Final Report

Wenqiang Chen
Hamilton, Canada
chenw25@mcmaster.ca

Carolyn Chong
Hamilton, Canada
chongce@mcmaster.ca

Kevin Ly
Hamilton, Canada
lyk2@mcmaster.ca

Meraj Patel
Hamilton, Canada
patelmu2@mcmaster.ca

ABSTRACT

A background on related work and a usability evaluation on a design prototype, called Streamer, is presented here. The topic is a design of a video on demand web application. The introduction provides an overview of the software. The related work section introduces four existing web applications. Each web application is critiqued based on its navigation and organization. A software prototype was designed and then evaluated on its usability. The evaluation consisted of five participants performing a series of tasks and then answering a questionnaire. It was found that despite participants performing better on the tasks using a similar software, participants preferred the user interface of Streamer.

INTRODUCTION

An increasing number of consumers in Canada are supplementing their media entertainment through online video on demand services. As such, more competition is emerging to meet the demand of this new market. Some services are free, while others require a subscription fee.

A better user interface for a video on demand web application was designed. Major issues around this topic included poor organization and frustrating navigation of the applications' media content. More specifically, we felt the navigation of these applications was inefficient because it took too long for a user to navigate from the main page to their desired content. It took too many mouse clicks to find what you were looking for, and often it was unclear how to begin the search for what you wanted. It was a confusing process plagued by the lack of organization of the overall site and its content. The layout of web applications was not intuitive and there were too many methods to go about reaching the same goal. On some applications, the user was not able to personalize the organization of content. This was especially troublesome and tedious for a user browsing for content with specific preferences in mind.

Watching media should be a source of entertainment, not frustration. Improvements on existing software should include a fast and simple process for users to access what they want whenever they want. This improvement could include a decrease in the number of button clicks, or a minimalist interface that makes the user experience more straightforward and intuitive. Furthermore, the improvements should include an interface that users of all ages and experience with computers can easily understand and navigate.

Ergo, our unique design has improved on two high-level goals of video on demand web applications: searching for specific content and browsing for content with preferences. An example of searching for specific content is navigating to an episode of a TV series that a user intended to watch. An example of browsing is using filters and search features based on user preferences to narrow displayed results. These goals are common tasks between all video on demand applications. The tasks the user must perform to achieve these goals include navigating the site and an understanding of the site's organization of content.

The related work section introduces four popular video on demand web applications. They are Crackle, Netflix, Plex, and Shomi. Each survey was critiqued on its ability to address the two high-level goals: searching for specific content and browsing for content with preferences.

RELATED WORK

Crackle

Crackle was an online distributor of movies and TV shows. It was a free video on demand web application with commercials. The navigation of the Crackle interface was overall straightforward. As shown in Figure 1, there was a simple, uncluttered navigation bar at the top of every web page that contained a search bar and quick links to the TV shows library and the movies library. The titles were displayed in rows that spanned the width of the screen. A user browsed all titles by scrolling vertically down the page. The scroll was a never-ending scroll. A user could also opt to use the search bar to quickly find a desired title, actor or genre. Playing a title was simple; a user clicked on the icon and the user was redirected to a page that immediately began playing the title in a player.

The organization of the Crackle interface was cluttered and not intuitive. Media titles could be browsed by categories and

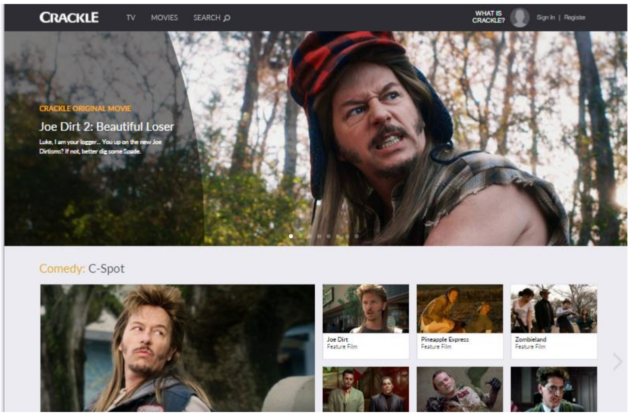


Figure 1. Crackle's home page.

