**Project Report - Cosc 341**

**Andrew Johnston - Hans Fuhrmann - Daulton Baird - Damyn Filipuzzi**

**Major Tasks & Changes From Paper Prototype:**

* Find Friends Page: This page provides the main functionality of the app; finding friends. You have the ability to view potential friend’s profiles and either “like” or “dislike”. If you like a profile, and they like you, they will appear in the messaging page. If you dislike a profile, they will not appear in messaging.

**Changes:** The main change that happened here is the ability to tap a profile to expand it. We realized this functionality was unnecessary as there is enough room on the page to show all the user’s details without expansion.

* Connections Page: This page provides a menu of all potential friends that have both liked you, and you have liked. This menu provides the option to open the messaging screen for each potential friend.

**Changes:** Two changes were made on this page. Firstly, we removed the search bar. We were not taught in creating dynamic elements, thus the time commitment to make this work was not feasible. Secondly, we removed the picture thumbnail from each user’s profile for similar reasons.

* My Profile Page: This page shows off how your page will look to potential friends when they enter the Find Friends page. This page also gives you the option to change non-vital parts of your information such as hometown and bio.

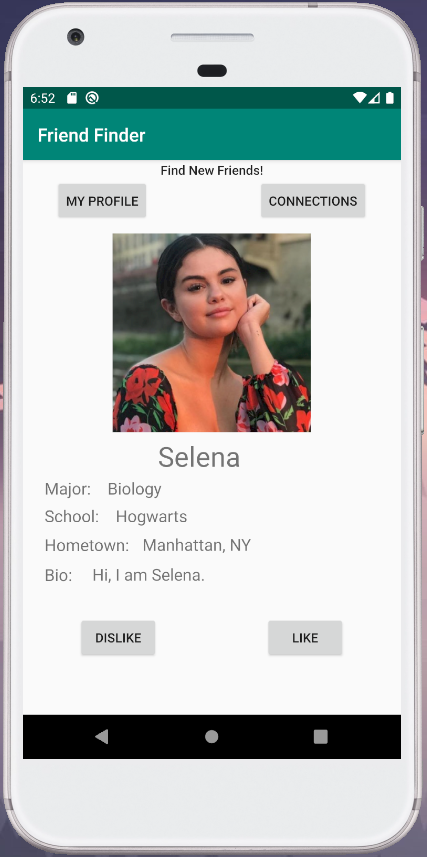
**Changes:** One very minor change here is that instead of a new menu appearing when selecting the “edit info” button, the current info simply becomes modifiable. This worked better aesthetically and functionally for our goals.

* Messaging Page: This page is the chat interface for whichever user you have selected to chat with. It shows past messages, and offers a text entry for new messages.

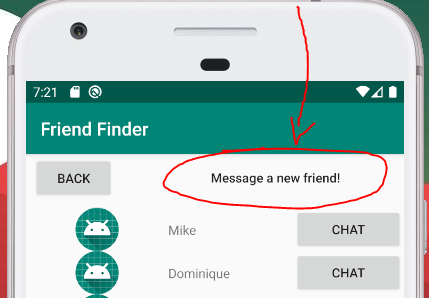
**Changes:** Similar to the connections page, the picture thumbnails were removed for simplicity and time constraints.

