User and Product Manual will first give an overview of the product and conventions and cautionary information about the product will be provided. Then the document will go over the steps for the first use of the product and how to set up the initial check in.

This is followed by the steps and information for continued use of the product, including how to create a new check in time, how to modify the font size and how to add additional security questions. This is followed by information on any troubleshooting that the user may need to do and steps on how to do it.

The Manual provides the information necessary to reconstruct the application at its current prototype state. This will allow future users to rebuild and improve upon the current version of the application. The information included will include the code required to build the application along with the third party services used to properly run the application.

# **2 Overview**

The problem we have set out to solve is that for people who live alone, it could be a

significant amount of time before someone realizes something has happened to them. Therefore, the team had to come up with a product whose main goal was to keep track of the safety of the users, and notify their loved ones when their safety is at stake. The unique part about this solution is that it is a simple process, yet, very efficient in keeping track of the safety of the user and notifying the emergency contacts.

A few of the client's fundamental needs are that a system is needed to alert close family of friends if something happens to them. Check-in notifications can be customized by the user to work around a specific schedule, meaning that the user must be able to set a time for the daily check-in. Also, checking in cannot be done accidentally. This requirement is important because it covers an edge case which can definitely happen, and it means that the check-in process should include a few steps that make sure that the user is aware of the decision he's or she's making and that it is actually him/her completing the check-in. The app should have a notification system with failsafe measures in case a single checkup is missed. In other words, the app should have a process that gives the user extra chances to complete the check-in, and should they not be able to complete any of them, the application should proceed with contacting the emergency contacts.

The product introduces an optimal solution that ensures the safety of its users. It performs an automated process that asks the user to check-in once daily. If the check-in is not performed, backup check-ins will be sent and after emergency contacts will be notified. Our solution respects the privacy of our users and requires no additional hardware. It is also very easy to use with a less than 5 second check-in time and is an automated process.



