

Rochester Institute of Technology

**B. Thomas Golisano College
of
Computing and Information Sciences**

Master of Science in Human Computer Interaction

Project Approval Form

Student Name: Andrew M. Fagan

Project Title: Student-centered Duplicate File Finder Interface

Project Committee

Name

Signature

Date

Chair

Committee Member

Student-centered Duplicate File Finder Interface

by

Andrew M. Fagan

**Project submitted in partial fulfillment of the requirements for the degree
of Master of Science in Human Computer Interaction**

Rochester Institute of Technology

**B. Thomas Golisano College
of
Computing and Information Sciences**

Department of Information Sciences and Technologies

12/15/2016

TABLE OF CONTENTS

Acknowledgements	4
Abstract	5
Introduction.....	5
Background.....	6
Current Products	6
Prior Research	9
Motivations	10
Preliminary Work.....	10
Research Questions.....	11
Pre-Prototype	11
Post-Prototype	11
Significance.....	11
Methodology	12
Overview.....	12
Participants.....	12
Interviews	12
Interview Questions	13
User-centered Design.....	13
Prototyping.....	14
Prototype Testing	14
Introductory Script	15
Tasks & Scenarios	16
Interview Findings	16
Interview Summary	16
Affinity Diagram.....	16
Design Considerations	21
Functional Requirements	23
Consolidated Sequence Diagram.....	24
Prototype Design	24
Storyboards	28
Prototype Test Results	29
Usability.....	29
User Experience.....	30
Conclusions.....	34
Future Work	36
Challenges & Limitations	36
Barriers & Challenges	36
Limitations	36
References.....	37
Appendix A - SUS Instrument	38
System Usability Scale	38
APPENDIX B - Prototype Test Notes	39

Acknowledgements

First and foremost I want to dedicate this project to my family and loved ones. Your support and encouragement have kept me motivated through the challenges of graduate studies at the Rochester Institute of Technology.

I would like to thank my capstone advisors, Professor Michael Floeser and Professor Matt Huenerfauth. Thank you for believing in me as a student and for taking the time and putting in the effort to ensure my success with this project. I feel very fortunate to have received your guidance and am amazed at your expertise in your respective fields. You are both outstanding professors and mentors and I sincerely appreciate your passion for teaching and devotion to your students.

Thank you also to Theresa Pozzi, Matt Lake, and Jill Persson for helping me stay on track with my courses and the capstone process.

Finally, I would like to thank my friends for always believing in me and being patient and supportive as I devoted many hours to my schoolwork. I could not have done this without the support of all those mentioned above, and feel truly blessed.

Abstract

The process of identifying and managing local duplicate files is often necessary, but happens to be a tedious, complex, and frustrating task for many.

Even as file storage options become increasingly larger, inexpensive, and cloud-based, local storage of files is often unavoidable and necessary for anyone who uses a personal laptop computer. This is especially true for students, who may have multiple slightly different copies of a variety of files to manage (i.e. downloads, assignments, source code, job application materials).

The overall goal for this capstone project was to gain a deeper understanding of how students manage the storage, identification, and removal of duplicate files.

The initial phase of this study consisted of gathering qualitative information from a sample of six Rochester Institute of Technology (RIT) students. The students were asked about their duplicate file management preferences, habits, and their opinions of the design of existing products (shown via screenshots). This data was then analyzed and consolidated into various models and used to develop the design of a duplicate file finder prototype.

This prototype was then evaluated during a usability test with four of the six student participants from the interview phase. The participants were asked to complete two scenario-based tasks and give feedback regarding their experience using the prototype and their impression of the design.

Introduction

The overall goal for this capstone project was to develop a duplicate file finder software solution that is user-friendly, easy to use, effective at the task, and addresses the unique needs of student users. The target population for this project is students at the Rochester Institute of Technology (RIT). For many students, their laptop is their primary computing device for academic and career-related work. While other professionals and individuals may benefit from a duplicate file finder program, students are especially likely to have duplicate files (i.e. downloads, assignments, source code, job application materials) to manage. Ultimately through interviewing, analysis, prototype design, and prototype testing, this study has yielded some design implications for a student-centered duplicate file finder program.

Background

Current Products

There are some products on the market aimed at scanning for duplicate files. These products have differing levels of complexity and number of features.

Double Killer software (Figure 1): This software uses a very utilitarian-style interface, but offers grouping of relevant search criteria and filters. This technique of grouping related options is a positive aspect of this software, and grouping was eventually incorporated into the prototype design for this current project.

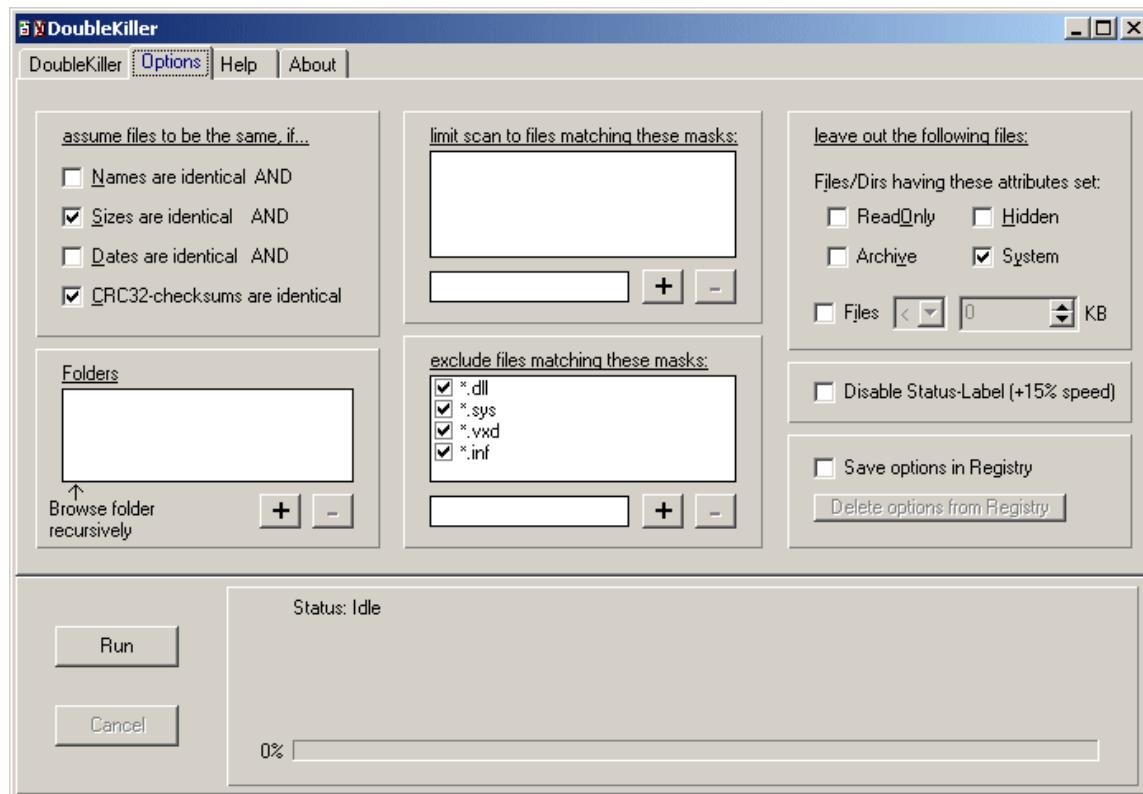


Figure 1 - Double Killer software interface with option grouping and utilitarian styling
(Image Source - <http://www.bangenterprises.de/en/doublekiller/>)

Auslogics Duplicate File Finder (Figure 2): This program offers a multitude of options, which are layered into the interface resulting in a visually appealing and simple-looking interface with advanced features for expert users. Multi-layered interfaces are often useful for simplifying an interface, and ultimately turned out to be a preference for all of the interviewees in this current project. Additionally, the highlighting of duplicate files in this existing program visually groups items for the user. This type of visual grouping was implemented in the current project prototype.

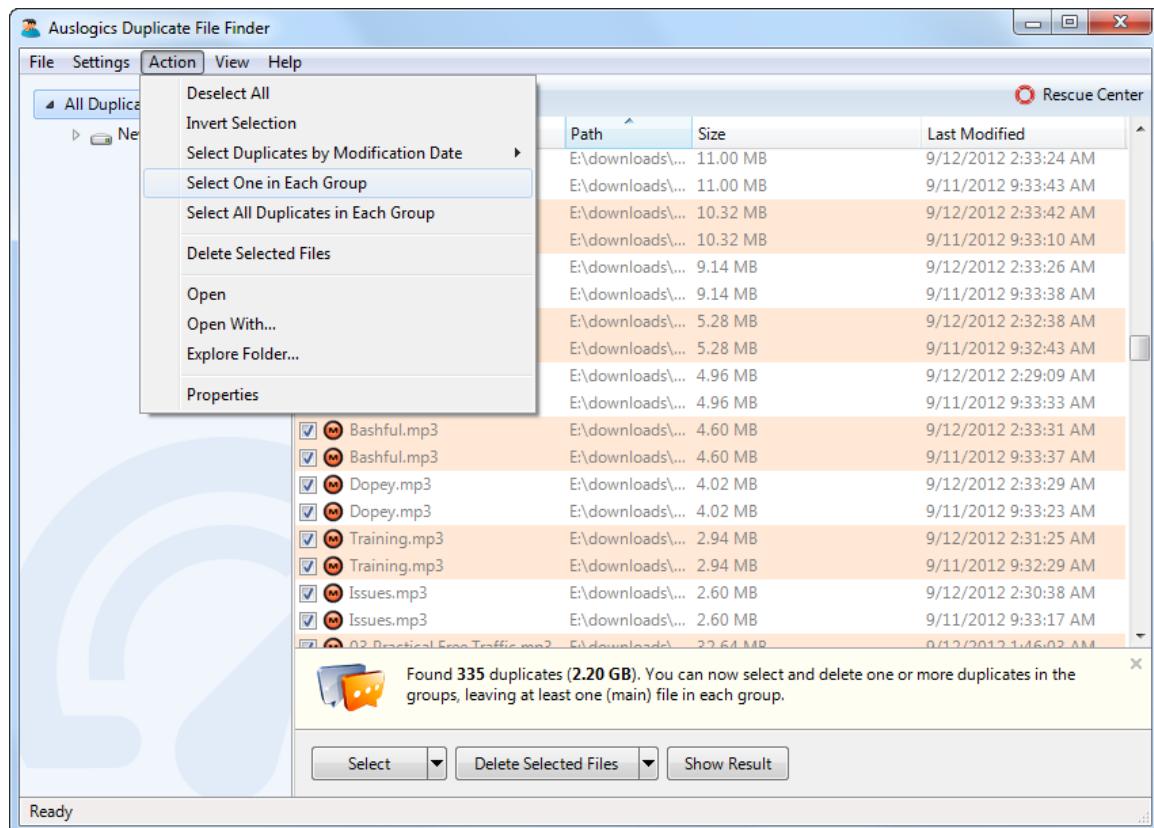


Figure 2 - Auslogics Duplicate File Finder duplicate grouping and scaffolding for advanced options
(Image Source - <http://www.auslogics.com/en/software/duplicate-file-finder/>)

Autosofted Duplicate File Finder 2.2 (Figure 3): With a minimalist design and enlarged buttons (i.e. “Scan”) for the critical tasks this interface appears to be quite user-friendly. This style of interface pairs well with the functionality considering the goal for many users of the program is to simplify and consolidate their digital files. The general layout and minimalist styling were influential in the current project prototype.

