

Bayn and Thayn Development

# Tier 6

A Pokémon Go Companion App

Tyler Bayn and Alex Thayn  
4-17-2018

## Table of Contents

<b>Section 1: Contextual Inquiry and Analysis.....</b>	<b>3</b>
System Concept Statement.....	4
How we Tailored the Scope .....	4
The Question .....	4
The Description .....	4
The Issues .....	5
Preparation for Interviews .....	5
People to Interview and Observe .....	5
Interview Questions .....	6
Interview Reflection .....	6
How we collected raw data .....	6
Photos and scans of work artifacts and notes .....	7
Building the WAAD .....	10
Pictures of us working .....	12
Workflow Diagram .....	13
Major Work Roles, Sub-Roles, and Machine Rolls .....	14
How we plan to “hook” our users .....	14
Monetization Strategy .....	14
<b>Section 2: Requirements and Design .....</b>	<b>15</b>
System Concept Statement.....	16
How we Tailored the Scope .....	16
Interaction Design Requirements.....	16
Requirements Extraction Process.....	17
Models Used and Why .....	18
The Models .....	19
Work Roles and Sub Roles .....	19
User Classes .....	19
Flow Model .....	20
Hierarchical .....	21

<b>Task Interaction Models .....</b>	<b>22</b>
<b>Barrier Summary .....</b>	<b>28</b>
<b>Section 3: Design .....</b>	<b>29</b>
<b>System Concept Statement.....</b>	<b>30</b>
<b>How we Tailored the Scope .....</b>	<b>30</b>
<b>Personas .....</b>	<b>31</b>
<b>Ideation and Sketching.....</b>	<b>35</b>
<b>Workspace and Materials .....</b>	<b>37</b>
<b>Sketches.....</b>	<b>37</b>
<b>Physical Mockups .....</b>	<b>41</b>
<b>Designers Mental Model, conceptual design, and how it links the User and Designer .....</b>	<b>46</b>
<b>Storyboard .....</b>	<b>49</b>
<b>Wireframes.....</b>	<b>51</b>
<b>Section 4: Development .....</b>	<b>52</b>
<b>Development .....</b>	<b>53</b>
<b>Screenshots .....</b>	<b>53</b>
<b>Reflection .....</b>	<b>56</b>

# Section 1

Contextual Inquiry and Analysis

## **1. System Concept Statement:**

The game of Pokémon Go is way more complex than initially meets the eye. With every creature in the game having different battle types, strengths, and weaknesses, there is always a need to keep data straight. The Tier 6 app will recommend what types to use in in-game battles known as raids and recommend effective teams for you to use. The Tier 6 app watches the screen live from the background and when it recognizes in-game Pokémon Battles it will show an on screen notification of what Pokémon and Pokémon types you should use. When inside the app it will allow you to view all the current Pokémon found in raids and recommend battle parties based on type or Pokémon.

## **2. How we Tailored the Scope:**

The most noticeable thing we have done to help with the scope of this project is to start playing the game. Ty has been playing the game since Summer 2016, and Alex has been playing for a few weeks. By doing so we will be able to see the issues from the perspective of the users.

For our field observation and interviews we decided to do those while we were out playing the game with a group. This allowed them to tell us about the issues they had as they happened live in-game.

We also have constant contact with our user base. Ty owns a Discord Server (essentially Skype for gamers) for just Pokémon Go in Ephraim, Utah. Using that we always have a way to ask questions and get ideas and concerns from real users.

### **The Question:**

How can we give Pokémon Go a streamless and intuitive way to know what Pokémon or types are effective during raid battles?

### **The Description:**

In the game Pokémon Go, there are in-game battles known as raids. When a raid joins you are presented with a 120 seconds on a waiting lobby where you can choose which Pokémon you want to use to battle. The current system will try to analyze which Pokémon you have that the system thinks would be effective for you to use. It makes this decision based off of moveset types, damage per second (DPS), and Pokémon type.

### **The Issues:**

There are currently two issues that need to be addressed with Pokémon selection in raid battles. The first is that there is no way to see a Pokémon's type when starting a raid. The most common question asked at a raid is "What type of Pokémon is effective against this boss?" With no way of seeing the types of Pokémon, creating a team on the fly can be difficult and can lead to an ineffective battle experience. The second is the auto selection of teams. The game will try to give a team it thinks is best based off of moveset types, damage per second (DPS), and Pokémon type. Although this system is decent, it still has many problems. The most noticeable is that it will recommend teams with a higher defense rather than a strong attack. This will keep Pokémon alive for longer, but ultimately less damage is done.

### **3. Preparation for Interviews:**

In order to most effectively get true, emotional responses, we decided that the best way to interview people was to interview people who play the game while playing the game. We also figured that we should get familiar with the game so that we could share those emotional responses with the game as well. So in preparation for interviewing we did the following:

- **Start playing the game:** Ty has played the game since its release in the summer of 2016, Alex just recently started but is already starting to understand some of the issues.
- **Get to know the users:** Ty Owns a Discord server that Pokémon Go players use to communicate and plan events. This was used to make connections and getting to know the users.
- **Prepare questions to ask:** We prepared questions that would allow us to get as much information as possible about what the users wanted in an app.
- **Scheduled a night:** We scheduled a night where both of us could join a group that was going out to play Pokémon Go. This was done so that we could get responses from the users that were in real time as they were playing.

### **4. People to Interview and Observe:**

Kara Bayn – Dedicated Player, Typically Raids 7x/week

Britney Mork – Dedicated Player, Typically Raids 5x/week

Kaitlyn Cushing – Casual Player, Typically Raids 7x/week

Nate Bridges – Dedicated Player, Typically Raids 5x/week

(Because both of us play, we counted ourselves)

Alex Thayn – Casual Player, Typically Raids 1x/week

Ty Bayn – Hardcore Player, Typically Raids 14x/week

## **5. Interview Questions:**

- What types are good against a raid boss?
- How many raids do you do each week?
- Do you know the type of Pokémon you are fighting?
- Do you know the type of Pokémon you are using to fight with?
- Do you know which types are effective against each other?
- Do you know all the currently available types of Pokémon?
- During or before a raid, have you ever asked the question: “What is the best type to use?”
- Would you like to see in-app suggestions for Pokémon, or on-screen notification while you are playing?
- Do you know what a “Glass Cannon” is?
- Would you rather have a list of recommended types or a list of recommended Pokémon?
- How much space do you currently have on your phone for a new app?

## **6. Interview Reflection:**

Our initial meeting and interview went very well. There were a lot of things that we were able to learn from the responses that ultimately caused us to change the purpose and design of our app. We created a good connection with them and they are excited to help us test different versions of the app.

## **7. How we collected raw data:**

We mainly collected our data from the questions that we asked. But we also collected data by watching the users play the game and listening to what they said while playing. During certain points they would say “the app should do something here.” At that point we would take a screenshot of the screen and write down a few notes regarding that screen.

8/9. Photos and scans of work artifacts and notes:

- Interviews / Field Visit  
Nate (alchemist)  
Kaitlyn (teverbloom)
- What is good against a raid?
- How many raids per week 6-12 (7)
- Do you know the types? of the pokemn you are fighting or using  
not always  
after I know because they are iconic
- Do you know which types are effective against which  
no usually
- Do you know all the types that are available  
yes most types
- Have you asked, what's the best type to use  
yes
- In game list or notification on-screen  
no notification
- Do you know what a glass cannon is?  
no don't use in app
- Recommend a list of pokeman or a type  
List! because I don't always know what types pokeman are
- Option to turn off pokeman recommendations for don't have from the list Yes! That would be nice

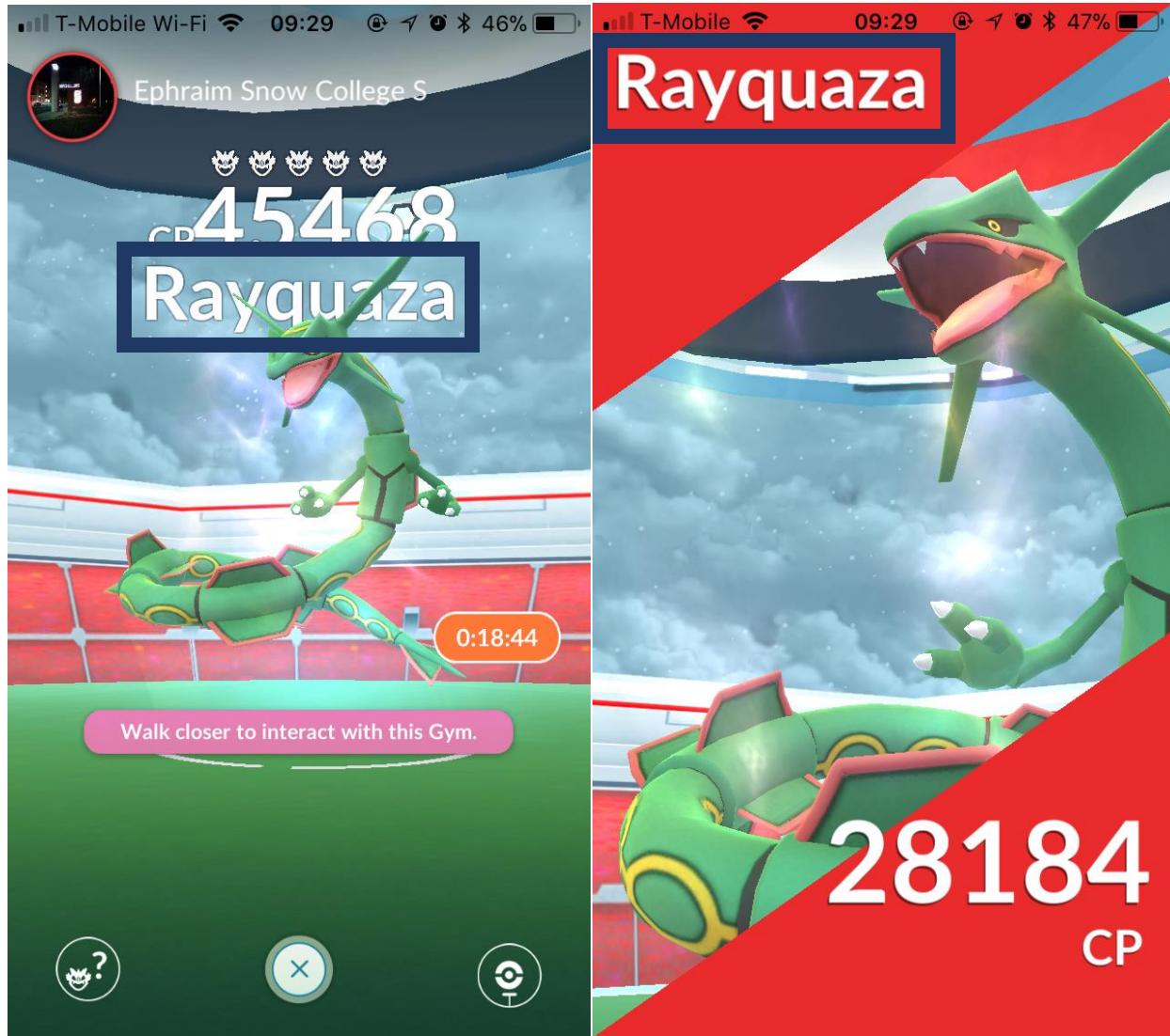
how much space on your phone  
is a whole lot

What is too big  
< 1gb I try not to download a lot  
limited space at this point

Focus on  
Just the raid assistant

- Not track shinies for now

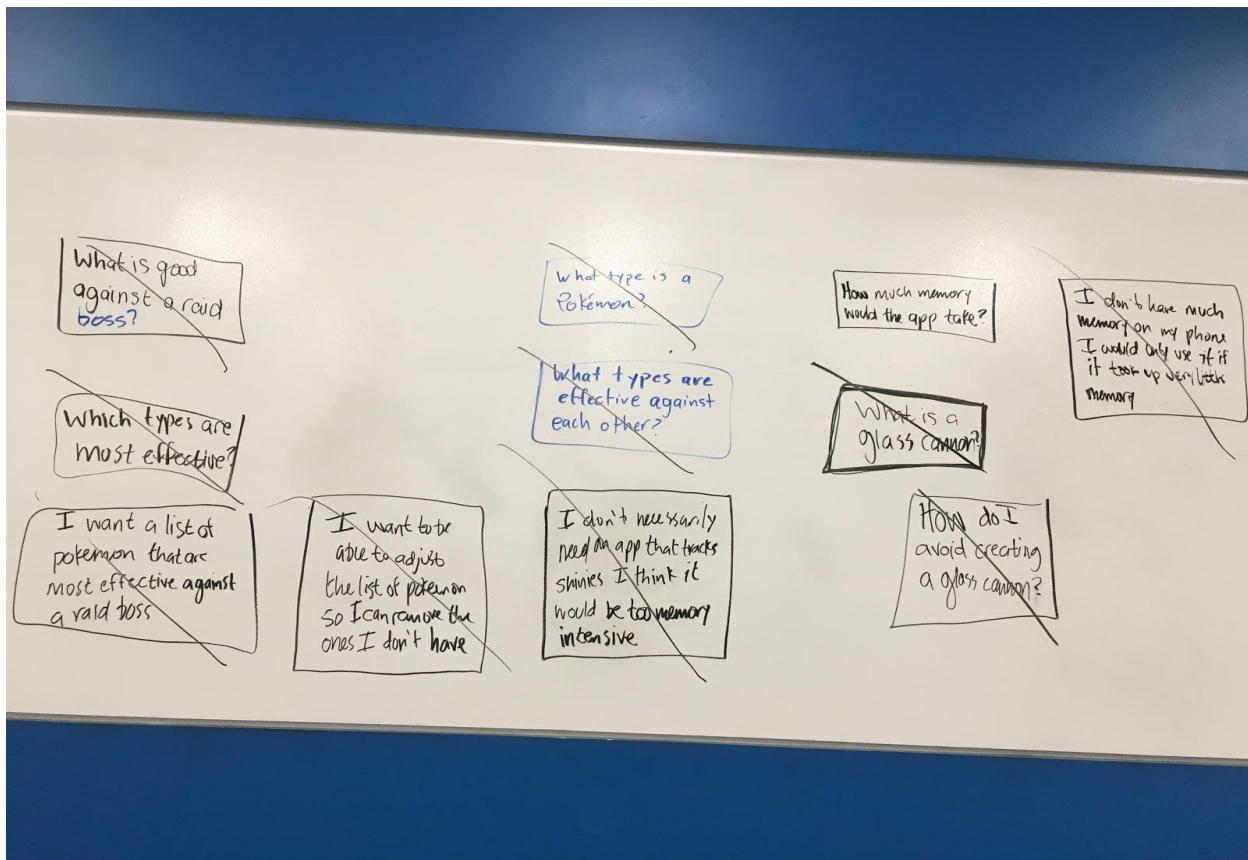
The following screen shots show where data would need to be recognized and a drop down notification shown (text to be recognized surrounds by blue).



