Deliverable G:

Prototype II

University of Ottawa
GNG 2101 B
November 5th, 2020

Adam Walters 300109768
Eric Chen 300136076
Brendan Sommers 300115531
Jarett Goodwin 300074553
Tianchen Cai 300127732

Table of Contents

Introduction

The purpose of this document is to analyze and understand customer needs. And Introduce the equipment improved by this group to customers and laboratory leaders. Therefore, in this document, we have a Client Feedback part of collecting and analyzing the needs, then prototype and test parts to Full display of our models and test results. We divide our team into two parts. One group responsible for software; another for the hardware. In the software prototype part, we used Android Studio to simulate an Android phone on a Windows computer to achieve a software test. For the hardware team, we retrieved components from the internet and Makerspace, and then assembled them. Because of COVID-19, our two teams are in different cities, and as such our testing was done in two independent groups.

Client Feedback

We had an excellent third meeting with our client, where we showed them our current prototypes, and they were very impressed with the progress we had made. Initially, we discussed with them if it was possible to schedule another meeting to show them our second set of prototypes, as we want feedback on them before Design Day. We are still in communication with them and will hopefully be able to do one in the coming weeks. In the next part of the meeting, we showed them our first prototype and estimated budget, and they liked it. They also appreciate our chosen name, Pathfinder. They had one question, how will we know when a beacon is low on power. We have now planned to include an LED light that shows the battery's status, as we deemed using Wi-Fi to transmit it would have significantly increased our battery consumption. To conclude the meeting, we discussed the critical locations within the library. We concluded that we would prioritize the main entrance and service desk as key navigation locations for our final prototype. Going forward, we will try and develop a well-built physical prototype with a battery indicator and an app that functions well.

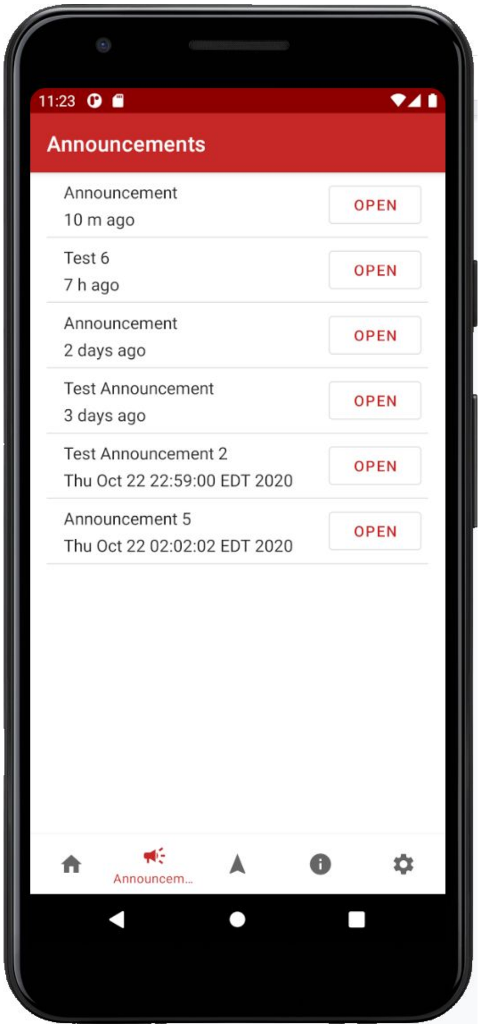
Prototypes

Software Prototype

For this second software prototype, our main goals were to test the viability of the mockups created for Prototype 1 in *Deliverable D*, the efficiency of using an Android app as our platform as chosen in *Deliverable C*, and the reliability of storing data (such as announcements) online.

To test these three objectives, we moved from the Adobe Illustrator-based UI mockups of the previous set of prototypes to an actual Android app, the source code for which is available on GitHub here: https://github.com/ricetech/pathfindr. To focus our testing where it was needed and to make sure not too much development would be wasted in the event of needing to switch platforms, we did the testing of all three objectives listed above by only implementing the displaying of Announcement Titles and Timestamps retrieved from our online database provider, Firebase Cloud FireStore. We have decided to **defer** the implementation of the other functions (Navigation, Location Info & Employee Access) until we determine that the platforms that we have chosen are proven to be reliable enough for our clients through the testing performed in this deliverable.

