

Mikhaila Friske's Portfolio

E-App ID: IU002272039
Email: frisk028@umn.edu
Phone: 608-293-0937

I certify that the work included in this portfolio is my own original work. Work included which was conducted as a part of a team or other group is indicated and attributed as such- the other team members are named and a true description of my role in the project is included.



Mikhaila Friske

12/31/2017

Personal Projects.....	2
Student Unions and Activities Web Projects.....	2
Information Desk Redesign.....	2
MCSE Redesign.....	2
OCL Google Calendar Design	2
History Page Redesign	2
ProDUCT Lab Research.....	4
Senior Project (CSCI 4950) Inventory App	5
User Interface Design (CSCI 5115) Semester Work	6

Personal Projects

A list of a few of my personal projects can be found on <http://mikhailafiske.com/projects> as well on on my Github page (<https://github.com/mcfiske/>).

Student Unions and Activities Web Projects

Information Desk Redesign

This design was to create easier access and search of the resources offered at the Student Union and Activities Information Desks. The images for each panel were designed by a student designer within the Student Unions and Activities Marketing space. The production and deployment to the website were done by myself. The design can be found on the following page: <http://sua.umn.edu/info>

MCSE Redesign

Execution of the MCSE redesign was done by myself, employing new ideas using CSS Grid into the design of the footer for better mobile responsiveness. The images for the headers were created by a student designer within the Student Unions and Activities Marketing space. The design can be found on the following page: <http://mcse.umn.edu/>

OCL Google Calendar Design

The entire backend design on the application for adding the calendar, the design, and the execution were done by myself. The calendar was created using Django Wagtail, Jinja Templating, and CSS styles (most significant being Flexbox and Grid). The design includes a responsive design into tiles of events for mobile. The design can be found on the homepage of the ocl website: <http://ocl.umn.edu/>

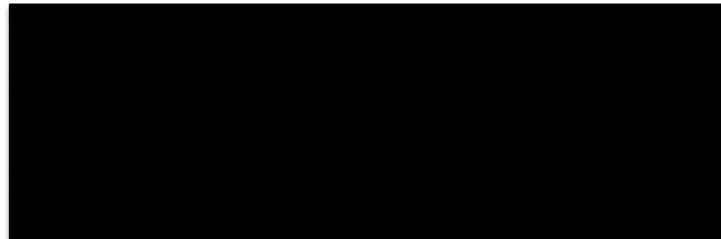
History Page Redesign

This is a current project in production at the moment. The project has been designed and being deployed by myself with the guidance of those who request the content to be displayed and accessible on the Student Unions and Activities Website. On the following page are a few major designs; details of the content have been hidden as per NDA for non-production ready material.

History of Student Unions on Campus

[Coffman Memorial Union](#)[West Bank Skyway](#)[St. Paul Student Center](#)[Village Union](#)[Farm Union](#)[Nicholson Hall](#)[Shevlin Hall](#)

West Bank Skyway

**1962 Union for the West Bank**

History of Student Unions on Campus

[Coffman Memorial Union](#)[West Bank Skyway](#)[St. Paul Student Center](#)[Village Union](#)[Farm Union](#)[Nicholson Hall](#)[Shevlin Hall](#)

Coffman Memorial Union

Lotus Delta Coffman

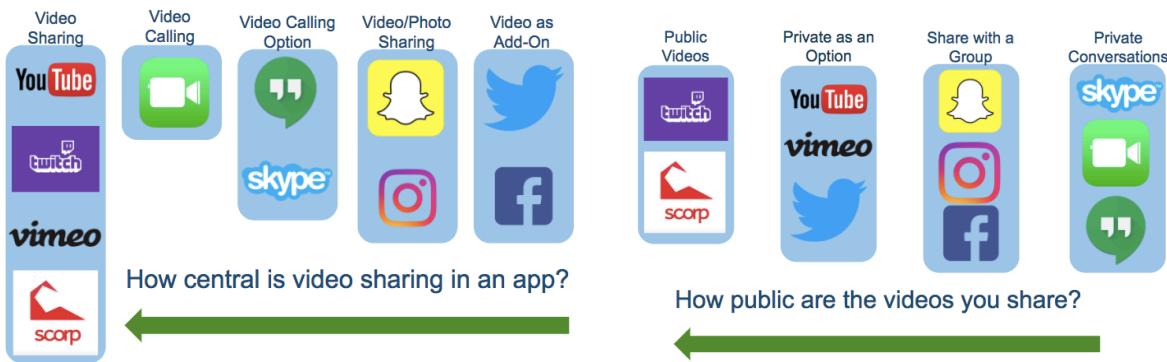
1940s A Place to Shape Development**1950s A Common Space**