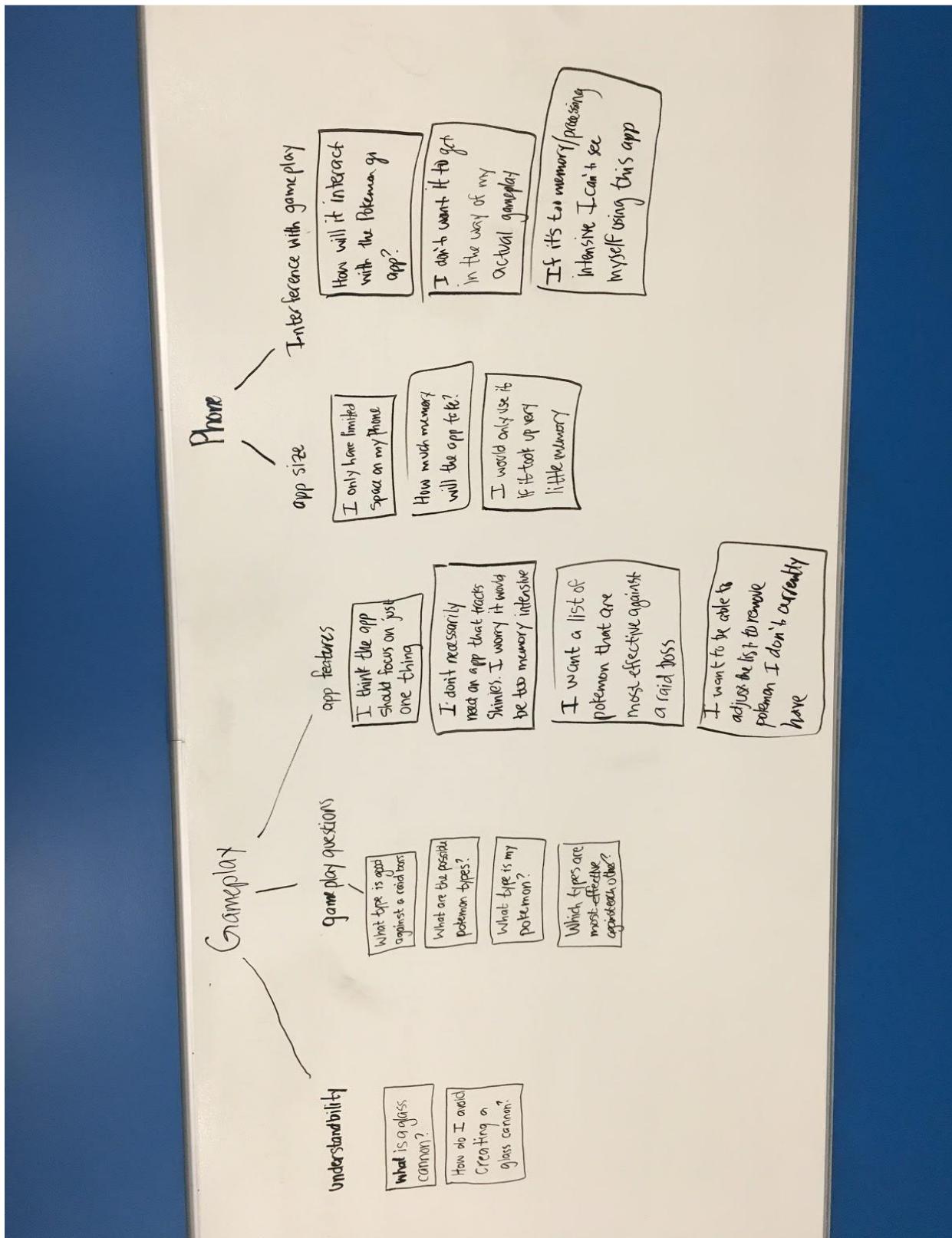
### 11/12/13/15. Building the WAAD:

Once we finished up with contextual inquiry we began extracting work activity notes for the WAAD. To start we found as many user stories as we could pull from our notes and wrote them up on the whiteboard. After putting up all the work activity notes we began to notice patterns and started to sort the notes into different categories in our WAAD. We identified two main categories: Gameplay and Phone issues. Inside each each of these main categories we sorted the notes into subcategories.

This is the Work Activity Notes that we pulled from our raw data and notes.



This is the Work Activity Affinity Diagram that we created based on our work activity notes.

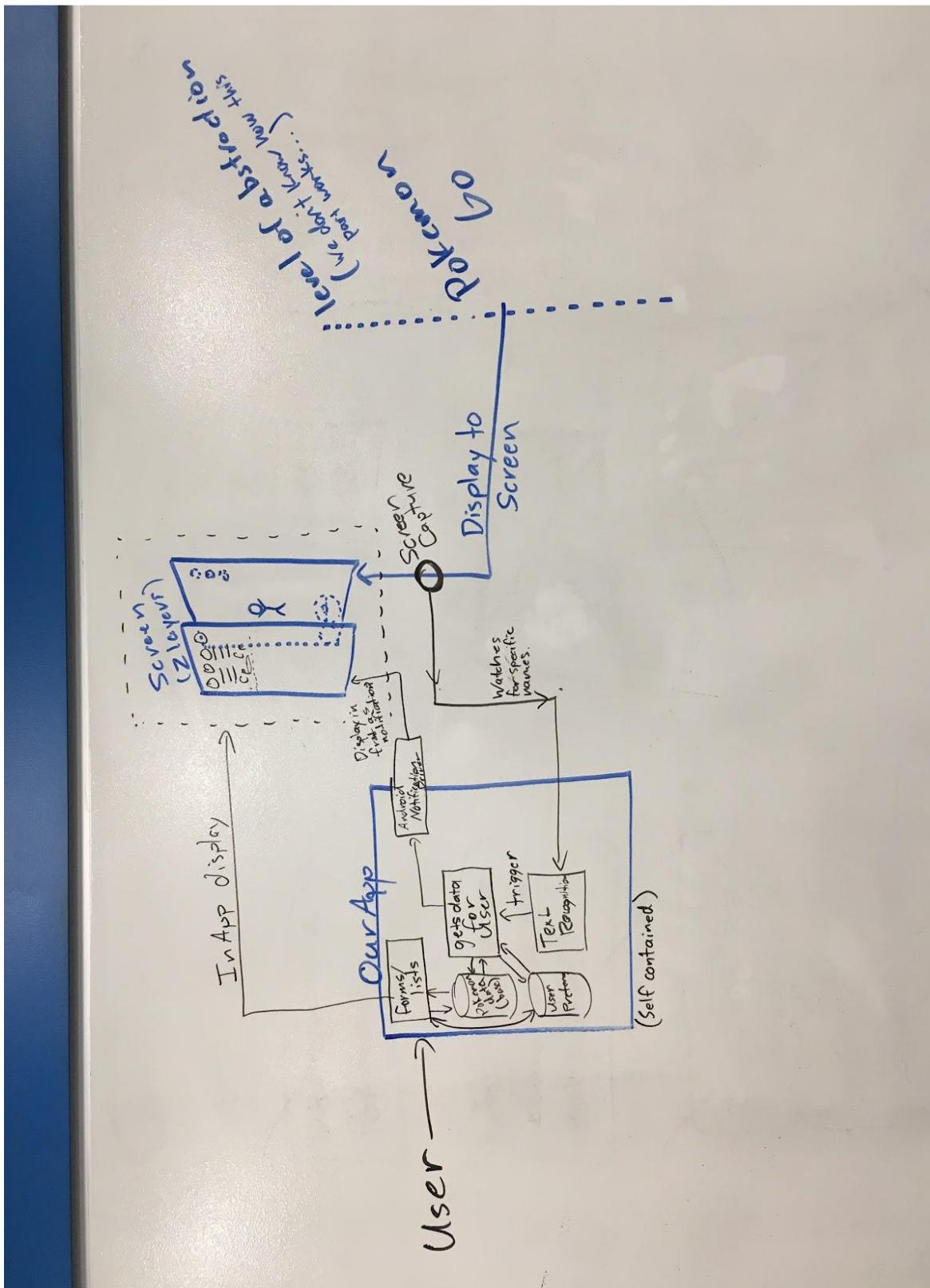


**14. Pictures of us working:**



### 17/18/19/20/21. Workflow Diagram:

We created a work flow diagram to show how the different parts of the system would interact.



## **16. Major Work Roles, Sub-Roles, and Machine Roles:**

Major Work Roles:

- Pokémon Go players: the player of this game are going to be our main users.
- Developers: Creates and maintains the app.

Sub-Roles:

- Google and Apple: Allow for the app to be placed in their respective Application Libraries and Stores.

Machine roles:

- Database: the machine roles we identified were the databases that we need to store within the app. These databases will contain all the information needed to run our app and store user preferences.
- Phone Notification System: The app will send requests to this to display messages on the screen.
- Screen Capture: Allows the Pokémon Go game to be actively watched from the screen.
- Text Recognition: Reads the screen capture, looking for specific Pokémon names.

## **22. How we plan to “hook” our users:**

To hook our users, we simply plan to provide an app that will solve the issues of knowing what types of Pokémon to use during raid battles. We figure that if a user has a better experience in Pokémon Go with the aid of our app, they will continue to use it and other players will notice and download it as well.

## **23. Monetization Strategy:**

Initially we do not plan to charge any money for the app. After there is some level of success we might decide to charge \$0.99 for the app or place very small advertisements within the app to generate revenue.

# Section 2

Requirements and Design Analysis

## **1. System Concept Statement:**

The game of Pokémon Go is way more complex than initially meets the eye. With every creature in the game having different battle types, strengths, and weaknesses, there is always a need to keep data straight. The Tier 6 app will recommend what types to use in in-game battles known as raids and recommend effective teams for you to use. The Tier 6 app watches the screen live from the background and when it recognizes in-game Pokémon Battles it will show an on screen notification of what Pokémon and Pokémon types you should use. When inside the app it will allow you to view all the current Pokémon found in raids and recommend battle parties based on type or Pokémon.

## **2. How we Tailored the Scope:**

For this assignment, we really had to focus on what models would be best for our app. As most of our is only used by one person with hardly any support from outside sources, we had no need for any social interaction models and very few User models as well.

The only change we made was to the Hierarchical Task model. Since our app has two modes (inside the app, and the app running in the background), we decided to separate those into two separate trees.

## **3. Interaction design requirements:**

We extracted the following requirements from the Work Activity Affinity Diagram:

<b>Explain Glass Cannons</b>  While using our app, the user should be able to learn what a Glass Cannon is.	<b>Find Good Type Advantages</b>  While in the app, the user should be able to find recommended Pokémon types against certain raid bosses.	<b>Recommend Good Type Advantages</b>  While running in the background, the app should be able tell the user recommended Pokémon types against certain raid bosses.
<b>Find Good Teams</b>  While in the app, the user should be able to find recommended Pokémon teams against certain raid bosses.	<b>Recommend Good Teams</b>  While running in the background, the app should be able tell the user recommended Pokémon teams against certain raid bosses.	<b>Maintain a PokéDex</b>  The user should be able to keep a current record of what Pokémon he currently has.

<b>Use Limited Space</b>  The app should only use a small amount of space.	<b>Single Subject</b>  The app should only provide service for one topic, don't include shinies or maps.	<b>Reminders</b>  When displaying recommendations, the app should also remind the user to use certain in-game items.
<b>No Interaction with Pokémon Go</b>  In order to protect the user, the app WILL NOT access Pokémon Go in anyway other than on-screen recognition.	<b>Player Profile</b>  The user should be able to keep track of XP, user name, and other settings for a personalized experience.	<b>Maintain Prebuilt Teams</b>  The user should be able to assemble teams in the app that the app would then recommend while in-game.

#### 4. Requirements Extraction Process:

We extracted all of the requirement from our work activity affinity diagram. We organized the extracted requirements into two sections: blue for interactions with the user, green for requirements of the app itself. There are two requirements that we felt were more important, these requirements have been outlined in red as they can affect the user and the phone, not just gameplay.

## 5/6. Models Used and Why:

- **User Models**
  - **Work roles:** We needed to identify the users and how they would interact with the app.
  - **Sub-roles:** We needed to identify third party users that would influence the app as well.
  - **User Classes:** We need to differentiate between different levels of users.
- **Flow Model:** We wanted to show the interaction between the user, the app, the game, the data bases, and how all of them work together.
- **Hierarchical Task Inventory:** There are many different paths that the user could go, there is also the idea that the app does something different between being in the app and using the app in the background. We wanted to show all possible uses and paths for both.
- **Task Interaction Models:** Both of the following got mixed together to show a task, the end goal, and a step-by-step usage guide on how to achieve those goals. The paths of those goals were created using the Hierarchical Task Inventory.
  - **Usage Scenarios**
  - **Step-by-step task interaction model**
- **Barrier Summary:** There are a lot of odd things in our app including terminology and how to get the app running. We needed to address problems like this, so we created a barrier summary to do so.

## 7. The Models:

### Work Roles and Sub-roles

#### Major Work Roles:

- Pokémon Go players: the player of this game are going to be our main users.
- Developers: Creates and maintains the app.

#### Sub-Roles:

- Google and Apple: Allow for the app to be placed in their respective Application Libraries and Stores.

#### Machine roles:

- Database: the machine roles we identified were the databases that we need to store within the app. These databases will contain all the information needed to run our app and store user preferences.
- Phone Notification System: The app will send requests to this to display messages on the screen.
- Screen Capture: Allows the Pokémon Go game to be actively watched from the screen.

Text Recognition: Reads the screen capture, looking for specific Pokémon names.

### User Classes

#### • Hardcore Pokémon Trainers:

These players track and catch all Pokémon, looking for the best ones and focus on Individual Values (IV's). They raid often and spend lots of time hunting Shiny Pokémon and Rare Pokémon. Usually spend real money on the game. Varying levels of knowledge of phone use.