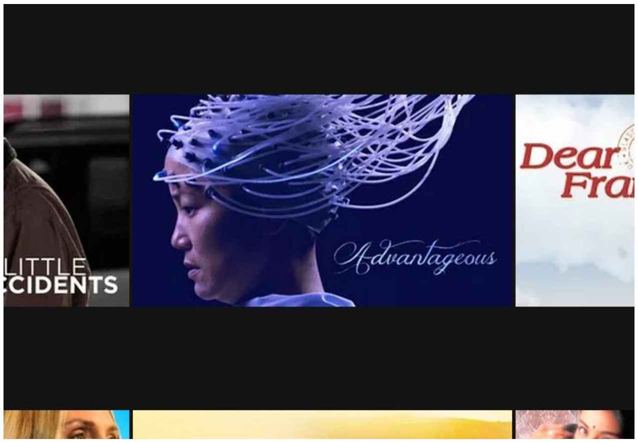


Figure 2. How titles are displayed on Netflix.

genres displayed on the main page, or by media type via the navigation bar.

The organization of the media on the main page of Crackle was not intuitive (see Figure 1). Across the top of the main page was a large carousel showcasing movies and shows from random genres. The carousel filled the majority of the screen. It was not clear to the user why these media items were selected to be showcased. Below the carousel were horizontal sections of different media categories and genres. There was no obvious organization in the listing of the categories, and it was unclear as to whether the categories and titles displayed represented the extent of Crackle's library. Some category titles were in a different colour font than the rest, for no apparent reason. This was poor design consistency [1]. Due to the large sizing and layout of the icons, the variety of selection displayed was limited. As a result, the user was forced to do a lot of scrolling to view the selection of items.

Aside from the main page, users could restrict their browsing using the navigation bar to movies or TV shows. Pull-down menus on the library page allowed the user to narrow their search by genre and length, however, a user could only choose one genre and one length to search as opposed to multiple genres and lengths. The user could then sort the titles alphabetically or by date added. There was no option to sort by popularity, trending, year or rating. Once a user selected an item to watch, they were redirected to a new page with the player. Below the player were facts about the movie or show, in addition to "You Might Like This" and "All New This Month" sections. These sections were only found on the player page, and nowhere else on the website. This new page was cluttered and full of content that was distracting to a viewer who had already chosen what to watch, consequently providing unnecessary feedback [1].

Netflix

Netflix was an online provider of on-demand streaming media to viewers across the globe. The organization of the Netflix user interface was simple and straightforward to navigate. The top of the webpage consisted of a menu bar that allowed users to browse media titles by genre. There was also a search bar for browsing by a particular title, actor or genre. On the main page, titles were organized by genre in the form of subsequent rows. Clicking on a genre would redirect you to another page consisting of media titles displayed in rows. To watch a video, the user had to click on a media title to be redirected to a page that would begin playing the video.

The organization of media on the main page attempted to appeal to audience interest, but lacked ingenuity. From top to bottom were rows of genres consisting of a carousel of media titles. A few of these rows depicted titles that Netflix assumed would appeal to the user. Most often, these did not. This was because they were based on titles watched, rather than titles liked by a user. Netflix lacked proper functionality to display media that appealed to the user's interests. This poor design choice resulted in the user having to search through the user interface, as no significant titles were made visible at first glance [1].

Netflix organized and displayed media titles by use of pictures. To view the actual title, the user had to place their cursor over the picture and a prompt would appear showcasing the name. This was a poor design choice, as some media pictures did not clearly display their title (see Figure 2) [1]. This left the user constantly having to move his/her cursor from picture to picture just to view the name. Netflix did not provide an option to display title names.

Media titles were displayed in a single page format. To navigate through titles the user simply had to scroll down the page. Titles would keep populating until there were no more to display. Although this method minimized the number of clicks for the user, it had its limitations. Given a slow Internet connection, a lot of time was spent rendering these media titles. As a result, an endless loop was displayed to the user. This was a poor design choice as it left the user at a standstill [1]. Additionally, having all these titles presented at once created latency issues for feedback in terms of navigating titles [1].

