

CS 3205: HCI IN SOFTWARE DEV.

FINAL REPORT UNIVERSITY OF VIRGINIA SPRING 2017

Muhammad Sareini
Nirali Shah
Alex Mangels
Muriel Sandel

ABSTRACT

The purpose of this paper is the documentation of the development process for a *Pet Lost and Found* mobile application. The first chapter begins with a theoretical overview of the system, including user roles, personas, a task analysis, analysis of the user base, and a list of anticipated user requirements and goals. The two central functionalities of the application are reporting a pet as lost (likely done by the pet owner) and registering a pet as found (done by any user who finds a pet). The goals and requirements determined for this product revolve around simplicity and ensuring that users with any level of experience can easily navigate through the system.

In Chapter Two, the paper demonstrates four different prototypes; two for reporting a lost pet, and two for registering a pet as found. The prototyping techniques chosen were paper prototypes and wireframes, providing a solid foundation and an efficient way of communicating our ideas for the system without requiring a high-fidelity model that would exceed the scope of the course. Each of the prototypes is included in the paper and described in detail, along with an explanation of the thought process behind it.

As the final part of the project and in order to gain feedback for our work, we conducted user tests with two of the prototypes. Chapter Three provides the script used during each of these sessions explaining the two tasks we aimed to test and improve, the questions each subject answered, and elaborate notes on the user's reaction to our interface designs including the time it took them to navigate through each task. The testing sessions revealed that, while some portions of our prototypes were intuitive, there were some specific things that could improve usability that we simply hadn't thought of. This valuable feedback would be essential for future work on the product. An analysis of these results including graphic representation of the user feedback can be found at the end of the paper.

CHAPTER 1

User Task and Analysis

OVERVIEW OF SYSTEM

Our system, in the most basic sense, is an application that links people who have lost their pet(s) to people that have found a lost pet. In order to achieve this, our system does a few things. Upon the first use of the application, users have the option to register their pet using a picture and our system's facial recognition software in the event the pet is ever separated from its owner. If the pet is ever found, another user can upload a picture of the pet and it can be compared against animals in the database, notifying the user who found the pet and the owner of the pet if a match is found. If it is not in the database, the found picture is added in hopes that the owner then uploads a picture of the pet that he or she has lost so a match can be made. When a pet is found, or lost, the user can post information about the pet such as current location, current state of the pet, where the pet was found to give users a visual of pets that have been lost and found in their area. This is to ensure if neither party has uploaded a picture to the facial recognition software a match can still be made.

This system is interesting from an HCI point of view because it has a simple task: link people who have lost pets to people that have found pets. Since it has such a simple task, there are no real restrictions on how our application will implement its function. This means the whole essence of the application, its whole identity, is how the interface is designed. The whole success of the application rests on the interface being designed around our users. Our project is going to focus on the ways we can link pet owners who lost their pet to people who have found a lost pet, while using facial recognition software to aid in the speed of connections and minimize confusion about whether the person who claims to own the pet really does.

USER ROLES / ANALYSIS OF USER BASE

Pet Owners

- A. Primary User
- B. All ages and builds (Can be no physical restrictions)
- C. Occasional use (registering pet, searching for lost pet)
- D. Will register their pets to be found if lost, or search for their lost pet

Shelters

- A. Primary User
- B. All ages and builds (Can be no physical restrictions)
- C. Frequent use
- D. Deals with a high volume of lost pets
- E. Posts found pets only (possibly dealing with multiple pets at a time)

Pet Finders

- F. Secondary User
- G. All ages and builds (Can be no physical restrictions)
- H. Scarce use (only when a single person finds a pet)

- I. Only will post/search that a pet is found
- J. Since not regular user, must be able to easily post a found animal without having to go through many hoops

The interface should reflect some of the things that are highlighted in this section. First of all, a majority of our users will not use this interface daily. Pet owners will use the application to register their pet and attempt to locate it if it is lost. Pet finders, as we described them, will only use the app to post that they found a pet. Shelters, unlike the other two main user roles, will use the interface a lot. They will have a constant stream of animals to put into the application (facial recognition and or postings) to see if they are registered or register that they have been found. The different frequency of use and how the different user groups are very important when determining usability goals and should be reflected in the design of this interface to ensure a worthy user experience.

This interface needs to work for a wide variety of users, so the interface must not hinder users do to physical attributes. Due to the fact that all different types of people own pets, this interface is unable to target a specific type of person / pet owner. There is no way to predict what type of pet owners will lose their pet or what kind of people find someone else's pet. This is a key idea that should go into the design of the interface; if it is not user friendly to a wide range of people, the application will not be used. Having many users that actively use the application is key to its success since its purpose is to link lost pets back with their families. If it is not used widely, links will not be made.

USER PERSONAS

- **Pet Owners**

- Bob is a loyal and loving dog owner, located in a large, urban city. With the chaos of raising his 4- and 6-year olds - the front door is always sporadically swinging open, with people running in and out. He wants to be extra cautious about his dog, Fluffy, in making sure she doesn't run out one day and get lost in the big city. He is a moderate to expert computer user, and has access to a home desktop, work laptop, and smartphone device at all times throughout the day.

- **Shelters**

- Cindy is an animal-aficionado and has devoted her career to helping and saving local pets. She works at a shelter in the city, where they take in lost or abandoned dogs found on the streets - taking care of them and nursing them back to health. She really wants to help willing and loving pet owners be reunited with their dogs and cats. Because this is her job, she has between work hours to access a laptop and smartphone device that the shelter provides her to be notified of and respond to lost pet situations. She is a low to moderate technology user.

- **Pet Finders**

- Debbie is a teenager attending the local city high school. On her way walking home from school, she noticed a stray dog on the street. She is unsure of

what to do or what avenues to use in helping the dog find its home. She herself owns a dog and understands what it must feel like to lose a pet. She is avid and determined to get the stray dog back to its owner. Debbie is an expert with her smartphone device and also has access to her family's desktop computer at home at any time during the day.

- **Conclusions**

- There is a clear connection between lost pet owners and lost pet finders/shelters. The two groups would like to be able to find each other, accordingly, and communicate, as they both have the same goal - receive/return the lost pet to the proper owner. To accomplish this goal on our interface, all groups of users will need 1) effectiveness, 2) utility, and 3) learnability.
 - Effectiveness - pet owners and pet finders/shelters will need the application to help them achieve the initial reason they came to/registered with the site, AKA successfully find and return lost pets to their owners. Shelters sole purpose and mission is to accomplish this goal, so during shelter workers' shift hours, they will need to be able to utilize this site to reunite pet owners with the pets that are turned into the shelter by Animal Control, etc.
 - Utility - pet owners will need tools and resources on the application that will allow them to register their pet, mark when their pet is missing, and communicate with pet finders/shelters when their pet is found. On the other hand, pet finders/shelters will need good utility of the application so that they can post when they have found the pet, the location of the pet, the condition of the pet, and communicate with the pet owner so that the goal can be accomplished and reuniting can occur. These tech tools will include a picture-facial recognition to register the dog (for both pet owners and pet finders/shelters), easy lost/found button switches, and a messaging component that will allow the two parties to contact and chat with each other.
 - Learnability + Memorability - because losing a pet does not occur frequently, pet owners will need easy learnability of the application so that when they do register or lose their pet, that they can do so quickly and without errors. Additionally, finding a stray dog or cat on the street does not happen very often, so local pet finders (unless they are pet hunters looking for cash rewards), will need to have easy-to-use features that allow them to swiftly post pictures, information, and notify the correct owner that their dog has been found. Shelter workers, on the other hand, will likely be the most frequent users of the application, and will need features that spawn quick memorability of buttons, pages, and their respective key functions. They will be using the application often, because they will be the receivers of the most lost pets, and will be doing lots of "found" animal postings and communication on the interface.

TASK ANALYSIS

High Level Description of Tasks

- A pet owner registers their pet on the system and connects to the facial recognition database
 - User Roles: Pet Owners
- A pet owner informs the system that their pet has gone missing
 - User Roles: Pet Owners
- A pet owner informs the system that their pet has been found
 - User Roles: Pet Owners
- Someone who has found a pet utilizes the application to find the owner of the missing pet
 - User Roles: Pet Finders, Pet Owners
- A pet owner updates their pet's information in the application
 - User Roles: Pet Owners
- A pet owner deletes their pet's information from the system
 - User Roles: Pet Owners
- A shelter, which has received a recent influx of animals, searches for matches with Missing Pets in the system
 - User Roles: Shelters
- A shelter registers pets which have no matches as Missing Pets
 - User Roles: Shelters, Pet Owners
- A pet finder contacts the owner of a missing pet once he or she has found a matching pet
 - User Roles: Pet Finders, Pet Owners
- A pet finder contacts a local shelter after no match is found with the missing pet
 - User Roles: Pet Finders, Shelters
- A pet finder registers their own pet in the system (after already being in as a pet finder)
 - User Roles: Pet Finders

Use Case #1: Registering a Pet and Connecting with Facial Recognition Database

- **A. Description**
 - This describes how a user can register their pet on the Lost and Found application, and upload a picture in order to connect to the facial recognition database
- **B. Actors**
 - Pet Owners, Shelters
- **C. Preconditions**
 - The member must have downloaded the Lost and Found app
- **D. Flow of Events for Main Scenario.**
 - 1. User accesses the "Register" functionality from the home interface
 - 2. The application brings the user to the registration interface, which has fields for name, address, pet's name, etc.
 - 3. User fills out the general information, and then submits the information to the system

- 4. The system stores the user's information, then proceeds to the "Photo" interface, which is connected to the user's phone camera
- 5. The pet owner takes a picture of their pet and uploads it to the application
- 6. The application utilizes its facial recognition feature in order to measure the overall facial structure of the pet, and characteristics such as distance between the eyes, nose, and jaw edges are stored in a public database. The interface then provides a "Finish" functionality
- 7. The user utilizes the "Finish" functionality
- 8. The system directs the user to the home interface of the application
- **E. Alternative Scenarios.**
 - 1. The picture that the pet owner uploads is too blurry, or cannot be processed by the facial recognition software. The user is sent back to step 5.
 - 2. The user only wants to register on the application in case he or she wants to report a found pet, in which the user is directed from step 2 to step 8.
- **F. Postconditions:** Once the user has finished registering, the information will be stored either in the facial recognition database and in memory, if a photo of the pet was uploaded, or just in memory, if a photo wasn't uploaded
- **G. Special conditions:** N/A.