#### • Dedicated Pokémon Trainers:

These players track and catch most Pokémon, looking for the best ones and focus on Individual Values (IV's). They raid often and spend some time hunting Shiny Pokémon and Rare Pokémon. Varying levels of knowledge of phone use.

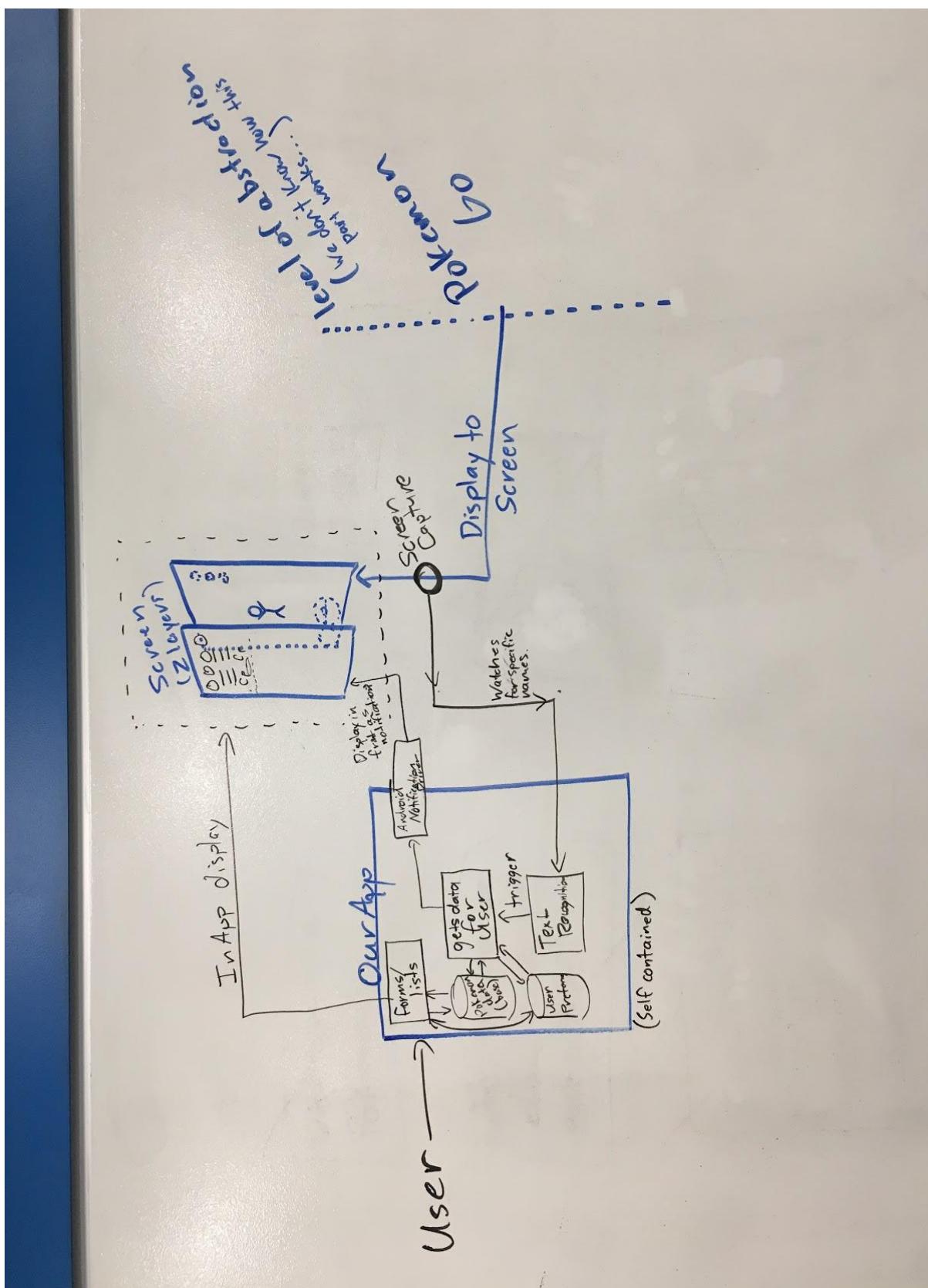
#### • Casual Pokémon Trainers:

These players Play for fun. Whenever there is a few spare minutes in a day they will catch whatever is around them. Varying levels of knowledge of phone use.

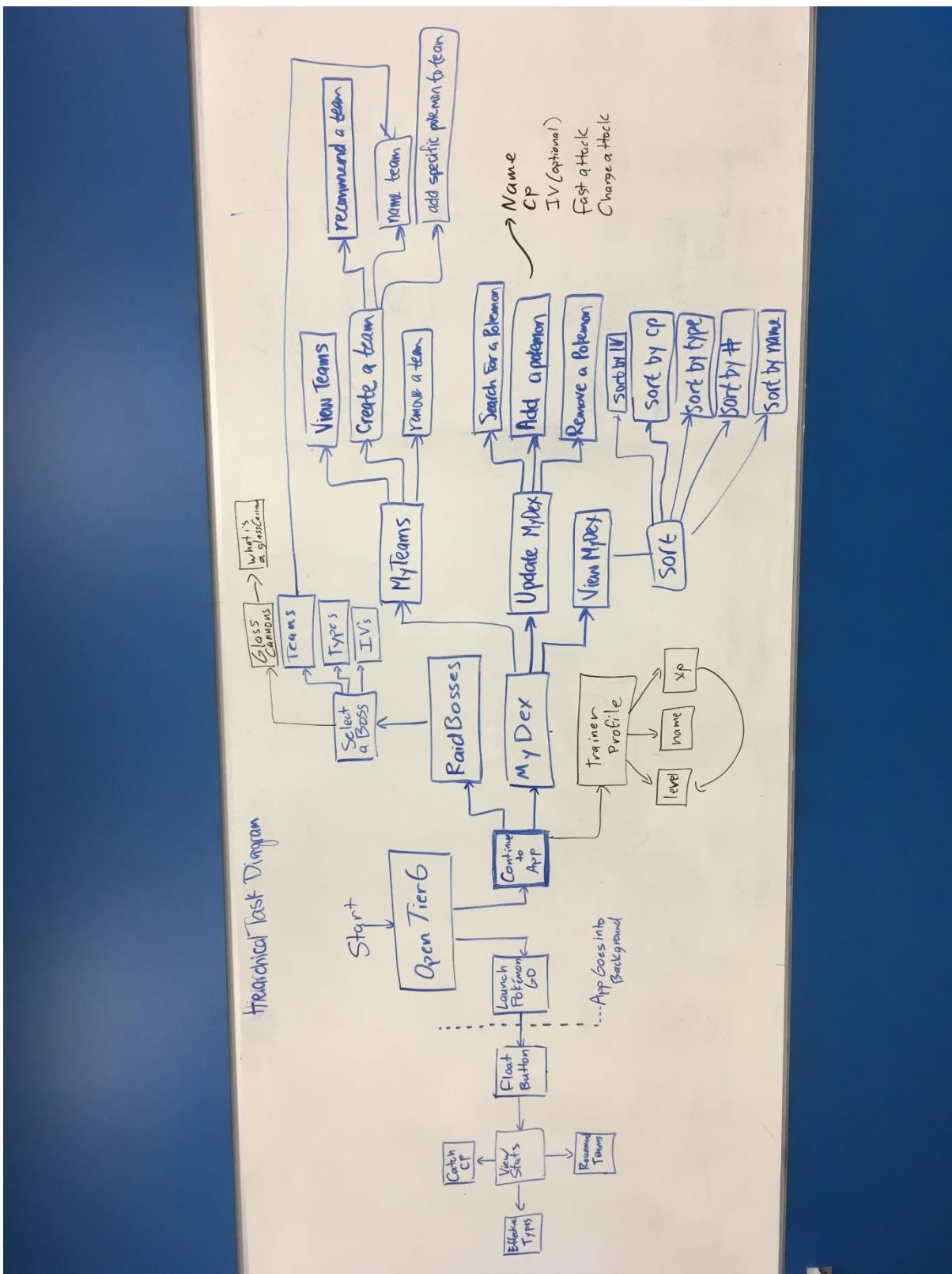
#### • Very Casual Pokémon Trainers:

These players play sometimes. No real dedication to any long term or difficult goals. Usually only play when bored or when others bring them along. Generally has lower knowledge of phone use.

### Flow Model



## Hierarchical Task Inventory



Task Name: I want to defeat Claydo)

Goal: I want to create the most effective team that will increase my chance of defeating the raid boss Claydo)

Trigger: open Tier 6 App

Continue to app

Select Raidbosses

Find Claydo!

Select team

Name the team that was recommended

Open Pogo app and create that team

Win the RAID!

Task: Raid Boss MaxCP's (Mewtwo)

Goal: See which catch CP's  
are 100% IV Raid Bosses

Trigger:

- Open App
- Continue into app
- Select Raid Boss
- Select Poké mon (Mewtwo)
- View Catch CP's

ded

7

Task: Ingame Recommendation

Goal: View a recommended team  
from within Pokemon Go

Trigger:

- Open App
- Start Pokemon Go
- Click on ingame raid.
- While viewing raid boss, click floating button
- View Team on dropdown

Task: update my Dex

Goal: I want to make sure that my Dex  
is current and only recommends  
Pokemon I currently has

Trigger:

- Open Tier 6 App
- Select my Dex
- Select update my Dex
- add all the pokemon I  
currently have
- deselect all the pokemon I  
do not have anymore

Task: Glass Cannon

Goal: Learn what a glass Cannon  
is

Trigger:

- Open App
- Continue into App
- Select Raid Boss
- Select Glass Cannons
- Select What is a  
glass Cannon
- Learn

Task: Update my XP

Goal: I want update my trainer profile  
to show my current in game XP

Trigger: Open App

Select continue to app

Select trainer profile

Select xp

Select update xp

Confirm

You are done!

## Barrier Summary

Barrier Summary		
Barrier	Model	Severity
1. Overlay will only work if Pokémon Go is started within the Tier 6 app.	Hierarchical Task Diagram	Extreme
2. Users may not know IV of their pokémon	Hierarchical Task Diagram	Low
3. Users won't know terminology (ie. GlassCannon)	Task Interaction models	Medium
4. Different Levels of Users (Both Pokémon & tech knowledge)	User Models	High
5. Incompatible phone... (iOS or outdated)	Knew this from beginning....	Medium

# Section 3

Design

## **1. System Concept Statement:**

The game of Pokémon Go is way more complex than initially meets the eye. With every creature in the game having different battle types, strengths, and weaknesses, there is always a need to keep data straight. The Tier 6 app will recommend what types to use in in-game battles known as raids and recommend effective teams for you to use. The Tier 6 app watches the screen live from the background and when it recognizes in-game Pokémon Battles it will show an on screen notification of what Pokémon and Pokémon types you should use. When inside the app it will allow you to view all the current Pokémon found in raids and recommend battle parties based on type or Pokémon.

## **2. How we Tailored the Scope:**

One of the hardest details that we had to work with was that we really don't have a "level of tech knowledge" that we can effectively use. So instead of including those in our personas and basing our sketches off of those, we decided that a better separator would be the level the user had with Pokémon Go. By doing this we could focus more on what the user knew about types and Pokémon rather than how to install the app on the phone.

### 3. Personas

Our personas are based on Pokémon Go Traveler Cards from TheSilphRoad.com. The cards used belong to real people, we have used their cards with their permission.



**Name:** Ty Bayn | **In-game Name:** ShinyHunterTy

**Bio:** 3<sup>rd</sup> year Computer Science student at Snow College (rural area). He lives a busy life with very few friends who play Pokémon Go. Although busy, he is a hardcore player logging over 20 hours a week. Has an extensive knowledge of the game and knows most Pokémon by type.

#### Frustrations:

- Hates the Random Number Generator, feels it's unfair
- Doesn't always know what Pokémon to use against raid bosses
- Not enough people around to take down larger raid bosses

This screenshot shows a player's profile from the game Pokémon Go. The profile is for a Trainer named "KEVERBLOOM" (Level 34) located in Ephraim, UT. KEVERBLOOM is a member of Team Mystic and has 4,885,754 XP, achieved 2 weeks and 2 days ago. The profile includes sections for "THE SILPH ROAD" (Traveler Card #B3FB8), "JOINED: 19 JAN, 2018", "MEETUP CHECK-INS: 2", "TRAINER LEVEL / 40", "POKEDEX / 368", and "PLAYSTYLE: CASUAL, WORKING ON IV HUNTING. TYPICALLY RAIDS 5X/WEEK." It also shows icons for Silph Co., Gym Badges, and a QR code for Discord.

**Name:** Kaitlyn Cushing | **In-game Name:** Keverbloom

**Bio:** 2<sup>nd</sup> year music student at Snow College (rural area). She lives a very busy life with very few friends who play Pokémon Go. Without the time to play often, she generally will only play between classes and when other friends are out.

#### Frustrations:

- Raids start and end too early in the day. By the time she is out of class, the raids are essentially done for the day.



**Name:** Kara Bayn | **In-game Name:** KaraBayn

**Bio:** 3<sup>rd</sup> year Computer Science student at Snow College (rural area). She lives a busy life with very few friends who play Pokémons Go. Although busy, she will play when she can. She generally only plays when her husband plays the game.

**Frustrations:**

- When given the Pokémons it recommends, it's gives you multiple Pokémons that are the same instead of a variety
- I wish the group size was based off of higher accounts than level 5 accounts.



**Name:** Alex Thayn | **In-game Name:** ImAGoofyChicken

**Bio:** Software Engineering student at Snow College (rural area). He lives a busy life with very few friends who play Pokémon Go. Very casual player, only plays occasional with friends. He is very new to the game and doesn't completely understand game and raid mechanics.