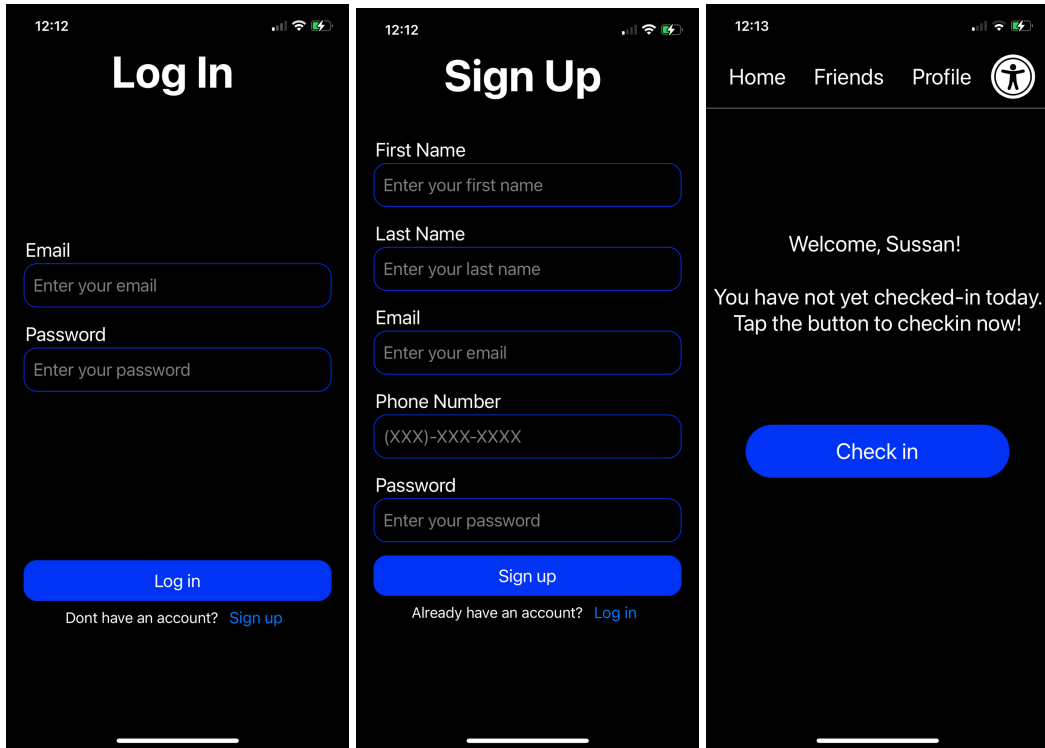


Figure 1. Login Page

Figure 2. Sign Up Page

Figure 3. Profile Page

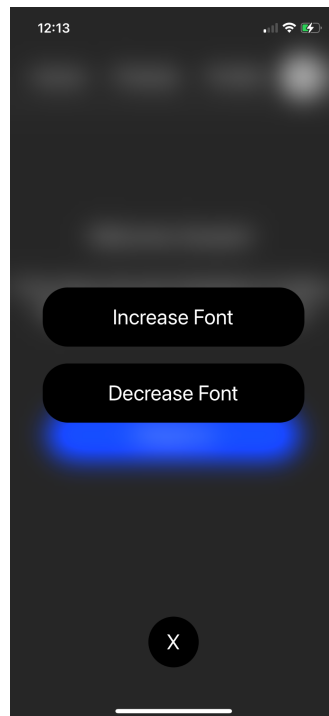


Figure 4. Changing Font Size

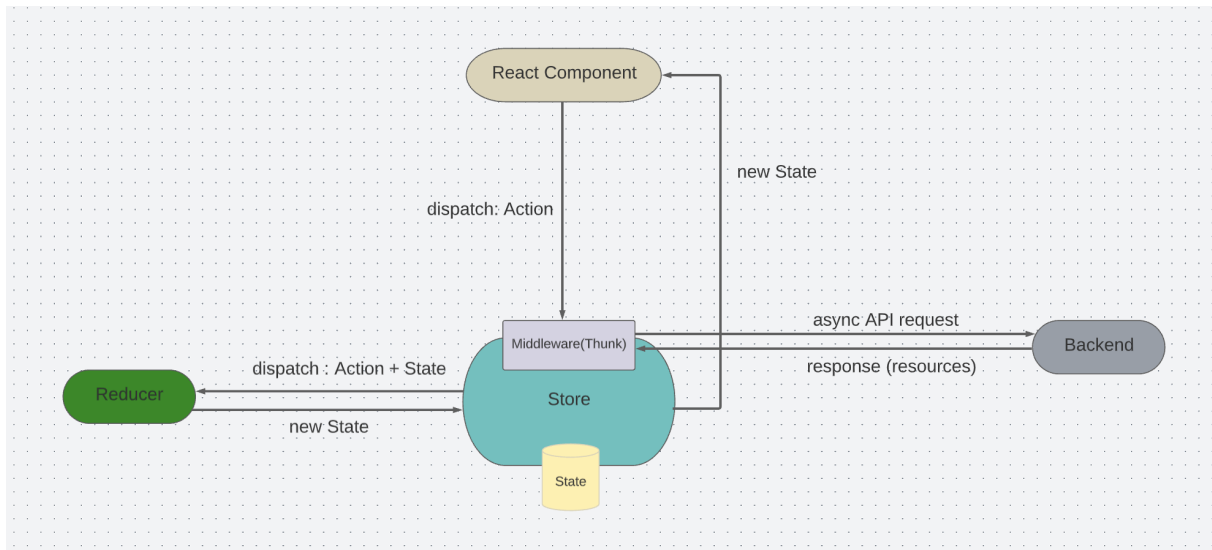


Figure 5. App States and API Communication

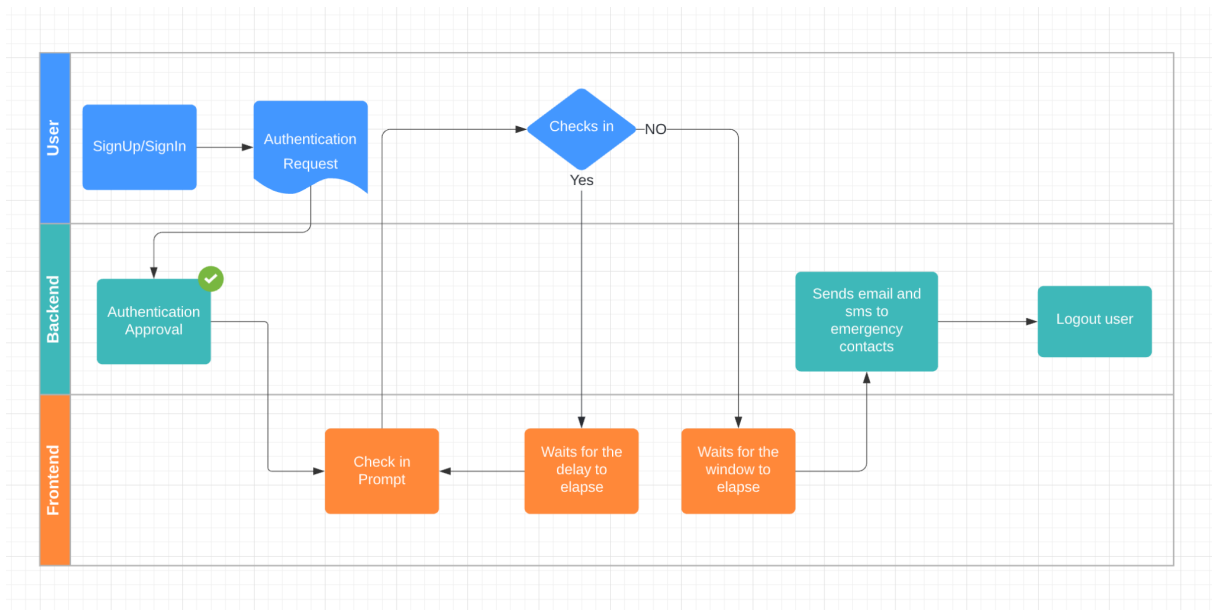


Figure 6. Front and Back End Workflow

## 2.1 Conventions

Not applicable.

## **2.2 Cautions & Warnings**

The personal safety app was developed to be safe and effective. This guide is intended to assist the user in navigating the application more easily. This application is not intended to be used in risky situations. If this is chosen to be ignored by the reader, the application may be used at your own risk.

## **3 Getting started**

First of all, to ensure good security of our application, the implementation of an authentication page is crucial. Thus, we designed a sign up screen for new users. They will need to enter their email, password and username. After entering their credentials, they will be able to be able to set up their customized settings for the daily check-in. These settings consist of the time of the