Use Case #2: Informing the application that a pet has gone missing

- **A. Description**
 - This describes how a user can notify the Lost and Found application that their pet has gone missing
- **B. Actors**
 - Pet Owners
- **C. Preconditions**
 - The member must have downloaded the Lost and Found app, and registered their pet with the facial recognition database
- **D. Flow of Events for Main Scenario.**
 - 1. Pet owner accesses the "Missing" functionality from the application's home interface
 - 2. The application brings the user to the missing pet registration interface, which provides functionality to use information that has already been registered, or to update a pet's information
 - 3. Pet owner utilizes the functionality to use their pet's current information
 - 4. The system finds the pet's facial recognition information in the database and the pet's name, location, and physical description in memory. The missing pet interface then allows the user to enter information about the location where the pet was last seen.
 - 5. The pet owner enters the exact address where their pet was last seen
 - 6. The system stores this information along with the rest it has already gathered, and then sends a notification to any user of the app within a 10 mile radius of the home address and the last-seen location, including the uploaded picture of the pet, and all of its information

- 7. Once the information has been processed and sent out as a notification, it is displayed on the Map interface of the app as a “Missing” case, allowing all other users of the app to view the missing pet
- 8. The system directs the user to the home interface of the application
- **E. Alternative Scenarios.**
 - 1. During step 2, the user wants to update their pet’s information and the system redirects to the Registration interface, where the user can update any information they want, and when they submit this new information, they proceed to step 3
 - 2. If the missing pet has already been found and registered as a “Found Pet,” the system will inform the user that there is a match during step 7, and then will release the contact information of the Pet Finder to allow for communication
- **F. Postconditions:** Once the user has finished registering their pet as missing, the information will be stored in memory
- **G. Special conditions:** N/A.

Use Case #3: Informing the application that a pet has been found, and connecting with the owner of the found pet

- **A. Description**
 - This describes how a user can notify the Lost and Found application that they have found a lost pet, and how they can contact the owner of the missing pet
- **B. Actors**
 - Pet Owners, Pet Finders (Can be a pet owner or an ordinary person)
- **C. Preconditions**
 - The member must have downloaded the Lost and Found app
- **D. Flow of Events for Main Scenario.**
 - 1. A user finds a pet who is missing, opens the Lost and Found app, and accesses the “Found” functionality from the home interface
 - 2. The application brings the user to the Found interface, which contains a list of all currently missing animals and provides a “Match Using Facial Recognition” functionality
 - 3. The user proceeds with the “Match Using Facial Recognition” functionality
 - 4. The system brings the user to the Photo interface, which is connected to the user’s smartphone camera. The interface contains two different services, one of which is “Upload Pet to Database” and one which is “Match Pet”
 - 5. The user proceeds using the “Match Pet” functionality
 - 6. The system directly accesses the user’s smartphone camera to allow them to take a picture of the found pet
 - 7. The user takes a picture of the pet, and uploads it to the system
 - 8. The system processes the photo using the facial recognition functionality, then searches through the database for a match. It informs the user whether a match was found or not
 - 9. If a match was found, the user can access the phone number of the owner of the matched pet, or message them through the system. If a match is not found, the app informs the user that it wasn’t found, and then stores the

information in the facial recognition database under found animals, and updates the map interface to include the found location of the animal, and its characteristics.

- 10. The user uses the provided functionality to contact the found pet's owner if a match is found
- **E. Alternative Scenarios.**
 - 1. The person who has found a missing pet has not yet downloaded the Lost and Found app, in which case they download the app, and then advance to step 1
 - 2. If a match is found but the pet finder cannot successfully contact the pet owner, the pet owner is notified that their pet has been found, and the pet's information is not removed from the "Lost" or "Found" data until the owner confirms the notification
- **F. Postconditions:** If a match is found for the pet, and the user notifies the system that the situation has been resolved, the pet's information will be removed from the public "Lost" and "Found" data
- **G. Special conditions:** N/A.

Environment

Because the purpose of this application is to help reconnect lost pets with their owners, most users will be accessing the system from their smart phone. The two main tasks are reporting a lost pet or reporting a found pet, and in both of these scenarios, the user will most likely be in a rush, and will be away from a laptop or computer. The case where this could occur, however, is when a shelter is using the system. Whoever is reporting the found pets from the shelter would most likely have access to a computer, and would use it rather than a smartphone. Because the users of the app will often be in a panicked state and will be trying to access and use the system as quickly as possible, it is important that the tasks are clear and easy to carry out.

USABILITY GOALS

Many pet owners like Bob, who was described above in the section User Personas, have busy and somewhat chaotic lives. Especially because it may be difficult for them to keep an eye on their pets at all times, they should be able to easily register them. Our usability goal for pet owners with busy lives is that a pet registration takes no more than four minutes when done by a person with moderate experience using computer technology. This way it can be done easily between daily tasks, and more pet owners will be encouraged to use our system.

Employees and volunteers at animal shelters are likely not required to have much experience with computers. For a devoted and busy person such as Cindy, this could become a real issue when trying to quickly use our system. One of our goals to ensure the success of shelters returning pets to their owners is to limit the required number of clicks or taps in order to upload a picture and use the "Match Pet" functionality to ten or less, on average. This should include opening the app, choosing "Found", proceeding to the "Match Using Facial Recognition" functionality, taking a photo, uploading it, as well any corrective

measures that need to be taken if one of the previous steps is performed incorrectly (i.e. if "Upload Pet to Database" was chosen instead of "Match Pet").

Because losing a pet is a stressful experience and pet finders may be in a hurry, it is important to provide feedback quickly to either side once a picture has been uploaded. Another one of our usability goals is therefore to optimize our database and the search algorithms to take no longer than 15 seconds in the matching process. As the database grows, this may be extended to 25 seconds.

One of the important requirements for the app, as described in the next section, is safety. We not only want to keep the user's (specifically the pet owner's) data and identity in the system safe, but we want to make sure they *feel* safe as well, while still keeping up the required speed and simplicity. We will therefore create an initial link to a second account (email, Facebook...) or a phone when they first register, and only require them to re-confirm this every three months. When a match is made, they will need to input a code, which is automatically sent to that second account or phone before being able to communicate with the pet finder. This way we ensure that using the app stays simple and quick but the user knows that nobody else will be able to pretend to be them and take their pet. The user should rate overall feeling of security at least a 4 out of 5.

Because it is essential for the system to be effective that as many people as possible upload their pictures, whether of pets lost or found, we need smartphone users to be aware of the app and want to use it. Arguably one of the most efficient and inexpensive ways to advertise a product is by word of mouth. We will therefore aim to make the overall experience pleasant enough for at least 80% of our users to be likely to recommend the app to a friend. Whether we achieve the last two goals will be determined based on user feedback.

USABILITY REQUIREMENTS

As previously stated in the User Personas section, there are several usability requirements that will be essential for our system. This section will discuss those mentioned earlier in more detail and add new usability requirements.

The first, and arguably most important requirement is Effectiveness. All users will rely on the system to perform the desired functionality, whether it is to find a beloved pet or return a lost animal to its owner. If the system is not effective at doing this, none of the other requirements will be able to make up for that drawback. The app will be essentially useless if Effectiveness is not made the utmost priority. This criterion is met if all of the system's goals are achievable. If there is any part of any user goal that is not achievable, then the overall functionality is damaged and the app is no longer fully effective.

The requirement of Utility goes hand in hand with the goal of Effectiveness described above. It is not simply enough for there to be *some* way to achieve a certain goal, but it must be in a way conducive to the whole process. The different options for actions on the interface must be logical and each command must have the correct functionality.

Commands include taking pictures, uploading pictures, navigating between the different screens within the app and every other potential action that has been described in this paper. If any of the items implemented in the system do not have the desired functionality,

this will hurt the Utility of the whole app. For example, let's assume a button that says "Submit" saves a picture to the phone instead of submitting the file for a pet to the system's database. Whether or not there is a different way to submit the file somehow, this significantly impairs the user's ability to complete the desired action. It is essential that users in each role can navigate through the necessary actions using the utility items we provide.

Another major concern for our app is Safety. The app must be able to confirm a person's identity before allowing them to communicate about a pet's location or arranging for the pet to be returned. Especially for more expensive pets, such as specific breeds of cats or dogs, there is a potential for users to falsely identify themselves as the owner in order to be given the animal without having to pay for it. Owners need to be able to feel safe with the system, or else they may not want to register their pet in the first place in order to prevent any possibility for theft. This is somewhat easily accomplishable by adding password protection to each account and requiring a confirmation of identity from a second account or phone number, such as email, text codes or possibly a Facebook login.

Because neither pet owners nor finders are likely to use the app very often, both Learnability and Memorability are very important. A user who is looking to register their pet will want to be sure that the whole process has been completed correctly in order to avoid any errors when trying to find the pet. Someone who has found a pet and is trying to return it should run into as few obstacles as possible, because they may give up on the task if they are in a hurry and it takes too long. To make users more confident and show them exactly what to do, the interface must be designed in a way that makes it easy to learn, regardless of what action should be performed or how much experience the user has with computer systems. For shelter workers, Memorability will be more important because they will be using a variety of functionalities on a relatively regular basis, and will not want to re-learn how to use the app each time, even if it is easy to learn. The interface must therefore also be laid out in a way that is externally consistent or have other characteristics that will make it easy for a user to remember how to perform a task.