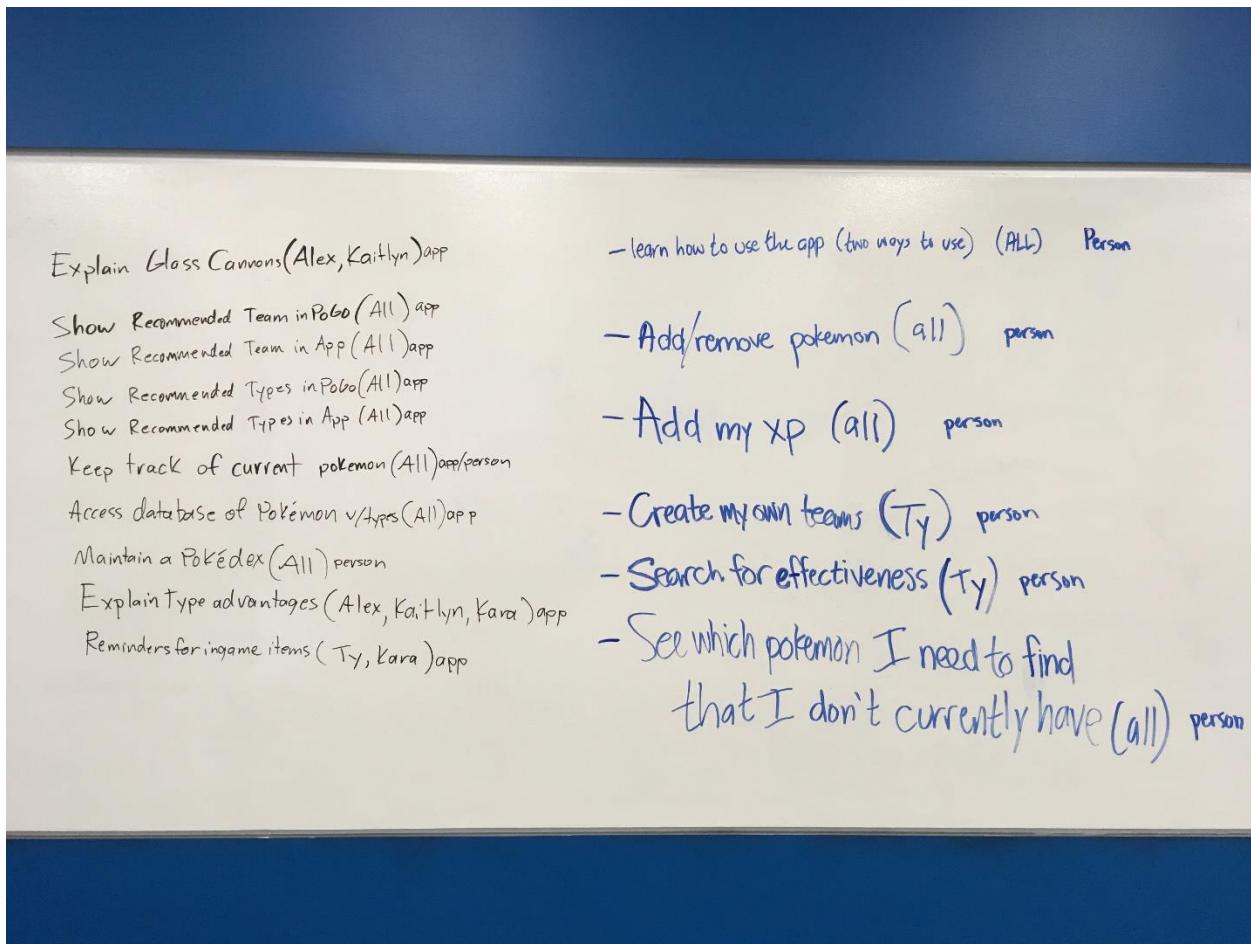
#### Frustrations:

- Dislikes that raids are sometimes too hard to do alone
- Has no idea who to use against a raid boss
- Doesn't like that he has very limited raid passes
- Doesn't know which raids are more important than the others

We decided that all of our personas should be based on real people that play the game since many of these frustrations can't be made up. The four personas chosen really provide a very wide variety of players and play types. They range from Hardcore players that know a lot about the game to very casual players who are just starting. The main points of the persona are already provided by TheSliphRoad's traveler cards made by these real people.

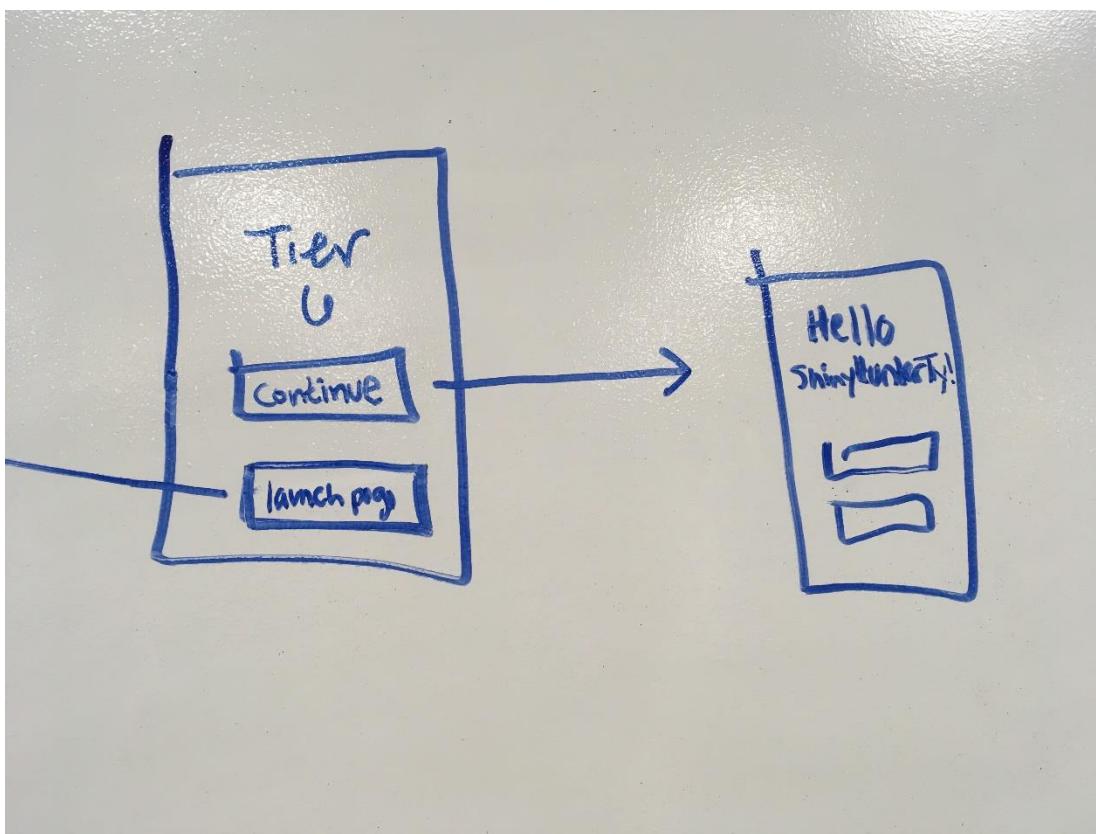
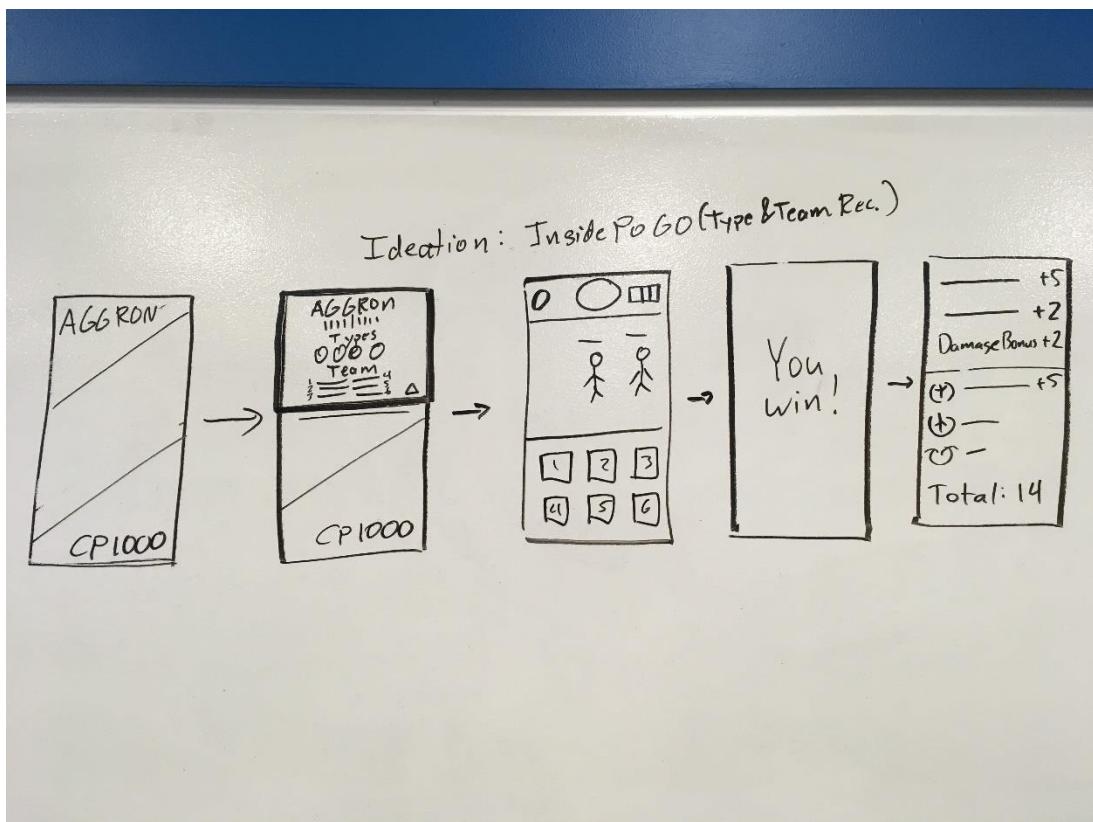
#### 4. Ideation and Sketching:

We started by having a brainstorming session where we simply listed all of our ideas. Since many parts of this app are going to be used by both hardcore and very casual players alike, we wanted to make sure that each idea would provide the information needed to new players without annoying experienced players. For every idea we labeled who would need the feature most based on our personas. Using the frustrations from the personas and the barriers we determined in our analysis, we also categorized the ideas into categories.



Once we had created ideas and requirements, we began sketching ideas on another white board, allowing the ideation to direct the sketches.

From the ideas and requirements, we knew that the app needed to be self-contained so that the data was local. We also knew that the app needed to be able to have half of it run in the background while the user is playing Pokémon Go, and half of it be usable while the user is actually using the app. Both parts would allow for the recommendation of teams based on Pokémons and types. The inside part of the app would also allow the player to keep track of personal data such as a Pokédex and experience in the game. Although there would need to be a lot of explanation for newer or inexperienced players, we figured that could be added to be easily accessible but not so apparent as to annoy experienced players.