Plex

Plex organized personal media and provided streaming to multiple platforms. This platform efficiently transcoded the video and audio so playback could occur on any device. Plex organized and populated common meta-data for television

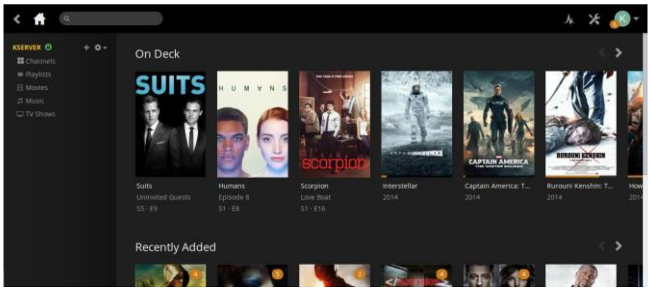
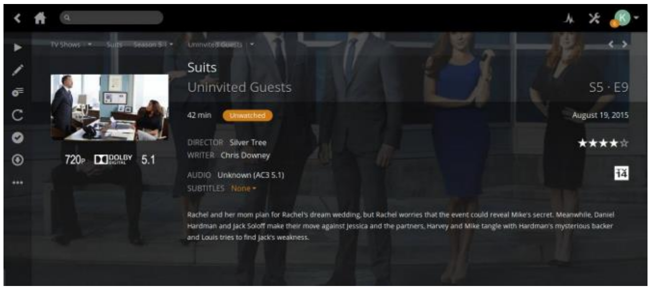


Figure 3. Plex's main page.



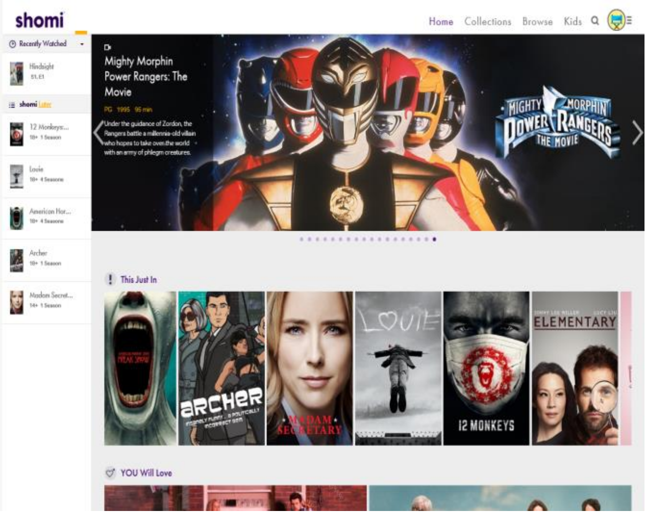
Rachel and her mom plan for Rachel's dream wedding, but Rachel worries that the event could reveal Mike's secret. Meanwhile, Daniel Hardman and Jack Soloff make their move against Jessica and the partners, Harvey and Mike tangle with Hardman's mysterious backer and Louis tries to find Jack's weakness.

Figure 4. How Plex displays episode facts.

series and movies. Plex also enhanced the streaming platform with additional features: unwatched episode indicators, ability to download/sync a title to any connected device, and library access control. Plex focused on an easy to navigate platform where the user could find specific content quickly. Movies and television shows were separated into libraries, as shown in the left panel in Figure 3. Content recently added and content still being watched were categorized in the main page.

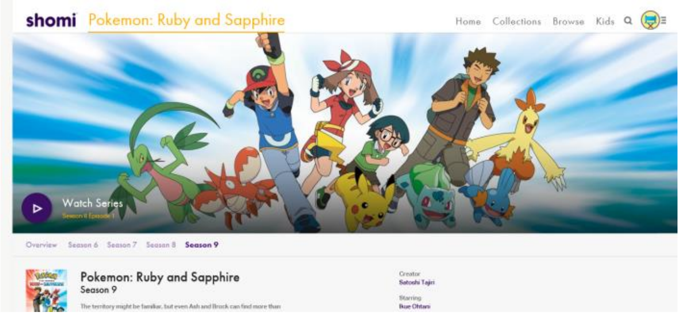
Navigating to a latest season was not possible from the main page, as shown in Figure 3. Viewing a television season required the user to select the television show from the library, which then rendered the layout in Figure 4. This was not useful [1]. Each title in the "Recently added" and "On deck" categories had the name and season number if applicable, however, navigating via these descriptions was not possible. Navigating to a specific title would provide a brief description, as well as other media properties such as audio, subtitles and quality, as shown in Figure 4. The location of the title was at the top of the screen, where these links created a natural mapping to a file manager [1]. The media file could be played via the toolbar on the left.