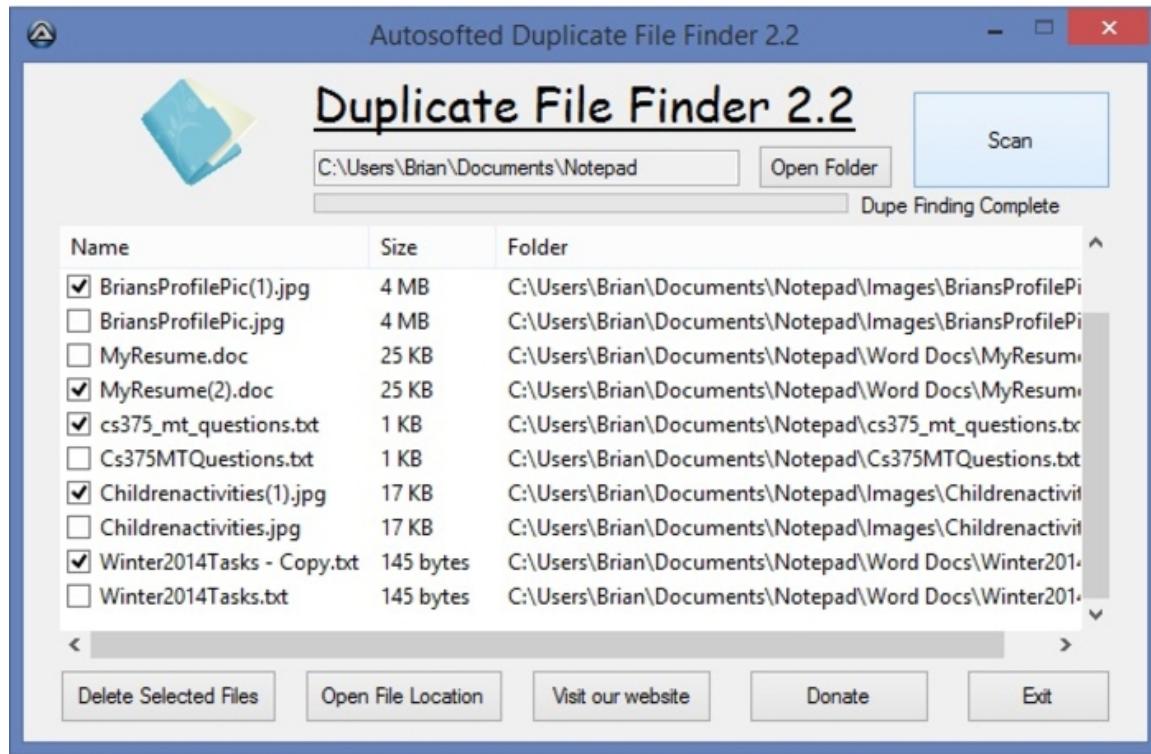


Figure 3 - Autosofted Duplicate File Finder 2.2 minimalist styling
(Image Source: http://www.autosofted.com/duplicate_file_finder/)

Duplicate Detective (Figure 4): This interface appears to show highly visible feedback of the scan results (i.e. number of files scanned, duplicates found, and size of duplicate files) and also offers the each of dragging and dropping the directory to be scanned. The design is modern and minimalistic, but exclusive to MAC OS. The large scan results section served as an influence for this project's prototype.



Figure 4 - Duplicate Detective for Mac, with large results display
(Image Source - <https://fiplab.com/apps/duplicate-detective-for-mac>)

Prior Research

I was unable to find any peer-reviewed research specifically aimed at duplicate file management interfaces; however, research has been done in some related areas which were useful to consider for this project.

Feedback and Visibility of System State

One particular study showed the effectiveness of using techniques such as icon highlighting, hover menus, and search directed navigations as they relate to file retrieval [2]. This may be relevant to consider regarding a file management application that deals with duplicate files. It will be important that a user is able to accurately decipher the appropriate files and determine what is a duplicate, what they have selected to search,

and what they have selected for deletion. While no hover menus were used for this prototype, highlighting groups of duplicates with different colors and search navigation of file results were incorporated.

Transparency and clear feedback may be important to consider in designing a program that involves some level of risk of inadvertently removing important files [3]. It was important to build a system that provides effective visual feedback (i.e. confirmation dialogs) and visibility of system state (i.e. checkboxes) considering the delicate nature and functions of this type of software.

Multi-Layered Interfaces and Scaffolding

Other research on multi-layered interfaces and interface scaffolding is also appropriate to consider depending on the unique needs of novice and experienced duplicate file finder program users. Not all users will require the same depth of features, but to increase error prevention and recovery, this current prototype solution must be first-use intuitive for users from a variety of skill and experience levels. Multi-layered interfaces have been shown to be generally effective at satisfying the needs of both novice and expert users [4].

Motivations

The aim of this project was to gain a better understanding of how students manage the storage and removal of their local files, with specific attention to duplicate files. While products do exist for locating duplicate files, this study focused on extracting design considerations based on the actual needs of student users. This project involved user-centered design processes that helped condense the needs of students and uncover relevant design implications for this specific group of users. This type of program may help students, especially those in computing majors; remain organized and free up more time for academic efforts.

Preliminary Work

The bulk of my preliminary work includes viewing and testing current products that address duplicate file management. These products, detailed above, range in complexity and features.

In order to gauge interest in this project I communicated casually with 5-6 RIT students to begin to understand their duplicate file management habits.

Regarding the technical aspects of this project, I have discussed the program requirements with Professor Floeser and so far have written a working program (in Java) for scanning and outputting information on duplicate files in a user-entered directory. This program scans the directory first by file size, and then by CRC, to identify duplicate files by content in an efficient manor. The program outputs duplicate file names, sizes, and CRC codes to a text file.

The focus of this project will be on designing and testing an interface specifically geared towards college students. The design implications and data gained from this project is useful in implementing a full working duplicate file finder software, and may be used in the future for development of a fully functional duplicate file finder software program.

Research Questions

Pre-Prototype

1. How do users currently manage duplicate files and limited local storage?
2. What does the process of “cleaning up” one’s laptop computer entail?
3. What is important to users when deciding whether or not to delete a duplicate file?

Post-Prototype

4. How do users rate the usability of the duplicate file finder prototype?
5. What do users find confusing or frustrating about the prototype design and how can the design be improved?
6. What are the positive aspects of the prototype design?

Significance

The results of this user study will provide insight into the systems and techniques that users currently employ for managing duplicate files. Organization can be a key component to a student’s time-management, and ultimately their academic success. With many electronic documents to manage and limited storage space on laptop computers, managing files is essential. This study may help further the understanding of how students currently manage the process of sorting through and clearing out duplicate files, and aid in the discovery of useful design implications for improving this process.

Methodology

Overview

Overall this study represents a cycle of iterative design from user interviews and requirements gathering to consolidation and prototyping, and finally evaluation of the prototype. The goal was to gather input from interviewees, analyze this information and utilize it in designing an interface prototype and then to test the prototype with participants from the interviews to evaluate the design.

Participants

Participants for this study were identified via networking with professors, students, and staff and recruited via in-person conversations with students around campus. Students were initially casually questioned about their interest in duplicate file management programs, and out of approximately nine conversations, six participants agreed to participate in the interview portion of the project.

Of the six interview participants, there were four males and two females. Three of the participants were majoring in STEM (Science, Technology, Engineering, & Math) majors while the other three were majoring in liberal arts majors, specifically Communications, Media Arts, and Criminal Justice.

Participants ranged in age from 19 to 26, and represented both graduate (two participants) and undergraduate (four participants) students.

Of the original six interview participants, four completed the prototype test portion of the project. For these participants, two were male and two female, with three in STEM majors and one in a liberal arts major. The age range for these participants was still 19-26, with two being graduate students and two undergraduate students.

One participant from interview and one from the prototype test portion were selected randomly to win a \$25 Amazon gift card. Consent forms for the interviews, user study, and gift card receipts were signed and submitted to the Institutional Research Board (IRB). The signed IRB form cover page is attached in Appendix #.

Interviews

Contextual data was gathered through individual interviews with six interviewees. The Interviewees were selected based on their interest and experience in managing their local duplicate files and were interviewed about their habits, needs,

preferences, concerns, and other factors related to the design of this type of software application. Students were recruited via in-person invitations after casual conversations about their interest in the topic. Interviewees were screened for genuine interest in managing their duplicate files via in-person discussion about the topic and their current duplicate management needs.

Interview Questions

If you have local files with multiple *identical* copies, what types of files are these?

What is your reasoning behind having multiple copies of a file?

If you have multiple *slightly different* copies of any local files, what types of files are these?

What is your reasoning behind having multiple copies of a file?

Do you have a system for cleaning up files on your laptop? If so, what does this involve?

What factors do you consider before deleting a file?

Suppose you have just realized that the local storage on your laptop is nearly full, how do you go about solving this issue?

What concerns do you have about deleting any of your files?

What design features (i.e. confirmation boxes, highlighting, etc.) are important to you while managing/evaluating/deleting files?

User-centered Design

The data gathered during the interviews was then consolidated into several models that were used to illustrate the interview findings, understand the users, and design an appropriate prototype. Specifically, this phase of the process involved sequence modeling, affinity diagramming, and storyboarding. These models can be found below in the Interview Results section. These techniques were helpful in further understanding the needs and preferences of users. The storyboards illustrate specific use-cases of the software solution and some unique problems these users face during duplicate file management.

Prototyping

A functional prototype was designed according to the consolidated information gathered from the interviews. This prototype was used during the user study to gather information on the usability and general design impressions in order to further refine and influence future design guidelines.

The prototype was built with HTML structure, CSS presentational styling, and JavaScript interactivity. This allowed for optimal interface customization, file data incorporation, and rapid prototype development.

The following libraries and plugins were used to rapidly develop the prototype:

- Bootstrap v3.3.6 Minified CSS (<http://getbootstrap.com>)
- Bootstrap Theme v3.3.6 Minified CSS (<http://getbootstrap.com>)
- jQuery JavaScript Library v1.12.3 (<http://jquery.com/>)
- DataTables 1.10.12 jQuery Plugin (<https://datatables.net/>)
- Featherlight jQuery Plugin v1.5.0 (<http://noelboss.github.io/featherlight/>)

Prototype Testing

Once the prototype was fully developed, it was tested with four of the six interviewees to gather data on the usability the overall user experience with the design. Specifically, the System Usability Scale (SUS) will be used to gather a standardized measure of overall usability for the prototype [1]. This instrument is attached in Appendix A.