Timeline:
1940
1950
1960
1965
1970
1980

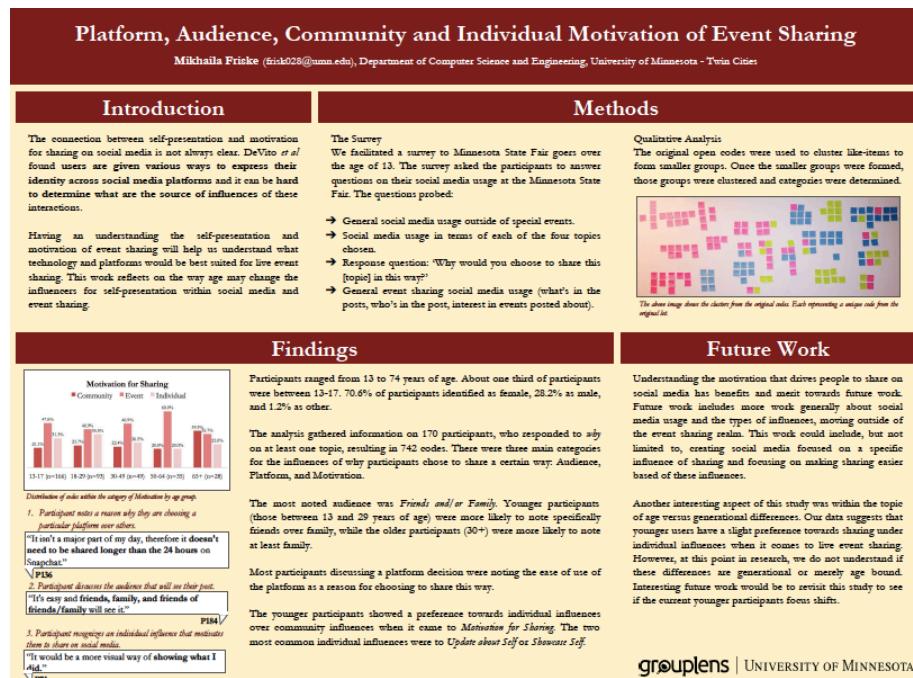
ProDUCT Lab Research

The ProDUCT Lab is a social computing lab with GroupLens at the University of Minnesota. Below are a few of the major items from my time working with a graduate student on their participatory design and video sharing work as well as event sharing. The research for each of these projects is ongoing.

Below are a few short taxonomies of video sharing applications for summer participatory design workshop presentation; work was evenly completed with another undergraduate student within the lab:



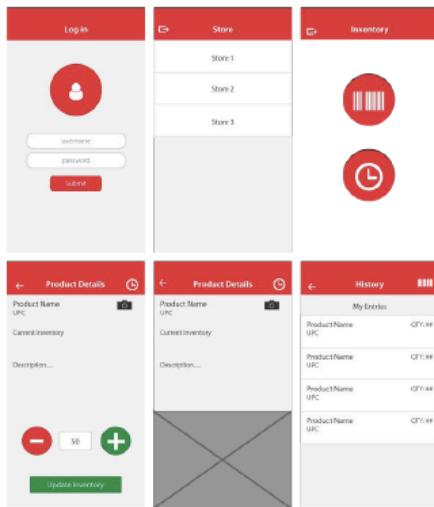
Below is the poster for my submission to CHI 2018 Student Research about event sharing motivation and the affinity diagram from the open coding of the survey. The survey used in the research was distributed to the Minnesota State Fair in the summer of 2017.



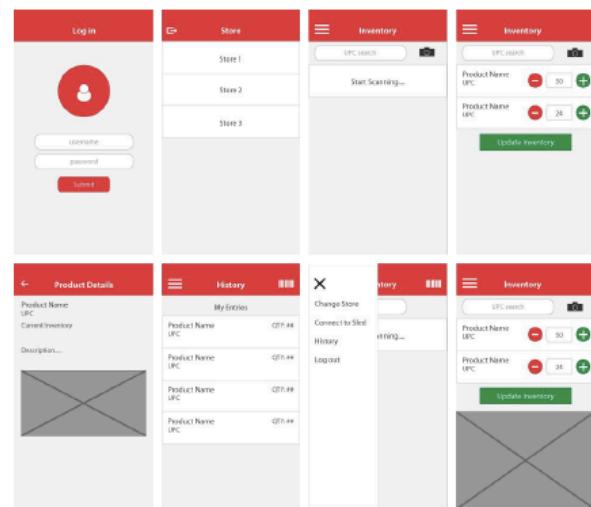
Senior Project (CSCI 4950) Inventory App

The following is some information on the Inventory App created with a small team of Computer Science students during the school year of 2016-2017. The full details of the project and work can unfortunately not be included due to an NDA agreement. However, below are the initial draft designs and a diagram of work completed by each student pulled from the final presentation.