Organization of content was a primary concern for television streaming services. Titles were sorted by series, season and episode that were displayed in either a grid layout or a list.



Under the guidance of Zordon, the Rangers battle a millennia-old villain who hopes to take over the world with an army of phlegm creatures.

Figure 5. "Recently watched" contents and "bookmarked" contents on Shomi.



The territory might be familiar, but even Ash and Brock can find more than

Figure 6. Search in Shomi.

All the content by default was sorted alphabetically, with an indicator for episodes that have not been watched. Episodes could be further ordered by first air date, rating, and date added. Adding specific filters on items, like genre, was hidden behind layers of options. A brief description for each season and episode was displayed upon traversing the library. Custom organization, like filtering by "currently following" or children shows, could not be created in Plex. This was poor organization, as the user would have to locate the titles they wanted to watch every time they visited [1].

Shomi

Shomi was a Canadian subscription on demand video service. Navigating through Shomi was extremely difficult and inefficient. The main page of Shomi suggested contents that may interest the user based on their view history and grouped them into categories such as "You will like" and "This just in". It also displayed contents that the user recently watched and bookmarked to watch later. "Bookmark" and "Recently watched" were helpful features because they reminded the user what they planned to watch and where they last left off on previously watched titles (see Figure 5) [1]. These features, however, were only accessible from the main page and were disabled once the user navigated to another page. This hindered the user experience because they could not see what they had "bookmarked" to play directly on their current page.

Another flaw with Shomi was the search feature (see Figure 6). Shomi's searching feature lacked functionality. It only filtered by title, actor, director, and creator name. For example, if the user wanted to watch *Fast and Furious*, they would not get any result when searching for "race" or "car". This limited the user to only the content of which they knew the name. Furthermore, search filters could not be cleared once they were set. For example, if the user searched for "Pokémon" and later wanted to see all results, they would have to start over from the main page.

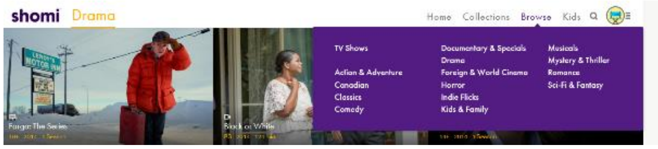
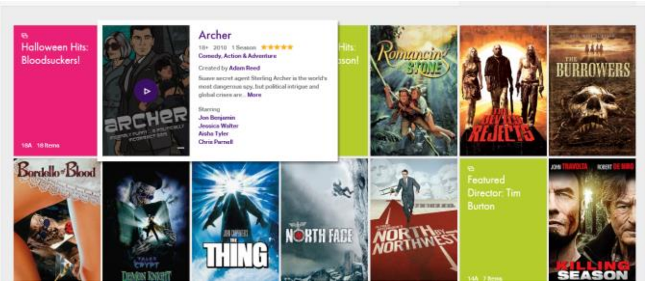


Figure 7. Genres in Shomi.



Archer

18+ 2010 1 Season ★★★★★

Comedy, Action & Adventure

Created by Adam Reed

Suave secret agent Sterling Archer is the world's most dangerous spy, but political intrigue and global crises are... More

Starring

Jon Benjamin

Jessica Walter

Aisha Tyler

Chris Parnell

Figure 8. Shomi's grid view.