As discussed, this prototype began with an Android app developed from scratch. While the UI is loosely based on the mockups created previously, the app is developed entirely from the ground up to confirm around Google's Material Design Guidelines. We were able to successfully implement a design similar to the mockups with the slight change of using the newer bottom navigation menu instead of the slide-out navigation menu seen in previous versions of Android. The design of the app and the Announcements page can be seen in the screenshot below:



Announcement
10 m ago

Test 6
7 h ago

Announcement
2 days ago

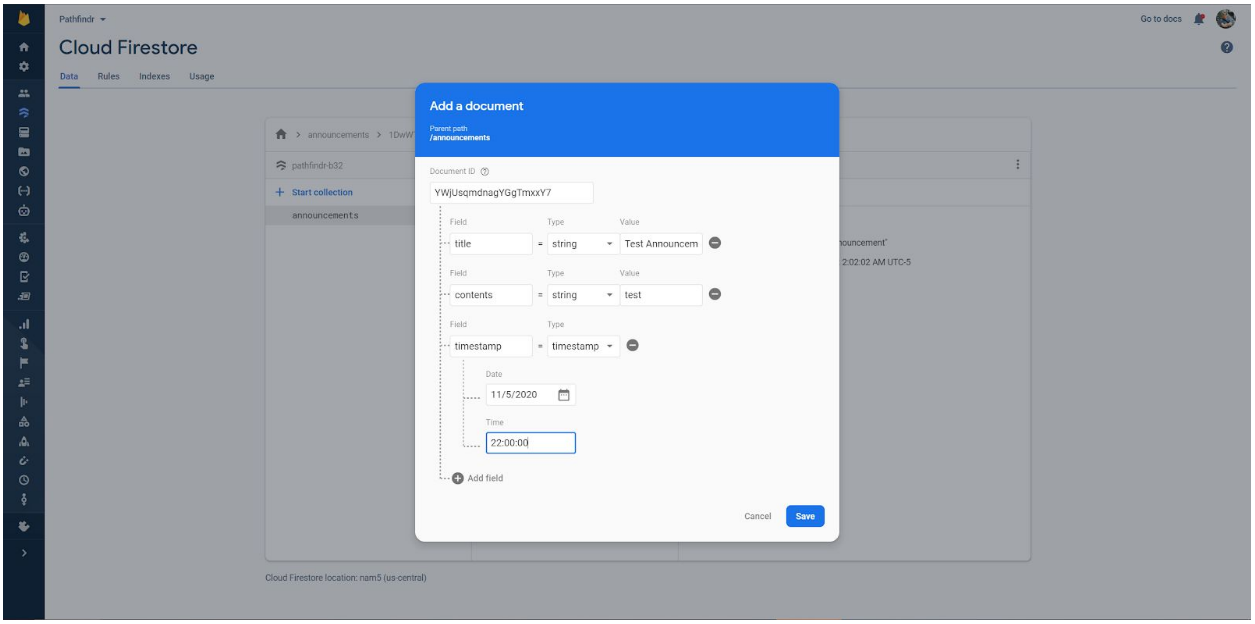
Test Announcement
3 days ago

Test Announcement 2
Thu Oct 22 22:59:00 EDT 2020

Announcement 5
Thu Oct 22 02:02:02 EDT 2020

To test online data storage, we created a new Firebase project housing a Cloud FireStore Database, which stores data in the NoSQL format. We chose this because our group has had experience with this service before and knows that it is reliable and updates quickly whenever new data is added.