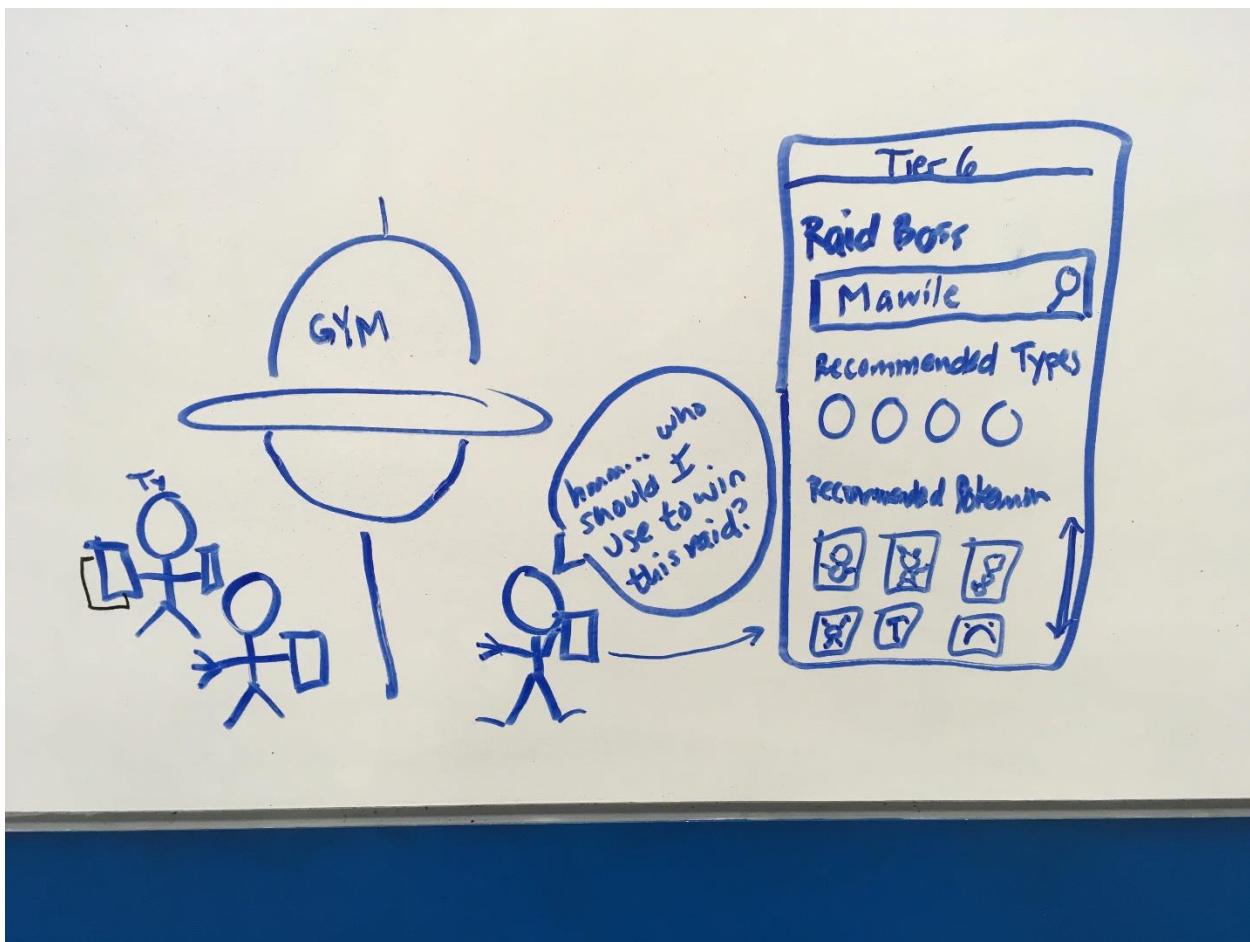


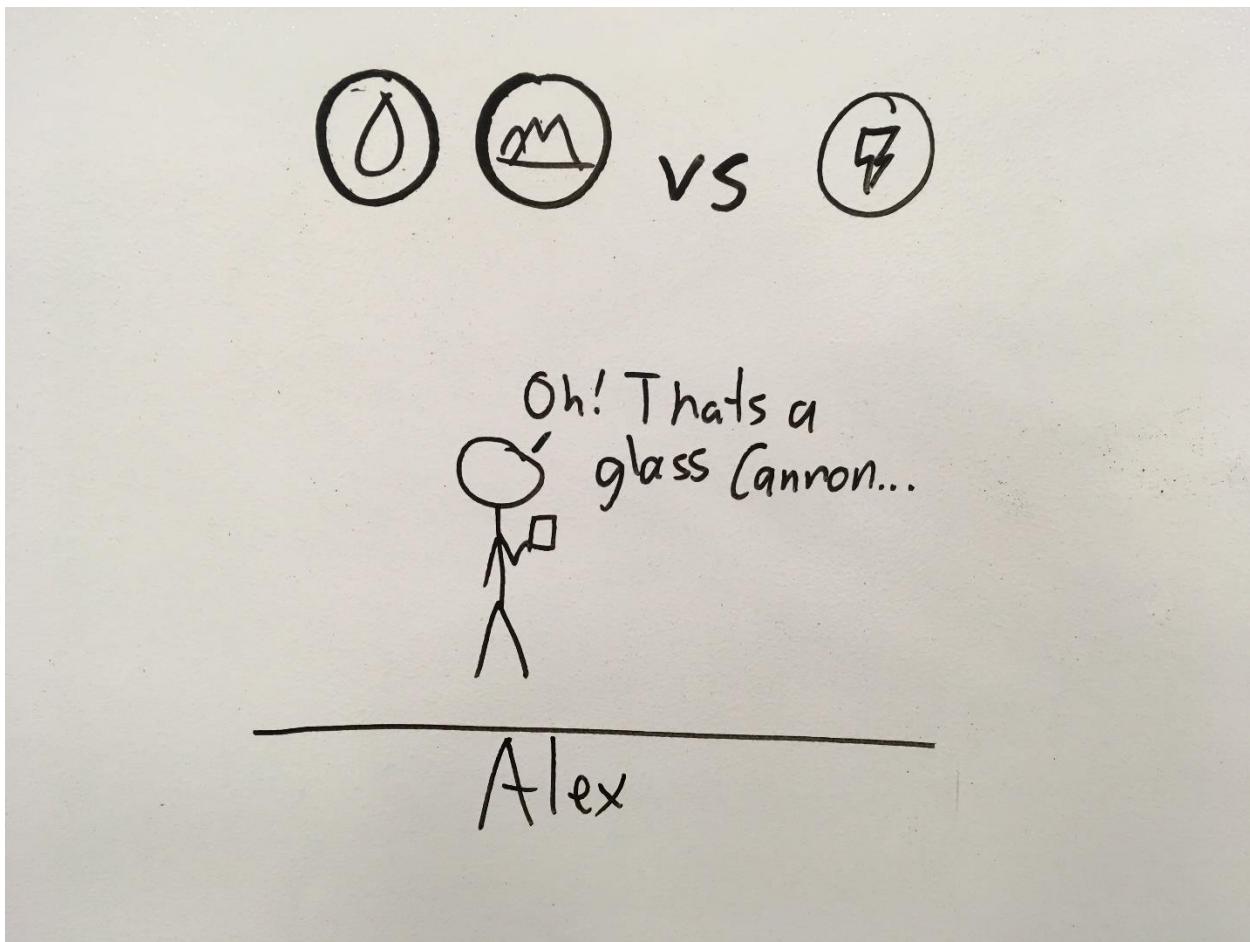
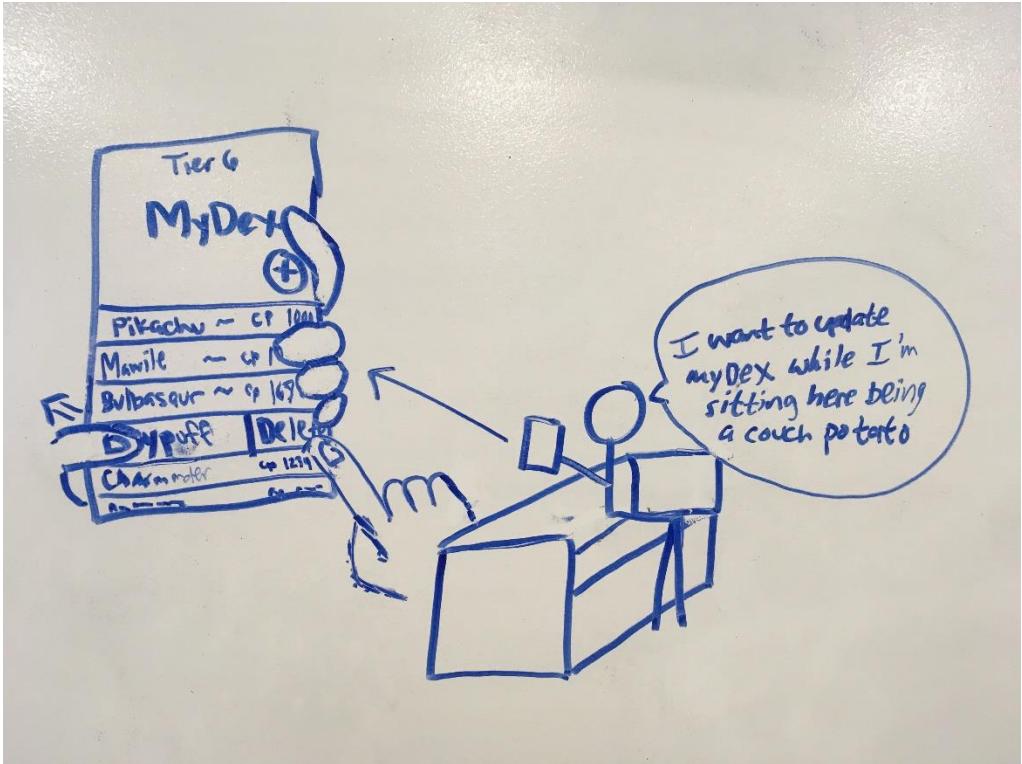
## 5. Workspace and Materials

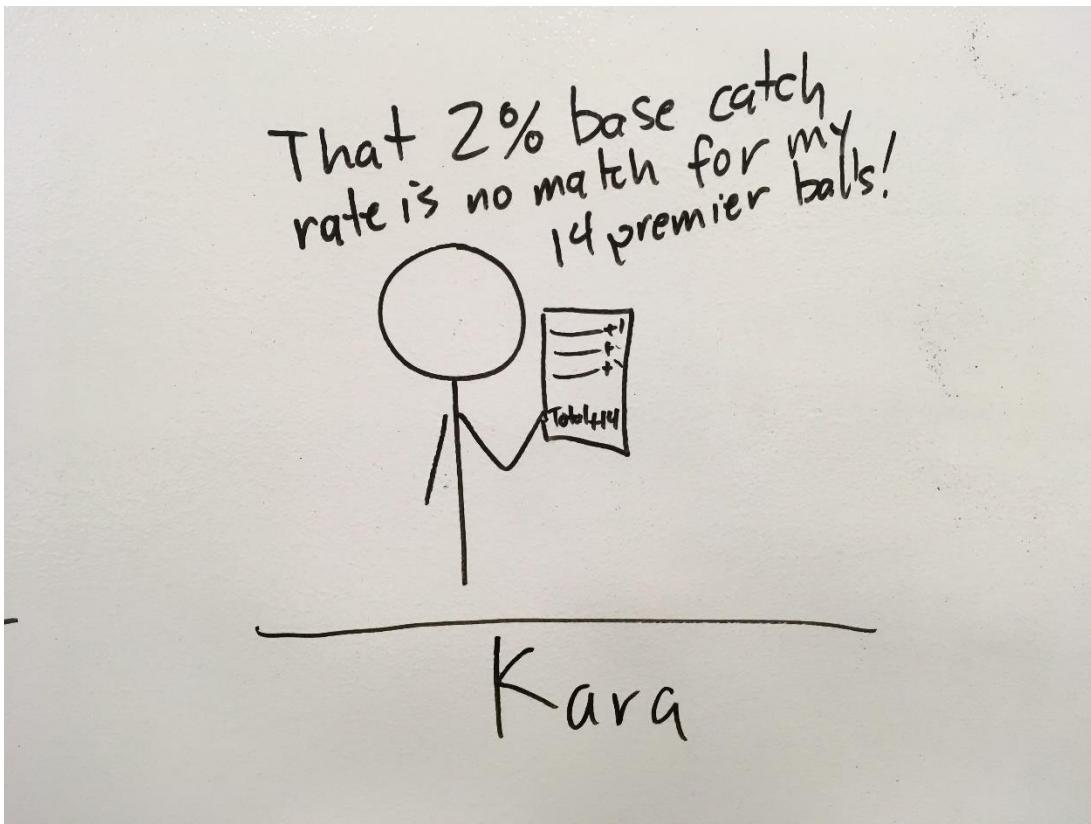
We prefer to collaborate with a white board. We generally each have a marker and we are up, discussing ideas, drawing sketches, commenting, and working together. This time through we did use a desktop to create mockup images of some of the screens that would be used.

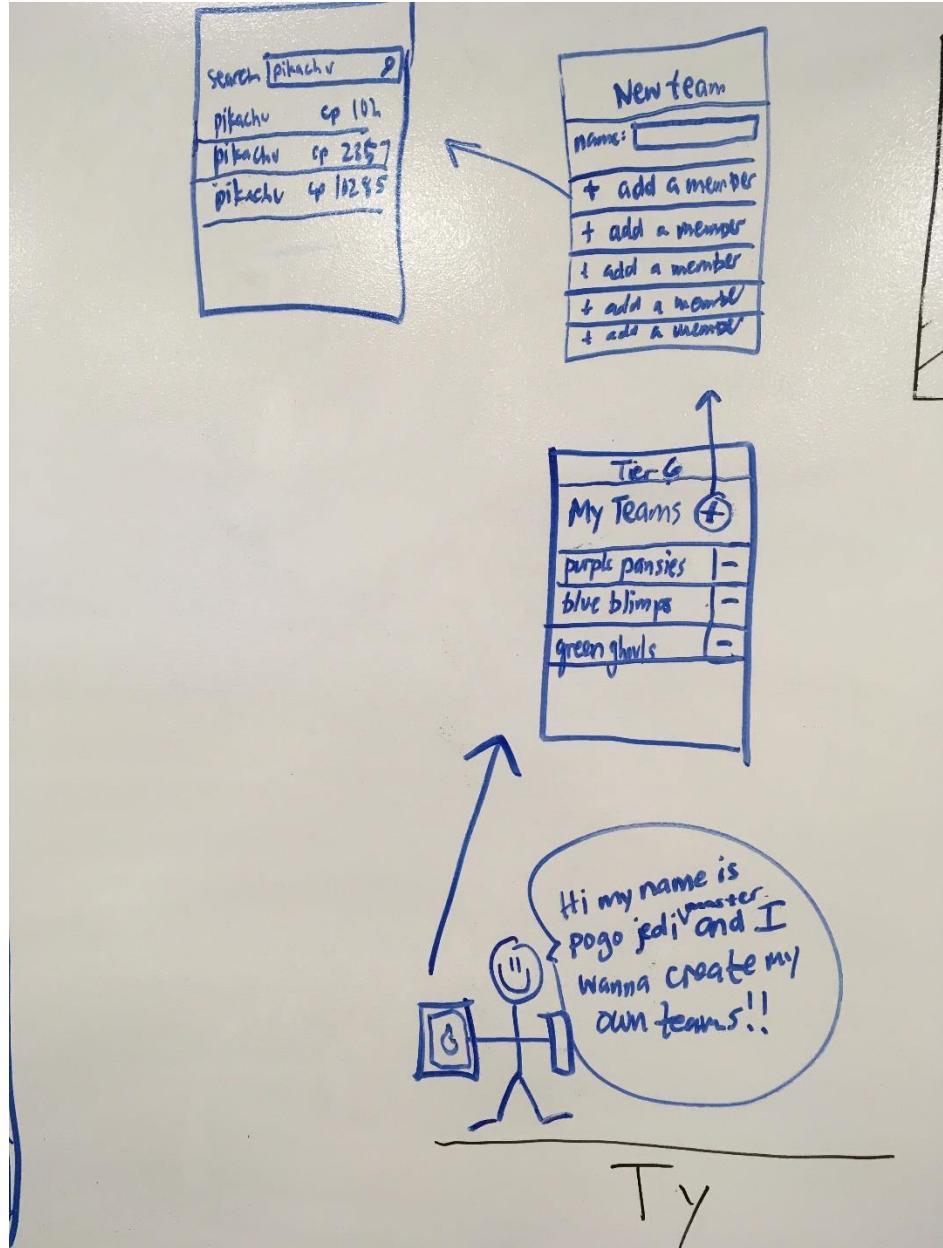
## 6. Sketches:

We made multiple sketches to show the use of the software and the emotional impact of the software on the chosen personas.