CONCEPTUAL MODEL

The nature of our interface's conceptual model is one that is based on activities. The users will want to perform an activity, and they will interact with the interface. Some abstract characteristics of the interaction would be uploading the picture of a pet to the database, the interface linking a lost post to a found post, and using the interactive map that shows where a pet was lost and or found.

Our user base will understand their interaction as exploring and browsing. They will search for lost animals, browse lost and found animals, and explore maps that show where these pets were lost and or found. Users will expect their requests to be quick and seamless. They will not be concerned with how or why the application is doing what it is, as long as it works. Users will want the functions they desire to be easily accessible so they can interact with them without a lot of effort. Also, users will expect our application to present accurate and true data. They will not want to be notified their pet is lost when it is home, or that it has been found if in fact it has not.

The conceptual model of our system that our users will have will definitely influence the design of the interface. Users will primarily be searching for or browsing through lost and found pets. This means the functionality of the application needs to be clear. Searching for a pet should be easy, obvious, and painless, as should linking with someone who's found or lost a pet. The interface should also have perceived affordances that invite the user to perform the action that the interface provides.

CONCLUSION

From our thorough research, user discovery, potential task-completion implementations, and ideal conceptual model, we have found an interactive and connected way to help lost pet finders communicate with lost pet owners. We have deeply understood users - their different roles, possible personas, and needs and requirements from the interface. There are many different tasks that can be carried out by users, the most important of which are notifying that a pet has gone missing, or informing the system that a pet has been found. For these tasks, the entire user interaction with the system is important to consider in order to assess the functionality of the system. Through this analysis, we have also have revealed the most important usability requirements that we must keep in mind when designing an interface. We realize that for our application to be successful, it must emphasize: effectiveness, utility, safety, and memorability. The steps taken in this report were key to identify our user base, and give us the basis that user centered design can be built upon.

CHAPTER 2

Low Fidelity Prototyping

PROTOTYPE 1: LOST / FOUND PET

USABILITY REQUIREMENTS

For this prototype, the usability goals were the major influence on how it was designed. The usability requirements that were used, and can be evaluated later, are usability, effectiveness, memorability, and utility. In order to make sure we meet these goals, this prototype was designed with the following usability requirements in mind:

1. A user who has not used the application in three months is still able to report a lost or found pet.
2. The user is able to search for a lost pet in two ways (searching, facial recognition).
3. A first-time user should be able to successfully navigate through the steps required to search a lost/found pet 90% of the time.
4. The user should be able to use the search feature to find a list of lost/found pets in 2 clicks 90% of the time.

DESCRIPTION OF PROTOTYPE

This is a prototype to show a possible way of using the software to report and search for a found or lost pet. For this prototype, the focus was on using a search based method to report and find lost pets with the aid of an interactive map for a visual. The prototype was designed to be easy to use, clean, and simple. The buttons on the screens invite the user to press them, and there is no confusion what something does because everything is labeled. The headings below are used to group together screens that link chronologically.

Screen 1:

This is the layout of the home screen. It is simple with only three possibilities (there will be account links, etc. but these were not important for what I am trying to prototype). It has a search bar, a map that is not interactive but becomes interactive, and a report lost and report found button.

Screen 2:

This screen shows what happens when you click the map. It zooms in, and it becomes interactive. This screen also allows you to interact with the lost and found pins, showing a picture of the pet if available, and the type of pet (dog, cat, etc.). The bottom of the screen has also changed. Instead of reporting lost and found, there are now lists of lost and found animals near your current location. The user is able to interact with the list and get more detailed information of the animal upon selection.

Screen 3:

This screen shows what happens if you click the search bar on the interactive map. Upon selection, a drop down list appears. This gives the user to search for a type of lost pet (dog, cat, etc.) and location. The user is also able to select a search radius. This search criteria moves the map accordingly to matches that are found. The near you list is updated as well,

and the user has the arrows in the bottom right of the map to toggle between matches with their search criteria for easy movement.

Screen 4:

This screen shows what happens if the user clicks the main search bar from the home screen. This is the most detailed search criteria on the application, and this is the main functionality that this prototype is testing. The list contains things like name of pet, color, location, etc. as well as a radius. Once the user is finished narrowing the search, they press the search button and continue with the search-based design.

Screen 5:

This screen is the what the user sees after they press the search button on screen 4. Two lists are shown to the user: found and lost animals based on the search criteria. Above the list, there is a radius that quickly allows one to change the search radius. The user can go back to the previous screen if they wish to. The lists contain a picture (if available) and a short description of type of pet, name, color, and exact location. All of these pictures are interactive to allow the user to easily obtain more information on a pet if they wish.

Screen 6:

This is the screen if you press one of the potential matches from screen 5. This screen contains an enlarged picture of the pet, and a description entered by either the owner who lost it or the person who found the pet. This is the most detail that is available to the user about any given pet. On the screen, there is a mark as found button that can be clicked to begin the process of notifying the owner that his/her pet has been located.

Screen 7:

This is the screen if you press mark as found from screen 6. At the top of the screen, is the owner information taken from the owner's profile if the dog is registered with a certain owner. Under that, there is a location for the person who found the pet to enter their contact information such as name, email, phone, etc. and a short description or photo if they want with the location. Under that, is a link to message the owner, again if they have the pet registered with their profile.

Screen 8:

This is the screen that shows up if the user presses the lost/found button that is located on the home screen. The screen in the prototype is the same for both, for the sake of prototyping. There would be very small differences in the two in implication, but the differentiation was not necessary in the prototyping phase. This screen has a place for the user who found the pet to select the type of pet, the location where the pet was found, and space for additional comments if the user deemed necessary. There is also an option for the user to allow this pet to be linked to their profile. This allows the owner to contact them easily with their information that was provided upon signup. At the bottom of this is facial recognition, and it is required to slide down to it.

Screen 9:

This screen shows the facial recognition process from screen 8. To get to this screen, the user must scroll down on their device. This allows the user to upload a picture of the found pet which puts it into the facial recognition software. If there is a match, the app takes the user directly to the owner's profile (since they must be registered for there to be a match). If not, the picture is added in the event that the owner registers their dog after it is lost to help aid in the speed of the match process. At the bottom, there is a report button. This is clicked, and the process is finished.

DESIGN RATIONALE

There were a few reasons that I chose this design. The functionality that this prototype is testing is the application's lost/found pet, as described earlier. This application also uses facial recognition. Since the idea of facial recognition and photo uploading could be confusing, I leaned toward a searched based interface to aid in learnability. This does hurt effectiveness slightly, however this prototype needed to be easily usable for all groups since pet owners are so diverse. The search based interface is something that is familiar to most people, therefore it is what I decided to use. I chose paper prototyping because of how time effective it is when taking into consideration what you are learning. While paper prototyping, you can easily make changes and scrap ideas as a whole quickly while learning the most information for the time being put in. I believe it was the best choice, I even made changes while making this prototype because I saw things were really not going to work. The user base influenced all aspects of this prototype. The user base influenced the search-centered functionality, as described earlier in this section. To account for my variety of users, I had to design an interface that would be easily learnable and memorable, while still having a good amount of utility (again, what led me to choose the search-based interface). The conceptual model for this interface is a conceptual based on activities, more specifically, exploring and browsing. The interface invites the user to search, and allows them to "explore" and "browse" through the lost and found pets around them. While working on my prototype, I kept the design principles of affordances in the forms of buttons, and visibility since the interface was kept as simple as possible to not confuse the user.

Prototype 2: Lost / Found Pet

USABILITY REQUIREMENTS

When designing my prototype, I kept the usability requirements that we had laid out previously in the second part of our project. The specific usability goals that I catered to with my prototype were utility, efficiency, learnability and memorability. A usability requirement for utility is that at least 98% of the time, the user must be able to successfully upload a picture of a pet to the facial recognition database, without encountering any errors. A usability requirement for efficiency is that the user must be able to match a found pet, or register a found pet, within two minutes of opening the application. Additionally, for learnability, a user must be able to familiarize themselves with the lost and found

functionality within a minute of downloading the app and opening it for the first time, and only have to click 5 buttons in order to do so. Lastly, in order to achieve memorability, an infrequent user must be able to re-familiarize themselves with the features of the lost and found interface within 30 seconds of re-opening the application.

DESCRIPTION OF PROTOTYPE

My prototype describes a scenario in which a user has found a lost pet, and attempts to match this found pet with a lost pet by means of the facial recognition software that our application offers. This prototype is similar to the previous prototype in the sense that it demonstrates how a user interacts with the lost and found interface, but it is more specific because it shows how a particular aspect of the interface (facial recognition software) functions. My prototype starts from the “Found Pets” home screen, and continues through the process of using facial recognition to return this pet back to its rightful owner, including alternative scenarios that might occur.

Screen 1:

The first screen of my prototype shows the home screen of the Found Pets interface on the application. The pets that have been found recently in the user’s area are displayed, and if one of these is clicked on, it will bring the user to a screen with a description of the found pet, including its type, color, name and size. At the bottom of the interface, there is a button to upload a found pet to the facial recognition database in an attempt to match it with a pet that has been registered as lost. When this option is clicked, the user is brought to the second screen of my prototype, which is connected to the camera on the user’s phone.

Screen 2:

The purpose of this screen is to allow a user to take a picture of the found pet, which will then be uploaded to the facial recognition database. The user would take a picture of the pet, and if a match is found, would be brought to the third screen.

Screen 3:

The third screen clearly conveys the message that a match has been found, and lists a few key pieces of information about the lost pet, which are when it was lost, and how far away it was lost. At the bottom of the screen, there is a button for the user to find the pet’s owner, and a button to view the pet’s information, as registered by the pet’s owner. If a user presses the “Find Owner” button, they are brought to screen 4.

Screen 4:

Screen 4 is a simple description of the registered information of the owner, including the name, age, address and phone number. The purpose of this screen is to supply the user with information that could help track down the owner of the missing pet. At the bottom of the screen are two buttons. The first button allows the user to directly message the owner through the app, and the second button connects directly to the calling capability of the user’s phone in order to reach the owner. If the direct message option is chosen, the system continues to screen 5, which is the direct messaging interface.