**Design Principles**

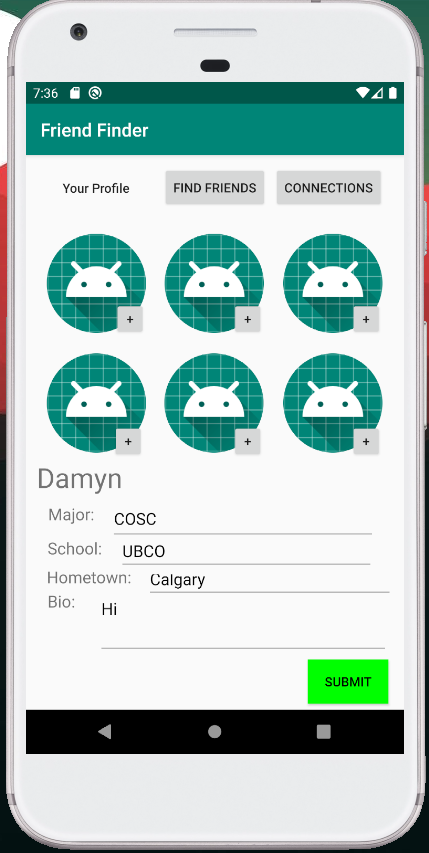
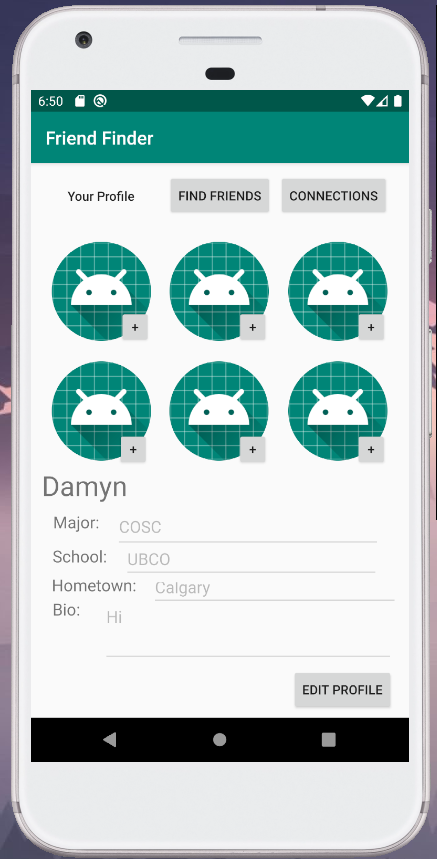
**Visibility:** The visibility principle is the basic principle which describes how the visibility of an element will play an importance on how often the element is used. Below is a picture from the matching activity. In the activity users can decide whether or not they would like to match with the user. This is done by either selecting the “Like” or “Dislike” buttons. These buttons stand out in the scene making it easy to distinguish for the user.

****

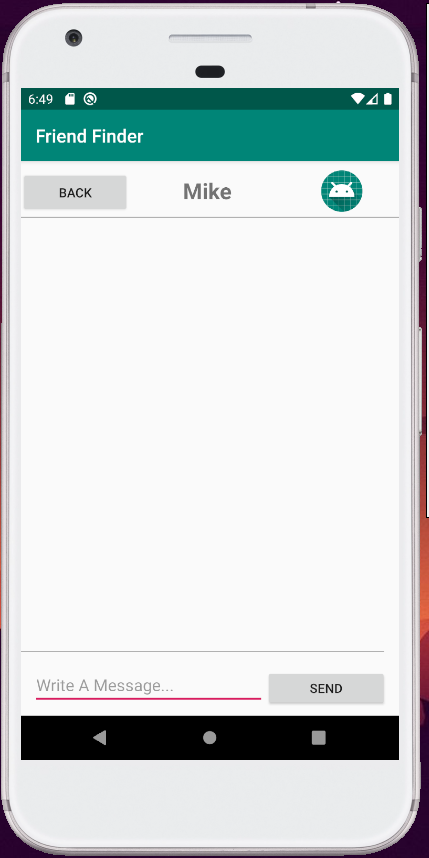
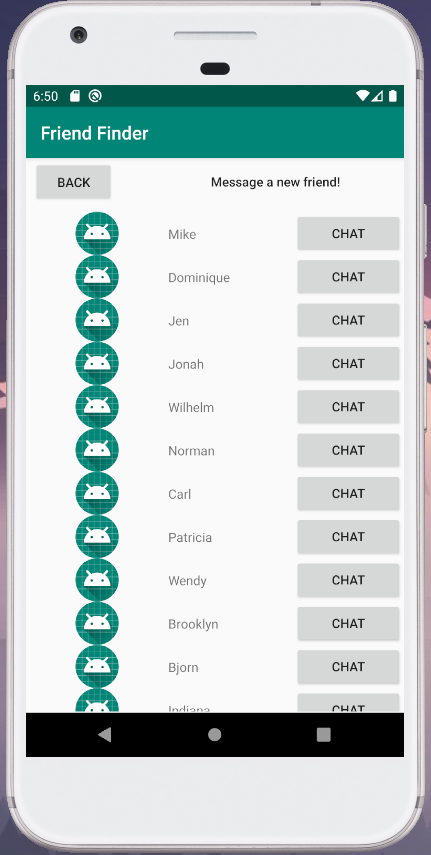
**Feedback:** Since there are no loading times for this application there does not require much in terms of feedback, however for each activity there is a prompt to inform the user on what they can do in the given activity. For example the below picture below.