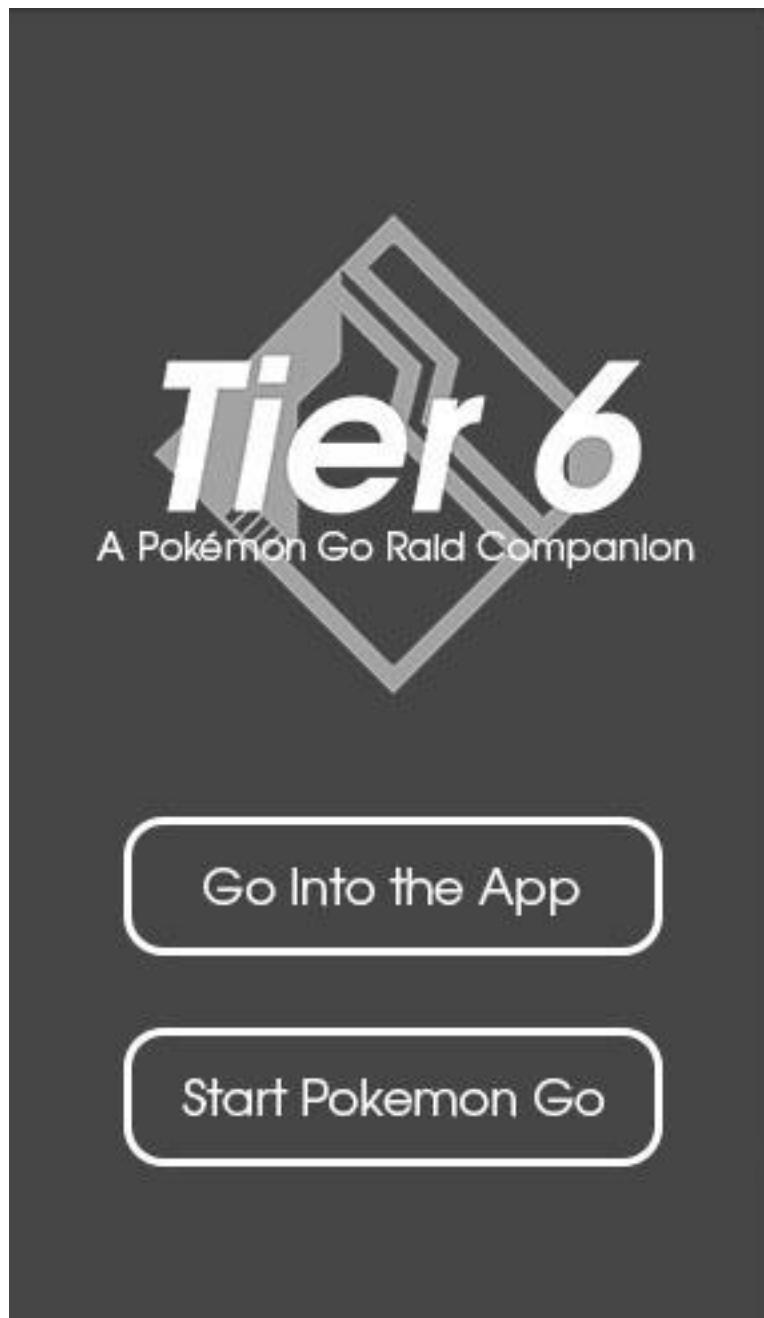


After the brainstorming and sketching session we looked through all of the ideas and sketches that had been made. We actually decided to keep every idea that was thought of except for the kiosk at every building. We decided to scrap this concept because if a student or teacher can access the system and scheduler from their phone, the kiosk would be redundant. But from all the other ideas, we were able to start sketching more and finalizing our designs.

## **7. Physical Mockups:**

We made a few physical mockups using Adobe Photoshop to show what the app would look like when being used while playing Pokémon Go, and also what the screen would look like for the main opening screen.

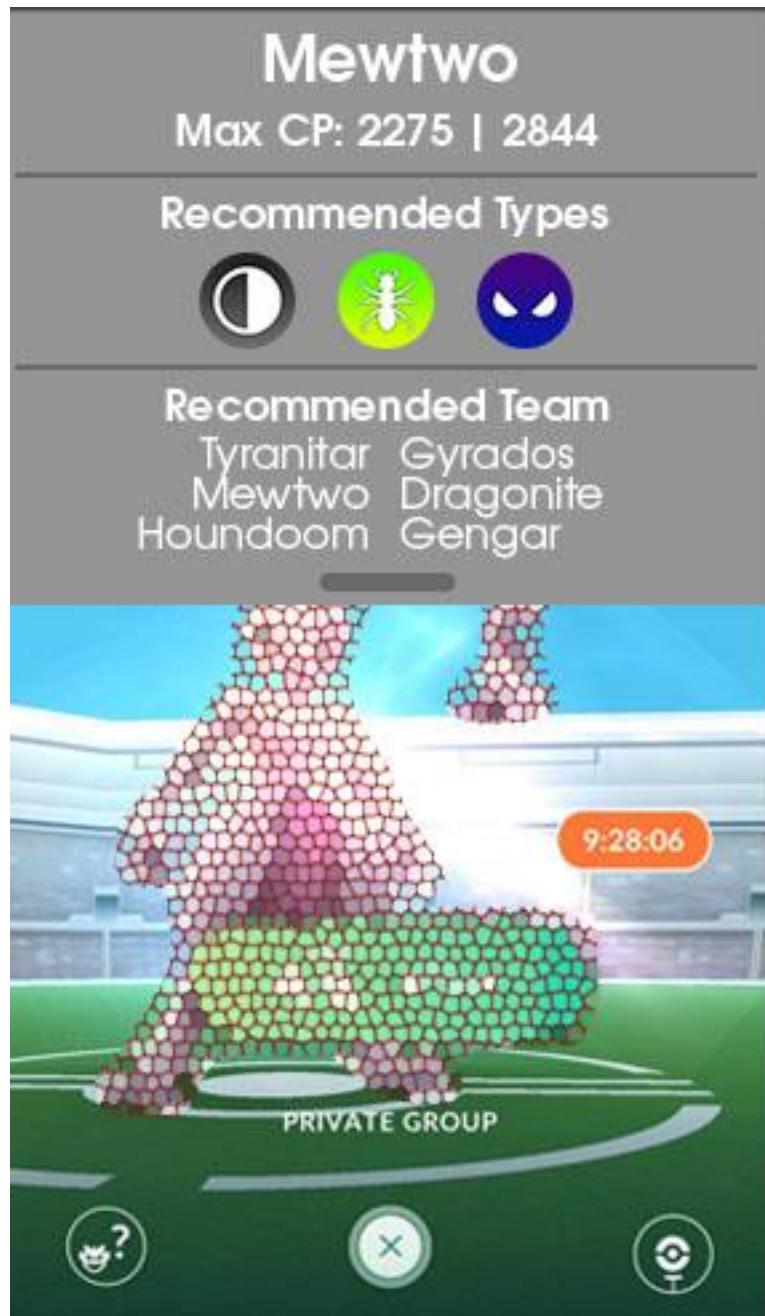
**Opening Screen:** This is the screen that the user will see when they first open the app. It will allow them to open Pokémon Go or to continue into the app itself.



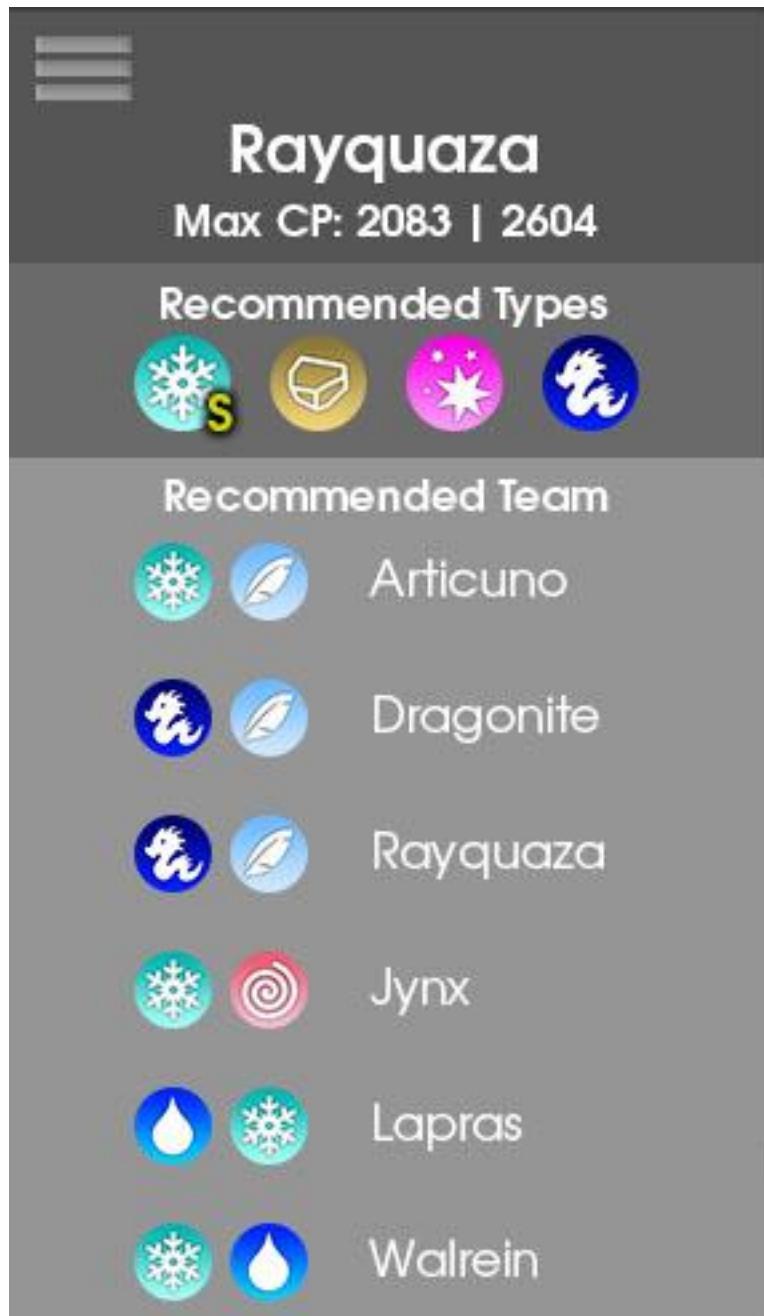
**Overlay of In-Game Raid (Screen 1):** This screen shows what the drop-down menu would look like in game when a raid boss is encountered.



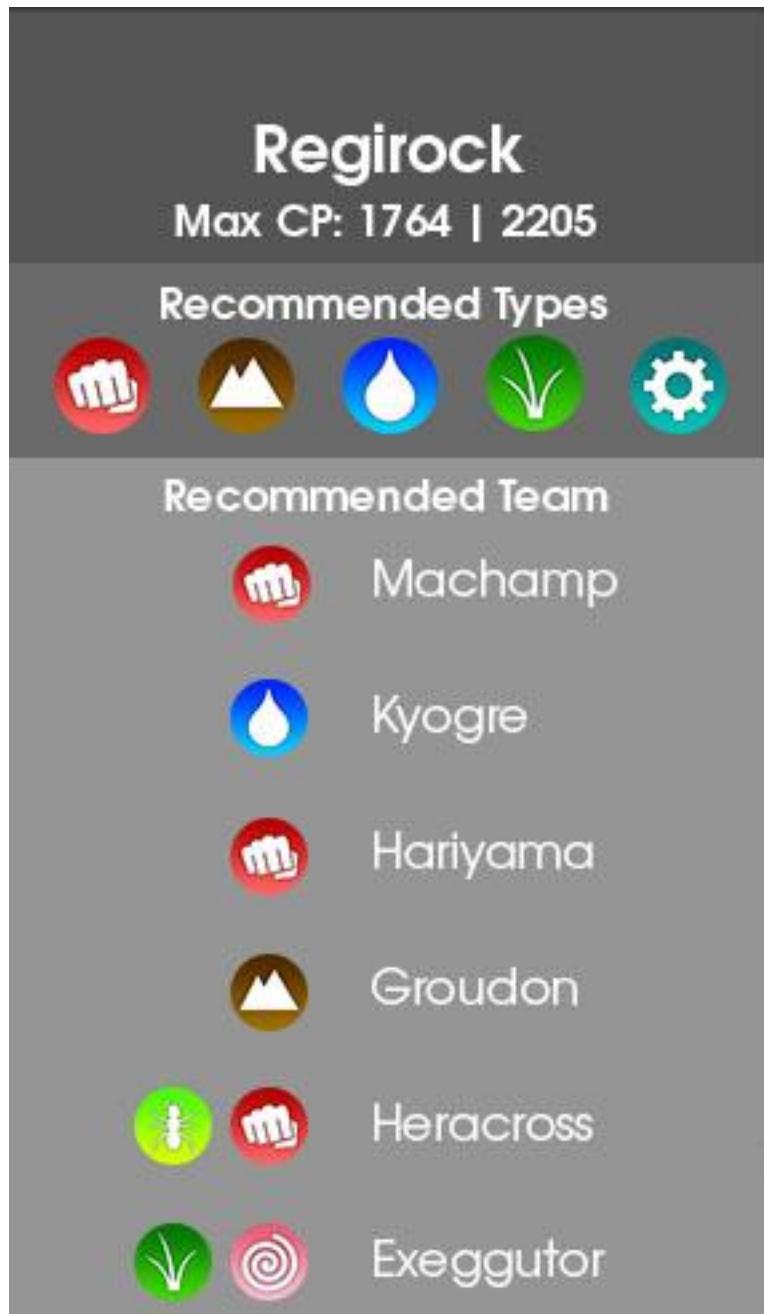
**Overlay of In-Game Raid (Screen 2):** This screen shows what the drop-down menu would look like in game when a raid boss is encountered.



**In-App screen of recommended types and teams (Screen 1):** This screen shows the recommended team and types the user should use during a raid.