Draft Design 1



Draft Design 2



Roles

Mikhaila

- Wireframes
- iOS App
 - Navigation
 - History

Tianhao

- Server, data model
- iOS App
 - Authentication
 - Network connection

Everyone

- Requirements
- SRS
- Use Cases
- Deployment

Sheldon

- Wireframes
- Product lookup
- Scanner

Andi

- iOS App
 - Database
 - Scanner
 - History

User Interface Design (CSCI 5115) Semester Work

The following is the final write-up completed in the course User Interface Design. The work was completed as a group. My personal major contributions consisted are listed below. If another teammate also helped along with these, it is noted as well.

Project Contributions

- Initial ideas and brainstorming (worked along with team).
- Creating wireframes from team ideas.
- Creating the survey to gain user information
- Front-end design of the prototype app. Can be seen in this YouTube video:
https://www.youtube.com/watch?v=9_gMIkgUJkk
- Organization of team, logs, and presentations

Write-Up Contributions

- Summary
- Overview of User Testing (Contributions from Ben Vanderkin)
- Giving Feedback on Given Awk-Rating
- Reflection (Contributions from Michael Rutz)
- Design Log

The report begins on the following page.

Ben&Ms Awk: Final Report

Group 24

Mikhaila Friske; frisk028@umn.edu

Max Marchionda; [REDACTED]

Michael Rutz; [REDACTED]

Ben Vanderkin; [REDACTED]

Mina Yacoup; [REDACTED]

Summary

The app we decided to develop is Awk. This app allows users to determine if a movie is going to have scenes that are going to be awkward to watch with their family or friends. The purpose is to help people not be scarred for life after watching an awkward scene with someone they would not want to see that with. Expanding on the user population, it could also be an app used by parents to determine if the movie is something they want their children to see.

Throughout the process we made many changes to our user interface based off our intuition and knowledge of the users through research, cognitive walkthroughs, heuristic evaluations and peer feedback throughout the semester. The major changes focused on simplifying our design and the length of the tasks within the app. Our goals were to create an intuitive app that was easy for new users to pick up and learn as they used.

Moving forward, we have devised design changes based on user testing as well as necessity in maintaining a long-term application. In regards to the user testing these changes include creating a better natural mapping for Awk ratings, clarifying the purpose of customization settings, and allowing the user to give more input on a movie's rating (as opposed to just agreeing/disagreeing). In addition to these, some miscellaneous feature additions include adding advanced search functionality, such as filtering, to allow for more user flexibility. We would also like to hook the back-end up to a third party movie database such as TMDB or Guidebox in order to get more movie information in general but also to diversify the sources from which base Awk ratings are derived.

Reflecting on all of the reports and the creation of the app, we have learned some key lessons on what makes up a good interface and how to effectively test it. Some of the hardest concepts to grasp are often the most important, such as allowing for change. Even if the majority of the group members believe an idea to be the best, sometimes that one person asking for change is right, so it is important that everyone is open to new ideas throughout the whole process. We also learned how important user testing is for this process to be a success. Even though our user testing was rushed and with friends, we were still able to gather a bunch of ideas on what else should be implemented.

This document will expand on the ideas above. The following also includes a link to a video of the current prototype that was implemented using PhoneGap, Bootstrap and NodeJS. Within the [Appendix](#) will be the raw user testing plan, user testing notes, post-testing brainstorm notes, design log and the final paper prototype design used for implementation.

App Documentation

The app we decided to create is a movie search app that provides information on how awkward a movie would be to watch with your family or friends. Our motivation for this app was in order to keep our users from being scarred from watching awkward parts of movies with other people. The main functionality of our app consists of three tasks. Finding a movie, adjusting the awkwardness level, and giving feedback on the awkward rating.

A video on the app can be found by following this [link](#) or by typing the following url into your browser: https://www.youtube.com/watch?v=9_gMIkgUJkk.