check-in, the number of check-ins per day, and a set of security questions. These security questions will later be used as they are part of the check-in authentication. For those who already have an account, they will need to click on the “Login” button which leads to the login screen to enter their email and password.

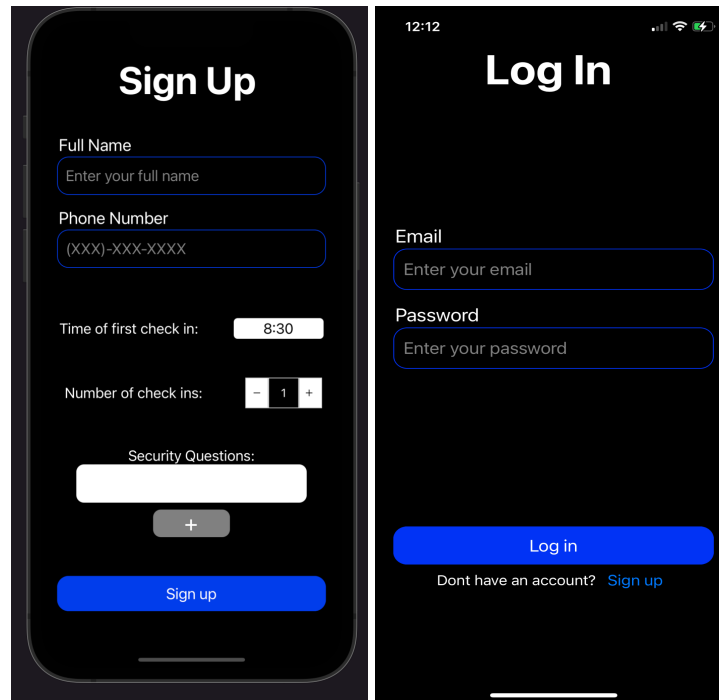


Figure 7. Sign Up and Login Pages

### 3.1 Configuration Considerations

There are no configuration considerations the user should be aware of.

### 3.2 User Access Considerations

The vast majority of our users will be people aged 50 or older that have mobility issues. The only restriction for our users will be for those who are blind as the Text to Voice feature which allows the user's device to read text on the screen is not currently available. The user needs to have an android or IOS phone in order to download the app.

### 3.3 Accessing/setting-up the System

In order for the user to set up the app, they need to download it from their operating system application store (App store & Google play), create an account through the sign up page, and they're ready to use it.