Contents in Shomi were not well organized. The user could not pick multiple categories to filter results. If the user wished to watch contents that fell in Comedy, Horror, and Action, they would have to select one of the three genres and manually browse until they saw contents that fit the other two (see Figure 7). There was also no option to search by rating or popularity within the current genre. Furthermore, Shomi displayed its content in a grid view with no text visible (see Figure 8). This kept the page clean and simple but it also hid critical information from the user. This design slowed down the user because they had to hover their mouse over the content to see more details [1].

USABILITY TEST

This section describes a test plan for conducting a usability test during the development of Streamer – a video on demand web application.

Participants

The participants' responsibilities were to attempt to complete a set of representative task scenarios presented to them in as efficient and timely a manner as possible, and to provide feedback regarding the usability and acceptability of the user interface. The participants were directed to provide honest opinions regarding the usability of the application, and to participate in a post-session subjective questionnaire.

Five participants were selected from Thode library at McMaster University. Two females and three males participated. They were between the ages of 18-22 years old. Participants were recruited by approaching students during day-time hours while they were studying. Participants were given a background on the project and were told the methodology of the test. Participants were then asked if they would volunteer their time to participate. No certain skills and/or background were required, however, it was expected that participants were familiar with using a web browser on a computer. No experience using video on demand web applications was necessary.

Training

The participants received an overview of the software and written instructions of the usability test procedure. No training was delivered on how to use the software; only a link to the landing page was given. The test environment was that of where the user was located at the time of recruitment, since that was most convenient for participants. All participants took the test seated at a table on the first floor of Thode library. The first floor of Thode Library is known to be loud relative to a typical library setting. All participants were surrounded by non-participants talking at a conversational noise level. The library was considered 'busy', as every seat at every table was occupied at the time of testing.

Procedure

Participants took part in the usability test on the first floor of Thode library at McMaster University. A laptop running Windows OS with the Web application and supporting software was used in the library environment. The laptop's mouse and keyboard was used. The participant's interaction with the Web application was monitored by the facilitator seated beside the participant at the same table.

The facilitator briefed the participants on the Web application and instructed the participant that they were evaluating the application, rather than the facilitator evaluating the participant. Participants were informed that participation was voluntary and that participation could cease at any time. The facilitator asked the participant if they had any questions.

Participants completed a pretest demographic and background information questionnaire as part of Task 1. The facilitator explained that the amount of time taken to complete the test task would be measured. Participants were instructed that exploratory behavior outside the task flow should not occur. At the start of each task, the participant read aloud the task description from the printed copy and began the task. Time-on-task began when the participant started each task.

The facilitator instructed the participant to 'think aloud' so that a verbal record existed of their interaction with the Web application. The facilitator observed and recorded user behavior and user comments in a text editor.

Task 2 required the participant to search for the pilot episode of *Breaking Bad* on Streamer and then on Shomi. Once located, the participant was required to play the episode in the default media player. The facilitator explained that the number of button clicks and the amount of time taken to complete the test task would be measured.

Task 3 required the participant to browse for any title from the 'Action' genre on Streamer and then on Shomi. Once located, the participant was required to play the episode in the default media player. The facilitator explained that the number of button clicks and the amount of time taken to complete the test task would be measured.

Task 4 was similar to Task 3 except the user was required to find the oldest title in the movie library.

After all task scenarios were attempted, the participant completed the post-test satisfaction questionnaire as part of Task 5 [2].

Due to the range and extent of functionality provided in the application, and the short time for which each participant was available, the tasks were the most common and relatively complex of available functions. Furthermore, the tasks represent the two high-level goals of the software discussed in the Introduction. The tasks were identical for all participants in the study.

A sample of the usability test instructions and questionnaires given to participants is attached.

Success Metrics

Success metrics referred to user performance measured on Streamer against user performance on a similar software, Shomi. Task completion success rates, task completion times and subjective evaluations were used.

Completion rate was the percentage of test participants who successfully completed the task. In other words, the completion rate represented the percentage of participants who, when they were finished with the specified task, had an "output" that was correct. If a participant required assistance in order to achieve a correct output then the task would not be scored as completed.

A completion rate of 100% was the goal for each task in this usability test.

Task completion time was the time to complete a task. It was measured from the time the person began the task to the time he/she signaled completion. An average task completion time of five seconds greater than the average task completion time of Shomi was the goal for each task in this usability test.