User Testing Report

Overview

Our general process started off by the proctor giving the general instructions on what our app is about, what the story behind the task is, then asking the user to perform the task. After the user finished the task, we would ask a few quick questions, then move on to the next task. All of our users were able to successfully perform each task with only small bumps along the way. The note taker would mainly just sit and watch what the user was doing on the phone given. Along the way, the notetaker would jot down anything they said that was useful, or something they were doing that was interesting. Please note that the full user testing plan may be found in the [Appendix](#).

Through our user testing we confirmed a lot of our design decisions, especially the one to remove our profile system. However, the user testing also led us to find out what we still forgot to implement from our current design after the heuristic evaluations. It also gave us some ideas on what to change.

There were two major downfalls from our user testing. The first is that we were only able to have the proctor and only one note taker (instead of the two we originally had planned for). The reason for this was that scheduling was difficult at the end of the semester because many people had other commitments they needed to take care of. Another downfall is that we were using our friends. Although we did receive some nice results, there is a high chance that some of the feedback given was filtered. Ideally we would have liked to have tested our app on even some strangers, given their consent of course. But similar to the reason for only one notetaker, the scheduling would have been difficult.

Tasks

Determining if a movie will be awkward with family

This task also included initial app customization. The users had a few issues during this stage.

One of our users had trouble with using the sliders, as he felt they weren't very responsive.

Another user did not understand if the ratings she was setting were for her, her family, or both.

The next stage of this task was to search for the movie. Our users were able to successfully find the movie *Deadpool*, and liked how the search was similar to other search engines. After this, the users were brought to the movie details screen. All of the users were unsure of whether or not a high awkward rating meant if the movie would be awkward or not. One of the users was unsure of where the numbers were coming from. The users seemed to like the overall design, and one noted that he liked having a picture of the movie on the screen. Some of our users did not understand that the category ratings could be expanded, but others discovered it without a problem.

Changing customization settings

In this task, the user had to change their customization settings to reflect a different view of “awkward” when watching a movie with their friends. Our user had no problem finding the customization screen, as he instantly pressed the profile icon in the upper left. His only issue was that when he finished adjusting his sliders, he expected a submit or ok button. He pressed the X button, and needed to go back to the customization screen to confirm that he successfully changed the settings. He liked how the process of finding the customization screen was intuitive.

Giving feedback on the given Awk-Rating

In this task, the user had to give feedback on whether or not they agreed with the awkward rating. This task was not completed during the first round of user testing because we realized that we forgot to implement the functionality. However, it was performed the second time around. The testers seemed to be a little confused about the label “Agree?”. They did not realize that it was asking the a question and thought that you would only click that when you agreed. But they tested it out, slightly unsure, and then got the prompt with the actual question. The testers wished that there would be more feedback after answering the question, such as a thank you or given the option to give a new rating.

Reflection

After performing the user testing, we were quite short on time, so we were only able to implement small changes. The small changes, based on things we had forgotten to implement, were actually added before the second and third users were tested. These include adding labels for the slide bars so they know which direction to go and adding the ability for them to complete the third task. Overall, the user testing was a great way to find out our major issues in a short

period of time. This really demonstrates how important it is to always get an external opinion on creative projects.

After user testing we brainstormed the major changes that would need to eventually need to be implemented. Most of the changes we discussed were based on having our app give more feedback to the user. These include giving the user confirmation that the user awkwardness preferences have been changed and letting the user know that their opinion has been taken into account when they give feedback. Some of the other changes were more based on clarifying and ease of use. These include improving the design of the drop-down categories so users know they can click on them and making it clear to the user that the Awkward-Ratings are out of 10. The raw notes of our brainstorm session can be found in the [Appendix](#).

Design Log

Overview

Throughout the process we made several changes to the design. The following shows some of the major changes made throughout the app development process. We started with two various sketches compiled from the rough sketches of the group members. From those two sketches, we combined into the initial design. From there, we made major changes after the cognitive walkthrough and then once again after the heuristic evaluations.

The initial design combined the two sketches with the major design components that we felt enhanced the functionality. This included having sliding bars to set up the customized awkwardness level, bottom feature bar, the ability to filter through the search result screen, and having an awkwardness rating and message like OK or Not-OK.

The cognitive walkthrough shed light on some major issues within our app. The biggest change that was made after the cognitive walkthrough was removing profiles from the functionality. This was done because our other tasks had shorter scenarios and the flow was more intuitive to complete the task. With profiles, a new user could get easily confused on what click on to complete the task or even where they were trying to get to.