As we have not implemented the employee-side interface for this prototype, we tested our “time to broadcast announcement” metric by manually creating a database entry through the Firebase web interface. It’s important that this web interface **<u>will not</u>** be used (nor accessible) in the final app, but is simply being used for testing purposes so that we can input values into the database. The storage format of the data can be seen below on a screen displaying the Firebase Web Interface:



Add a document

Parent path
/announcements

Document ID: YWjUsqmdnagYGgTmxxY7

Field	Type	Value
title	string	Test Announcem
contents	string	test
timestamp	timestamp	

Date: 11/5/2020
Time: 22:00:00

To test, we measured the time between clicking the “Save” button (which uploads the data into the database) and the announcement appearing in the list of announcements on all three of our testing devices running the same version of the app (A OnePlus 3, a Samsung Galaxy S9+, and an emulated Google Pixel 3a running on a Windows Computer) 5 times and took the average, with a result of 1592 ms - well below our ideal value of 5000 ms.

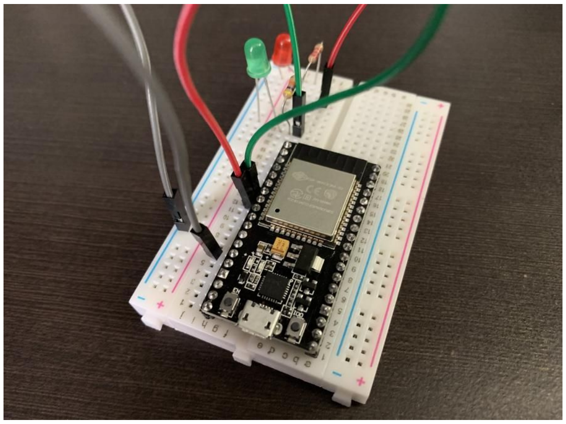
This time is not 100% accurate compared to the final product, as the final product will verify all of the inputted values for validity before sending it to the database. However, it still gives us a good idea of how Firebase performs in general, and we are confident that the time measured in the final app will not exceed our ideal values.

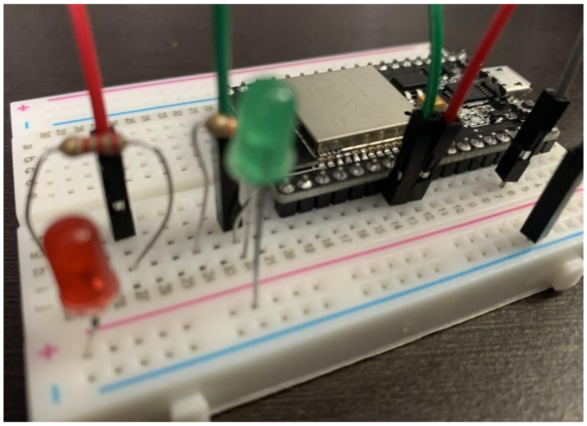
Based on the successful testing conducted in this deliverable, we will continue to develop the app using our current system of an Android-based app storing data in Firebase Cloud FireStore as it has proven to work quickly and reliably. We will also work on getting feedback from the client regarding the final format of our app now that it is in development, as well as implementing the remainder of the app’s required functions in a user-friendly manner.

Hardware Prototype

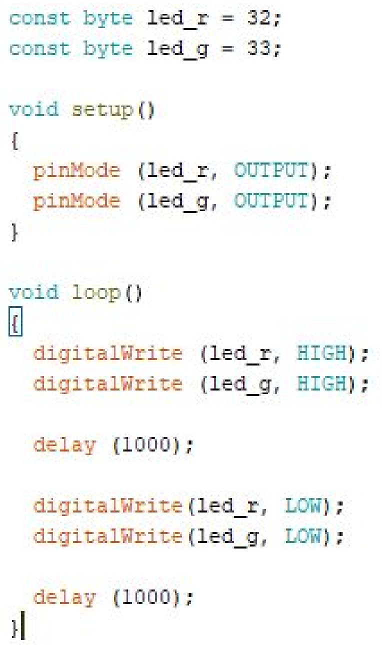
In order to continue the development of the beacon, we've worked to create a higher fidelity prototype which will more closely reflect the final version of the beacon case. In order to do this, the dimensions of the protoboard acquired by the team and the AA battery holders were used to determine the overall size of the case. Additionally, we determined which screws will be required to create this beacon case, and included those within our CAD model. For future iterations, we plan to optimize the shape of the case by reducing its maximum dimension to reach our target specification. We also plan to minimize the weight of the case in the final prototype to get closer to the target specifications defined in *Deliverable B*.