### 3.4 System Organization & Navigation

#### 3.4.1 System Organization

The app is divided into two parts:

**Front End:** Is the code that is executed on the user's phone and it allows the rendering of the UI components that the user can interact with. We used a state manager for the UI components, this allows us to include accessibility features that require applying changes to the whole app in an efficient manner, such as: font size modification.

**Back End:** Is the code that is running on our servers and that ensures that the app's behaviour is maintained even if the app is closed on the user's phone. Initially the front-end was responsible for keeping track of when the user needed to check in but we realized that it is not a scalable solution and it caused the app to be tightly coupled.

Moving it to the back-end allowed us to easily integrate emergency sms and email sending capabilities, since each functionality was isolated as an independent service which allowed us to perform changes without worrying about the other services. As for the database, we went with PostgreSQL, a relational database. Setting up the schemas for the tables was the hardest part. But after it was just a matter of storing data.

### 3.4.2 Navigation

The app is composed of 4 pages:

**Sign up:** this page allows the registration of the user, it is also where the check in preferences are configured initially .It only appears once, when the app is opened for the first time. There isn't a sign in page - this was to adhere to our client's needs, as she didn't want to deal with logging in everytime she wants to use the app.

**Home:** After signing up the user is immediately taken to the Home page where the check it is done. Performing the check in only takes up to 10s; Clicking on the button, answering one of the security questions the user setup and using biometrics information (FaceID/TouchID).

**Profile:** In case the user wanted to change his check in preferences (number of check ins, time of first check in and security questions) they just need to click on the profile button on the navigation bar on top of the screen.

**Accessibility:** If the user wants to adjust their font size they just need to click on the accessibility icon on the navigation bar on top of the screen. Now they can click on the increase or decrease the font button until the font size is appropriate.

Note: the navigation bar is always available on top of the screen to allow the navigation between the 3 main pages.

### **3.5 Exiting the System**

Exiting the system depends on the phone and that software version that user has. Usually, there's a button on the bottom of the screen that displays all the active apps and the user can terminate ones from there. On new iPhones, all you need to do is swipe up from the bottom of the screen, swipe left to find the app you want to close, then swipe it up and off the top of the screen. There's however no built-in way to terminate the application from within the system as we figured it isn't essential.

## **4 Using the System**

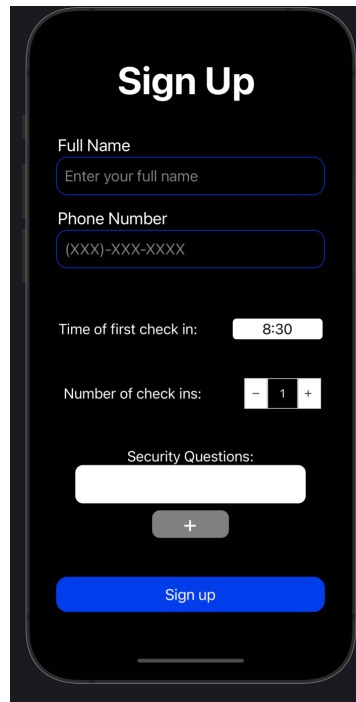
The following subsections provide detailed, step-by-step instructions on how to use the various functions or features of the Personal Safety application.

### **4.1 Start Page**

When the application is opened for the first time the user will be prompted to enter their full name as well as their phone number. Only 10 digit phone numbers will be accepted by the application. The time of the first check in can be set by clicking the time and scrolling to the desired hour and minute for the check in.

The number of subsequential check-ins throughout the day can be changed by pressing the '+' or '-' buttons to increase and decrease the number of check-ins, respectively. The number of check ins throughout the day will be evenly distributed throughout the 24 hour period. Check-in security questions can be set by clicking on the '+' button underneath the security

question and entering the question in the top box and the answer in the bottom. Once all the fields have been filled, the user can press the ‘Sign up’ button to save their information.