Since this study is exploratory in nature, data regarding ease of use, frustrating aspects of the design, and user satisfaction were gathered in via qualitative participant responses and comments. Participants were questions and encourage to voice their opinions of the prototype design during and after the prototype testing session.

Testing was held at the Rochester Institute of Technology B. Golisano Building in the Usability Testing Lab on November 2nd, 4th, and 5th of 2016. The testing environment setup is pictured below in Figure 5.

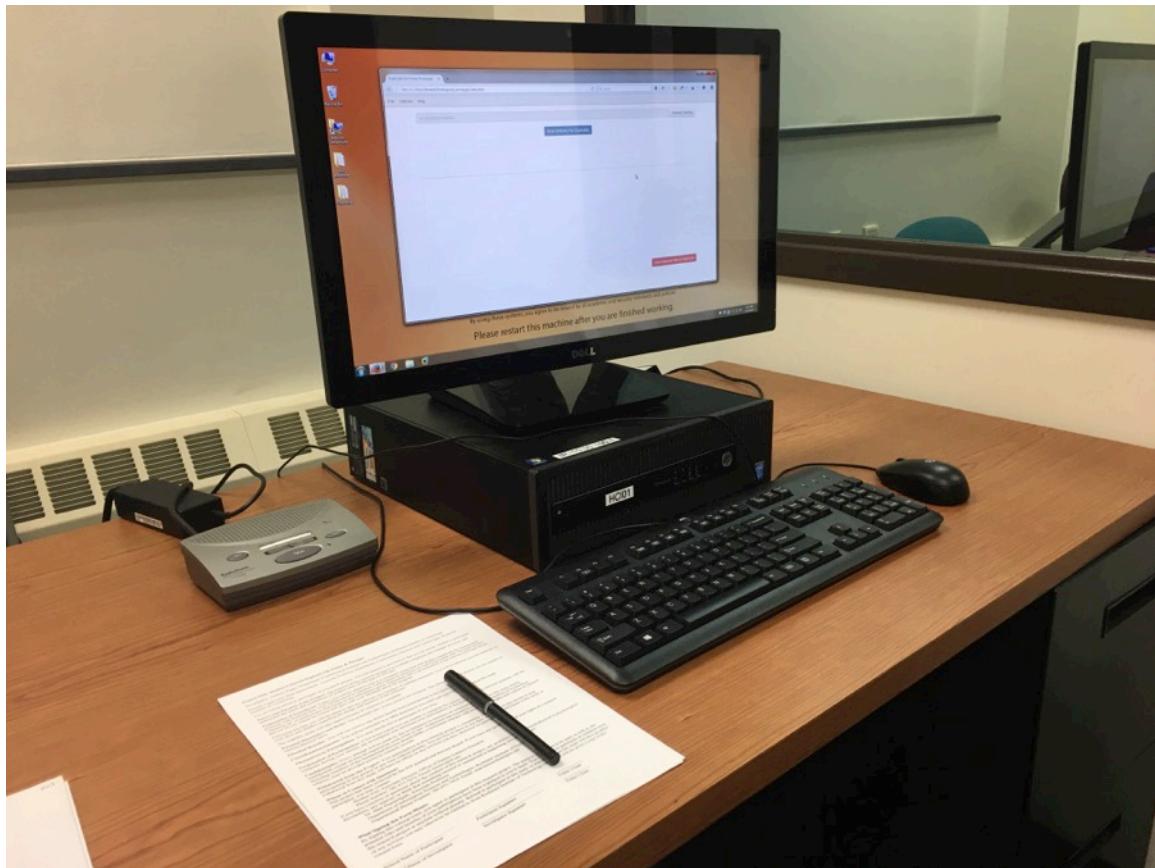


Figure 5 - Rochester Institute of Technology B. Golisano Building in the Usability Testing Lab setup

Introductory Script

The following script was read to participants prior to starting the tasks:

“While completing the tasks please ‘think out loud’ and walk me through your thought process, reactions and feelings about the design of the program. For example: ‘I am clicking this button because I expect it to do X’ or ‘I am looking for X, but do not see it where I expect it to be located in Y’.

This is just a prototype and will have limited functionality as a result. The program will not allow you to actually delete any of the files so do not worry about that aspect. For testing sake the directories for the two scenarios are located on the desktop of the computer in a folder called ‘Directories.’

Remember, we are testing the design and not you. Please feel free to be as honest about the design as you would like.”

Tasks & Scenarios

1. Suppose you are working with some large files for a school assignment and receive an alert stating that your laptop hard drive is nearly full. You decide you would like to clear some space in your “Downloads” folder. When you feel you have completed the task, please say so.

2. As you prepare for career fair, different versions of your resume have been piling up in your “RIT” folder. Recently you accidentally sent the wrong version to an employer, and have since decided to remove older copies. When you feel you have completed the task, please say so.

Interview Findings

Interview Summary

Interviews were conducted with six RIT students. Five of the interviews were completed in-person on the RIT campus, and one was completed via Skype. The original time allotment for the interviews was 30-45 minutes each, however due to interest in the topic, many participants spent 45-60 minutes sharing information about their duplicate file management habits, opinions, and experiences.

The interviews were conversational in nature, and therefore flexible, however the interview questions from the methodology section were used to add structure to the interviews. These questions were also used to gather specific data to assist with the prototype design and inclusion of functionality.

Affinity Diagram

The information gathered from the interviews was consolidated into an affinity diagram to organize the data and illustrate thematic ideas, opinions, and experiences of the interviewees. Figure 6 shows the overall scope of the complete diagram. This diagram is segmented into three sections, that describe the file system, software preferences, and file removal concepts shared by the interviewees. These segments are shown in detail in Figures 7, 8, and 9, respectively.