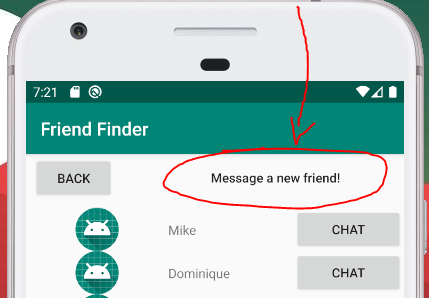
**Constraints:** The picture below shows how a user can only change their info once the edit profile button is active.



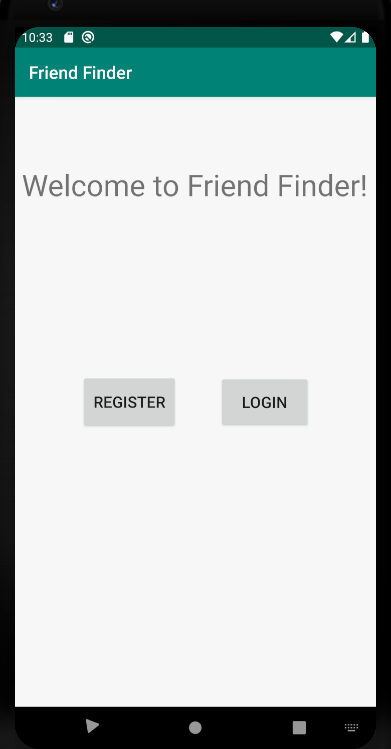
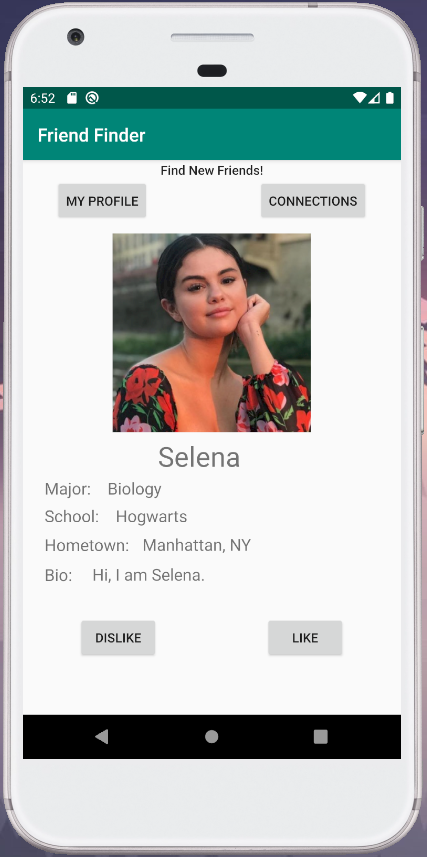
**Consistency:** The two pictures below show how consistent practices are followed with regards to the back button. This principle is followed throughout the entire application. In the two pictures the back button is always in the upper left hand corner of the screen.



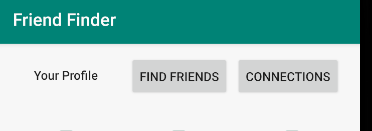
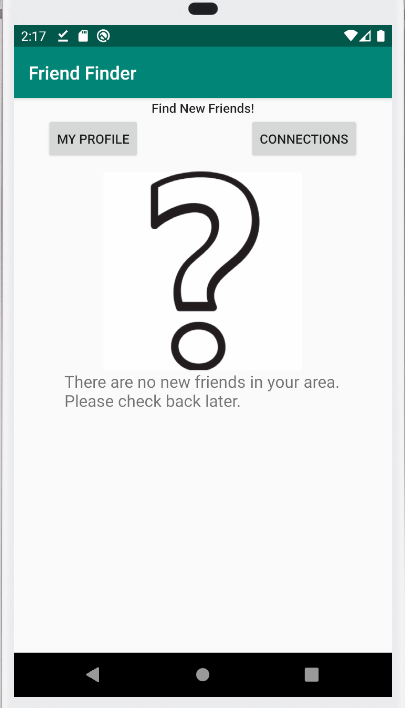
**Affordance:** The principle of affordance describes how the application should be used. This is done by prompting the user at the top of the activity with information describing what to do on that specific activity.