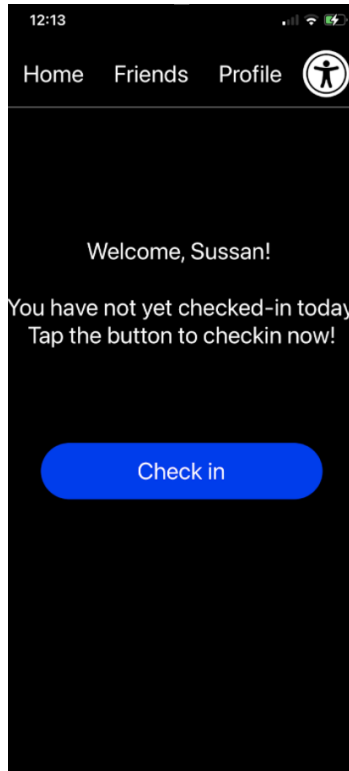


*Figure 8. Sign Up Page*

## 4.2 Home Page

Once the user has completed the information on the start page, or when the user opens the application and has already completed the start page information, the application will default to the home page. From the home page the user can complete their check in by pressing the blue button in the middle of the page. They are also able to navigate to the profile page using the ribbon at the top of the page. Lastly, they can access the accessibility tooltip by pressing the icon in the top right corner of the page.





*Figure 9. Home Page*

### **4.3 Check in**

Once the user presses the 'check in' button on the home page, the user will be prompted to enter their touch ID, face ID or phone password. Once this is successfully completed, the user will be prompted to choose one of their predefined security questions and enter the answer for the given question. Once the answer has been verified, the check in is complete and the application will return to the home page, with the page saying that the check in has been completed for the time period.

### **4.4 Accessibility Tool-Tip**

The accessibility tool-tip is navigated to by pressing the icon in the top right corner of the home page. From the accessibility tool-tip the user can increase or decrease the font by pressing

the 'Increase font' and 'Decrease font' buttons respectively. The buttons can be pressed as many times as the user would like until the font size is ideal for them. These changes to the font size will be applied to all text throughout the application.

## **4.5 Profile Page**

From the profile page, the user can enter and edit the details that were initially input on the start page. The time of the first check in can be changed by clicking the time and scrolling to the desired hour and minute for the check in.

The number of subsequential check-ins throughout the day can be changed by pressing the '+' or '-' buttons to increase and decrease the number of check-ins, respectively. The number of check ins throughout the day will be evenly distributed throughout the 24 hour period. Additional check in questions can be added to the check in question list. The user just clicks on the '+' button underneath the security question and enters the question in the top box and the answer in the bottom. Once all desired changes have been made the user must press 'save' at the bottom of the page to save and apply all changes.

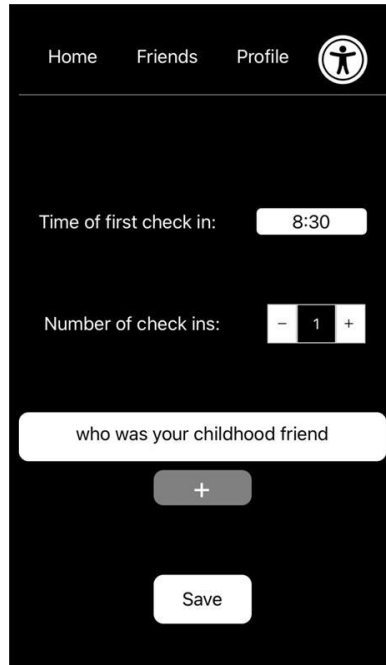


Figure 10. Profile Page

## 5 Troubleshooting & Support

Given that our application has been designed to be simple to use, we hope that the majority of the error messages expressed allow the user to solve the problem without having to consult this document. However, it is also sometimes difficult to understand or see the errors made. Thus, this document will explain all the error possibilities that the reader can receive.

### 5.1 Error Messages or Behaviors

#### 1. Email not found:

This error means that the email entered doesn't exist in the database. This might be caused by the fact that the user is entering a wrong password, the user hasn't created an account with this email yet, or the email format is incorrect.

## **2. Password incorrect:**

This error means that the password entered doesn't correspond to the email entered.

## **3. Full name is empty:**

This error occurs in the sign up page when the field of the full name is left empty.

## **4. Phone number is empty:**

This error occurs in the sign up page when the field of the phone number is left empty.

## **5. Phone number is incorrect:**

This error occurs in the sign up page. It might be caused when the user enters a phone that doesn't have 10 digits.

## **5.2 Special Considerations**

There are no special considerations the user should be aware of.

## **5.3 Maintenance**

In order to keep our application constantly interactive, modern and with as few bugs as possible, our team will ensure that Personal Safety is updated regularly. This will not only prevent failures but also add the new features that Adalo offers each month in terms of design and functionality.