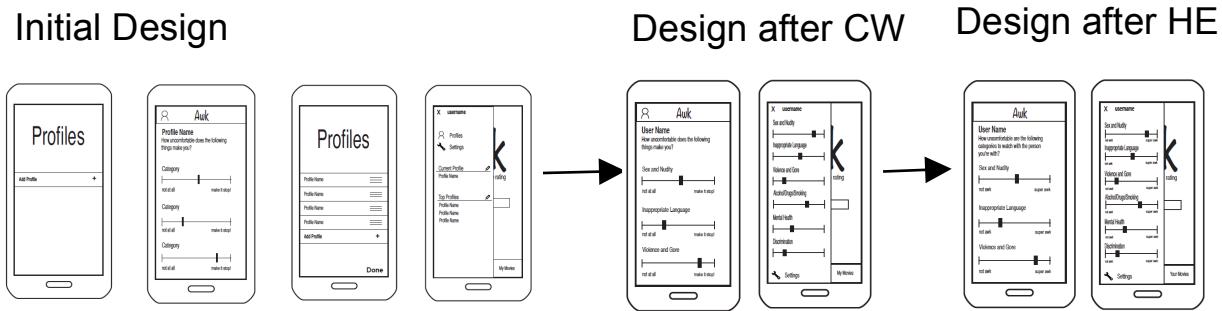
During the heuristic evaluation, we found areas within our app where the design could be improved to make things clearer for the user. Many of these changes had to do with the heuristic evaluation Recognition instead of Recall. After the heuristic evaluation, we realized our design was missing a lot of user feedback that would allow them to realize what they had just completed.

A full list of the changes can be found in the Design Log Table and the final paper prototype before implementation can be found within the [Appendix](#). However, below we are going to

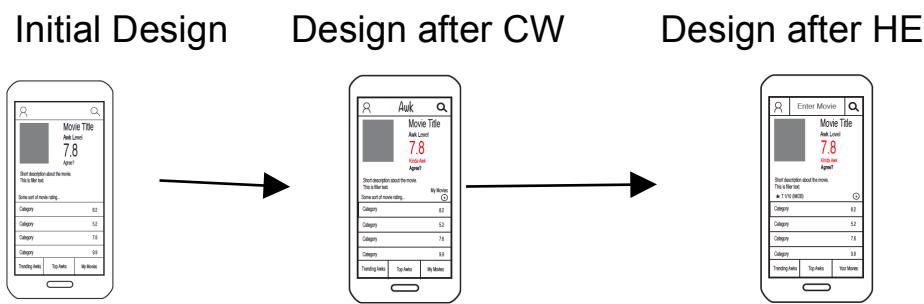
highlight some of the changes mentioned above through our initial design and each stage of the paper prototype.

Major Changes

The following figure shows the changes made to the way the user customizes the user-specific awkwardness levels. These levels are used to determine the weight of the awkwardness ratings of each category which affect the overall Awkward Rating that is given to the user to determine if the movie is awkward or not.



The next figure shows some changes that were done to the movie detail page. The first one was adding color to the Awkward-Rating. The deeper red, the more awkward it is. If it were green or yellow that means it is not as awkward. We also added a message underneath to help show that the higher numbers were the more awkward. In addition, we realized that we forgot a way for the user to add a movie into Your Movies. Another change that was added later is changing My Movies to Your Movies.



What's Next

Changes prompted by user testing

During the user testing phase, it was clear that many users did not see the rating system as being very intuitive. The rating scale (low numbers representing low awkwardness and high numbers representing high ones) seemed, to users, to contradict the more common scaling standards (low numbers representing lower quantity / quality). We believe this is because of the negative

connotation associated with “awkwardness” which would generally associate it with this lower quality. In future development, we will consider changing the rating system to better reflect this natural mapping.

In addition to this, users found the customization feature to be unintuitive as it was not clear who the customization was for. Many believed it to be for themselves rather than for a significant other / family member / friend / etc. We plan to add labeling to this feature to make its usage clear.

Furthermore, the “agree” feature was not very usable to the user. Many were not aware that it was clickable. And even when they did interact with it, they did not get any feedback responding to the agree/disagree action. We plan to implement a more detailed and responsive rating system which allows the user to input not only whether they disagree/agree with a rating but why they do and indicate which categories are most relevant.

Other features to be implemented

In addition to things that need to be added based on user testing, there are certain features we would like to implement to enhance the usability of the interface. One of these features is advanced search functionality. During our implementation phase, we were not able to add filtering as we originally planned. Filtering by name (alphabetically), awk level, and imdb rating will give the user more flexibility in finding desired movies.