Subjective evaluations regarding ease of use and satisfaction were collected via a questionnaire. The questionnaire utilized rating scales. This data was used to assess attitudes of the participants.

RESULTS

The usability of Streamer was evaluated by asking test participants to complete three different tasks on both Streamer and Shomi, as well as a pre- and post-test questionnaire. The participants' performance was measured by the total time to complete each task. The average times of all participants to complete each task on Streamer was compared against the average times of all participants to complete each task on Shomi. 'Think-aloud' results provided subjective feedback on the user experience of Streamer. The post-questionnaire provided subjective feedback on Streamer itself, as well as Streamer compared to Shomi.

The task completion rate was 100% for all tasks on both software applications, therefore the success metric was met for completion rate (Table 1). In Task 2, Streamer had a better average time by 6 seconds (Table 2). In Task 3, Streamer and Shomi had comparable average times (Table 4). In Task 4, Stromi had a better average time by 17 seconds (Table 7). Therefore, the success metric for task completion time was only met for Task 1.

The most interesting result directly visible from the data is that despite not meeting the success metrics on tasks 2, 3, and 4, all participants agreed or strongly agreed in the post-questionnaire that they would use Streamer over Stromi.

Participant	A	B	C	D	E
Age	22	18	22	21	20
Gender	Male	Female	Female	Male	Male
Experience	Once	Once	Monthly	Weekly	Daily

Table 1: Task 1 Pre-Questionnaire Responses.

Application	Streamer	Shomi
Completion Rate	100%	100%
Avg. Time (s)	33.02	39.2

Table 2: Average time for Task 2.

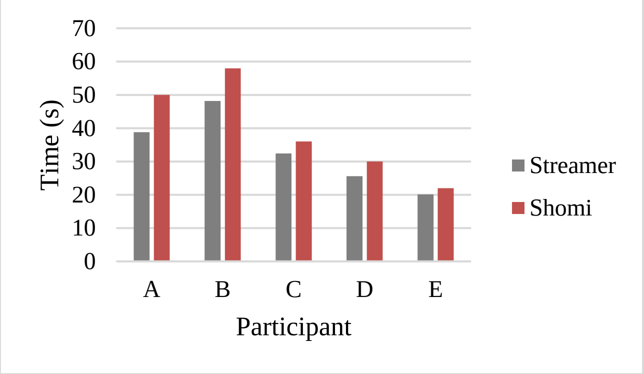


Table 3: Task 2 completion times for each participant.

Application	Streamer	Shomi
Completion Rate	100%	100%
Avg. Time (s)	20.46	19.0

Table 4: Average time for Task 3.

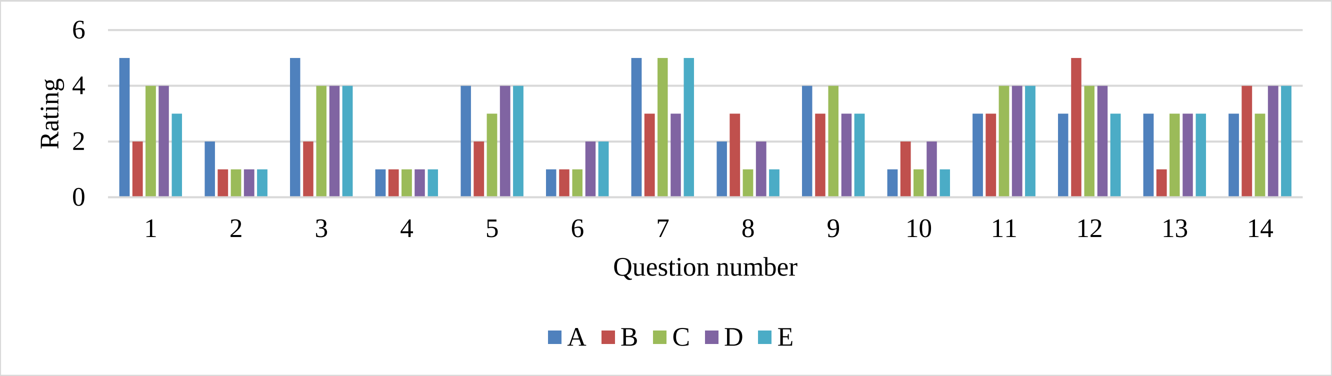


Table 5: Task 5 Post-Questionnaire Responses.