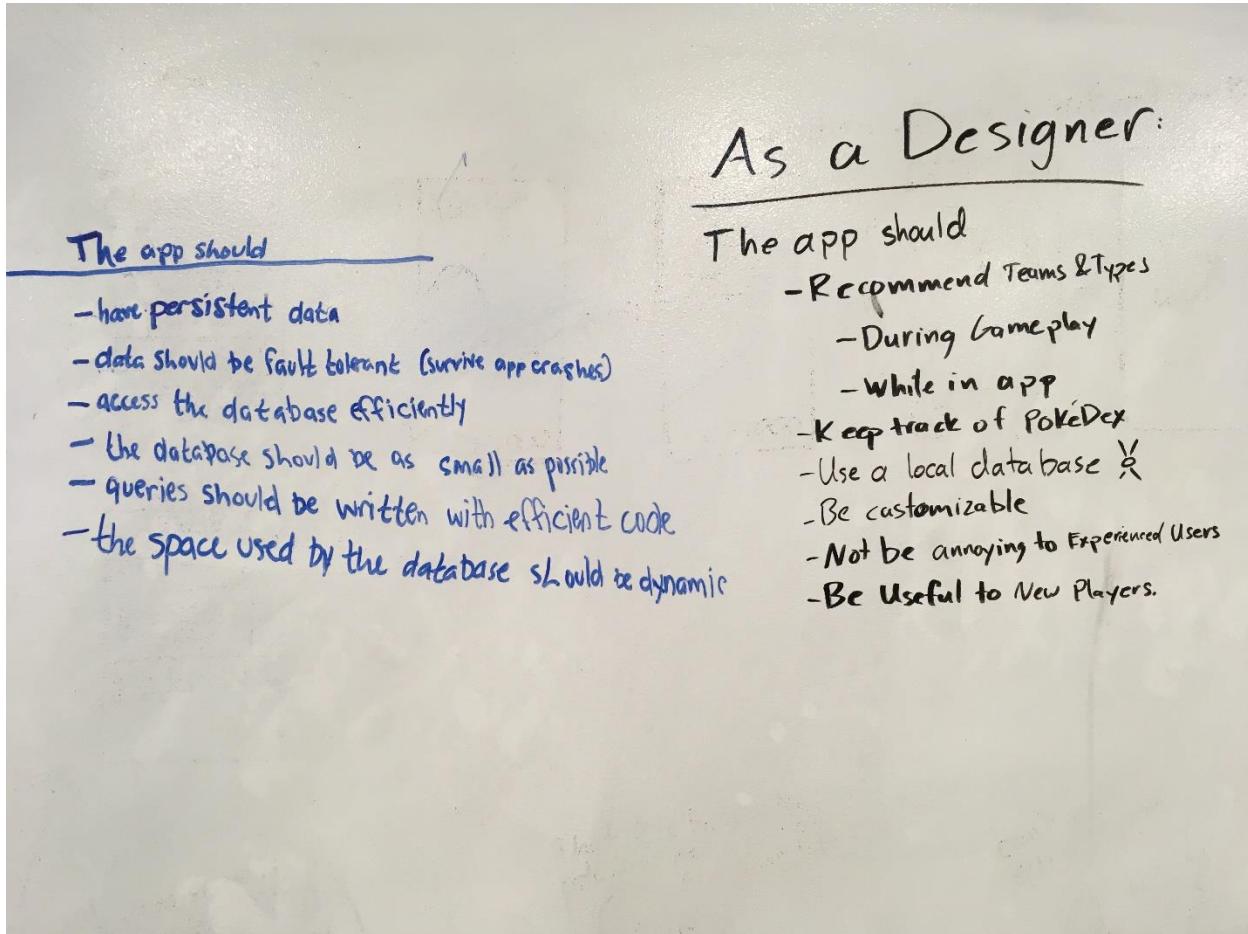
**In-App screen of recommended types and teams (Screen 2):** This screen shows the recommended team and types the user should use during a raid.



## 8. Designers Mental Model, conceptual design, and how it links the User and Designer:

To get the designer mental model we wrote down what we (the designers) expected from the product. We then took a friend of the designer group (Kaitlyn Cushing) and asked her what she expected from a Pokémon Go Raid Assistant/Companion to get the user model. (We also looked at our contextual inquiry and analysis to get the user model).

Designer Mental Model:

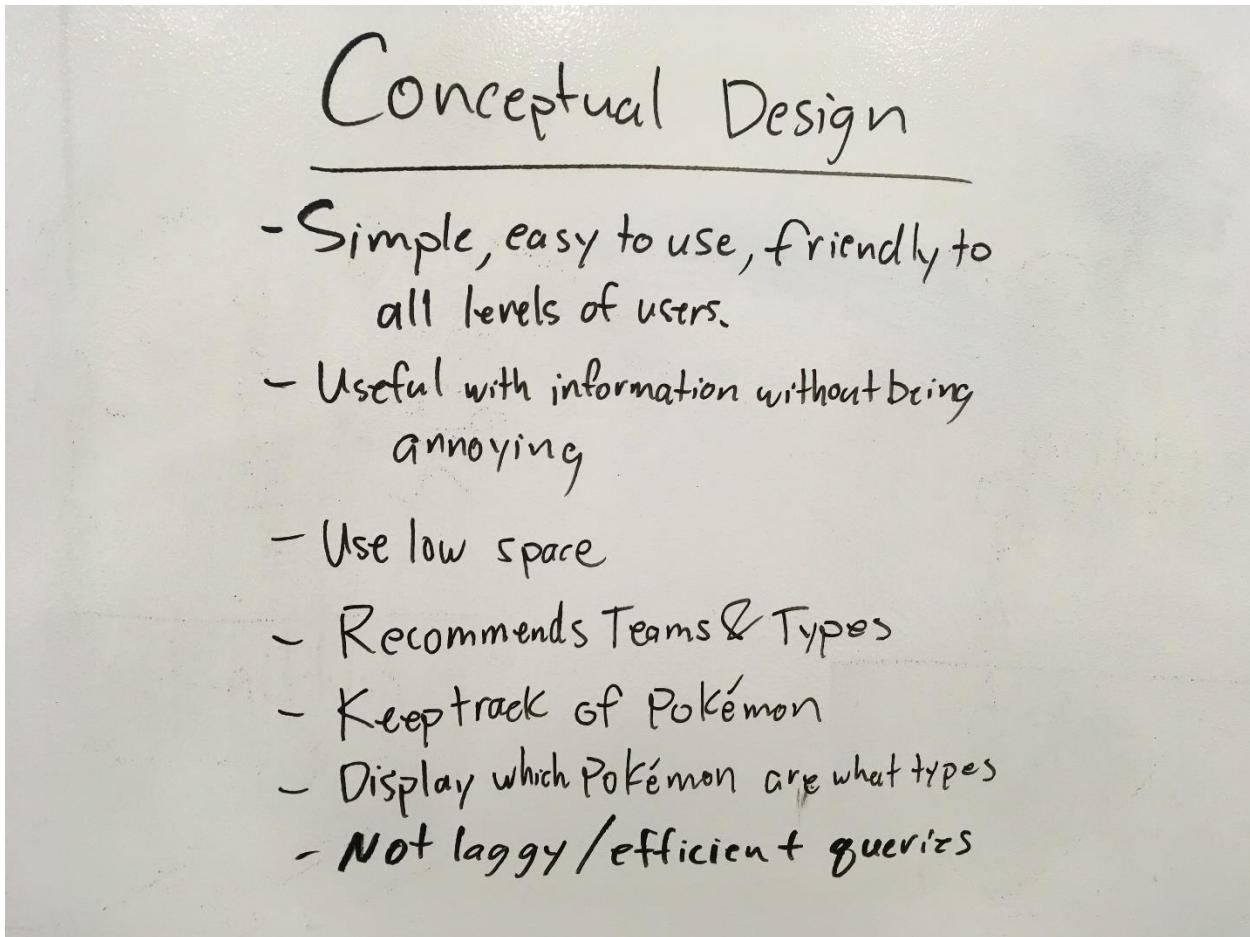


User Mental Model (Drawn up by a potential User)

## As a User:

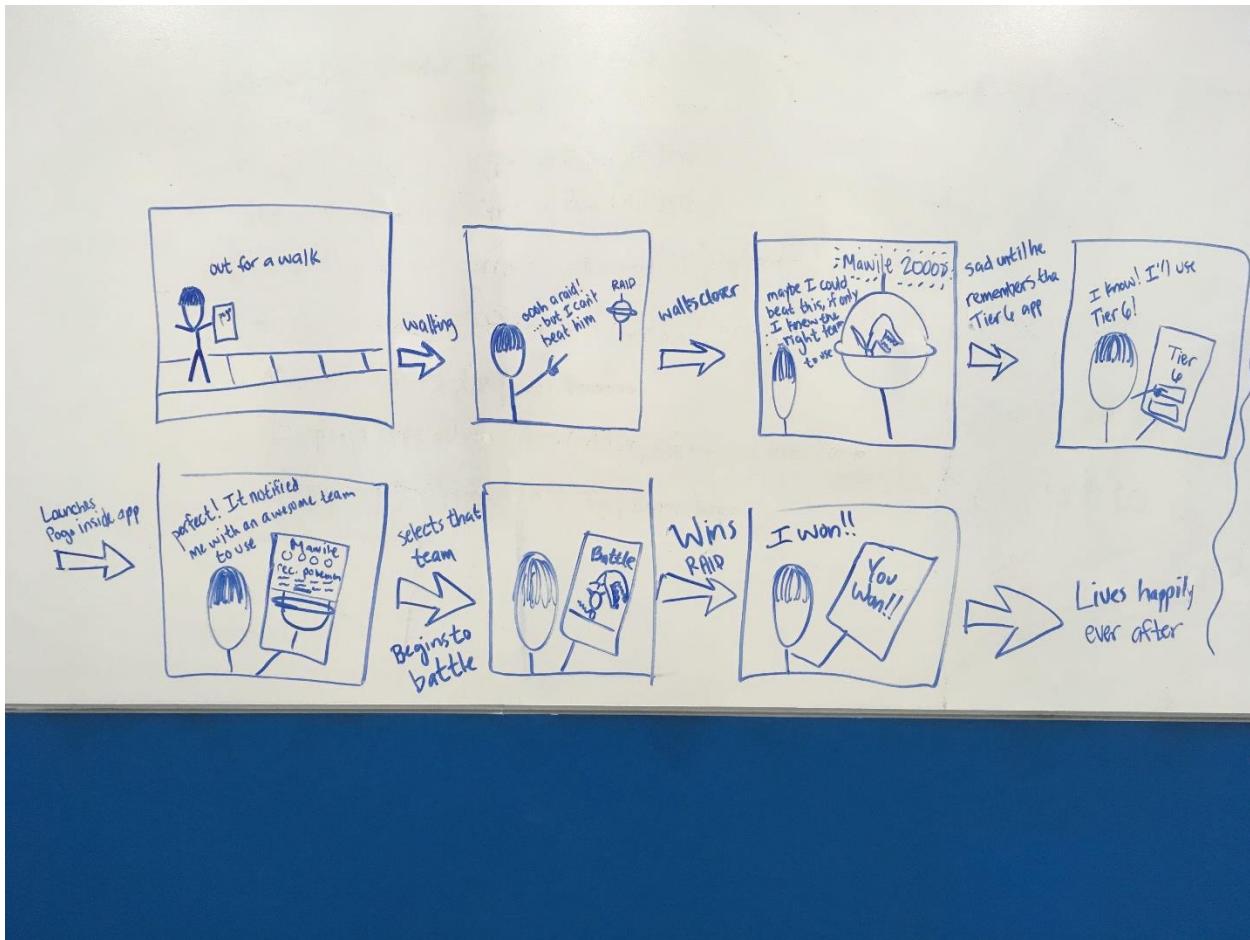
- Not super laggy (pls)
- Simple & easy to read/use - not put a lot of clutter on the screen
- Not a lot of storage space :)
- Informative but not too 'wordy'
- Recommend teams / types
- Have an easily accessible list of which pokémon are certain types (?)

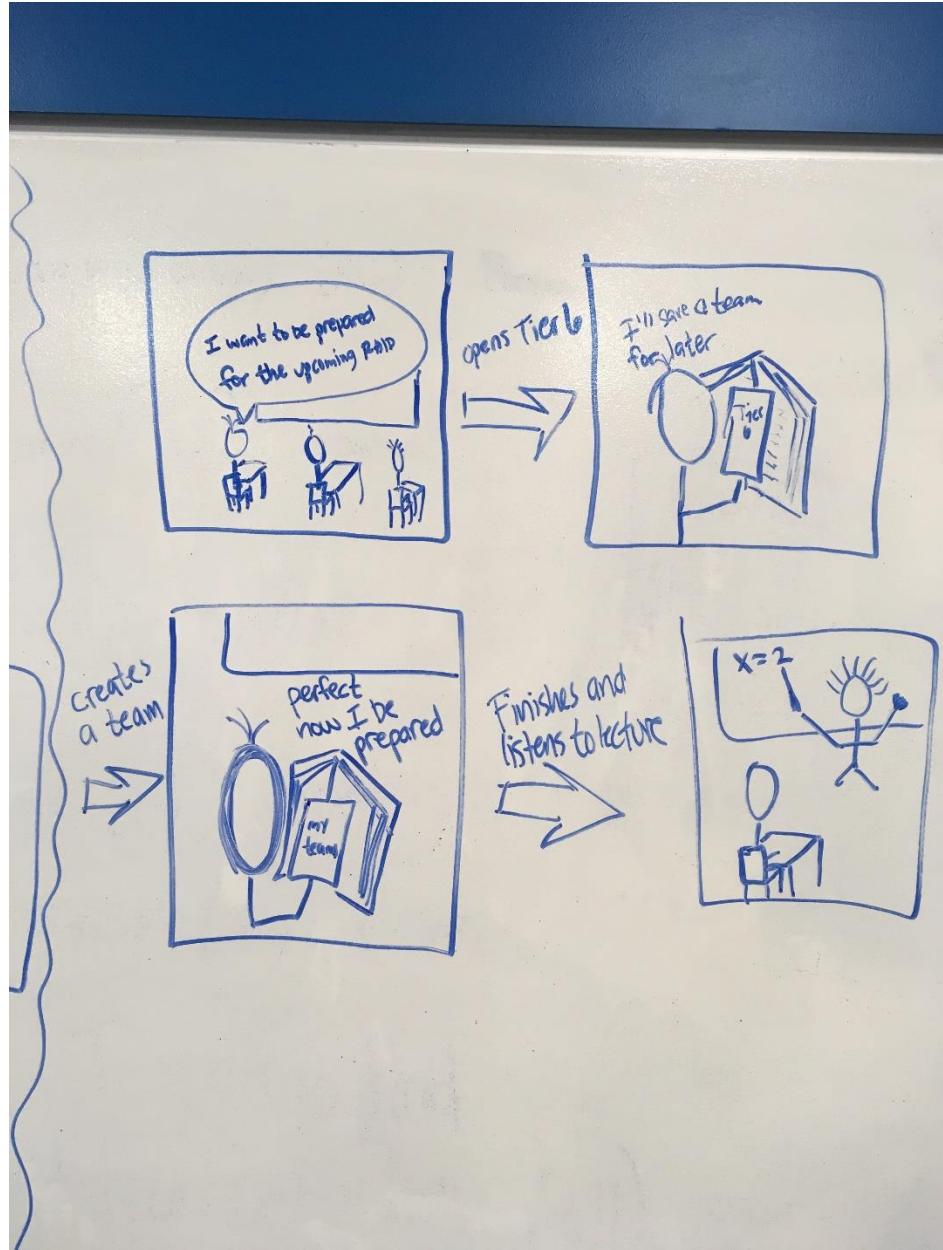
Conceptual Design: This was used to tie the two mental models together, to make sure the requirements, wants, and ideas match.