Screen 5:

This interface is similar to an iPhone's iMessage interface. At the top of the screen, the name of the person who is being contacted is displayed, as well as information about the last time they used the application. There is a box at the bottom of the screen for the user to begin typing a message, and the user's messages are displayed on the right side of the screen, while the owner's messages are displayed on the left side of the screen. One option that the direct messaging interface offers is a "Find Me" message that can be sent by the owner of the pet. If this occurs, the user is brought to screen 6.

Screen 6:

Screen 6 is the map interface of the application. It connects directly to the "Maps" feature on the user's iPhone, assuming that access is granted by the user. The address of the owner is used as the destination, and the interface provides navigation until the user has arrived at his or her destination.

The last three screens of my prototype demonstrate an alternative scenario with regards to the facial recognition software being accessed by the application. After a photo of the found pet has been uploaded to the facial recognition database, the system will either alert the user that a match has been found, or hasn't been found. If a match is not found, the user is directed to screen 7.

Screen 7:

This screen promptly displays a message to the user stating that a match could not be found in the facial recognition database. There is a button towards the bottom of the screen which prompts the user to register the found pet to the application. When this button is pressed, the application transitions to screen 8.

Screen 8:

Screen 8 is the found pet registration interface of the application, and is relatively straightforward. There are fields for the name, type, and color of the found pet, as well as the location that the pet was found at. Beneath this is a text box for any additional comments that the user might have about the status of the found pet. The user is prompted to press the submit button at the bottom of the screen upon completion. When this information is submitted, it is uploaded to the "Found Pets" home interface, and will be displayed on other users' applications when opened.

Screen 9:

The user is then brought to screen 9, which informs the user that their submission has been received and that they will be notified if a match is found with a lost pet. At the bottom of the screen is a button to return to the home interface on the application, because the user's task has been completed.

DESIGN RATIONALE FOR PROTOTYPE

I chose to use an iPhone screen skeleton for my prototype because this was the most accurate portrayal of what our product will be, since it is an IOS application. The skeleton

looks professional, and allowed me to fill in exactly what I thought each interface for my prototype should look like. To make my prototype realistic, I tried to provide various buttons on most of my screens which would direct users to different functionalities within the lost and found interface of the application. I decided to use paper prototyping because it is easy to do and can be done quickly, while still providing the user with a very accurate depiction of what the final product is going to look like. Additionally, it is a relatively mistake-free process, because any time that I wanted to change what a screen looked like, or add a minor detail, I could easily erase what I had done and trace over it. When constructing my prototype, I tried to keep the user base in mind as much as possible. The purpose of this application is to reconnect lost pets with their owners, so at its core, it should be very learnable, easy to use, and simplistic. When constructing my paper prototype, I kept these things in mind, which is why my design is not too aesthetically pleasing, but is self-evident and intuitive to use. There are buttons clearly labeled, which allows the user to know exactly where they are going if they press a button. Because anyone can find a pet, or lose a pet, this design had to be tailored to a wide variety of users. This is another reason why my design is aimed to be simple and clear. A conceptual model for the system is a conceptual model based on activities, specifically exploring and browsing. Users need to be able to match a found pet with a lost pet, or vice versa, which requires browsing. A design principle from class that I focused on was mapping. A user can look at my design, and naturally deduce what each screen or icon is supposed to provide functionality for, without having to think about it.

Prototype 3: Registering a Pet

USABILITY REQUIREMENTS

The usability goals I adhered my prototype to were, specifically, utility, learnability, and memorability. While creating my prototype, these were highly important in coming to decisions about which features to add/place into the interface, which design objects (i.e. buttons) to repeat across screens, etc. In particular, from the requirements that my group concluded upon from Part 2 of the project, I chose to address the requirement of “pet registration taking no longer than four minutes”. When designing my paper prototype, I kept this usability requirement in mind - assessing how I could invoke good practices of usability goals like learnability and memorability into the interface - in order to keep users moving at a quick pace through screens and completing registration in our desired time limit. My thought was that, if users can easily remember, locate, and utilize functions of the system (toolbars, buttons, account features, etc.), then they will be able to execute user tasks like registering a pet or finding a lost pet in a very short amount of time.

DESCRIPTION OF PROTOTYPES

Screen 1:

This is my design for the home screen of the user interface. The bar at the bottom, similar to a bookmark or home tab, will later show consistency and assist towards memorability of specific tasks and interactions with the interface. The bar at the top shows the company

logo and has a fun, interactive affordance (button) that invites the user to click-and-register their pets with the system. The bulk of the rest of the homepage is post-based: including nearby lost/found postings. This is a similar feel to Facebook - providing an engaging conceptual model that invites user scrolling, swiping, clicking and exploring of the application, upon immediately entering the app.

Screen 2/3/4:

Screen 2 holds a simple yes/no interaction, guiding the user task appropriately within the system. If the user has an existing account, they will be directed to the Sign-In page (Screen 3) and then moved to access their previously registered pets (to the My Pets page). If they are new users to the system, they will be directed to a Sign-Up page (Screen 4) where they will be asked to enter very minimal information and continue into their My Pets page. Both paths will eventually lead users to the same My Pets page (Screen 5).

Screen 5:

This screen is the My Pets page, previously mentioned. This page shows potentially prior registered pet(s). It shows the facial recognition image of the pet and a few quick details about the pet (enough to distinguish this pet from potential others). There are a couple buttons on the page that invite user tasks: 1) edit button and 2) new button. Each button clearly portrays via text and relative location (i.e. the new button is in the shape of a new pet picture, signifying that clicking it would create a new pet in their My Pets page/account.

Screen 6:

This screen is the first of the series of Register a Pet pages. This screen would appear upon the user clicking “new pet”, on the previous page (Screen 5), as to create a new pet in the system, they must register that pet in the facial recognition system. This screen reflects the first step in the swift process of registering a pet on the interface. There are text boxes and buttons that allow the user to answer simple questions about their pet. This basic information will be stored in the system to later be cross-checked against local lost pets. The zip-code textbox will also include a geographic plug-in that will portray a dropdown menu of local cities, as most applications do. When the user is done entering information, they will click the next button at the bottom right of the screen.

Screen 7:

This screen is the second of three Register a Pet pages. The user will enter this screen upon completion of the previous page (Screen 6). Here, there is minimal text, but instead a camera plug-in that allows the user to take a picture of his/her pet. There will be grids that help align the facial structure of the dog/cat in order to properly function with our facial recognition system. Once the user has completed capturing the image of their pet, they will click submit at the bottom right of the screen.

Screen 8:

This screen is the last of the three Register a Pet pages. This page’s main purpose is to provide the user with feedback. It shows text with either “Congratulations” or “Please, try again!”. “Congratulations” signifies that the user has successfully entered all info required for registering a new pet in the system and has taken a functional picture that is compatible

with the pet facial recognition database. “Please, try again!” will signify that the user has either 1) failed to enter a piece of critical information requested on Screen 6 or 2) needs to retake a picture of their pet on Screen 7. If the user receives the notification “Congratulations”, he will also be prompted to register another pet, if he/she wishes to do so. This appears above the, now reappeared, home/bookmark bar at the bottom of the screen.

DESIGN RATIONALE

First and foremost, I utilized the strategy of paper prototyping to be able to create a low-fidelity design that focused on functionality of the application, rather than looks, sizing, creativity, etc. It is very easy for designers and engineers to get caught up in the “perfect” layout or building of an application when creating prototypes. To ensure that I was completely centered on creating an application that met our desired usability goals and requirements (learnability, memorability, and utility), I designed a prototype that was very simplistic, but holistic in including all screens for user task completion of Registering a Pet in the system. In relation to our user analysis, this model of the user interface applies to all users. The three sets of users - Pet Owners, Pet Finders, and Shelters - all must be able to complete the user task of Registering a Pet in the system. Because of this, I created a very simple and universal task completion path/design that allows a user of any user type to Sign-In or Sign-Up, and swiftly, in three easy steps, register a pet that is either 1) owned by the user and desired to be prevented from being lost or 2) found by the user and desired to be returned to its rightful owner. This is not only helpful for implementation of the application, but also for testing: we can allow any individual from any set of users, or all sets, to functionally test or evaluate our product, while quickly being able to make changes (as nothing from this paper design is set in stone!). Further, I utilized buttons and textboxes that are good *affordances* - inviting users to easily see (good/easy-on-the-eye *visibility*), click, and use them when progressing through the user task of Registering a Pet - accomplishing our usability goal of learnability. Additionally, I provided the user with *consistency*: a home/bookmark bar at the bottom of several screens that can easily help them navigate the core/critical pages of the interface, and remember how to reach those pages with just the click of a button - accomplishing our usability goal of memorability. Finally, the last design principle I utilized was *feedback*. I made it a point to provide the user with positive or negative feedback when they had run through every step of the user task. If they did not successfully complete a portion of the registration, I notified them upon entering the last stage of the registering process. This is important for the user to learn/remember how to complete principal tasks on the interface, while also being emotionally gratifying.

PROTOTYPE 4: REGISTERING A PET

USABILITY REQUIREMENTS

We made it one of our requirements to limit the time the registration takes to no more than four minutes when done by a person with moderate experience using computer technology. With this prototype, the steps are made simple and fast, using clear labels or

designs that the user is already used to such as the camera with the round button at the bottom to take the picture. The only thing that may take a substantial amount of time if this prototype is followed will be the facial recognition, which will be implemented and optimized without the user being able to see it. The steps that require an action from the user will be doable in less than a minute following this prototype, assuming that taking a picture of the pet does not bring any complications with it (pets not holding still etc.). This prototype demonstrates Effectiveness and Utility by offering tools that the user is already familiar with, such as camera and camera roll, and presenting unfamiliar features in a logical manner in order to make the goal of registering a pet achievable. Learnability and Memorability are also taken into account and supported by the simplicity of the commands and the minimalistic design of the app. Eliminating any feature or design element that is not necessary makes the application easier to navigate and remember.