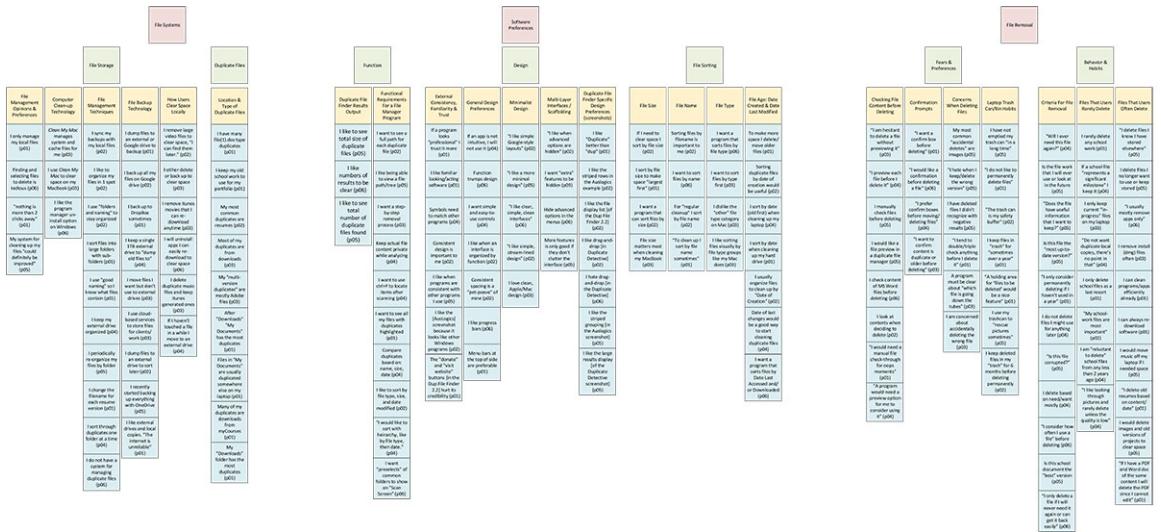


Figure 6 - Overall snapshot of the Affinity Diagram

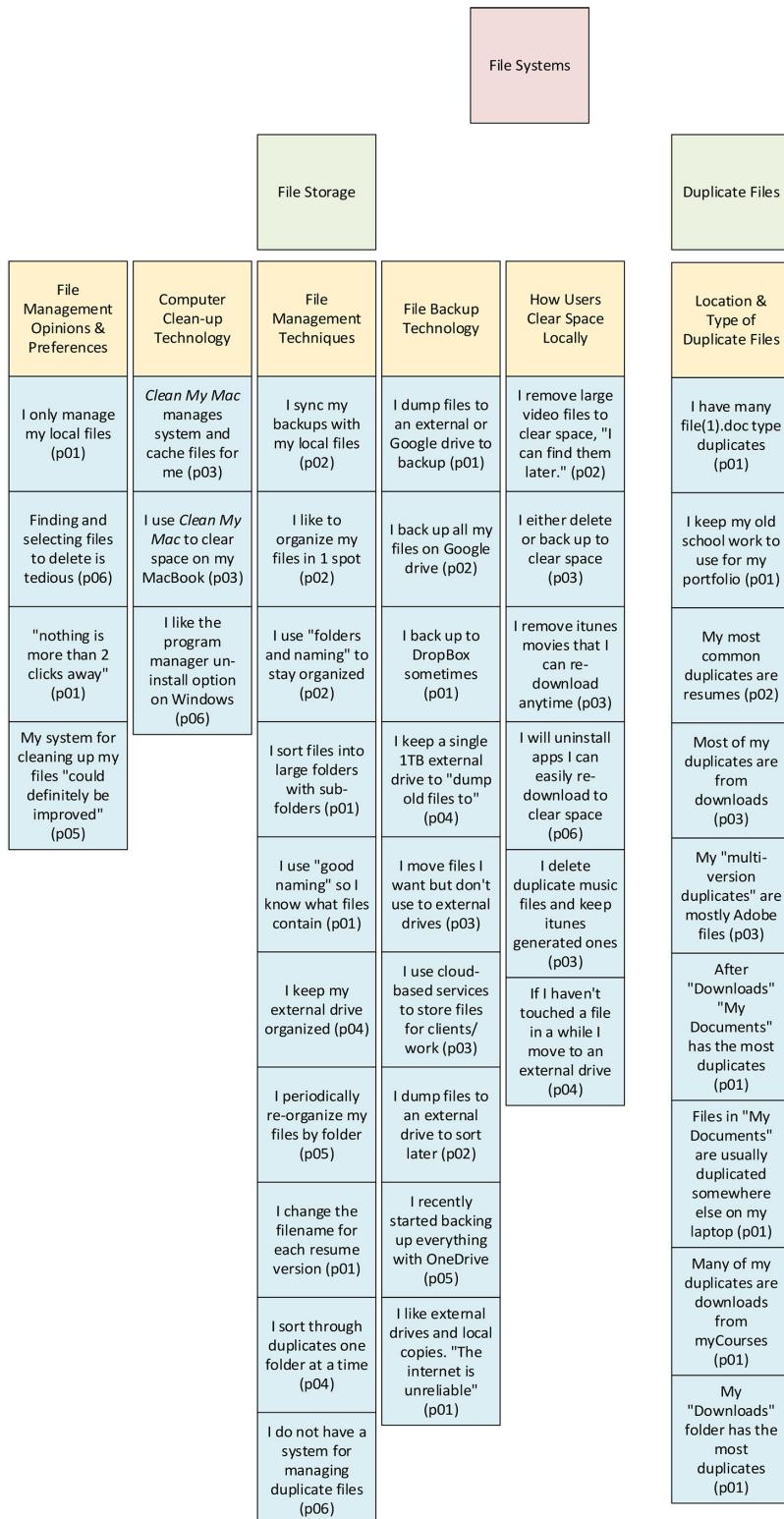


Figure 7 - Affinity Diagram "File Systems" segment

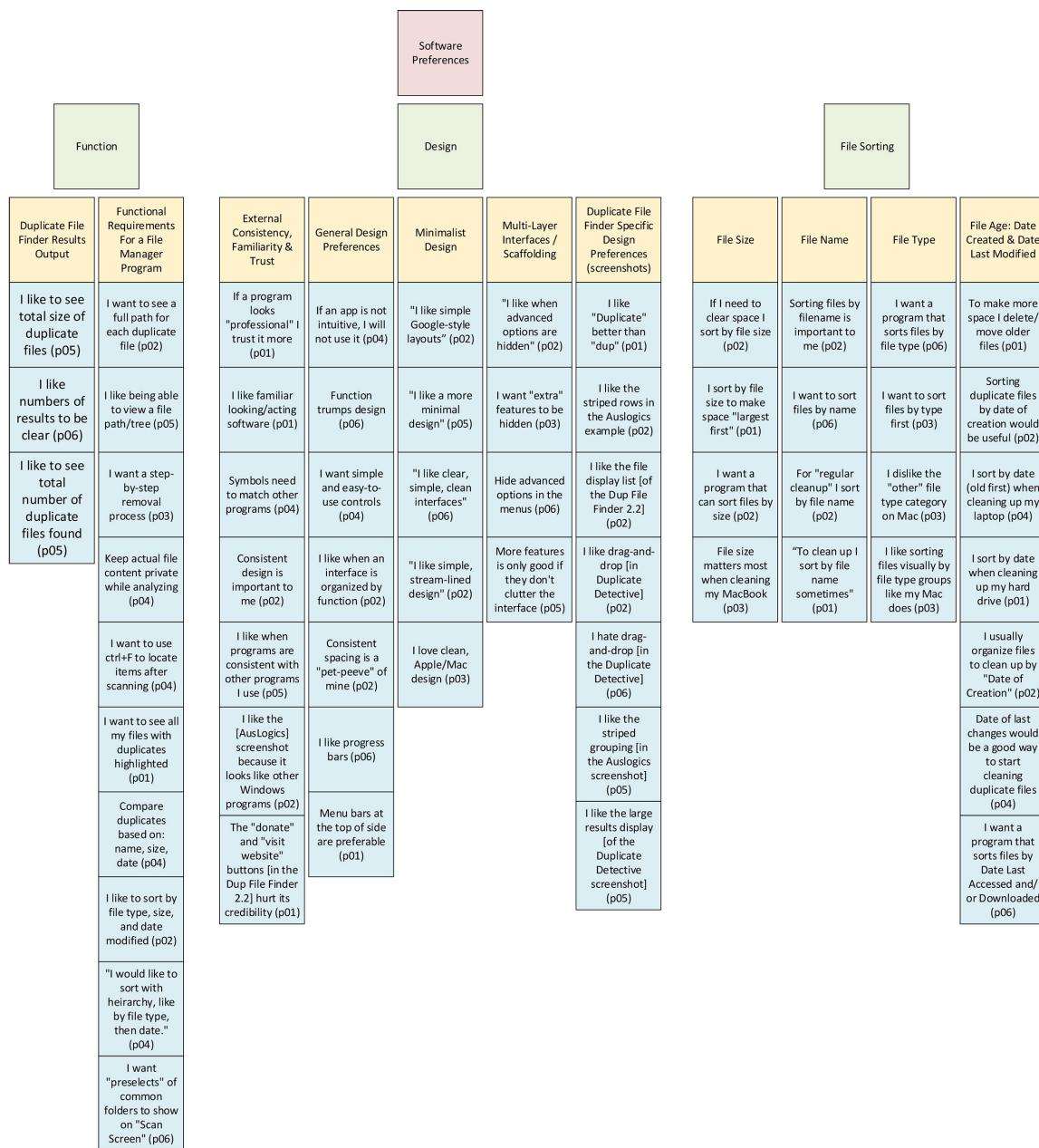


Figure 8 - Affinity Diagram "Software Preferences" segment

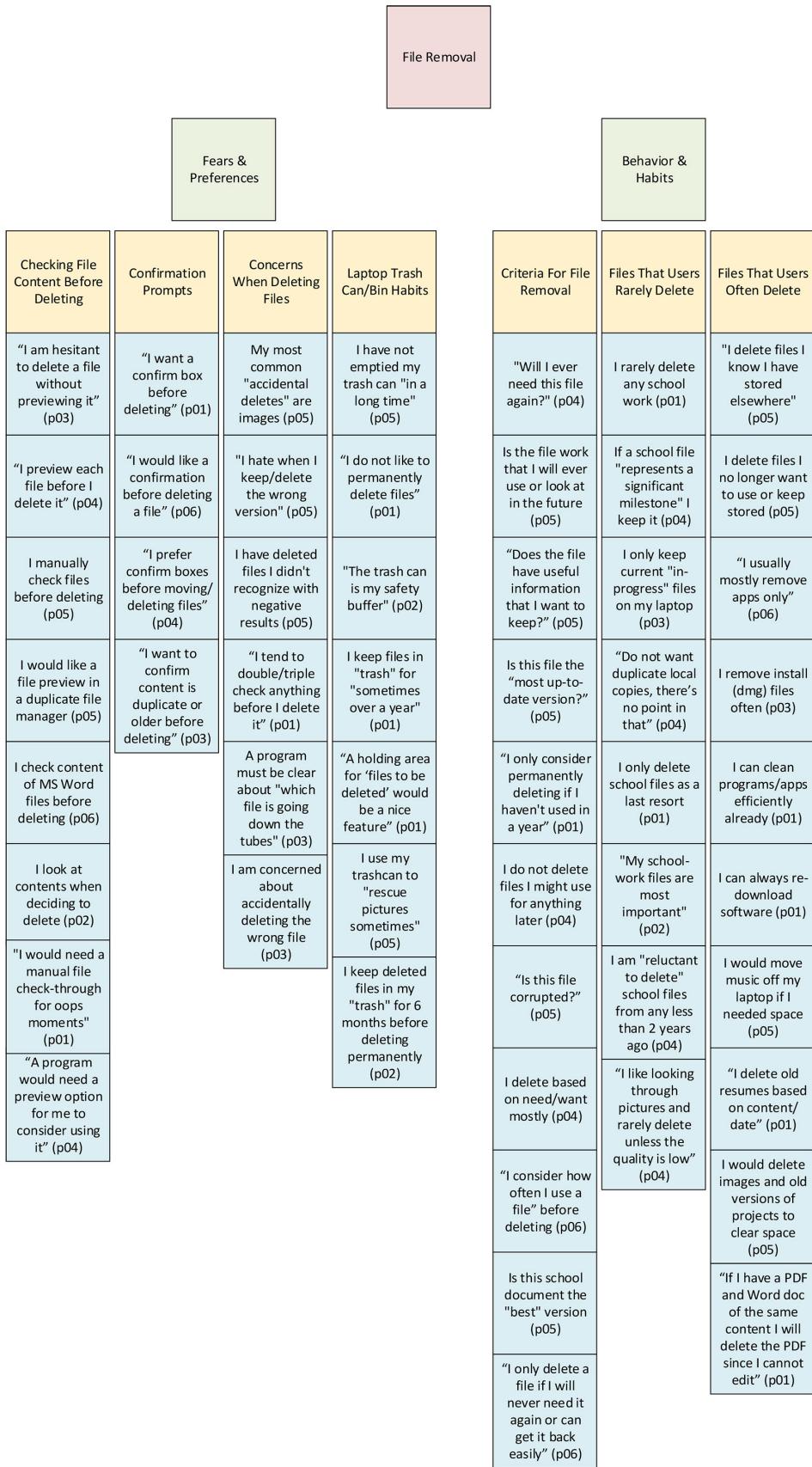


Figure 9 - Affinity Diagram "File Removal" segment

Overall the affinity diagramming helped uncover some key design considerations for the duplicate file finder program. These specific design considerations are discussed in detail below, with examples of interviewee comments to illustrate interviewee preferences, opinions, and program wants/needs. Additionally, functional requirements based on the collectively expressed needs of the interviewees are presented below.

Design Considerations

Error Prevention:

During the interviews most users indicated that they manually check their files before deleting. Overall, considering the sensitive nature of file removal, users showed a preference for mechanisms (i.e. confirm messages and file preview) that can assist with unintentional file deletion, and prevent accidental selections or deleting an unintended file.

The following interview comments further illustrate the concern for error prevention:

"I am hesitant to delete a file without previewing it" – Interviewee 03

"I would like a confirmation before deleting a file" – Interviewee 06

"A program would need a preview option for me to consider using it" – Interviewee 04

Error Recovery:

Five out of the six students interviewed stated that they would prefer a duplicate file finder program not permanently delete their duplicate files. The mechanism of simply moving selected files to the trashcan/trash bin (native to the laptop's operating system) provides a way to prevent accidental permanent removal of files and recover from accidental file removal while using the program.

The following interviewee comments illustrate the overall preference toward moving files to the trash can/trash bin (or a temporary holding area) to enhance the error recovery of the program:

"I do not like to permanently delete files" – Interviewee 01

"A holding area for 'files to be' deleted would be a nice feature" - Interviewee 01

"The trash can is my safety buffer" – Interviewee 02

External Consistency:

With a delicate task that involves removal of files, it is no surprise that interviewees showed concern about trusting the program. Some of the interviewees expressed strong opinions on how the design of a program and hurt or enhanced trust in the program. While discussing the existing software solutions (using screenshots of the interfaces), one participant (Interviewee 01) commented on an unexpected design element in a negative way: “The ‘donate’ and ‘visit website’ buttons in the Dup File Finder 2.2 hurt its credibility.”

Other interviewees made comments regarding the importance of external consistency:

“Symbols need to match other programs” – Interviewee 04

“I like familiar looking and acting software” – Interviewee 01

“I like the [Auslogics] screenshot because it looks like other Windows programs” -

Interviewee 02

Multi-Layer Interfaces:

Interviewees showed a strong preference for uncluttered, minimalist interfaces during the interviews. All interviewees commented on this preference, and some remarked that they prefer advanced features to be hidden. While some interviewees showed a preference for inclusion of advanced features, these users (specifically Interviewee 05 and Interviewee 03) also stated that they would like these features to be “hidden” in the appropriate menus.