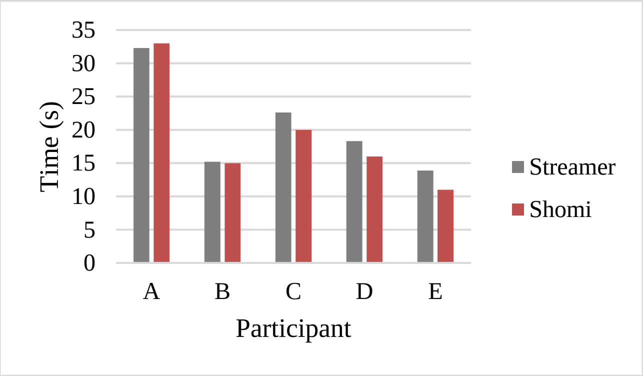


Table 6: Task 3 completion times for each participant.

Application	Streamer	Shomi
Completion Rate	100%	100%
Avg. Time (s)	45.18	28.2

Table 7: Average Time for Task 4.

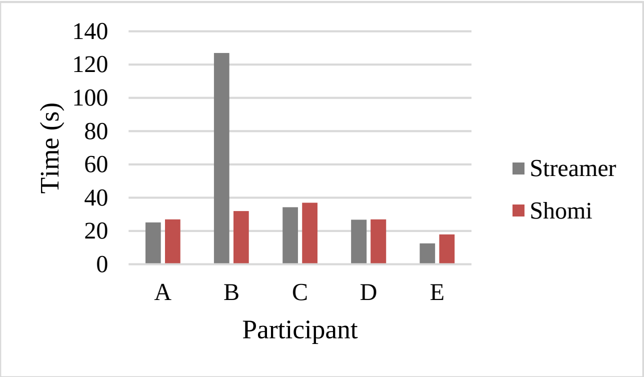


Table 8: Task 4 completion times for each participant.

DISCUSSION

The usability evaluation proved there were many positive aspects of the Streamer user interface. Every participant was able to complete all three tasks in both Streamer and Shomi. Participants were able to finish Task 1 6 seconds faster on Streamer than on Shomi. Streamer's navigational tools successfully allowed the participants to navigate to a desired title faster than a competitor, which was one of the high-level goals. The questionnaire results showed that participants generally agreed that Streamer was easier to use than Shomi due to its intuitive and easy to learn design. Even first time video streaming participants acquired the skills quickly. The participants of the usability test unanimously agreed that the organizational and navigational features were implemented seamlessly into the system, both of which supported our high-level goals. Many participants noted that the windows used to display information of the selected title enhanced the user experience. By the end of task 3, all participants were comfortable using Streamer.

Streamer was not without its problems though. Streamer performed poorly on task 3 and task 4, both of which tested the organization of Streamer by asking users to browse for a title using the genre filter and sorting filter. Experienced participants of streaming applications noted during the 'talk-aloud' that they were accustomed to the organization of the other applications and found it challenging to transition to the different organizational structure of Streamer. Based on the results of the tasks, Streamer did not meet its high-level goal of organization. Interestingly, the results of the usability test showed most participants preferred the user interface of Streamer. This result, however, had low external validity as the demographics of our participants were narrowed to students studying at Thode library, the science and engineering library at McMaster University. Consequently, all of our participants were quick learners of new technologies. Furthermore, the success metric for task 3 and task 4 was not met. In task 3, Shomi out performed Streamer by 1 second. In task 4, Shomi outperformed Streamer again by 17 seconds. The results of task 4, however, could be attributed to the fact that participant B had trouble finding the sort feature in Streamer during their test. The order effect may have played a role in skewing task 4 results because the user did the same task on Shomi soon after. In fact, the order effect may have skewed all of our results as the order of software was not randomized. Instead, Streamer was always tested first, followed by Shomi.