Finally, we would like to hook our backend up to an existing movie database. Currently, we have a mock database with a very limited amount of movies. Connecting our search to databases such as TMDB and Guidebox would allow not only for more movie data sources but also more sources to build base Awk ratings from.

Project Assessment

Project Conception, Design, and Implementation

We believe our project was very well done, we were able to successfully implement a pretty original and useful app concept, while at the same time following the principles of good user interface design and development. Starting off with the concept of our project, an app that determines the amount of awkwardness that may be encountered while watching a movie with others is a relatively original idea and very applicable to families today. Our original target audience was primarily teenagers who want to pick out a movie to watch with their family without being subject to the inevitable awkward moments due to sex scenes, vulgar language, etc. As we thought about our idea more, however, we realized it could be applied to a wider range of situations, such as parents wanting to pick out a movie to watch with their kids, or a couple wanting to find a movie to watch together.

The bottom line is there are plenty of people out there who would find our app useful, and our app's concept would be appealing to many. Although we are not the first to attempt to rate the awkwardness of movies, we are the first to come up with changing the awkward rating based on the type of people the user is watching the movie with. This dynamic awkward rating using user customization via sliders for each awk category is what makes our concept unique and a step above the competition.

The main reason our project was successful though was the successful implementation of our idea with a good user interface design. We were able to create an app with all of the features we originally intended, although we did change quite a few things to make things easier on the user. When a user opens up our app, they are able to login or create an account, set their preferences for awkward levels for each category, then search for a movie to determine how awkward it would be based on those preferences. This was the core idea of our app, although we added other functionality we thought would be useful to the user, such as an IMDB rating of the movie, a plot summary, and descriptions for each category on why it's awk rating is what it is. All of these features were successfully implemented in our app, as well as a lot of minor ones that don't need to be mentioned here.

As beneficial as it is to have all these nice features, the app isn't useful at all if the user can't navigate them, which brings up the most important part of our app, the user interface. The user interface was probably the most challenging part of designing this app, mainly because a lot of issues came up that we didn't even think about initially. An example of this is how during the user testing we realized a lot of users didn't understand our rating system, they didn't understand that the rating was out of 10 and if a high rating meant a movie was more or less awkward. We accounted for this by adding a color to the rating, a green color meant less awkward, and a red number meant more awkward, as well as adding a brief text below the rating explaining how awk it was, for example 10/10 would say super awk, and 5/10 would say kind of awk. We changed many other features of our app based off of the user testing we did, so we could have the best possible user interface. After we made all these changes, we were left with an application with a very intuitive and fluent user interface that most users should find easy to use.

Because of all the reasons listed above, good concept, good implementation, and good user interface design, we believe that our project was very successful, and deserves an A.

Project Process and Group Dynamics

Our group was made up of all friends, so the typical challenges from group work of not being able to contact one another, or being able to meet up periodically, was generally not an issue. We had a really good thing going in our group chat where someone would bring up an issue that needed to be addressed and we would collectively tackle it as soon as we were available. The

response times were generally good so we could solve problems relatively quickly. It probably would have been helpful to start some of the things earlier so we didn't have to cram so much near the end, but it ended up working out just fine.

Everyone in our group seemed to be quite busy from work, other classes, and project due dates overlapping, so with having a somewhat large group, it was sometimes difficult to schedule times to work on stuff so a lot of what we did through the semester was done on Google Docs and GitHub. With that, we were able to talk frequently amongst each other to sort out small problems before they became too large to handle. A lot of the work done for this project was through various types of reports, which means we were able to be quite successful while not necessarily working in the same room. This also led to a challenge of dividing up which parts each group member should do. For some of the smaller reports, sometimes only a couple of group members would end up contributing to it. However for larger tasks, such as this final report, all of the group members worked on different pieces and categories. Our actual app was also easier to divide up amongst group members. After creating the initial framework, we assigned each section of the application to a different group member. This allowed us to quickly build our application without stepping on each other's toes.

From this experience, we all learned some valuable lessons. It's easy to get defensive, especially if it is something you designed and put effort into, but you need to be open to making changes because that's how the interface will grow and be enjoyable for the user. For example, the use of profiles initially sounded like a good idea, but quickly became confusing and convoluted. After cognitive walkthroughs, heuristic evaluations, and user testing we realized that it did not have the same effect/functionality we wanted it to have.