DESCRIPTION OF THE PROTOTYPE

Screen 1:

Screen one shows the first view that appears once the "Register a Pet" function has been selected. It requests information about the owner and the pet from the user in a simple and straightforward manner. This part is especially important if the pet gets lost and is found by another user, because it will help that person contact the owner and ensure that the pet hasn't been confused with someone else's pet. After entering this information and submitting the form, the user will see Screen 2.

Screen 2:

Screen two reassures the user by confirming that the information has been added and prompts him or her to either take or upload a picture of the pet that is being registered. Again, the minimalistic design makes it easy to see the options for the next step and choose one without getting distracted and with close to no possibility for error. If the user selects the "Take A Picture!" option, they will be taken to Screen 3. If they decide to upload a picture from Camera Roll instead, Screen 4 will appear.

Screen 3:

Screen 3 accesses the phone's camera and displays what the camera sees to the user in a format that will likely look familiar. The round button at the bottom of the screen for taking pictures is consistent across many apps, and implementing the picture taking functionality in this way will make it easier for even inexperienced users to continue the process efficiently and not get sidetracked.

Screen 4:

Screen 4 accesses the phone's camera roll and, like Screen 3, shows the user a view that he or she is likely already relatively familiar with. It will let them choose a picture to upload. This is especially useful if the pet owner didn't register the pet until after it got lost or is unable to take a picture of the pet at that moment for any reason. Once they tap a picture it will be highlighted with a thickened border around the square. It can be deselected by tapping again or by simply choosing a different photo. After the best picture has been found, one can upload it by pressing the "Select" button at the bottom of the screen.

Screen 5:

Screen 5's only functionality is to inform the user that the system is working and processing the picture to be stored in the database. As a result, the user may be less

impatient or worried if the app doesn't seem to be doing anything for a few seconds of minutes after the photo has been uploaded. This screen does not require or allow any user action and will be automatically replaced by Screen 6 once the photo has been successfully processed and the features have been stored.

Screen 6:

Screen 6 lets the user know that the process has been completed successfully and their pet is now registered. Tapping "Finish" will make the application return to the home screen.

DESIGN RATIONALE

The main advantage of this design is that it keeps the app organized and simple. There are no excess design components or functionalities that could distract or even mislead the user. The color scheme consists of only two or three different colors that are each used for the same functionality throughout the app (orange signals buttons etc.) and the shapes are chosen to be rectangles throughout the prototype, with the exception of the round button for the camera.

This design accommodates for the variety of potential users by making the app easy to use regardless of level of experience. The minimalistic design and the color choices make the application more accessible to pet owners with impaired vision or motor-impaired users, as it is simple to navigate and there are no extra items to accidentally click or unnecessarily increase the difficulty of making out the shapes and text on the screen.

PROTOTYPE COMPARISONS

Since we did two prototypes for two separate tasks (total of 4), we decided to compare them separately as well. We did not think it would be appropriate to compare prototypes that were not testing the same functionality, since they are separate completely.

LOST/FOUND PET (PROTOTYPES 1 & 2)

The task that the lost and found prototypes were seeking to demonstrate was linking missing pets with their rightful owners. In order to do this, we employed two slightly different approaches to prototyping. Prototype 1 focused mainly on the searching capabilities of the interface, while prototype 2 explored the functionality of the facial recognition software. In prototype 1, there was only a search option from the first page. It takes a few clicks to get to the facial recognition, but this was done on purpose. This was to allow the possibility of this feature but only to those who felt comfortable with it. This prototype took advantage of the use of a map to give users a visual along with the list results from a search. Prototype 2, on the other hand, began with accessing the facial recognition feature of the application almost immediately, and then followed through with the process of matching a missing pet in this manner. These differences are very important because they are two completely different approaches to the same problem: finding a lost / found pet. When these prototypes are tested out, we would be able to observe which approach works best for a variety of users in unique scenarios. For example, if there are more casual users who are accessing the application in order to browse missing pets in the

area, we would steer towards prototype 1, while more urgent users who need to upload a photo of a found pet would potentially prefer prototype 2. From this feedback, we would pick which general approach for the main functionality works best (search or facial recognition) and then tie in aspects of each of the prototypes that users preferred most for the final product. These two prototypes took completely different approaches to the same problem, with different minor functionalities as well. This gives us a lot of data, which will allow us to produce the best interface for our diverse user base.

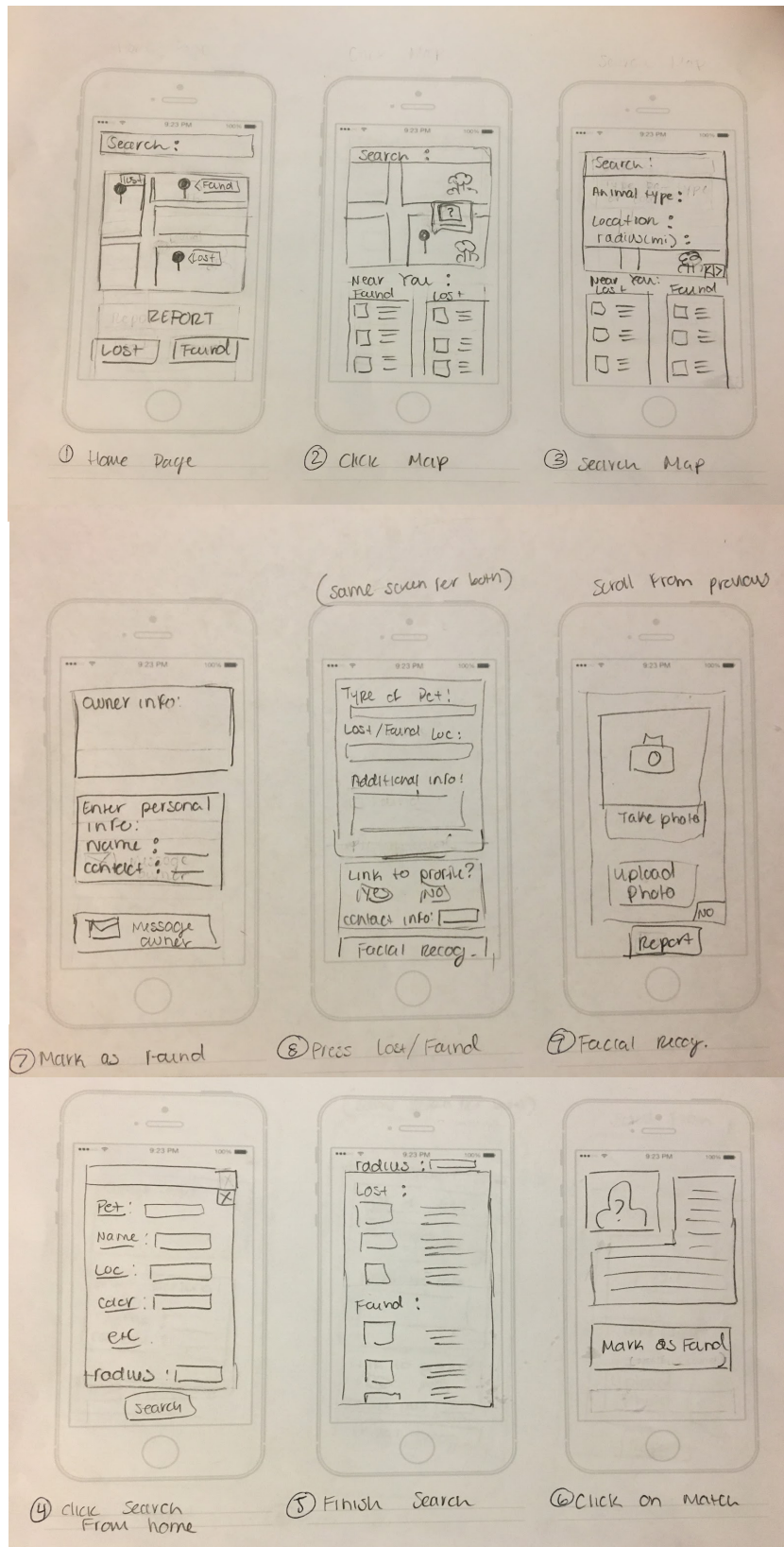
REGISTERING A PET (PROTOTYPES 3 & 4)

The first and largest difference between the two prototypes is that one utilizes the paper prototyping strategy and the other the wireframing strategy. It is beneficial to see both potential low-fidelity and high-fidelity prototypes, as it provides different stages of completion in the build of our application. It may be interesting to see how users interact with each (i.e. with different mannerisms). Prototype 3 focused more on a holistic path of the user from homepage to success page of registration, while Prototype 4 honed in and centered their design on the user task of registering a pet, at hand. While the focuses diverged, the basic steps for Registering a Pet were very similar. Both prototypes utilized a basic text description page and facial recognition page. Additionally, each prototype had their own style of feedback signifying that the user task had been successfully completed. Where the prototypes differed, again, was the specifics of the facial recognition page and its particular features. Prototype 4 included an extra plug-in that allowed the use of the user's own smartphone camera roll to upload a picture to the facial recognition system, while Prototype 3 did not have that functionality. These differences are important to note, as during evaluation stages, it will be crucial to the application's success to allow users to interact with different versions of user task paths, and provide feedback on which was more desirable. Will more features be pleasant to users in handling tasks with strong utility and effectiveness or will they be confused in other aspects, like visibility of the interface?