The following comments further illustrate a preference for minimalist design:

“I like simple, Google-style layouts” – Interviewee 02

“I like clear, simple, clean interfaces” – Interviewee 06

“I like when advanced options are hidden” – Interviewee 02

Grouping:

During the interviews the screenshots of existing duplicate file finders (examples shown above in Current Products) were used to evoke opinions from interviewees. This portion of the interview was especially useful in gathering information about interviewee aesthetic preferences. A common preference of those interviewees was to group the duplicates together to make file selection and removal more stream-lined.

The following interview comments demonstrate how interviewees prefer duplicates to be displayed:

“I like the striped grouping [in the Auslogics screenshot]” – Interviewee 05

“I like the striped rows [in the Auslogics example]” – Interviewee 02

“I want to see all my files with duplicates highlighted” – Interviewee 01

Functional Requirements

File Preview

As mentioned above in the “Error Prevention” section, users showed a strong preference for a mechanism to preview files before selecting them for removal. Some interviewees even went as far as to say that they would only consider using the duplicate file finder software if a file preview feature was included. Since each of the six interviewees indicated that they manually check file content before deleting duplicate files, the file preview feature is included in the prototype for documents and images (the file types that interviewees indicated are accidentally deleted the most).

Results Sorting

Another unanimous preference among the six interviewees is sorting functionality. Users shared about their current duplicate file management practices with all six using some form of sorting during the process. The common file attributes that interviewees stated they sort by are: type, name, size, and last modified/created dates.

The following interviewee comments illustrate how the interviewees use sorting to eliminate unwanted duplicate files:

“To clean up I sort by file name sometimes” – Interviewee 01

“I want a program that sorts files by Date Last Accessed and/or Downloaded” –

Interviewee 06

“If I need to clear space I sort by file size” – Interviewee 02

Consolidated Sequence Diagram

The following consolidated sequence model (shown in Figure 10) illustrates the general process that the interviewees indicated they engage in while managing duplicate files. The diagram shows the steps and problems that the interviewees stated they encounter during the process.

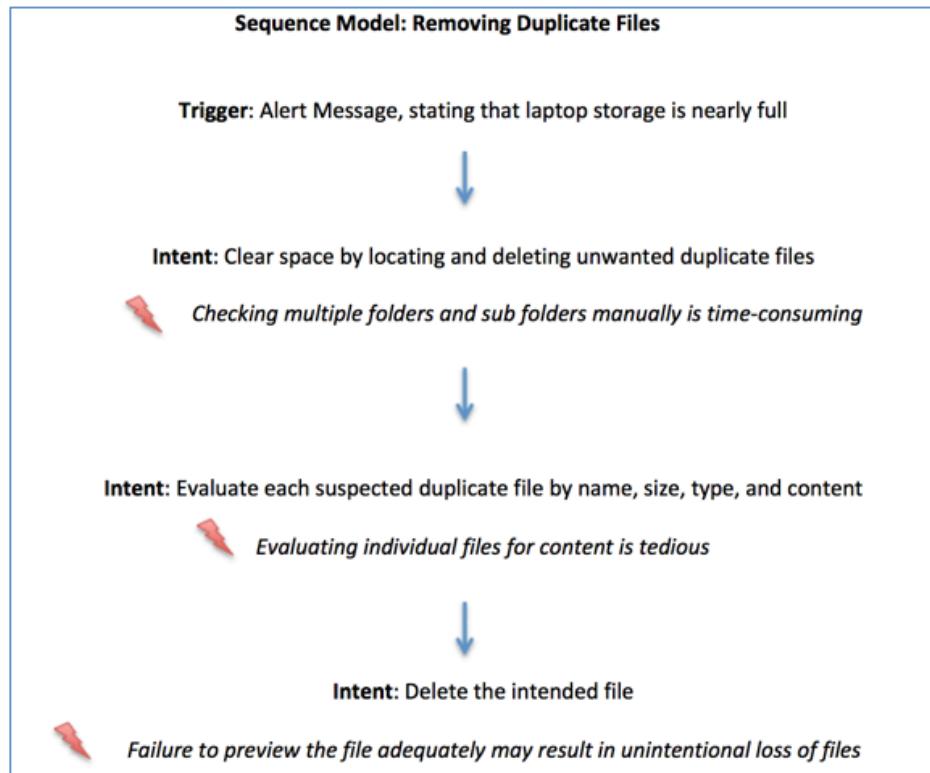


Figure 10 - Consolidated sequence diagram, based on interview data about duplicate file removal processes

Overall this sequence model demonstrates three areas in which process improvement needed to be addressed by the duplicate file finder design. More specifically, the processes that can be improved include: locating duplicate files based on content matching, visually evaluating the content of each file to determine need/want of the file, and removal of the intended files.

Prototype Design

The overall intent for the prototype was to incorporate design features and functional abilities that satisfy the needs, preferences, and desired workflow that the interviewees expressed. Ultimately, in developing the prototype, the data gathered

during the interviews served as the primary guide for both design choices and functionality.

The following design features and functional mechanisms were included in the prototype, and are based on the consolidated interview data:

Error Prevention

- “Scan Directory for Duplicates” button disabled (Figure 11a) prior to user selecting a directory and enabled on directory select (Figure 11b)
- “Delete Selected Files” button disabled until scan has completed
- Confirm message dialog provided before removing files (Figure 12)

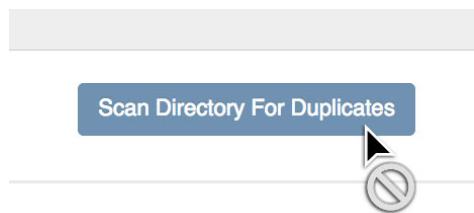


Figure 11a - Disabled interface button



Figure 11b - Enabled interface button

Error Recovery

- Files are not permanently deleted, and instead moved to the trashcan (Figure 12)

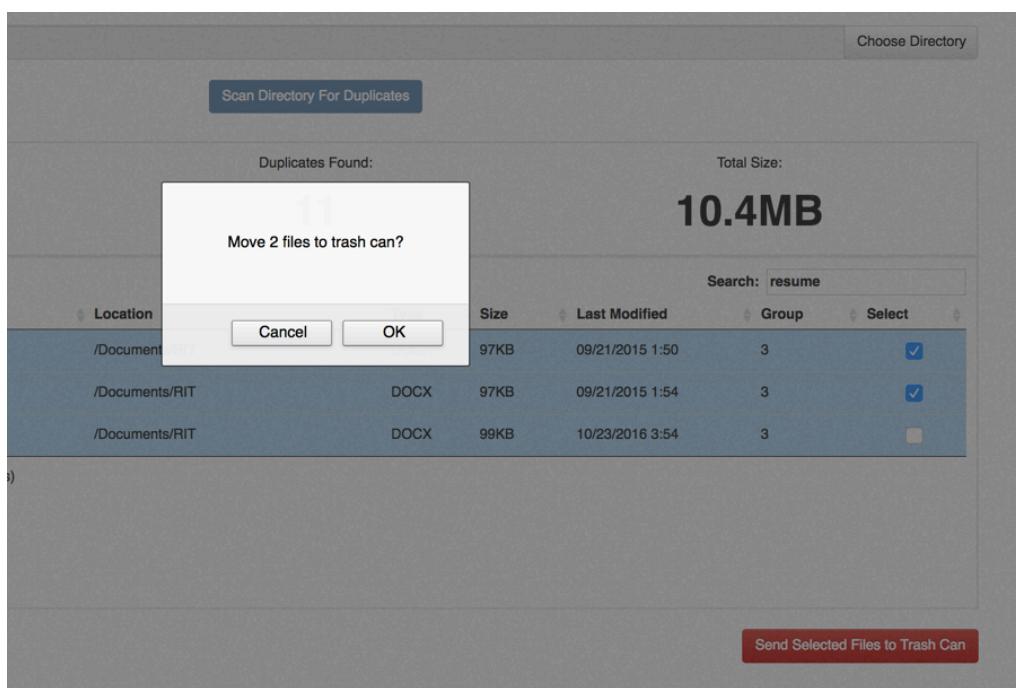


Figure 12 - Files are moved to the trashcan or trash bin instead of being permanently deleted

External Consistency

- Modern minimalist styling of buttons, search bar, and results table (Figure 13)
- Windows-style top toolbar with “File”, “Options”, and “Help”
- “Choose Directory” file selector consistent with common programs

The screenshot shows a software application window titled 'RIT'. At the top, there's a menu bar with 'File', 'Options', and 'Help' and a toolbar with 'Scan Directory For Duplicates' and 'Choose Directory'. Below the toolbar, the text 'RIT' is displayed. The main area has three summary statistics: 'Files Scanned: 126', 'Duplicates Found: 11', and 'Total Size: 10.4MB'. A search bar labeled 'Search:' is present. The central part of the interface is a table with the following columns: Preview, Name, Location, Type, Size, Last Modified, Group, and Select. The table contains six rows of data, each with a preview thumbnail and file details. The rows are color-coded by group: purple, green, blue, purple, green, and blue. A note at the bottom says 'Showing 1 to 11 of 11 entries'. At the bottom right is a red button labeled 'Send Selected Files to Trash Can'.

Preview	Name	Location	Type	Size	Last Modified	Group	Select
Preview	IMG_4631_copy.JPG	/Documents/RIT/pics	JPEG	1961KB	10/17/2015 7:35	1	<input type="checkbox"/>
Preview	IMG_4631_copy_2.JPG	/Documents/RIT/pics	JPEG	1961KB	01/24/2016 5:21	1	<input type="checkbox"/>
Preview	IMG_2593.JPG	/Documents/RIT	JPEG	1926KB	09/21/2015 8:54	2	<input type="checkbox"/>
Preview	IMG_2593(1).JPG	/Documents/RIT	JPEG	1926KB	06/20/2015 3:54	2	<input type="checkbox"/>
Preview	resume.docx	/Documents/RIT	DOCX	97KB	09/21/2015 1:50	3	<input type="checkbox"/>
Preview	resume(1).docx	/Documents/RIT	DOCX	97KB	09/21/2015 1:54	3	<input type="checkbox"/>

Figure 13 - Modern minimalist styling for interface form elements

Grouping

- Duplicates grouped by colored rows and by group number for accessibility (Figure 14)

The screenshot shows a software application window titled 'Downloads'. At the top, there's a menu bar with 'File', 'Options', and 'Help' and a toolbar with 'Scan Directory For Duplicates' and 'Choose Directory'. Below the toolbar, the text 'Downloads' is displayed. The main area has three summary statistics: 'Files Scanned: 322', 'Duplicates Found: 9', and 'Total Size: 8.62GB'. A search bar labeled 'Search:' is present. The central part of the interface is a table with the following columns: Preview, Name, Location, Type, Size, Last Modified, Group, and Select. The table contains nine rows of data, each with a preview thumbnail and file details. The rows are color-coded by group: purple, green, blue, purple, green, blue, orange, orange, and orange. A note at the bottom says 'Showing 1 to 9 of 9 entries'. At the bottom right is a red button labeled 'Send Selected Files to Trash Can'.

