**Week One: Sept. 6th- Sept. 12th**

General Notes

This week, we discussed about which project we will work on as a team. We brought out 7 ideas in total but eventually decided on the Bus System project. Received great input during class presentation which gave us specific points of focus for our project. Meeting Coach Allen next week to discuss about the project.

Some points of focus for this project includes:

An app for riders to select which bus stop they are at to notify bus drivers to pick them up.

A button at bus stops for those who do not have an app to notify bus drivers that a person is there.

A button panel for riders to select destination while on the bus.

Minimize bus drivers interaction with the app so it won’t be a distraction.

Problem of the week

Our only problem this week is project selection. We had a meeting to select a project but after a 1-hour meeting, we narrowed it down to two projects. I had hoped we could come to an agreement on one, but we finally decided on the day of our presentation and selected the one we were more confident on that was original.

**Week Two: Sept. 20th- Sept. 26th**

General Notes

Project idea scope has pivoted towards schools. The purpose is to provide more transparency to administrators and parents for student pick-up and drop-off. This way, it will give parents peace of mind that their child is safely going to and from school.

Focused on learning Swift for the next couple weeks as it will be our programming language. Learning from Codecademy and LinkedIn Learning.

This upcoming week’s meeting, we must discuss about the required physical and technical requirements for this project.

Must account for application security and driving safety.

Follow FMSCA laws and regulations:

The use of a hand-held mobile telephone means:

* Using at least one hand to hold a mobile phone to make a call;
* Dialing a mobile phone by pressing more than a single button;
* Reaching for a mobile phone in a manner that requires a driver to maneuver so that he or she is no longer in a seated driving position, restrained by a seat belt.

Problem of the week

Problem this week and must figure out for next week is what to present to Maize School District. We’ll ideally need to secure all our requirements before we present our project idea to them. We’ll also must be open to ideas and suggestions to further improve our application and pinpoint areas they are highly interested in.

**Week Three: Oct. 11th- Oct. 17th**

General Notes

The market share of mobile operating systems in North America in June 2021 is 53.66% iOS and 46% Android. For that reason, we have decided to use Swift as our programming language to develop our application for an iOS system.

Graphical user interface, application, table

Description automatically generated with medium confidence

During our research for requirements, we emphasized on security and its features as we don’t want malicious parties to abuse our application but after our midterm presentation, we may step away from that. The idea was to have each parent see their child’s bus route only, but we have decided to make the application allow the users to see all the buses and their routes. We will however provide anonymity as to which bus it is and who is on the bus. This change would be more beneficial for school administrators and ease our security requirements when developing this application.

We also have decided to do more research on integrating a Raspberry Pi 3 or 4 as a GPS device that can be installed on the buses. This method would not require the bus driver’s phone as the key GPS tracking device as we initially planned.

Problem of the week

Our problem we are currently facing is finding a school to work with us on our application. I am reaching out to many schools in the local community to discuss about our project.

* Maize
* Wichita Public Schools
* Andover
* Haysville
* Derby
* Renwick
* Goddard

So far, Maize and Andover have replied and have told me that they already have a similar system from their contractor and won’t be working with us while Haysville has told me that they will discuss with their interim directory of transportation and get back with me. The rest have not responded. We are hoping to be able to get to work with a school but worse case scenario, we can still continue to develop our GPS device and application and be able to provide it to public transportation in general such as city-wide transit.

**Week Four: Oct. 25th- Oct. 31st**

­General Notes

Working on the frontend of the application to create the user interface using SwiftUI. The main screen should look like this. Account button will have user information. Settings will have accessibility options.

Text, letter

Description automatically generated with medium confidence

Block Diagram:

Diagram

Description automatically generated

Met with Haysville school district superintendent Jeff Hersh and they have shown willingness to work with us on our project.

Problem of the Week

Before Haysville can come on board and collaborate with us on this project, we now face legal work. We must have a contract created for Haysville legal counsel to look over and the school district to sign mainly regarding the data collection we will encounter with our application. We must draft up a legal document stating that we will not misuse user data and keep it secured which means no selling or sharing any of it.

**Week Five: Nov. 7th- Nov. 14th**

­General Notes

The general user interface has been created with buttons at the bottom. It features a main screen upon opening, a navigation button that shows the bus tracker, an account button, and a settings button. These buttons aren’t fully fleshed out and completely operational yet. Currently we’re allocating our attention on displaying the bus location on our map.

A picture containing text, electronics, screenshot

Description automatically generated

The idea is to fetch the location that is transmitted from our Raspberry Pi to our cloud server and to show the visualization of the route the bus is taking and where the bus is at on the map.

Will need to research and familiarize myself with MapKit for this step. Found an article that I will be looking over.

https://www.raywenderlich.com/10028489-routing-with-mapkit-and-core-location

This part is our main priority and will want to finish this in time for the fall showcase.