For the second prototype, we began to test simple circuits in order to get a better understanding of the ESP32 via practical experience by creating several circuits including one circuit with two LEDs in parallel connected to the ESP 32, pictured below:





To power the LEDs in this circuit, we wrote a code that was interpreted from the examples given to us in a previous lab where LEDs were powered by an Arduino UNO on a breadboard.



```
const byte led_r = 32;
const byte led_g = 33;

void setup()
{
  pinMode (led_r, OUTPUT);
  pinMode (led_g, OUTPUT);
}

void loop()
{
  digitalWrite (led_r, HIGH);
  digitalWrite (led_g, HIGH);

  delay (1000);

  digitalWrite(led_r, LOW);
  digitalWrite(led_g, LOW);

  delay (1000);
}
```

This code powers both LEDs simultaneously. In the future, the LEDs will not be powered using this code, since they will be used to indicate the remaining battery life of the beacon. Additionally, we hoped to test the Bluetooth range of the ESP32, however, we were unable to connect our phones to the Bluetooth signals from the module. Additionally, we measured the voltage and the amperage from the circuit which enables us to calculate the expected battery life. From this, we determined that the battery would last about three weeks using this circuit. To improve in the following iterations, we plan to make changes in the circuit by using less powerful resistors and optimizing the path of the circuit to reduce the battery consumption since the Bluetooth signals will decrease the battery life significantly.

Prototype Testing

Original Target Specifications (from Deliverable B)

#	Imp	Need #	Metric	Marginal	Ideal	Units
1	5	1, 2, 6	Time from app start to navigation start	$x \leq 30$	$x \leq 15$	Time (s)
2	5	1, 6	Customer Satisfaction	$x \geq 3/5$	$x \geq 5/5$	Subjective
3	5	3, 5, 9	The effective range of the beacon	$x \geq 6$	$x \geq 10$	Distance (m)
4	5	4	Battery Life of beacon	$x \geq 1$	$x \geq 5$	Years
5	4	7	Time to reprogram beacon locations	$x \leq 420$	$x \leq 300$	Time (s)
6	4	7, 9	Cost per beacon	$x \leq 70$	$x \leq 30$	CAD$
7	4	7, 9	Beacon Weight	$x < 750$	$x < 500$	Weight (g)
8	4	7, 9	Beacon Size	$x \leq 25$	$x \leq 10$	Size (cm)
9	4	8	Effectiveness of notification	$x \geq 4/5$	$x \geq 5/5$	Subjective
10	3	10	Time to broadcast announcement	$x \leq 6000$	$x \leq 5000$	Time (ms)

(continued on the next page, as the table does not fit)

Prototype 2 Benchmarks

#	Metric	Value Achieved	Units	Achieved
1	Time from app start to navigation start	N/A	Time (s)	NOT TESTED
2	Customer Satisfaction	3	Subjective	YES (MARGINAL)
3	The effective range of the beacon	N/A	Distance (m)	NOT TESTED
4	Battery Life of beacon	3 weeks	Years	NO
5	Time to reprogram beacon locations	N/A	Time (s)	NOT TESTED
6	Cost per beacon	17.36	CAD$	YES (IDEAL)
7	Beacon Weight	692	Weight (g)	YES (MARGINAL)
8	Beacon Size	20	Size (cm)	YES (MARGINAL)
9	Effectiveness of notification	N/A	Subjective	NOT TESTED
10	Time to broadcast announcement	1592	Time (ms)	YES (IDEAL)

Conclusion

In this deliverable, we successfully translated previous ideas and prototypes to match the Android Material Design standards, and in the software part, we met the expectations that were tested in this deliverable. However, we are still working on implementing the remainder of the app functions. In the shell part, we successfully assemble everything together, while the prototype is a little bit overweight, however, we can solve the problem by using better material, and use 3-D printing to reduce weight. To sum up, we made the prototype with an extremely high degree of completion and accuracy. And our prototype works well, it basically achieves the predetermined goal.