Preview	Name	Location	Type	Size	Last Modified	Group	Select
N/A	StarWarsEpisodeFour.MP4	/Downloads	MP4	2400MB	12/17/2015 8:34	1	<input type="checkbox"/>
N/A	StarWarsEpisodeFour.MP4	/Downloads	MP4	2400MB	12/17/2015 7:35	1	<input type="checkbox"/>
N/A	Breaking_Bad_Season04_Episode_03.MOV	/Downloads	MOV	849MB	12/18/2015 5:21	2	<input type="checkbox"/>
N/A	Breaking_Bad_Season04_Episode_03(1).MOV	/Downloads	MOV	849MB	01/21/2016 8:54	2	<input type="checkbox"/>
N/A	Breaking_Bad_Season04_Episode_03(2).MOV	/Downloads	MOV	849MB	01/22/2015 3:54	2	<input type="checkbox"/>
N/A	itunes.exe	/Downloads	EXE	262MB	07/21/2014 1:50	3	<input type="checkbox"/>
N/A	itunes(1).exe	/Downloads	EXE	262MB	7/21/2014 1:54	3	<input type="checkbox"/>

Figure 14 - Duplicate groups indicated by coloration of rows

File Preview

- For documents and images “Preview” button is displayed to show file content in light box-style popup display. Figure 15 shows a preview for a resume file.

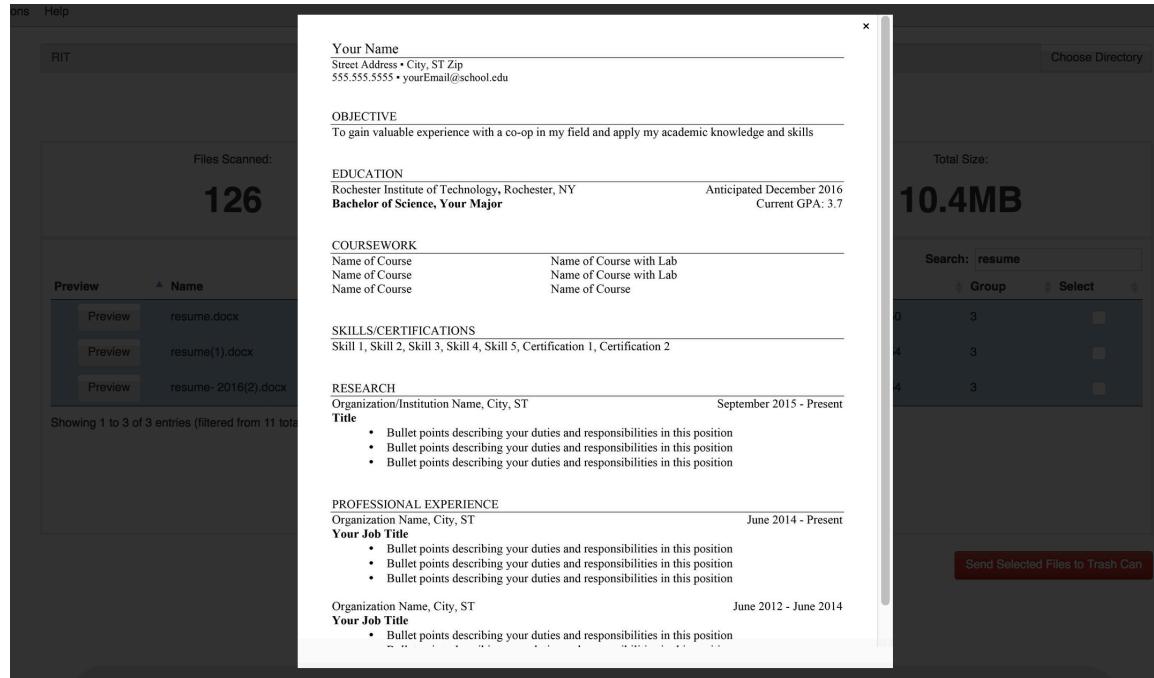


Figure 15 - File preview mode for a sample resume file

Results Sorting/Multi-Layer Interfaces

- Results table is sortable by all fields: Preview, Name, Location, Type, Size, Last Modified, and Duplicate Group (Figure 16)
- Only basic features included on primary interface
- Search and sorting features discoverable via clicking, but unobtrusive

Search: <input type="text"/>					
Type	Size	Last Modified	Group	Select	
DOCX	97KB	09/21/2015 1:54	4	<input type="checkbox"/>	
DOCX	99KB	10/23/2016 3:54	4	<input type="checkbox"/>	

Figure 16 - Sorting features for the results table

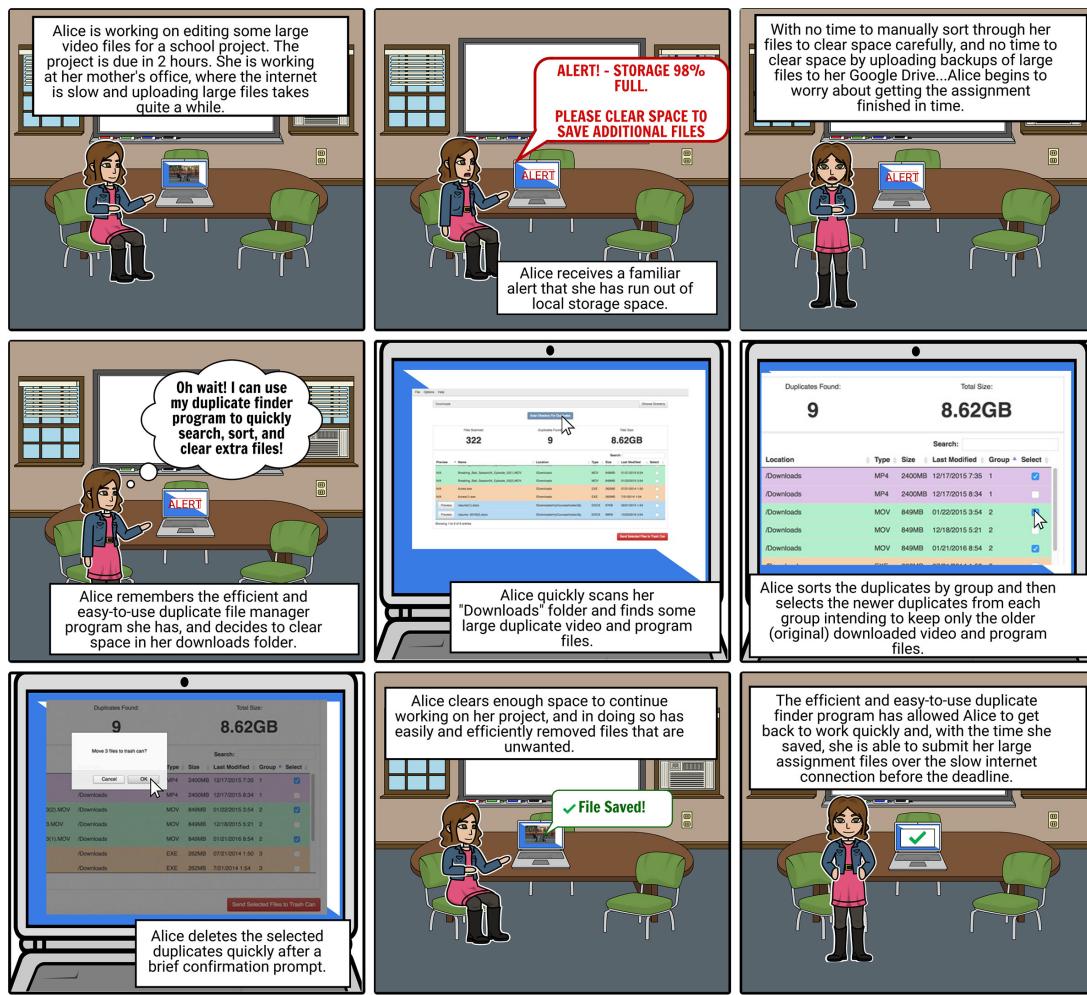
Storyboards

The following storyboards illustrate two main problem areas and how the design and functionality of the duplicate file finder software prototype can address them.

The first storyboard, shown in Figure 17, illustrates a scenario in which the user has run out of space and needs an immediate way to carefully remove unwanted duplicate files, by size and name.

In the second storyboard, shown in Figure 18, the user is faced with a file organization problem, and utilizes the duplicate file finder software to clean up files with similar content based on file age.

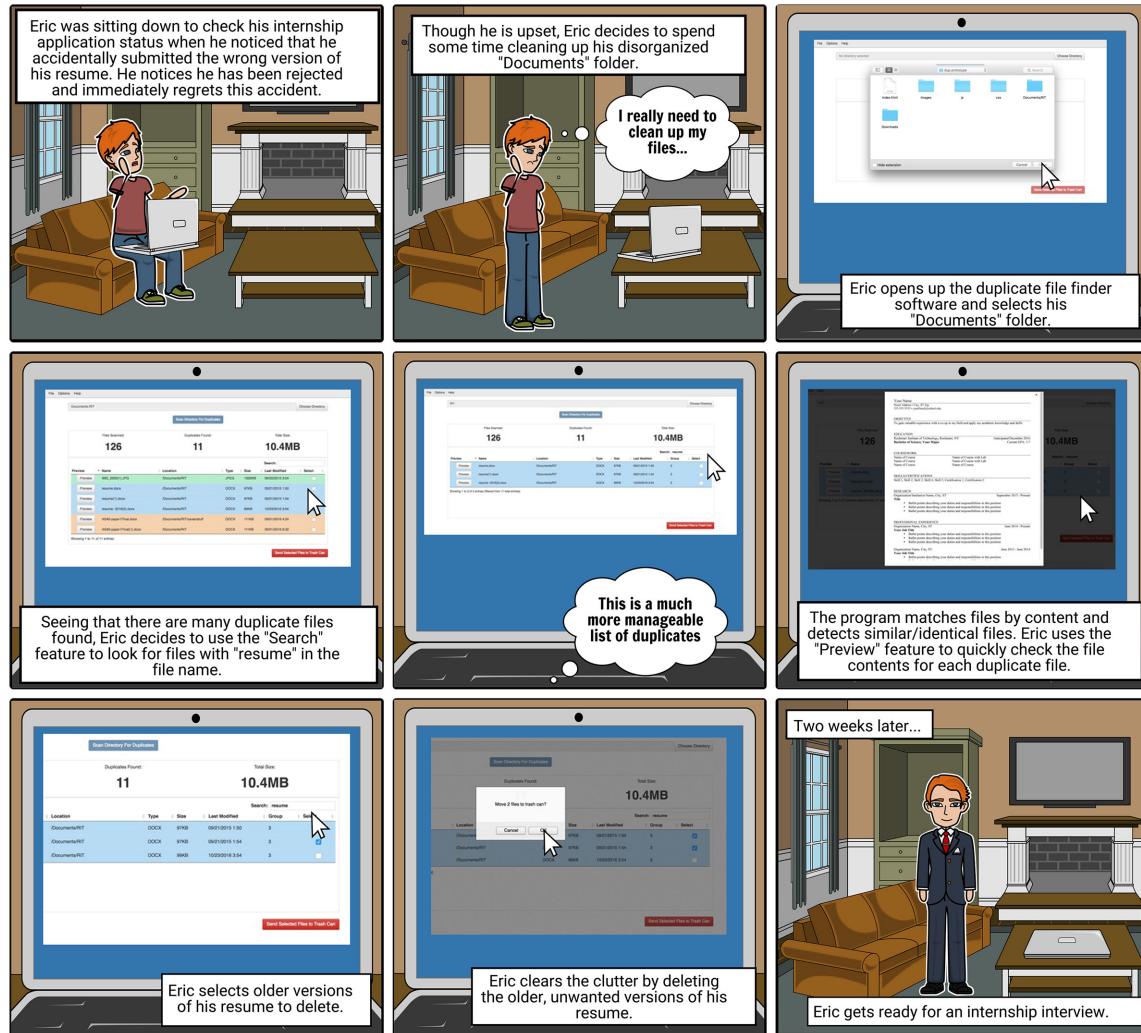
Participants described scenarios similar to these during the interviews, with one participant actually showing the “storage full” alert on her laptop during the interview.