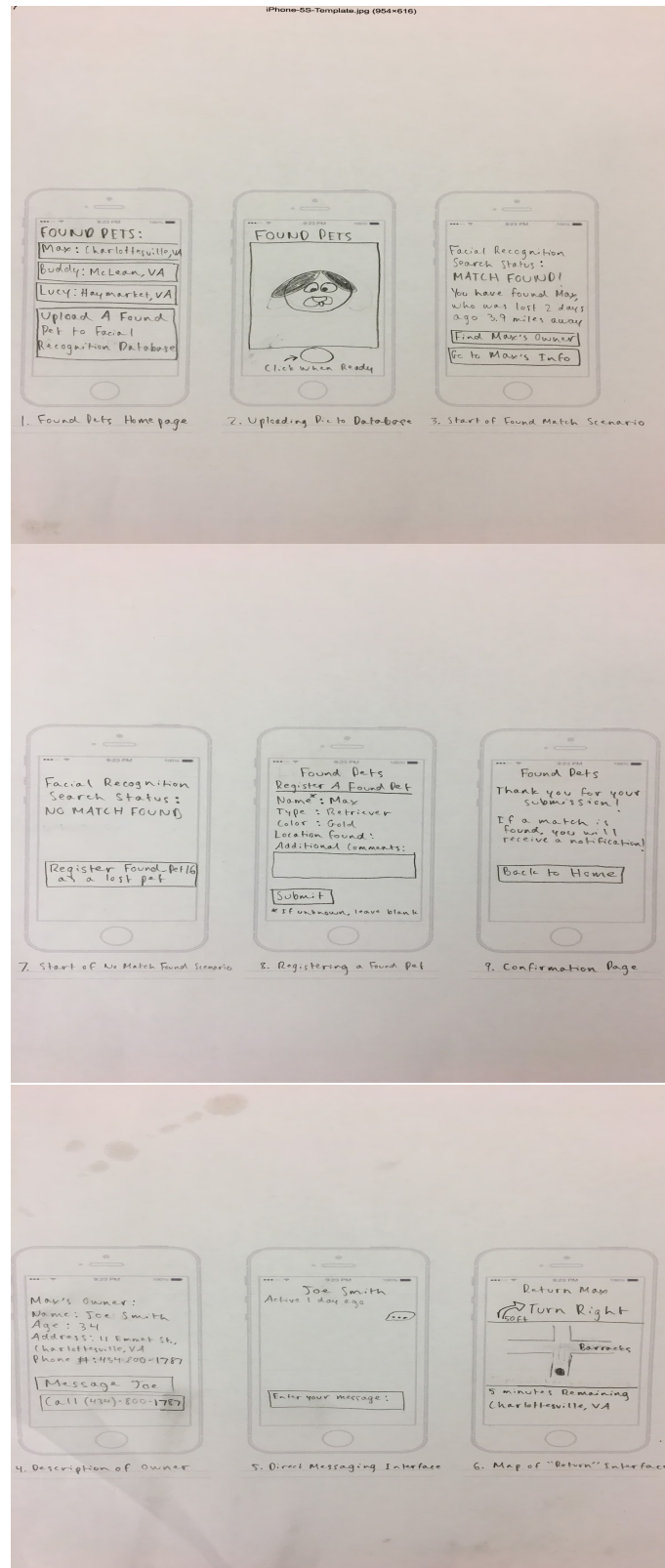
CONCLUSION

This part of the project mainly served as a way to analyze and test our ideas. Prototyping the two main functionalities in two different ways each brought to light the differences in our thought processes, as well as advantages and disadvantages of each approach. The different prototypes explore very different designs and goals, from making the apps as fun as possible to use to providing the user with a minimalistic application and therefore a simple experience. Having more than one prototype for each functionality also allowed us to explore how the two different processes would be embedded in the system as a whole, while keeping the level of detail we wanted for the process in question. The gain from this part of the project is an improved understanding of our system and an impression of the issues we might run into. It also provides us with insight into the user experience, which will be important to keep in mind as we continue on to the next phase of the project.

Prototype 1



Prototype 2



Prototype 3



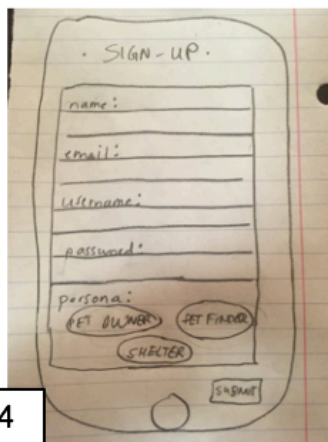
1



2



3



4



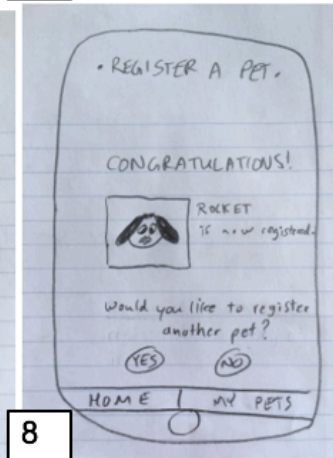
5



6

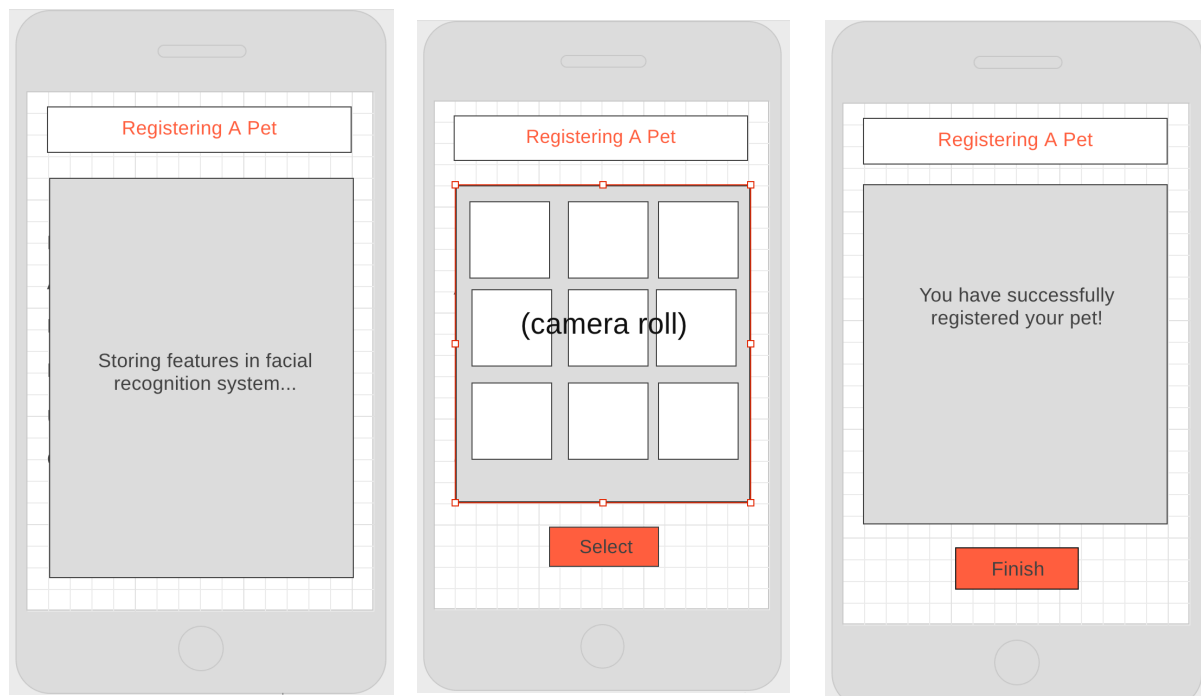
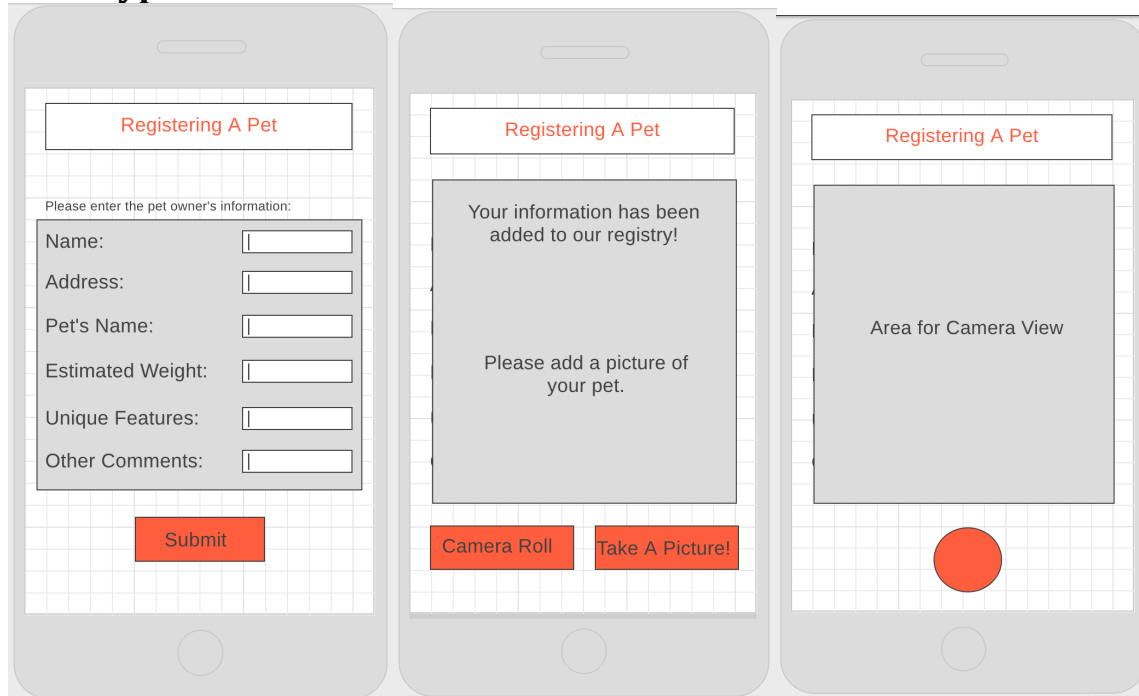


7



8

Prototype 4



CHAPTER 3

Evaluation

EVALUATION PLANNING

Our primary goal with our evaluation period was being able to come to conclusions regarding certain usability ambiguities across multiple prototypes. Most of our questions revolved around the usability questions of efficiency, visibility, and learnability. In other words, we desired to discover what potential users thought about their interactions with our system in terms of how quickly they could accomplish tasks, how familiar and easy to repeatedly use our interface was, and how clear the main UI was in displaying functionalities of the application. We decided to test these usability questions by asking users to complete basic user tasks - like finding/registering a lost pet - while using the “think-aloud protocol” practice. This ensured that we could best comprehend what users were really *thinking*, not just physically doing, about/with how easy to use and visually pleasing the application was.