## 9. Storyboard:

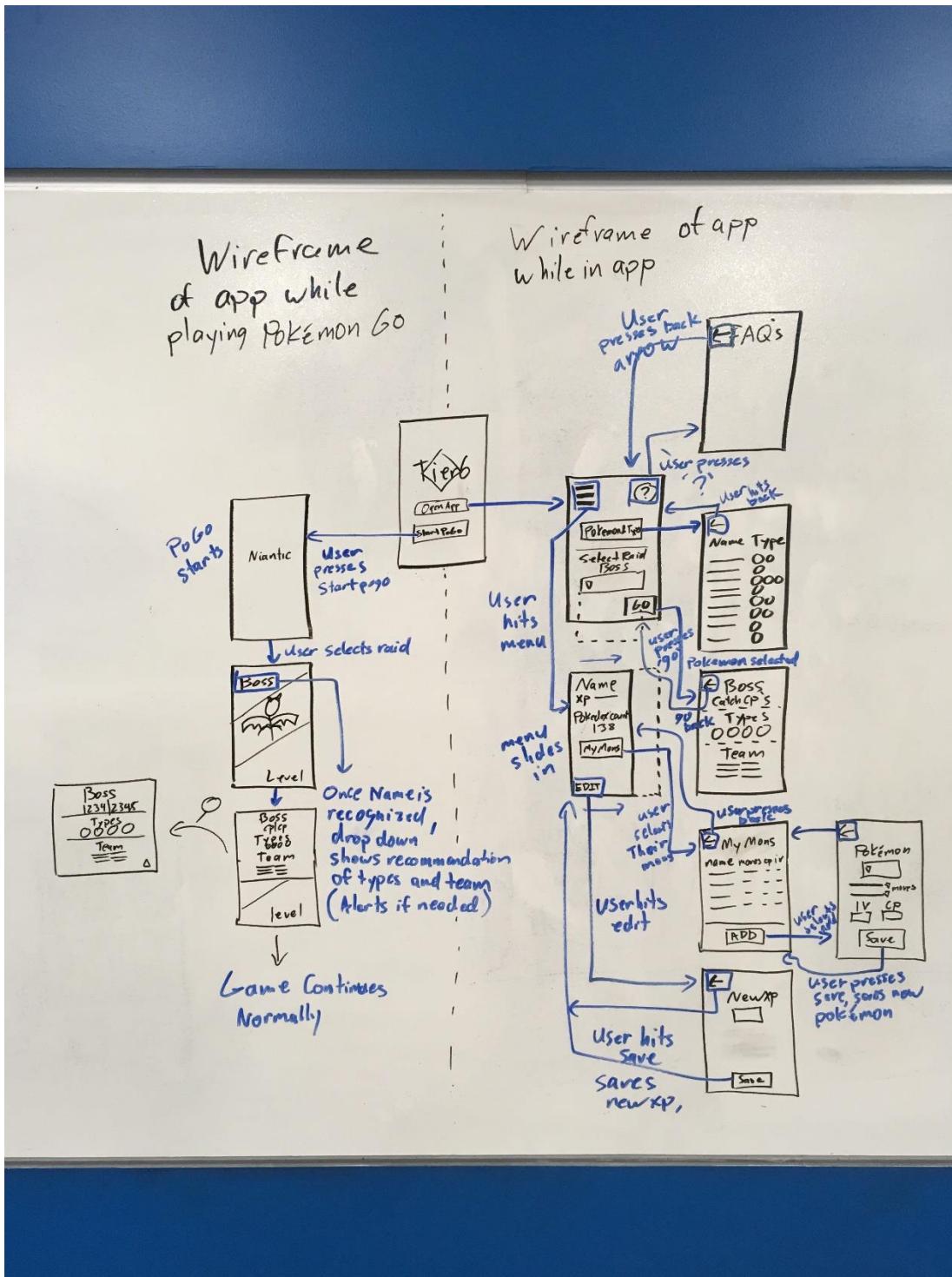
We drew out a storyboard to show what the user would do when using the app. We really focused on the user interaction during gameplay. Since it can be complicated how the app will work while playing Pokémon Go.





## 10. Wireframes:

We drew a wireframe based on our Hierarchical Task Diagram and physical mockups that we created.



# Section 4

Development

## **1. Development:**

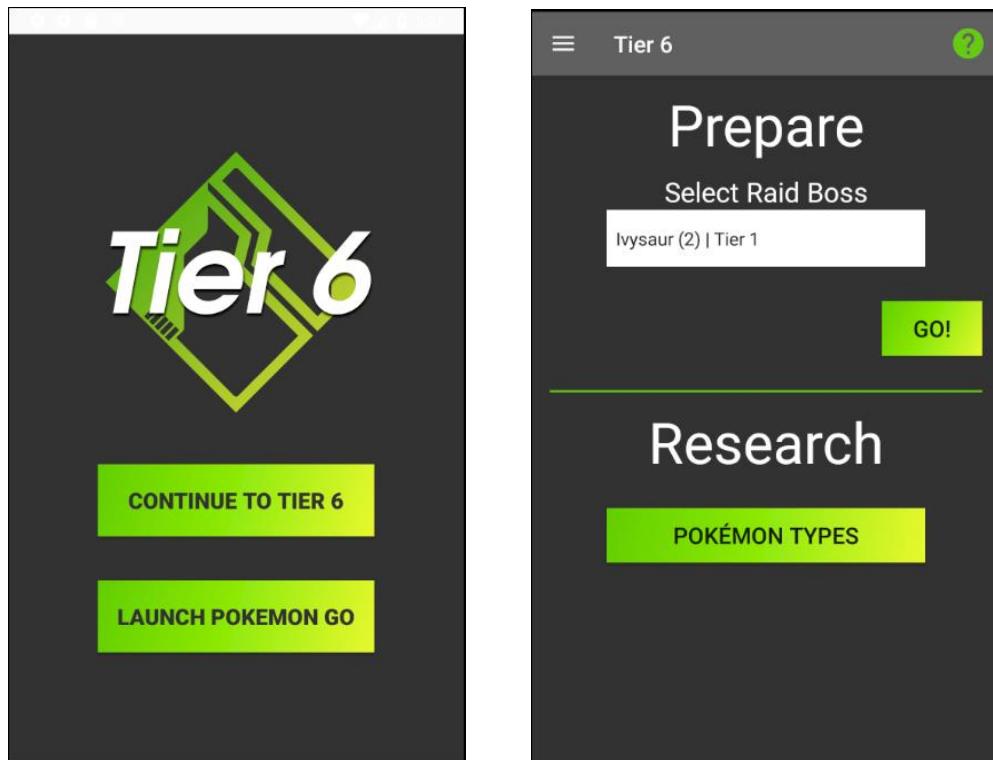
We based our project strongly on the wireframes that we built, as well as some early mockups that we had made for the web class. Along the way (mainly due to time and limitations of our knowledge) we decided to forgo the creation of the screen watching aspect of the app and only implement the internals. But it was very cool to watch the project come together and work.

For the most part, Alex worked on front end and getting activities ready while Ty worked on the back end of the project (databases, images, etc.). Toward the end we finally married the two portions of the project and were able to see a working app that was controlled by the front end of the application, but still implemented the back end of it as it was designed.

Other than a couple of items being skipped, the app works as we intended it to.

## **2. Screenshots:**

Below are some screenshots of the final version of the app.



≡ Tier 6 ?

## Prepare

- Ivysaur (2) | Tier 1
- Charmeleon (5) | Tier 1
- Wartortle (8) | Tier 1
- Metapod (11) | Tier 1
- Magikarp (129) | Tier 1
- Bayleef (153) | Tier 1
- Quilava (156) | Tier 1
- Croconaw (159) | Tier 1
- Wailmer (320) | Tier 1
- Swablu (333) | Tier 1
- Shuppet (353) | Tier 1

## Wailmer

Max CP: 814 | 1017

---

### Recommended Types



---

### Recommended Team

Raikou	Jolteon
Ampharos	Sceptile
Meganium	Tangela

## Latios

Max CP: 2082 | 2603

---

### Recommended Types

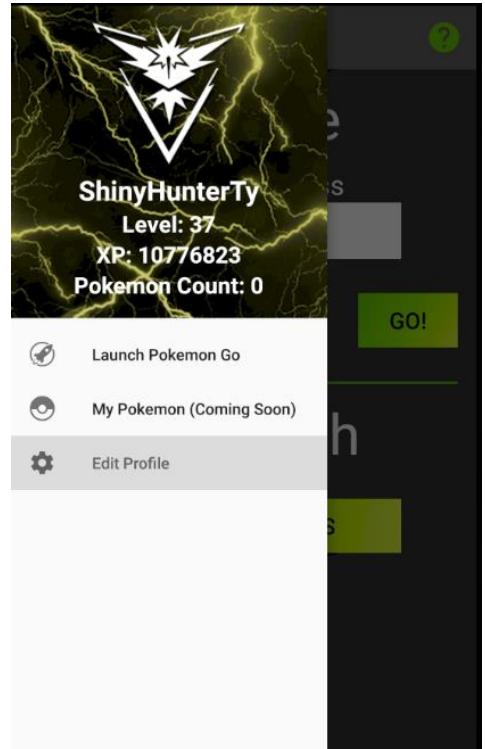
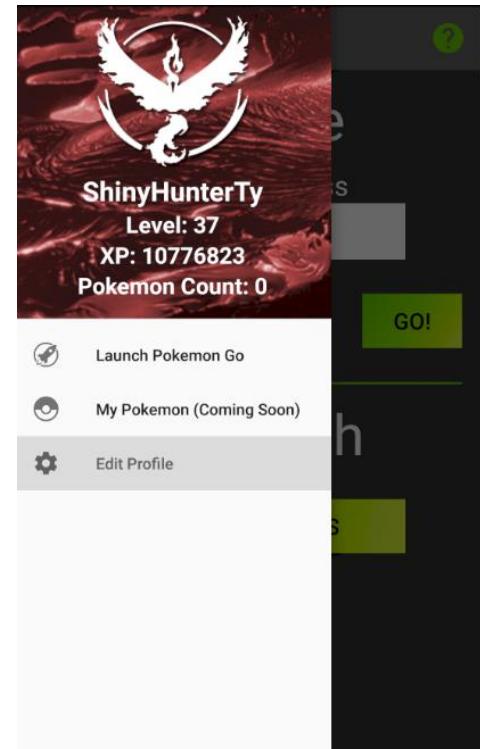
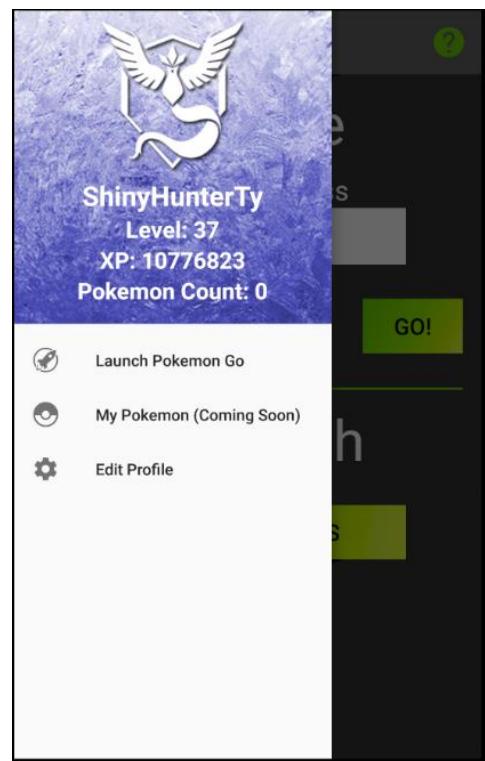
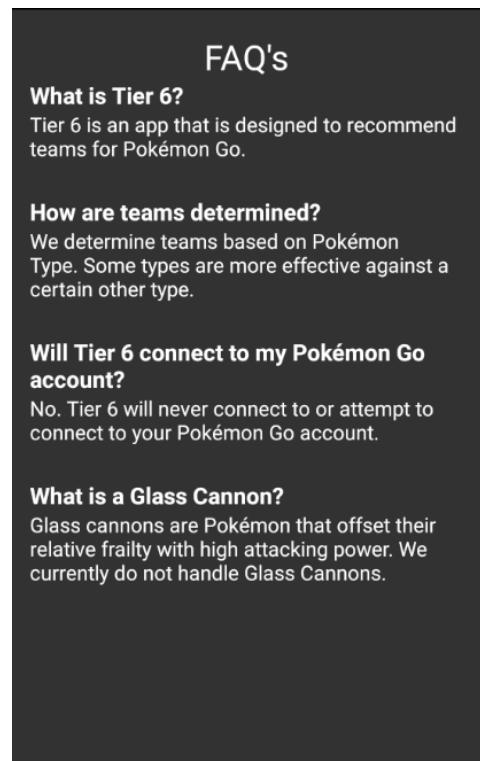


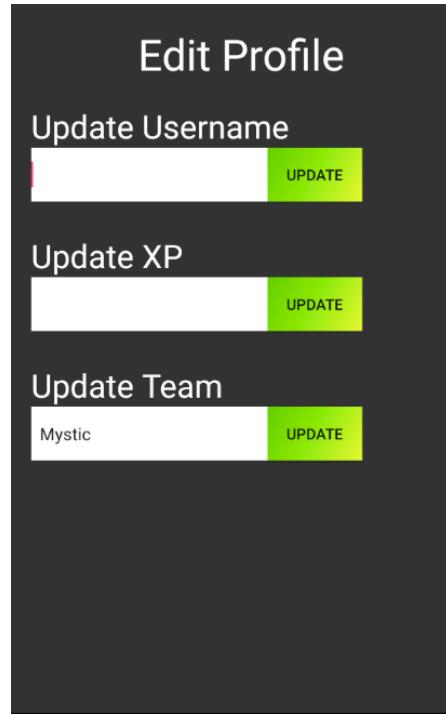
---

### Recommended Team

Pinsir	Banette
Shelgon	Glalie
Misdreavus	Dragonair

Aerodactyl	 
Snorlax	
Articuno	 
Zapdos	 
Moltres	 
Dratini	
Dragonair	
Dragonite	 
Mewtwo	





### 3. Reflection:

Overall, we are able to learn a lot from doing this project. Not only was it a good review of what was learned in our User Experience and Design class, but also about versioning and programming. We will openly admit that this project was probably a little out of our league, and we realized that when we could not implement the screen watching aspects. But we are glad that we took on such a large project, cause it forced us to learn the language, the skills, and forced us to continue researching further and we are both better app designers and developers because of it.