Personal Reflection

We were able to get an interview with parents, a student, and an outside perspective. Based off these interviews, everyone seemed to quite excited about the prospect of such an app. Parents were excited and accepting of using a bus tracker app for their child, and students liked the idea to reduce the worries of their parents and not having parents always checking up on them and their whereabouts. The outside perspective, Mr. Stallard comes from a background of also having dealt with school bus transportation in the past and thought it was a great idea. He also gave suggestions about our project and how to make it more cost-effective with the idea of using cheap smartphones as the GPS tracker instead of a Raspberry Pi. These smartphones already have a GPS on these phones and won’t require the more complex and technical work that is needed for the Raspberry Pi. Although I see the benefits of this idea, I still lean towards having the Raspberry Pi in our product in the long run simply because I feel like it would be a more marketable product. Overall, it made me happy that there are potential users of our app and for it to be effective in improving school transportation that hasn’t changed for many decades.

Individual Work Package

For the Spring semester, a couple areas will need to be revisited to improve our product.

* On the hardware side of things, figure out which GPS tracking device is best for our project: a Raspberry Pi or a cheap, disposable smartphone. Will also need to have a housing in place for the device during transportation to prevent it shuffling around.
  + Raspberry Pi route: Will need to test the device while active and operational during a route to ensure it can send the location to a server consistently. This test will also ensure that power will keep it on and functional.
  + Smartphone route: Figure out how to have the phone transmit user location to the server as well as the test run that goes along with it.
* For the app itself, design operational features that we have put off during the first semester. These include:
  + Administrator View
  + Creating alerts as administrator
  + Push notifications to parents
  + Text messages to Android devices of parents
  + Accurate weather information display
  + Account and settings menu
* Another major undertaking for next semester would be account creation and handling. A major point during our first semester and our meeting with Haysville school district was safety and security. This also leaned into legal matters of who and how account information is stored and managed. For our product, if we were to be the ones managing user data, we will use Firebase as our database and login authentication system. This while require legal contracts to be created and signed by Haysville and multiple hours to implement as we cannot be negligent with handling data that the school district gives to us.

**Week Six: Jan. 30th - Feb. 13th**

General Notes

Did research for authentication for Firebase for iOS over winter break. These sources were used for research and development for authentication:

* [1](https://firebase.google.com/docs/auth/ios/manage-users) was used for managing users in firebase database which featured email verification and password reset instructions
* [2](https://firebase.google.com/docs/auth/ios/email-link-auth) was used for learning how to use auth.auth() to help with firebase account creations
* [3](https://firebase.google.com/docs/auth/ios/passing-state-in-email-actions) was used for handling buttons related to email verification
* [4](https://www.letsbuildthatapp.com/course_video?id=7135) was used for learning how to setup xcode for firebase authentication
* [5](https://stackoverflow.com/questions/49134297/send-an-email-verfication-email-to-a-new-firebase-user-in-swift) was used for coding email verification

Coded the following this week:

* UI change to reduce view changes specifically for functionality testing but can be re-added once functionality is complete
* Added sign up method that creates an account in Firebase users when given an unregistered email and password.
* Added a verify account functionality that emails a verification email to a newly created account.
* Added a sign in method that allows a user to sign in when given a registered email and its respective password.
* Added a forget password functionality that emails a user a reset password link.
* Included status message display for testing purposes to help with authentication debugging.

To-Do:

* Combine login screen with the rest of the app
* Fetch user data to display the correct views when user signs in
* Clean up login screen views to make it more user friendly
* Enforce email verification rules before user can sign into app

Problem of the week

Have not heard back from Haysville since late December and have sent some emails since in hopes for a meeting to discuss about solutions for the app in their environment. Another problem popped up about how to handle user accounts once created to ensure which account is for which parent. Will need to research further on viable methods of either email+phone verification or 2FA.

**Week Seven: Feb. 14th - Feb. 27th**

General Notes

Added comments for readability into loginView.swift. Combining login view with the rest of the views are next on the to-do list. loginView is currently using Swift Package Manager (SPM) instead of cocoapods like the rest of the app. Will need to carefully set up library folder structure. This link [here](https://quickbirdstudios.com/blog/swift-package-manager-cocoapods-support/#:~:text=It's%20crucial%20that%20we%20use,Manager%2C%20or%20vice%2Dversa.) may provide useful information during that process.

Team has also discussed on potentially allowing alternative sign-in methods such as Google/Facebook/Phone using Firebase UI. Further research must be done to determine viability.

<https://firebase.google.com/docs/auth/ios/firebaseui>

A screenshot of a phone

Description automatically generated with medium confidenceDefault Login View: Sign In View & Creation of Account:

Graphical user interface, text, application, chat or text message

Description automatically generated

Verification email: Graphical user interface, text, application, email

Description automatically generated