## **5.4 Support**

If there is any problem with the app, please close the app first and restart it. If that doesn't work, please restart your phone. If the application is still not responding, please contact our support department at the following email addresses : [riyad.azhari7@gmail.com](mailto:riyad.azhari7@gmail.com), [theonlyayman@outlook.fr](mailto:theonlyayman@outlook.fr) . The user having the problem can send a detailed description of it. Our

team makes sure to respond to our customers as soon as possible (1 day maximum for a simple problem).

## 6 Product Documentation

The design of our solution was based primarily on the needs of our client as well as other applications on the market. These two elements allowed us to have an idea of our solution (features, sections, security...).

**Back End:** Our team had to decide on a programming language that all the members are familiar with and that provides api services and frameworks that might be useful to our product. Our decision making process on this matter was based on the knowledge and level of comfort in coding. Our top pick was Ruby on Rails and that's what we decided to work with.

**Front End:** Deciding on a front end framework was one of the most crucial decisions made by our team. It required long discussions and brainstorming sessions in order to come up with an optimal tool. We decided to go with react native as most members of the team are comfortable with it and it provides the ability to build cross-platform applications.

**Notification System:** Since our application included a notification requirement, our team needed to figure out the best notification system within our budget. Luckily, we found a service called Pusher that provides 200k messages per day.

**Messaging System:** Since our application included a messaging requirement, our team needed to figure out the best messaging system within our budget. Luckily, we found a service called twilio that provides a free trial. This trial allows you to send thousands of text messages.

Finally, we had to incorporate all of these services and bring them all together into one product.

## 6.1 Subsystem 1 of Prototype

### 6.1.1 BOM (Bill of Materials)

Both APIs use essential services for the app, a notification system and a text messaging service. They are both cheap and can be funded with money from the premium service of the application. A free trial will be used for twilio for the time being, and pusher has a free “Sandbox” option that is used in the safety application.

Item	Description	Units of measure	Quantity	Unit Cost	Extended Cost	Link
twilio	SMS aPI	number of texts	50	\$0.0075	\$0 - free trial	<a href="https://www.twilio.com/sms/pricing/ca">https://www.twilio.com/sms/pricing/ca</a>
pusher	Notification aPI	number of notifications per day	50	\$0 for 200k messages per day	\$0	<a href="https://pusher.com/channels/pricing">https://pusher.com/channels/pricing</a>
<b>Total project cost without taxes and shipping</b>					<b>\$0</b>	
<b>Total project cost with taxes and shipping</b>					<b>\$0</b>	

### 6.1.2 Equipment list

As the project was entirely software and coding based, no equipment was needed in order to construct it, besides an IDE for the application.

### 6.1.3 Instructions

It's a built-in process within the app.