Moving forward, there is space for improvement within the organization and navigation of Streamer. In terms of organization, users gave feedback suggesting that media titles could be viewed in multiple viewing schemes. For example, some users may prefer viewing media titles in a list view without pictures, while others may prefer to view titles by pictures only in a row format with more details of the title listed. There could also be an option to toggle between movie and TV titles on the front page. This way, users could view suggested and popular titles for movies or TV-shows specifically. Regarding navigation, more options could be added to narrow down searches according to user preference. An example could be multiple genre selections. This would allow users to narrowly search media titles that stand in between genres (e.g., horror comedy). In addition, having a keyword search could allow users to search for titles related to a particular key. For example, typing in “robots” should display related movies like Transformers, IRobot or Iron Man. Having these improvements implemented would improve participant satisfaction.

CONCLUSION

Our topic was the usability of video on demand web applications. We presented a background of the topic and discussed related work. Overall, we felt existing user interfaces of these applications were frustrating to navigate and poorly organized. They suffered from a lack of simplicity and intuitiveness, and were good examples of putting “pretty” design ahead of usability. They were clunky and inefficient, and frustrated even a frequent user.

We also provided a usability evaluation on our design prototype. We attempted to improve on two high-level goals of video on demand web applications: navigation, as demonstrated by playing an episode from a specific television series, and organization, as demonstrated by browsing for a specific movie genre.

The results of the user evaluation showed that despite participants performing better on the tasks using Shomi, participants preferred the user interface of Streamer.

REFERENCES

1. Don Norman. 2013. *The Design of Everyday Things – Revised and Expanded Edition.* Basic Books, New York, 10-131.
2. *System Usability Scale.* Usability.gov http://www.usability.gov/how-to-and-tools/resources/templates/system-usability-scale-sus.html

Task 1 – Pre-Test Questionnaire

Please write your age: _____

Please write your gender: _________________

How often do you use video on demand web applications?

___ Never ___Once ___Once a month
___Once a week ___Every day

Task 2 – Playing a specific episode

This task begins on the main landing page. Please play the pilot episode of *Breaking Bad.* The amount of time to complete the task will be measured. Please do not do any exploratory behaviour of the software outside of this task. Please think aloud your thought process and interactions with the software.

Was the participant successful at completing the task? Y/ N

If the participant was successful, how long did it take them to complete the task? _________

Recorded observations of participant's 'think-aloud'?

____
____

Task 3 – Browsing a movie genre

This task begins on the main landing page. Please find and play any desired movie title from the Action genre. The amount of time to complete the task will be measured. Please do not do any exploratory behaviour of the software outside of this task. Please think aloud your thought process and interactions with the software.

Was the participant successful at completing the task? Y/N

If the participant was successful, how long did it take them to complete the task? _________

Recorded observations of participant's 'think-aloud'?

____
____

Task 4 – Sorting titles

This task begins on the main landing page. Please find and play the oldest movie in the library. The amount of time to complete the task will be measured. Please do not do any exploratory behaviour of the software outside of this task. Please think aloud your thought process and interactions with the software.

Was the participant successful at completing the task? Y/N

If the participant was successful, how long did it take them to complete the task? _________

Recorded observations of participant's 'think-aloud'?

____
____

Task 5 – Post-Test Questionnaire

Place a value, between 1-5, for each question. 1 is Strongly Disagree. 5 is Strongly Agree.

1. I think that I would like to use Streamer frequently _____

2. I found Streamer unnecessarily complex _____

3. I thought Streamer was easy to use _____

4. I think that I would need the support of a technical person to be able to use Streamer _____

5. I found the various functions in Streamer were well integrated _____

6. I thought there was too much inconsistency in Streamer _____

7. I would imagine that most people would learn to use Streamer very quickly _____

8. I found Streamer very cumbersome to use _____

9. I felt very confident using Streamer _____

10. I needed to learn a lot of things before I could get going with Streamer _____

11. I prefer the navigation of Streamer over Shomi _____

12. I prefer the organization of titles in Streamer over Shomi _____

13. I prefer the "look" of Streamer over Shomi _____

14. Overall, I would use Streamer over Shomi _____