However it is also important to not see a potential problem and immediately fix it without weighing out the benefits as well. For example, our movie detail page includes a lot of information. Though the amount of information caused some confusion during user testing, we felt the information was necessary and would be a requirement to satisfy a broad scope of users. For this, we would have to apply better content strategy and update the design to feel less crowded and provide the user with the most important information first. The last, somewhat broad takeaway from this project, is that by working in these somewhat large groups, we learn to collaborate in a setting similar to the ones found in many software development companies which in turn further prepares us for industrial work.

Appendix

User Testing Plan

Users Involved:

Connor and Spencer

Time: November 27th at 7:00 p.m.

Location: Max's apartment building

Facilitators: Michael, Max

Josh and Anushka

Time: December 11th at 8:30 p.m.

Location: Keller Hall Atrium

Facilitators: Mikhaila, Ben

What will be Tested:

The first task we will give the users is finding a movie, as it is fundamental part of our application. We will not give the user a lot of description, as we want to see if they can figure it out on their own.

The second task we will give the users is modifying their preferences so they will receive customized Awk ratings.

The third task we will give the user is to give input on the awkwardness level.

Testing Goals:

Determining whether or not getting rid of profiles in favor of global user preferences was the right choice.

Testing our assumption that the user will be able use our search engine to find the correct movie. Concluding if the user understands the usefulness of the “Your Movies” feature and how to use it

Instructions for the User:

Intro/Describe Role/Expectations

Hi, _____. My name is _____, and I'm going to be walking you through this session today. Before we begin, I have some information for you, and I'm going to read it to make sure that I cover everything. Let me go over why you're here briefly. We're asking people to try using an app to see that it performs as expected. The session should take about thirty minutes. The first thing I want to make clear right away is that we're testing the app, not you. You can't do anything wrong here. As you use the app, I'm going to ask you as much as possible to try to think out loud: to say what you're looking at, what you're trying to do, and what you're thinking. Any spoken thoughts will help. Also, please don't worry that you're going to hurt our feelings. We're doing this to improve our app, so we need to hear your honest reactions. If you have any questions as we go along, just ask them. I may not be able to answer them right away,

since we're interested in how people do when they don't have someone sitting next to them to help. But if you still have any questions when we're done I'll try to answer them then. We will be recording your screen, but we will not be recording you or your voice at any time. If you need to take a break at any point, just let me know.

Prototype Orientation

The app is called Awk. It is designed to assist you in deciding if a movie will be awkward to watch with the people you want to watch it with. When you first use Awk, you will be prompted to set your tolerance level for viewing sexual content, inappropriate language, violence, and discrimination. Awk will use your ratings of these categories, along with a user rating system, to determine if the movie will be awkward for you to watch. These tolerance levels can be adjusted for watching movies with different groups of people.

First task - setting their preferences and finding a movie

Okay, now I'm going to ask you to try to perform a specific task in the application. Let's say that you are about to watch *Wolf of Wall Street* with your parents and wanted to double check that it won't be awkward. Open the app and try and find *Wolf of Wall Street* and decide if it is appropriate. <- don't tell them how to do it

<Ideal path: user will be shown initial preferences screen, they will set their tolerance levels, then will search for the movie and view the ratings>

Second task - changing their preferences

Next, let's say that now you are going to watch *Wolf of Wall Street* with some friends, and you have a different definition of what is awkward with them. Try changing the Awk settings to allow you to receive a rating that would better suit you watching *Wolf of Wall Street* with them.

<Ideal path: user goes to screen where they modify preferences & sets new levels>

Third task - agreeing or disagreeing with an awkward rating.

For the third and final task, let's say you watched this movie and you thought the awkward rating wasn't exactly right. So now you want to let the app know.

<Ideal path: user gets to the movie description page (may still be on it), and they hit agree? Then they answer the prompt>

What will be Collected:

We will collect notes on the session from the designated group members, and collect any sketches from the participant. We will also meet up as a group after each test and take more notes on the discussions of our findings.

Responsibilities:

We will have three group members at each test, one person orienting the user and the other two taking notes.

Potential Problems:

If the user gets stuck or lost, we will move them onto the next task. We will also mark that down in our notes and identify the path they took to get where they were. "What were you expecting to see here?"

User Testing Raw Notes

Aayushka
- Thinks it's just for her, not for family/watching with others,
- Feels like higher rating is awkward, not sure.
- Confused by star
- Looks at plot to get more information on how awkward it is.
- Uses picture to determine awkwardness
- Didn't realize you could expand categories
- Not sure how to agree with rating at first, figured it out eventually, confused that it led to nowhere, rating didn't change, wanted to know how many people agreed or disagreed.
Tash
- No problems with sliders
- 7.8 kinda awk, sounds confused
- Wanted to know where ratings came from
- Maybe wanted Alcohol/Drugs/Smoking to each have own categories.
- Liked having picture
- Agreed with rating just fine.