Created with Storyboard That for amfagan

Image Attributions:
Players (<https://www.flickr.com/photos/wvs/2566384895/>) by wvs License: Attribution, Non Commercial (<http://creativecommons.org/licenses/by-nc/2.0/>)

Figure 17 - Storyboard regarding deletion of duplicate files to clear space



Created with Storyboard That for amfagan

Figure 18 - Storyboard regarding organization and “cleaning up” of files

Prototype Test Results

Usability

System Usability Scales (SUS) assessments were completed by each of the four participants. This scale was used to provide a standardized measure of usability for the prototype. Despite the fact that the prototype is only a representation of the duplicate file finder program, the functionality and interactivity closely mimic a full working program. For this reason, it was important to collect a qualitative assessment of the

prototype in order to benchmark the usability and discover any major usability issues to fix going forward.

The average score for the SUS assessment came out to be 88.75 with the following ratings for each participant:

Participant 01 – 92.5

Participant 02 – 95.0

Participant 03 – 77.5

Participant 04 – 90.0

This average SUS assessment score of 88.75 indicates that the usability score for the duplicate file finder prototype is above the average score of 68.0 [1]. Overall, these scores seem to indicate that the prototype usability is a positive aspect of it.

User Experience

Looking at participant comments often provides context and valuable qualitative information about the user's experience with the tested software. This happened to be the case for this particular project. Below responses from all four participants will be discussed, showing what users perceived to be positive and negative aspects of the duplicate file finder prototype design and functionality. The full, transcribed notes for each participant are attached in Appendix B.

Participant 01

How this user described the prototype:

“Straightforward” - “simple” - “distracting colors” - “it gets the job done”

Positive:

This individual stated that she liked how groups of duplicates were segmented and shared: “It helped me to identify that I should keep one file in each color.” She described the overall prototype as “straightforward,” and stated: “Simple is good. There is nothing to clog the screen or complicate the process.”

Negatives:

This participant indicated that she found the colored group to be somewhat distracting due to the brightness/saturation of the color, though she did immediately realize the purpose as evidenced by this quote at the start of Task 1: “Oh, okay, I guess colors mean duplicates” made approximately 1-2 seconds after clicking the scan button. This participant indicated that, since she was looking to delete all but one file in each duplicate group, that it would be nice if the program had the option to “flag the original file” and sort by date and name within duplicate groups.

Observations during the test:

This participant used the file names and sizes to select which files to remove. Specifically, in Task 1 she kept one file without “copy” or “(#)” in the file name for each of the colored duplicate groups. In Task 2 she specifically looked for “resume” in the file names and then, using preview and the “Last Modified” dates, determined that one file was a copy of the older resume version. She elected to keep the original copy of the older resume and the newest version.

This participant struggled significantly with the sorting feature. She became confused after attempting to sort by “Last Modified” date by clicking the arrows that corresponded with file size sorting. This type of error occurred with three of the four participants and illustrates a proximity issue with the headings in the prototype.

Another recommendation that this participant made relates to the text on the “scan” and “delete” buttons being too wordy: “Maybe use a trash can instead icon instead of so many words on the buttons.” Other participants commented on this as well.

Participant 02

How this user described the prototype:

“Simple” - “easy” - “clean” - “straightforward” - “colors somewhat distracting”

Positive:

This participant self-identified as “tech savvy” and completed both tasks very quickly. He shared that he regularly cleans up his files on his computer and syncs with Google Drive. He seemed to like the features and design of the prototype overall. Specifically, after scanning during Task 1, he commented: “I like that the results are clear about what the scan found.” This user also stated: “I like the interface. It is clean and simple.”

Negatives:

While this participant showed satisfaction with the prototype, there were a few areas where the participant stated that improvements could be made. For example, he stated: “There isn’t last modified at the top of each group. I would like that better” while referring to each duplicate group (and sorting within groups). Shortly after this comment he noticed that the results were sortable by clicking each heading, and was the only participant to sort by the intended attribute by clicking the heading title instead of the arrows next to it, which coincide with the heading to the left. Overall, he stated that his preference would be to have the option to sort by date within each group as opposed to only the entire results table.

Observations during the test:

Anecdotally, out of the four participants this participant seemed to “pick up” the program most quickly. He did not struggle with sorting files and seemed to like the grouping overall, but wanted the colors slightly paler. He had multiple comments that

gave some insight into changes that would improve the program. He was one of two participants to suggest removing the “scan” button altogether and, instead, having the program scan as soon as a directory is selected. As with other participants, this one also was a bit distracted by the colors and concerned at how many combinations could be used with hundreds of duplicate files. This participant’s stated that perhaps a single color striping would be useful if the sorting was based on each group and not the whole results table.

Participant 03

How this user described the prototype:

“Straightforward” - “confident I could figure it out” - “colors a little distracting”

Positive:

While this participant seemed to have more complaints than praises for the prototype, she did seem to appreciate the directory selector (“The directory selector feels natural”). She also indicated that she felt confident in being able to “figure it out” when referencing the overall program and stated “this program seems useful for others.”

Negatives:

The color grouping may have been a bit distracting for this participant, as illustrated by the following comments:

- “The purple color struck me as a little odd. It’s not a typical color for programs I use.”
- “I’m not sure what the colors mean. They are the first things I noticed. I’m guessing they mean duplicate sets?”
- “The colors seemed useful, but maybe a little distracting.”

This participant struggled with the sorting feature, and like others, clicked the sorting arrows to the left of each heading, which belong to the heading to the left. She clearly described the issue in the following quote: “The sorting arrows should be closer to what they sort since there are not borders between each word.”

Observations during the test:

This participant seemed to have a neutral outlook on the program with some skepticism on the software improving her current system and admitted hesitancy to change her system. This is illustrated in comments, such as: “I’m not eager to change my ways, but this program seems useful for others. I find it hard to trust any programs and like to manually check.” This combined with some of the usability issues that the participant encountered during the test (i.e. trouble sorting and distracting colors) might also partially explain why this participant’s SUS assessment came out lower (77.5) than the other participants (90.0, 92.5, 95.0) and overall average for all participants (88.75).

Participant 04

How this user described the prototype:

“Not complex at all” - “easy” - “straightforward”

Positive:

This participant initially commented regarding liking that he knew he had to select the directory first due to the “scan” button being disabled. Though he liked this, he commented that he would like the button removed and to: “Just immediately search the directory on select” in order to streamline the process even further. He also commented that the button text clarity is a positive and “Tells you exactly what they do.” In addition this user spoke positively about the results display, search, sorting, color grouping, and file preview:

- “I like the results at the top. They are helpful.”
- “Sort and search are nice features.”
- “I like the breaking up by color. It shows where each set ends nicely.”
- “I like the file preview for a quick look.”

Negatives:

Like other participants, this one also struggled a bit with the sorting arrow icons, but did not use them until after the tasks, while playing around with the interface. He stated that a border between headings or icons closer to the heading titles would fix the problem, but also wanted to sort by data within each duplicate group, as opposed to with all files.

Observations during the test:

This participant seemed to have a higher tolerance for accidentally deleting the wrong file than the other participants, and he zipped through the tasks seemingly with ease. He brought up a good point in that the software would not be something he would use constantly, but more likely just once in a while to clean up his computer. This participant also wanted and option to “Select All” and/or “Clear All” duplicates from a given directory: “I would like an option to Select All and nuke everything in a folder.”

Conclusions

Overall, there were two distinct reasons why participants stated that they spent time removing duplicate files:

1. Regular cleanup/organization
2. Running out of storage space.

For the interviewees, the majority of their duplicate files are in their “Downloads” and “Documents” folders.

1. How do users currently manage duplicate files and limited local storage?

While the interviewees revealed various methods of managing non-duplicate files (i.e. uploading to cloud-based drives and external storage devices), most interviewees reported that they did not have a satisfactory method of managing duplicate files without tedious searching and manual checking of files.

The most common solution for clearing space was to sort downloaded, video, music, and install files by size and remove those that they could recover easily, or those, which they had multiple copies of. Many interviewees reported that their “Downloads” folder typically contains the largest number of duplicate files. A program that scans recursively combined with a sorting feature would address the issue of analyzing content across multiple sub-directories and sorting by file size.

2. What does the process of “cleaning up” one’s laptop computer entail?

In order to clean up and organize their files many interviewees disclosed that they would manually check through many directories and files, opening each one, in order to evaluate the content and then delete or rename the file. With recursive directory searching, duplicate matching based on file content, and a visual preview feature this file finder software would improve the current process for these interviewees by reducing the time to complete the “clean up” process.

3. What is important to users when deciding whether or not to delete a duplicate file?

Interviewees shared a number of different factors that go into the decision to remove a duplicate file, including: the age of the file, whether the file can be recovered elsewhere, if the file is school work, how much work the file involved, the version of the file, and if the file is wanted in general. Since users analyze the value of a file in different ways, content-based analysis, sorting, searching, and preview tools were included in the prototype. As one participant stated: “This program gives you all the tools to determine if a file is actually a duplicate and/or worth keeping.” This is critical for this type of

program since some participants seemed hesitant to abandon the flexibility of their current file management methods.

4. How do users rate the usability of the duplicate file finder prototype?

As mentioned in the results section, participants gave the prototype an above-average (68.0) SUS score of 88.75. In general, none of the participants failed to complete either of the two tasks, and each participant only found minor parts of the prototype to be unsatisfactory. None of the usability issues prevented the participants from completing the tasks.