On the other hand, our research questions were focused on which of the prototypes were better suited for the ideal user. We agreed upon using two of our four prototypes in the evaluation sessions - prototypes 1 and 2. This was meant to ensure that our research could be more comparable, as prototypes 1 and 2 both tested the user task of losing/finding a pet. Further, we didn’t want to use all four prototypes, due to the fact that most of our subjects were likely to be busy college students who didn’t have more than a few minutes to complete evaluation sessions! Overall, we desired to seek the prototype that the user was able to complete user tasks on the quickest and most effectively - without errors, confusion, or excessive clicks.

Some of our experimental design hypotheses were centered around the core usability questions mentioned earlier. Specifically, one of the most critical hypotheses we formulated was: both prototypes 1 and 2 will allow users to complete user tasks 1 and 2 (finding and reporting a lost/found pet) in less than three clicks. This hypothesis embodies all three usability questions in one: testing the efficiency of the system, how aesthetically pleasing and easy to navigate the main user interface is, and what usefulness the repetitive features of the interface offers the user. Some other hypotheses we created are: prototype 1 will take less time to complete task 2, prototype 2 will take less time to complete than task 1, users will favor the report lost/found buttons on prototype 1 when compared against the map and search, and first time users will make less than 1 error on prototypes 1 and 2 (either task). These hypotheses are easily testable on our prototypes, because with observers in evaluation sessions, it is very easy to track basic metrics that falsify or prove our hypotheses: the number of taps/clicks users make on the interface, time taken to execute tasks, and user feedback on prototypes.

To factor out bias, my team made sure that we were completing all user evaluation sessions in a controlled environment. Loud outside conversations, distractions, and leading questions were all restricted, due to our holding of sessions in quiet areas that promoted focus and concentration on the tasks at hand (i.e. Rice meeting rooms and apartments). Additionally, we took small measures to prevent bias. One step we took was hiding our

timer from participants/subjects to ensure that they did not get panicked and rush through tasks. The specific tasks will be outlined in the evaluation procedure section below.

DESCRIPTION OF PROTOTYPES

We did not make any changes to our prototypes for the user evaluation sessions, aside from only including Prototypes 1 and 2 for reporting a pet as found, and excluding the prototypes for registering a pet to the database before it is lost. The usability requirements each prototype aims to fulfill, the explanation of the general purpose for each prototype and a description of each individual screen are included in Chapter 2.

EVALUATION PROCEDURE

Going into our evaluation period, our group focused on keeping our experiments as reliable and valid as possible. In order to do this, we did our best to eliminate external variables that could potentially alter our results. Our experiments were conducted in quiet, well-lit, and comfortable areas, to ensure that every user could focus solely on our prototypes, without any distractions. We conducted our experiments in two separate sessions. In one of the sessions, we observed three separate subjects, and in the other, we observed two members from another group in our HCI class. The first three subjects that we observed were three roommates of one of our group members. They are all 20 years old, and two of them do not have pets, while one does. We evaluated them in their apartment, which helped provide internal validity, as they were all familiar with their surroundings to the same degree, and we eliminated any possible distractions. This evaluation session was held on April 18th, and lasted for a little under an hour.

In our second evaluation session, we observed two members of our HCI class who are both advanced users of technology. This session took place in a conference room in Rice Hall on April 19th. One of the subjects is 20 years old, and is a dog owner. Although she has never lost a dog, she represents a potential user of the application. The other subject is 21 years old, and is not a dog owner. Living in a highly populated area, he represents a user that might download the app to try to locate lost dogs in his area. This evaluation session took place on April 19th, and lasted about an hour and a half, since both groups were testing each other's prototypes. Each individual user experiment took about 7 minutes, including the follow up questions we asked after they had finished using the prototype.

For each of our experiments, we had the users complete two different tasks on our interface, using two different prototypes that had similar functionalities. Because we had a set of prototypes that focused on registering a pet on the application, and a set of prototypes that focused on reporting a lost or found pet, we decided to focus in on one set of prototypes in order to gain more detailed insight. Because reporting a lost or found pet is the main functionality that the application offers, we decided to focus on these prototypes. At the beginning of each evaluation session, our group read the subject a short paragraph explaining some background for the experiment. The paragraph is as follows:

"You will be testing an application that allows you to search for local lost and found pets. We will be asking you to perform a basic user task on the main interface of the application. You will be performing this task on 2 different designs of the app. The user task to complete: searching for a lost or found pet. We would like you to talk aloud as you complete the tasks - while clicking buttons, scrolling through the page, etc. - so that we can better understand what you want/expect from the application. We will be using this data to enhance and perfect our prototype for deployment, one day."

After providing some background for the user, we then proceeded to read them the first task card. Because our prototypes demonstrated the functionality of the lost and found interface of the application, the first task focused on using the prototype to report that a pet had been found. The exact text on the card is:

"You are walking down the street back from class and you see a brown dog on the side of the road. You know on your phone, you have the pet lost and found application. Open the application and see if the dog has been reported lost. It is not, post that it has been found."

The second task was similar to the first, but demonstrated an alternative scenario for our prototypes, which was that a match was found, and the user needed to contact the lost pet's owner to let them know about the situation. The task card reads:

"You find a lost pet on your way to the store, and you see a pet. You check your Pet Lost and Found Application and you search for the pet. It is registered. Message the owner of the pet that you have found their missing pet."

During our evaluation sessions, each of our team members had a distinct role, in order to avoid confusion or overlapping of tasks. Because Muhammad and Alex made the prototypes that were being tested by the users, they were the facilitators. This meant that when a prototype was being tested, whoever made it would act as the facilitator to help walk the user through the task progression, and the other person would act as an observer until their prototype was being tested. The subjects did not know that the facilitator was also the creator of the prototype, in order to avoid biased reactions. Muriel and Nirali were the observers during the evaluation sessions. This largely entailed taking copious notes about the user's interaction with the application, including everything that they said, the sequence of buttons that they pressed and pages they went to, and the time that it took to complete each task.

A lot of important information about our prototypes was revealed through the "Think Aloud" approach that the subjects took part in. When an aspect of the interface was confusing or frustrating, this was revealed through the way that the participants elaborated on what they were thinking at that moment in time. For example, a lot of users did not understand what to do from the home page of prototype 2's lost and found interface, and

this was discovered through participants commenting that they were unsure what to do with the prototype at this point.

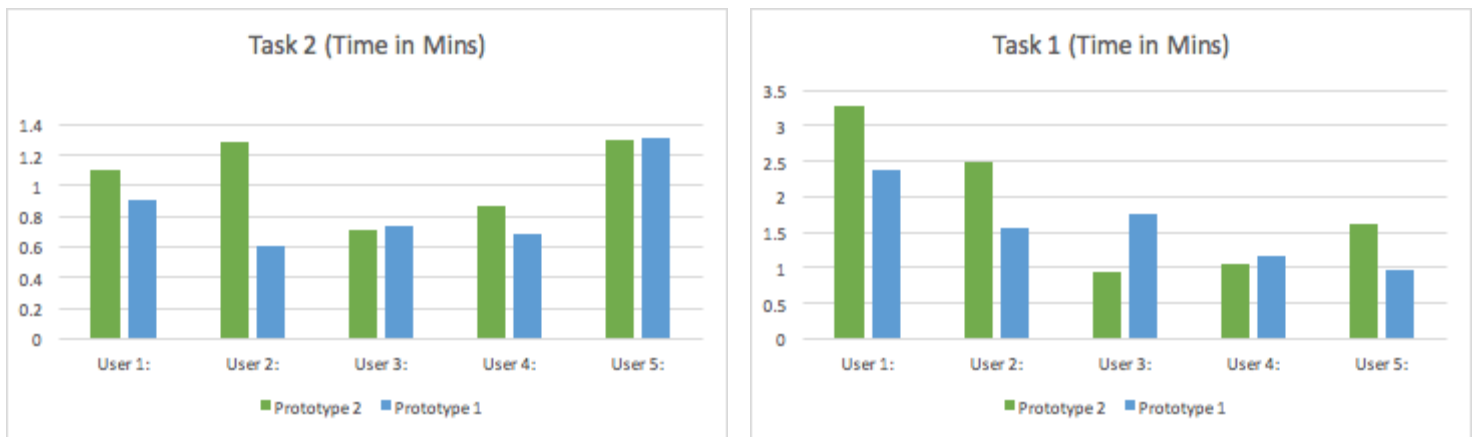
In order to gauge the subjects' overall sentiments about the prototypes, we asked some follow up questions after each evaluation session had ended. The first question that we asked was which prototype felt more natural or intuitive to use. There was a mixed bag of answers to this question, which provided insight into what is important for some users versus others. Some thought that prototype 1 was more natural because of its detailed home page, advanced search features, and lost and found map interface. Others thought that prototype 2 was more intuitive because of its relative simplicity, and clear instructions on screen after the home page had been left.

The second question we asked was what the users liked about each of the prototypes. This yielded similar themes that the first question brought to light, such as that prototype 1 allowed for a variety of ways to report a lost or found pet, prototype 2 allowed for a pet to be reported missing in only a few clicks, and both were pretty intuitive to figure out, or in other words, supported learnability well.

The third question we asked was what aspects of the prototypes were confusing or frustrating. Across the board, the subjects said that the home interface on prototype 1 had too much going on, and was confusing because it was unclear what to do in order to get started. Additionally, almost everyone agreed that the home interface on prototype 2 was confusing because of the button placement, and relative lack of direction.