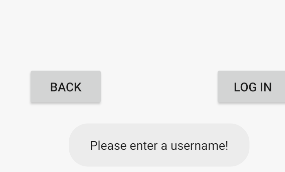
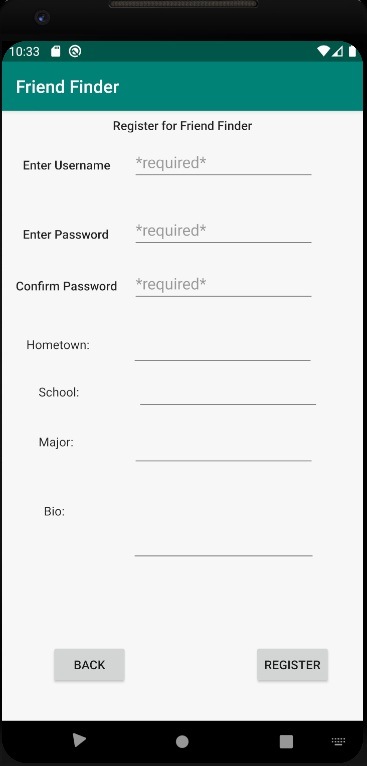
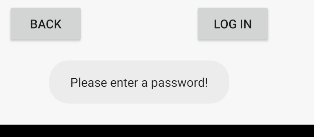
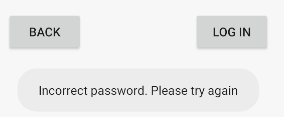


**Simplicity:** The UI on all of our pages is very simple. Buttons are easy to spot, and the labels are simple so the user understands where they go. The initial page only has two buttons (register or login) and a text box welcoming the user. These decisions were all for simplicity. The following screenshots are good visuals of how simple we made our activities.

****

**Matching:** The first screenshot shows how we speak the user’s language. When there are no more friends to match with in your area at the current time, the user is notified. In the second screenshot, we utilize meaningful abbreviations to direct the user. Instead of “talk to people you have matched with” we title the button “connections” for simplicity and user understanding.

****

**Help:** In the first screenshot is our registration page which outlines the required fields to help users and prevent errors. The other screenshots are some of the error messages when attempting to log in to an account. For example if a user enters their username correctly but their password incorrectly, the system notifies the user of this specific error.. 

**Problems Identified by Users, Heuristic Violations, Severity, and User Recommendations - Sorted by Severity Level**

1. **Error/Recommendation:** When registering for an account, you can’t tell which inputs are required and which are not. There should be some kind of indication for the information that is required, such as saying what is optional and what isn’t. It’s hard for a user to know without prior experience.

Violation/Severity: This error violates the “Help and Documentation” heuristic. We rank this as a 1 in severity as it is good for a user to know what information is required but at the same time the only information required is a username and password which is well known to users of various knowledge levels. Also, there is the ability to edit info so any mistakes can be corrected.

1. **Error/Recommendation:** When signing in with incorrect information, the error displayed is “user not found”. This error displays for any errors. If the password is wrong, if no information is entered, anything. The error message should change depending on the error so the user knows. Similarly when registering there is no indication the registration was successful. Messages such as “incorrect password” or “username not registered in system” would help.

**Violation/Severity:** This error violates the “Help users recognize, diagnose, and recover from errors” heuristic. We rank this as a 2 in severity as it is important for the user to know what is wrong when there is an error logging in so they can fix this error efficiently and effectively.

1. **Error/Recommendation:** When moving from page to page, you can’t really tell what page you are currently on. There is no indication that you are on the “my profile” page when you are on it other than the content. There should be clear indication what page you are currently on.

**Error/Recommendation:** When you edit your info on the my profile page there is no indication anything changed other than a very small color difference and the button text changes. There should be more obvious display changes so the user knows they are editing information.

Violation/Severity: These both violate “Visibility of system status” heuristics. We ranked these as a 3 in severity as it is important for a user to know which page they are on and it is equally important for a user to know how to edit their info in case they make a mistake.

**Change made based on heuristic evaluation:**

* Modified the error message so if a username doesn’t match it displays a message stating this, or if a password is wrong it says this, or if no information is entered it says this as well
* Registration page indicates to user if registration was successful
* Each page has a small title indicator so the users know what page they are on
* The edit info button on the “my profile” page indicates it has been pressed by changing color
* Required fields are bold and have a “required” tag

**Changes based on Professor’s comments during presentation:**

* Messaging Activity: Stores chat history for messages with other users in a text file. Pulls chat history when reopening messages with a user
* MyProfile Activity: Allows users to select images from drawables for testing purposes. We tried to fix this but it was more complicated than expected, so we left it to allow the user to select images from Camera and Camera Roll as that is more realistic.

**Video Link:**

<https://www.youtube.com/watch?v=3zb6JebTU_k>