User: Spencer Date: Dec 10, 2016
- Noticed type that should be removed
- Friend Deadpool w/ parents?
- Slider was hard to move
- "Don't really care about that" - referring to some user preferences
- Moved slider to left to indicate Awk
- Determined it to be "Kinda Awk"
Now w/ friends:
- Found user preferences on sidebar instantly
- "Did it freeze?" - "OH, I just have to hit 'x'"
- No feedback or change a preference on sidebar
- suggests a bar for agree/disagree
* I wouldn't want to make a profile, it should be simple.*

Brainstorming Session after User Testing

User Testing - Post Session

General

- more feedback
- help user understand ratings
- make clickable items, more clickable lacking
- give freedom to say awk-levels (when giving feedback)

Feedback

- let them know that they changed the preferences in the sidebar.
- give user feedback when they give feedback
(i.e. thank you, stat or who agrees,
or not, option to change awk-level, etc).
- let them know buttons work.

Usability

- look into styles of drop downs. People didn't know things were clickable
- help user understand ratings are out of 10. Maybe small "1/10"

Freedom

- give user ability to input awk-levels when they disagree with the rating they received

Content

- think about better organization on movie page.

Design Log Table

Change	Reason	Week	Stage
Login allows facebook and google connection	Some people may not want to go through the process of setting up an account.	6	Initial Draft
Set up Awk with sliding bars.	Allows for more customization and a more accurate rating.	6	Initial Draft
Search screen provides brief instructions.	Making it clear to the user what the page is meant to accomplish.	6	Initial Draft
Adding Profile editing screens. Drag and drop table for order customization.	Profiles allow the user a way to keep certain awk-ness saved within their profile. This allows for easy access.	6	Initial Draft
Add bottom features bar, but change items to Trending Awk, Top Awk and My Movies.	New Releases was too close to Trending Awk. People really seemed to want an option to know what the most Awkward movies are.	6	Initial Draft
Allow Filtering option on the list screen.	This is related again to people who may want to see what movies are the most awkward to watch with people.	6	Initial Draft
Adding rating to Movie Description Screen.	Like the Professor said, people may be willing to put up with an higher awkward level if the movie is really good. However, we did not want to make this decision for the user, so we just provide both the rating of the movie and the Awk-Level.	6	Initial Draft
Value for Awkward Level	More quantitative than just an OK or Not-Ok message.	6	Initial Draft
Slide Movie to the right rate option (and option to rate in movie description)	We thought about having a message to pop up to ask people if the awk-level	6	Initial Draft
Value for the Awkward Level changed to have color, also include a written indicator of awkwardness	Some people may be confused about what the Awk level corresponds to and if a higher number is good or bad. Adding the color and the words will clear things up for the user	9	Cognitive Walkthrough
Prompt for agree/disagree	It just asks if they agree then gives them yes/no	9	Cognitive

should remind user of the rating.	options. To help the user understand, we should include the name of the movie and awkward level again.		Walkthrough
Add option to add movie to My Movies within the movie detail page.	Currently no way to add movies into My Movies. An screen that is meant to help more experienced users.	9	Cognitive Walkthrough
Remove profile functionality to just give the user options to change the sliders in side nav.	The profiles added a lot of steps to customize the apps ability to predict the awkwardness. It was the task that had the most steps, and the most steps that were taxing on the user. There was also the potential for a user to change the settings a lot, therefore we wanted to streamline the process.	9	Cognitive Walkthrough
Change My Movies to Your Movies	The use of your instead of my.	10	-----
Add labels to feature pages (Trending Awk, Most Awk, Your Movies)	The user may forget what page they are on. This way they can recognize it instead of having to recall it.	10	Heuristic Evaluation
Add feedback to the user for when they add a movie into Your Movies	Besides the changing of the button from being unfilled with a plus to filled with a minus, the user may not realized that the movie is now in Your Movies. So with the feedback, it allows the user to understand what they did.	10	Heuristic Evaluation
Changing the indicators on the customization awkwardness sliders to make it clear which way to slide and add labels to side nav.	We realized that we did not have labels for levels on the sliders in the side nav, so we added those in order to help the user realize they are the same as in the customization. We also changed them to not-awk and super-awk because it was more clear than what was there before.	10	Heuristic Evaluation.

Final Paper Prototype