The fourth and last question that we asked was what the users would like to see changed in each prototype. Unsurprisingly, a lot of the subjects wanted the home page to be altered, especially on prototype 2, to make the distinction more clear between pets that were listed as lost, and how to begin registering a found pet. Another change that participants wanted to see was an improved map feature on prototype 1, so that it was clear what each map was supposed to represent. Lastly, users wanted to see a better confirmation page, where they could enter their personal information should a match be found for a pet that was reported as found.

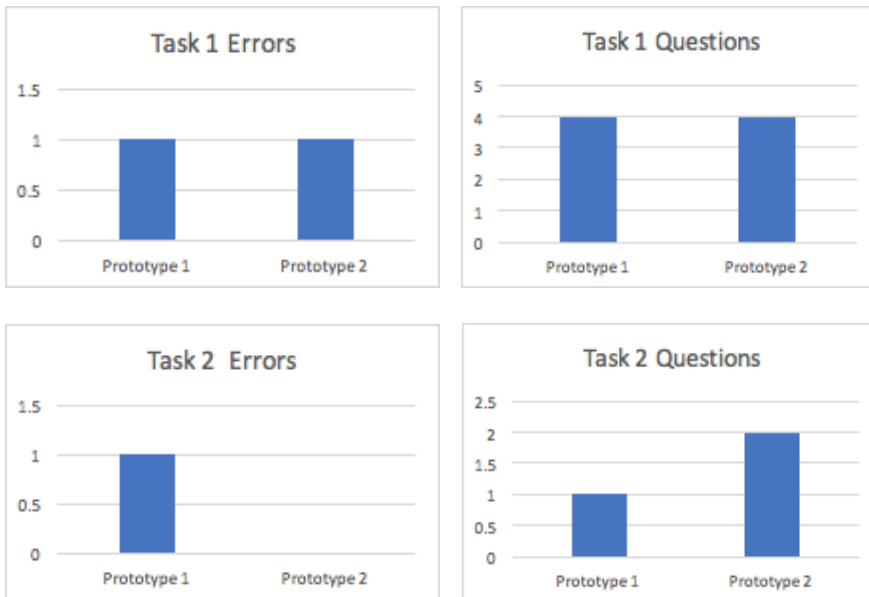
SUMMARY OF RESULTS



One of the usability questions we were trying to answer from these prototype evaluations was whether Prototype 1 or Prototype 2 was more efficient. For each task with each of the prototypes, we timed the amount of time it took the user to complete the task at hand. Above is the data that we collected. The left chart shows the times for each prototype according to user for Task 1, and the right chart shows the times for each prototype according to user for Task 2. As you can see, Prototype 1 outperformed, or performed at the same level as, prototype 2 every time except the case of User 3 completing Task 1.

It is important to note, however, that while timing our participants were allowed to stop and ask questions and clarifications. This could possibly have skewed our data as the timing, since we did not stop the timer for questions, if the user had any. The main reason we chose to gather the data this way is because we wanted to gather data about learnability as well and we thought this would be the best and most efficient way. To help mitigate this as much as possible, we averaged the number of clicks it took for each task for each prototype. For Prototype 1, there was an average of 5 clicks for task 1 and 4 clicks for task 2. For prototype 2, there was an average of 4 clicks for task 1 and 3 clicks for task 2. Even though prototype 1 took, on average, one more click for each task, it still outperformed prototype 2 when looking at time to complete the task.

Prototype 1, as described in Chapter 2, has three ways to complete the task of searching for a lost or found pet. Throughout all 5 users, only one user used the map functionality of the interface. The search bar functionality was pressed only once as well by a user, and that user went back to the home page after because they thought they were in the wrong place even though they could perform the task at hand from that screen. Prototype 2 did not have this issue, there was only one way to complete the task so all users completed the task in the same way.



Above are the total errors made and questions asked for each prototype from all 5 evaluation sessions. As you can see from the data, task 1 had more errors and questions than task 2, but this can be attributed to the fact that task 1 was the first time our users have seen and interacted with our interface. The real data from this the questions and errors of Prototype 1 compared to Prototype 2. For task 1, Prototype 1 and 2 had the same total errors and question asked across all 5 evaluation sessions. For task 2, prototype 1 had a single error with prototype 2 having none, however Prototype two had 2 questions and prototype 1 had a single question. This data shows that the two interfaces had no significant difference in number of errors or questions asked.

At the end of each evaluation, we asked each participant the same 4 questions. The first question (described in detail in the section Evaluation Procedure) asked the user which interface felt more natural to them. Out of 5, 3 preferred Prototype 1 and 2 preferred Prototype 2, giving us a split result of which interface our users would prefer to use.

Question 2 asked what the user felt frustrating about each prototype. The important results were as follows:

Prototype 1:

- The map seemed constricting and confusing
- The map wouldn't be accurate since animals move (a problem on our part, we did not explain thoroughly how the map worked)
- Search feature was confusing
- Scrolling was "unnecessary"

Prototype 2:

- Confusing register page
- No way to confirm if picture was taken

- The pets “around you” was confusing
- The pets “around you” had a radius was too big

Question 3 asked what the user liked about each prototype. The important results were as follows:

Prototype 1:

- Straightforward after starting
- Easy to contact the owner
- In depth searching
- Intuitive
- Links to found and lost pets automatically
- Map visualization

Prototype 2:

- Straightforward after starting
- Easy to contact the owner
- Easy to search for found pets
- Intuitive
- Links to found and lost pets automatically

Question 4 asked what, if anything, our user would change about each interface. The relevant responses are as follows:

- Make the home UI clear for both Prototypes
- Prototype 1 should have a search by area button, then the map interface after
- Prototype 2 should have a “near you” that shows pets around you (like prototype 1)
- Prototype 1 should make the map more clear, or move it somewhere else
- Both apps should notify users about lost pets in their area, similar to an amber alert
- Prototype 2 should change the words from “found pets” to “nearby pets”

This concludes the data that we collected that we felt was most necessary to complete the analysis of the two prototypes.

CONCLUSION

Part four of this project was very valuable because of the amount of information and feedback we were able to collect about our prototypes. Going into the prototype evaluation stage, we wrote some usability questions, which included which prototype was more efficient, in terms of both time and the number of clicks required for each task, which prototype had better visibility, and which prototype was more intuitive or natural to learn. From our observations and through the answers we obtained to our follow up questions, we were able to conclude that both prototypes were essentially equal in terms of efficiency.

Prototype 1 was slightly faster in terms of time, and prototype 2 was slightly more efficient in terms of the number of clicks required to complete a task. Both prototypes had their share of issues when it came to visibility. The layout of buttons on the homepage of prototype 2 was confusing to most users, and they often got stuck in their quest to report a found pet right off the bat due to this. Because there were so many different options on the home interface of prototype 1, there were a lot of complaints about visibility for prototype 1 as well. In general, the subjects said that prototype 2 was easier to learn because of its simple and direct functionality, but that prototype 1 had a lot of advantages for more familiar users because of the advanced search features it provided, and its unique map interface.

During our prototype evaluation, we were able to assess our original hypotheses. Our first hypothesis was that it would take an average of 3 clicks to complete tasks one and 2 on both of the prototypes. Our data showed that it took about 3 clicks on average to complete the tasks using prototype 2, and about 4 clicks on average using prototype 1. Our second hypothesis was that it would take less time to complete task 2 using prototype 2, compared to prototype 1. This hypothesis ended up being refuted during our testing, as prototype 1 actually took slightly less time on average to complete task 2. Our third hypothesis was that users would favor the report lost/found buttons on prototype 2 when compared against the map and search designs of prototype 1. Our evaluation showed that certain users agreed with this hypothesis, while others did not. Some felt that prototype 2 allowed for easy registration of a found pet because of its straightforward approach, while others liked the map and search functionality of prototype 1 because it seemed like a natural way to lay things out.

When looking at the data, a lot of things reveal themselves. The data shows that our users had things they really liked about each design and things that they really disliked about each design. In the summary of results section, Prototype 1 outperformed Prototype 2 in terms of speed every evaluation except one, even though it averaged one more click. Since efficiency was something we valued highly (nobody would use our app if it was cumbersome) we would continue with Prototype 1 and make changes that were desired. However, we would change the parts of the interface that Prototype 2 was preferred. Some certain scenarios one was favored. For example, searching for lost pets was easier on Prototype 2, while posting that a pet was found was easier on Prototype 1. For example, the search feature was not used so we would remove that from Prototype 1 to make it more clear. The home UI was a problem for both so we would only leave the report lost and found from Prototype 1, and a Near You button as well. The near you Button would have the map from Prototype 1, and the list of found and lost but presented in the way it was on the Prototype 2 home interface. The actual reporting of the pet as well would be changed. Users preferred the facial recognition software to the search based that Prototype 1 uses, so that would be the screen that would come up when a pet was reported lost or found, with a button to bring up the list-style reporting only if necessary (i.e. a picture could not be taken, etc.).

Our data gave us the information we set out to gather: which prototype is better at each step of the process. To continue, we use the data to combine the two interfaces to mitigate the weaknesses and improve its strengths, improving all of our usability goals.

CHAPTER 1 SOURCES

To gather our data, we primarily used the textbook and lecture notes. We used the texts as guidelines to what is expected for each section of the assignment. If we were unsure about what should be included in a section, or just wanted some more clarification, we turned to some outside sources which have been cited below.

"Animal Shelters." PETA. People for the Ethical Treatment of Animals, n.d. Web. 02 Mar. 2017.

"Face Recognition - Technology Overview." How Facial Recognition Works, Face Recognition, Facial Biometric Technology. Tomorrow's Technology, Today, n.d. Web. 02 Mar. 2017.

Lewis, Clayton, and John Rieman. "1. The Task-Centered Design Process." Task-Centered User Interface Design. N.p.: n.p., 1993. N. pag. Print.

Khella, Amir. "Knowledge and Mental Models in HCI." Theories in Computer Human Interaction. N.p., 1 Sept. 2002. Web. 02 Mar. 2017.