## 6.2 Testing & Validation

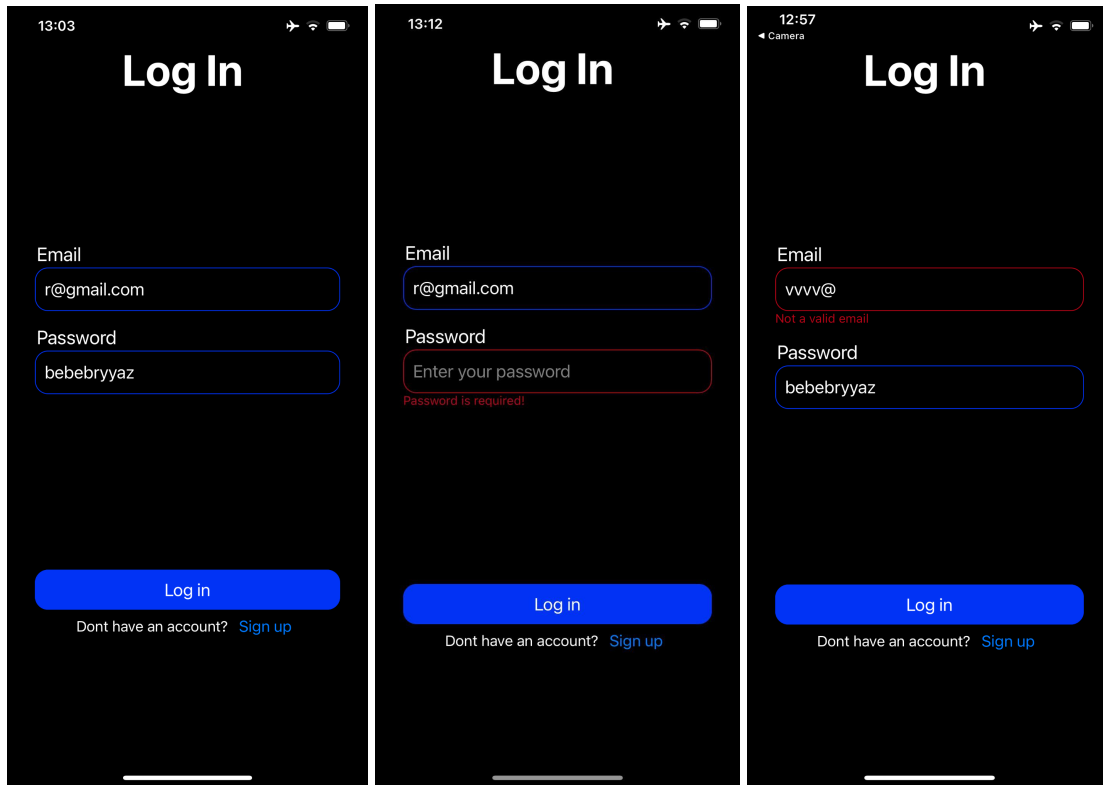


Figure 11. Log in with correct fields    Figure 12. Log in with incorrect password    Figure 13. Log in with incorrect email.

The figures 1, 2, and 3 represent tests related to the field validation feature for the Login process. As shown in figure 3, When the user enters a wrong email, the system updates the state of the field in real time, it changes the border to red and returns a red error message stating “ not a valid email”. As shown in figure 2, When the user enters an empty password, the system updates the state of the field in real time, it changes the border to red and returns a red error message

stating “ not a valid email”. As shown in figure 1, When the user enters a correct password and email, the system updates the state of the fields back to normal. Figure 8 shows the state of the fields when the login is ready to be processed.

12:57  
Camera

## Sign Up

First Name  
Enter your first name  
First name is required!

Last Name  
Enter your last name  
Last name is required!

Email  
Enter your email  
Email is required!

Phone Number  
(XXX)-XXX-XXX  
Phone number is required!

Password  
Enter your password  
Password is required!

Sign up

Already have an account? [Log in](#)

12:59  
Camera

## Sign Up

First Name  
x

Last Name  
x

Email  
@gmail.com  
Not a valid email

Phone Number  
(XXX)-XXX-XXX  
Phone number is required!

Password  
Enter your password  
Password is required!

Sign up

Already have an account? [Log in](#)

13:00  
Camera

## Sign Up

First Name  
x

Last Name  
x

Email  
r@gmail.com

Phone Number  
(XXX)-XXX-XXX  
Phone number is required!

Password  
Enter your password  
Password is required!

Sign up

Already have an account? [Log in](#)

Figure 14. Sign Up with Incorrect Field

Figure 15. Sign up- Correct Name

Figure 16. Sign Up Email and Name Correct

The figures 4, 5, and 6 represent tests related to the field validation feature for the Sign up process. As shown in figure 4, When the user leaves all fields empty and tries to login, the system updates the state of the fields in real time, it changes the border to red and returns a red error message accordingly. As shown in figure 5, When the user enters a correct full name, the system updates the state of the field back to normal. Updating it back to normal means that the value is



now accepted. As shown in figure 6, When the user enters a correct email, the system updates the state of the field back to normal.

13:00  
Camera

## Sign Up

First Name  
x

Last Name  
x

Email  
r@gmail.com

Phone Number  
1293019293

Password  
Enter your password

Password is required!

Sign up

Already have an account? [Log in](#)

Figure 17. Sign up all fields correct except password

13:00  
Camera

## Sign Up

First Name  
x

Last Name  
x

Email  
r@gmail.com

Phone Number  
1293019293

Password  
x

Password should be more than 5 characters

Sign up

Already have an account? [Log in](#)

Figure 18. Sign up password with less than 5 characters

13:00  
Camera

## Sign Up

First Name  
x

Last Name  
x

Email  
r@gmail.com

Phone Number  
1293019293

Password  
xxxxxx

Sign up

Already have an account? [Log in](#)

Figure 19. Sign up all fields correct

The figures 7, 8, and 9 also represent tests related to the field validation feature for the Sign up process. As shown in figure 7, When the user enters a correct phone number, the system updates the state of the field back to normal. As shown in figure 8, When the user enters a password with less than 5 characters, the system updates the state of the field in real time, it changes the border to red and returns a red error message stating “ password should be more than 5 characters”. As shown in figure 8, When the user enters a correct password, the system updates

the state of the field back to normal. Figure 8 shows the state of the fields when the sign up is ready to be processed.



Figure 20. Check In Page

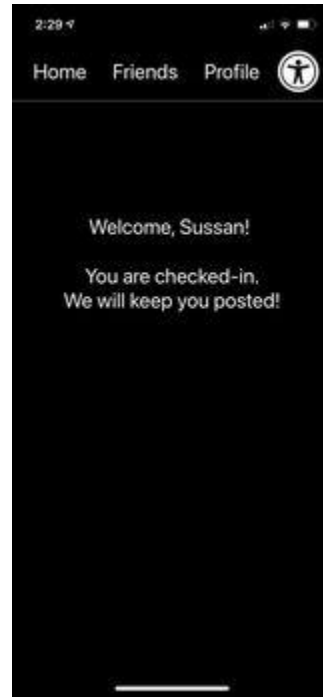


Figure 21. Check In Completed

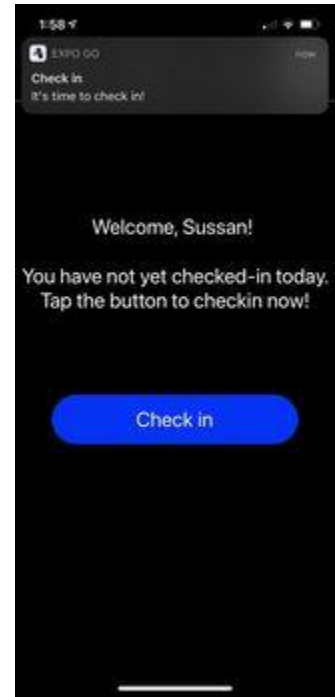


Figure 22. Check In Notification Received



Figure 23. Check In Notification Received 2

Figure 10 illustrates the state of the home page before the check-in is completed. After pressing the check-in button, and after a predefined amount of time has passed, the user receives a notification mentioning that it's time to check-in. There are two cases for the notification process, the first one is receiving a notification when the application is open and the second is when the application is closed. Figure 12 shows the first case and Figure 13 shows the second one. As shown in figure 11, When the check-in button is pressed, the message displayed in the home screen changes accordingly, letting the user know that the check-in was successfully completed. It emphasizes software feedback.

## 7 Conclusions and Recommendations for Future Work

A lot was learned throughout the development of the application. Firstly, the importance of a concept map and plan to ensure that all of the pieces will integrate together well. This will also allow for the ability to expand the application and add extra features in the future, should there be a need. This project pushed the team out of their comfort zone and to learn and acquire new skills. Each individual on the team was able to expand their current knowledge of coding and business practices. In the development of the application the group was challenged design wise to make the user interface as accessible as possible for as many users as possible.

Some key features that could be included in future updates are a friends section that allows users to remain in contact with their friends on the app and be able to see their location and check in status. Another update will include more customizability with the user interface colours. Lastly an update would include the ability for the user to delete possible check in questions and to modify their emergency contacts and the message that is sent to the contacts.

## **8 Bibliography**

## APPENDICES

### 9 APPENDIX I: Design Files

Table 3. Referenced Documents

Document Name	Document Location and/or URL	Issuance Date
MakerRepo Document	<a href="https://makerepo.com/riyadelazhari/1192.personal-safety-">https://makerepo.com/riyadelazhari/1192.personal-safety-</a>	March 27, 2022
Wrike Project Plan	<a href="https://www.wrike.com/frontend/ganttchart/index.html?snapshotId=tv1oBpWdW2VUuZpo2n5avHpm6sOmRQtx%7CIE2DSNZVHa2DELSTGIYa">https://www.wrike.com/frontend/ganttchart/index.html?snapshotId=tv1oBpWdW2VUuZpo2n5avHpm6sOmRQtx%7CIE2DSNZVHa2DELSTGIYa</a>	February 6, 2022

## **10 APPENDIX II: Other Appendices**