5. What do users find confusing or frustrating about the prototype design and how can the design be improved?

There were a few items about the prototype design that were prevalent issues among participants. These issues and possible remedies include the following:

- For some users sorting may be useful with all files, but most participants wanted the option to sort files (by file name, size, and date last modified) within each duplicate group. The option to sort within duplicate groups should be added.
- It was also clear from the prototype testing that participants struggled with the lack of borders between the results table headings. This lead to accidentally selecting the wrong set of sorting arrows and could be fixed by adjusting the proximity of the arrows to be closer to the column heading that they sort and/or adding borders between each heading title.
- While all participants reported satisfaction with the duplicates being grouped, tweaks in the coloration of the groups were also desired to avoid distraction and confusion. Perhaps, as one user suggested, the colors could be made more pale or turned off and a single color striping used to differentiate duplicate groups. The single color striping would make the most sense while sorting within groups.
- While there were some conflicting opinions on these, two other changes could be: reducing the button text length/using icons, and removing the “scan” button altogether to have the program search when the directory is selected.

6. What are the positive aspects of the prototype design?

Overall participants seemed to enjoy the simplicity and modern-looking minimalistic interface of the prototype. They frequently referred to the prototype as straightforward, simple, and easy. Some of the notable positive responses were regarding things like the search feature, duplicate grouping, confirmation dialog, file preview, signifier clarity, and overall external consistency of the prototype. This would likely be a program that is used “once in a while” to clean up one’s computer. Fortunately, the prototype test results seem to indicate that the program is intuitive,

which is especially important considering the long times between when users might use it to clean up duplicate files.

Future Work

The next logical step in this topic area would be to investigate ways to optimize the speed and technical aspects of scanning directories for duplicate files in order to develop a fully functional software program. Many of the participants in this study commented that the prototype was quick and easy to use. The prototype type simply loaded a small amount of JSON data, rather than scanning multiple directories of files, and used JavaScript timing to simulate a short loading period. While this does somewhat simulate a brief scan, it will be important moving forward to investigate ways to ensure this process is quick and the appropriate feedback (i.e. progress bars, messages) are used to maintain user satisfaction.

Challenges & Limitations

Barriers & Challenges

In developing the prototype it was challenging to develop a set of fictional, but relevant, files to use during the prototype test. Since modifying a participant files, as part of the prototype test would introduce a number of new variables and risks, this was not considered as a viable option. This made it somewhat difficult to make the testing personally relevant to the user, as generic test file data was used. Perhaps there is a way to incorporate actual participant files, but this seems to be one of the barriers or conundrums in this type of study.

Extracting requirements, needs, and ultimately consolidating these into a design or to test is a challenging task. There are many diverse needs for students based on personal preference, major, and other factors. A challenging aspect of designing this type of software is determining which features are most relevant for the main interface, and which can be included in deeper layers within the program.

Limitations

Perhaps the most considerable limitation was the number of participants. With six interviewees for the iterative design process and four participants in the prototype testing this project involved actual users, but with a small number, will not represent a population and therefore lacks the power to make any specific research claims.

References

1. Brooke, John. "SUS-A quick and dirty usability scale." *Usability evaluation in industry* 189.194 (1996): 4-7.
2. Fitchett, S., Cockburn, A., and Gutwin, C. 2013. Improving navigation-based file retrieval. *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '13)*. ACM, New York, NY, USA, 2329-2338.
DOI=<http://dx.doi.org/10.1145/2470654.2481323>
3. Marshall, C. and John Tang, J. C. 2012. That syncing feeling: early user experiences with the cloud. *Proceedings of the Designing Interactive Systems Conference (DIS '12)*. ACM, New York, NY, USA, 544-553. DOI=<http://dx.doi.org/10.1145/2317956.2318038>
4. Shneiderman, B. 2002. Promoting universal usability with multi-layer interface design. *Proceedings of the 2003 conference on Universal usability (CUU '03)*. ACM, New York, NY, USA, 1-8. DOI=<http://dx.doi.org/10.1145/957205.957206>

Appendix A - SUS Instrument

System Usability Scale

© Digital Equipment Corporation, 1986.

	Strongly disagree					Strongly agree				
1. I think that I would like to use this system frequently	<input type="checkbox"/>									
2. I found the system unnecessarily complex	<input type="checkbox"/>									
3. I thought the system was easy to use	<input type="checkbox"/>									
4. I think that I would need the support of a technical person to be able to use this system	<input type="checkbox"/>									
5. I found the various functions in this system were well integrated	<input type="checkbox"/>									
6. I thought there was too much inconsistency in this system	<input type="checkbox"/>									
7. I would imagine that most people would learn to use this system very quickly	<input type="checkbox"/>									
8. I found the system very cumbersome to use	<input type="checkbox"/>									
9. I felt very confident using the system	<input type="checkbox"/>									
10. I needed to learn a lot of things before I could get going with this system	<input type="checkbox"/>									

APPENDIX B - Prototype Test Notes

Participant-01

Usability Test Notes

Scenario 1 – selected “Choose Directory” immediately and clicked “scan”

- Commented “the directory is selected” because “it showed up in the box” (text field)
- Commented that they would like an option to clear the directory select/results before and after searching
- “Oh, okay, I guess colors mean duplicates” – after noticing group numbers then commented that these “should be moved to the left so they are spotted first”
- Stated she would prefer to see “sets of duplicates, as opposed to group numbers”
 - Elaborated: maybe symbols instead of numbers would help make them easier to group together because “the colors might be distracting”
- Commented that “maybe there could be an option to flag the original file. That’s basically the only feature that would make this process faster, but overall it’s very straightforward”
- Likes grouping in general, “helped me to identify that I should keep 1 in each color”
- Looked at file names and decided to remove files with “(#)” since the downloads folder was downloaded and “I trust the first copy the most”
- “I am looking to keep the originals”
- Eventually clicked on arrows next to “Last Modified” – attempting to sort by date and spent a while trying to figure out what had happened (Had clicked sorting arrows to sort files by size)
 - Poor proximity on arrows, and poor visibility due to time to discover
- Noticed the difference in names of resume, “I will check the preview since these have different names”
- Decided to keep older and newer copies of resume and delete duplicate of older copy because “these are small enough files and different, so I will keep both”

Scenario 2 – selected directory immediately again and clicked “scan”

- “I would like to see number of duplicates AND number of groups of duplicates in the results”
- Stated that resume deleting is “more scary” than clearing movies to make space
- “The same size, it’s probably fine to delete one, but I want to preview anyways. You never know.”
- “I like that the files go to trash so I have another chance to save them or double check.”
- Noted that if size and name are the same, they are probably the same
- “Maybe a renaming feature within the program would be nice”

- “I would be hesitant to delete a file with a different name and larger size, but files with (#) are easier to identify as true duplicates”
- “I can delete the (#) files without thinking”
- “resume – original, resume(1) – duplicate, resume-2016(2) – totally different”

During SUS

1. “Duplicate files and storage are current issues of mine, however I would likely only use this for folders like my Downloads and not on resumes and school files”
2. “It’s not complex at all” “It gets the job done” “Simple is a good thing”
3. “Simple is good. There is nothing to clog the screen or complicate it process”
4. “Might be tougher for older person, but not more difficult than any other app”
5. “The arrows for sorting should be closer to the word/thing they sort”
6. “I’m interested to see how many colors it would have for hundreds of duplicates” “Maybe use trash can instead icon instead of so many words on the buttons”
7. Might be difficult for elderly to notice “little things like sorting”
8. “The software is simple and does what it needs to”
9. Where there are 2 files of different name or size it is a little confusing (prototype simulates content matching – maybe show percentage?)
10. “This program is good at allowing users to make a well informed decision about files, and no matter what the person will need to take some time to sort, but this helps.”

Participant-02

Usability Test Notes

Scenario 1 – selected “Choose Directory” immediately and stated that it “is consistent with other apps that he uses” – clicked “scan”

- First noticed number of files/duplicates/size at top of results section – commented on this positively: “I like that the results are clear about what the scan found”
- “I am assuming color coded files are the same”
- “The colors make it easy to spot and delete the same files (duplicates)”
- “There isn’t last modified at the top of each group. I would like that better.” – has not noticed sorting options
- Discovers sorting is possible by clicking on “Last Modified”
- Talked about colors and being “somewhat distracting” stated that he would like to see them be “lighter and fade into background”
- Talked about number of duplicates and having enough colors. Discussed having them striped, but gray scale like in some tables
- “An option to delete all but most recent duplicates would be cool”
- Allow vertical resizing on list of duplicate files to reduce scrolling

- “I like the interface. It is clean and simple”
- “The color coding could be less distracting”
- “I like the delete confirmation”

Scenario 2 – selected directory immediately again and clicks “scan”, seems faster

- “Scan button is not needed. I would like it to scan as soon as I select a directory”
- Option - For each file group sort by last modified
- Seems familiar with interface after first task
- Noticeably quicker at task
- Sorts by Last modified and deletes older files
- Did not use preview, but stated it could be helpful with school documents
- Stated that preview “works well,” but needs file name on top of document or image to state which user is looking at
 - Option to compare files directly next to each other would be useful for this participant

During SUS

1. “I would use since I mirror my local files to a Google Drive”
2. Simple, easy, clean, straightforward
3. Okay with button text, but “Move to Trash” or “Delete Files” would “make buttons smaller”
4. Easy
5. N/A
6. “Scan button adds extra step.” “Button text could be shorter”
7. “It’s easy, straightforward, and each thing says what it does, except the sorting needs a tweak”
9. “There is no question about files being deleted” – easy to tell what is being deleted
10. “It is very easy to use”

“I like that it looks like other programs”

Participant-03

Usability Test Notes

Scenario 1 – selected “Choose Directory” immediately, clicked “scan” after selecting

- Noted that file chooser is “like other programs”
- “I’m not sure what the colors mean. They are the first things I noticed. I’m guessing they mean duplicate sets?”
- “The purple color struck me as a little odd. It’s not a typical color for programs I use.”
- Starts reading file names immediately after talking about colors
- States that she normally checks by name and then deletes older files, then continues to check dates of each file group

- Clicks size ordering when trying to “sort by date” – seemed confused by result. Not sure what sorting had done
- Noticed group numbers much later than colors “I was drawn to the colors immediately”
- “The colors seemed useful, but maybe a little distracting”
- “Colors a little confusing at first”
- “Nothing else really stood out right off the bat”
- “The sorting arrows should be closer to what they sort since there are not borders between each”

Scenario 2 – selected directory immediately again and “scan” clicked, quicker during this scenario

- “The directory selector feels natural”
- Begins to scan looking for the name resume, then notices and comments on “Search” bar. “I guess I could have just used the search tool”
- Uses the search to type in “resume”
- 3 results show
- Sorts by last modified
- “If there were more files, I would preview them.”
- Comments on delete button: “I don’t see anything wrong with extra clarity, but maybe reduce the number of words on the scan and delete buttons.”
- Recommends “Scan” and “Delete Selected”

During SUS

1. “I’m not eager to change my ways, but this program seems useful for others. I find it hard to trust any programs and like to manually check.”
 3. “The sorting threw me off”
 6. “I noticed a typo in the dates”
 7. “My grandma might have a tough time, but no worst than any other program”
 9. “I felt confident that I could figure it out”
 10. Described as “straightforward”
- Would like to see filename on preview, and be able to scroll via arrows through duplicates to double-check (especially with images)
 - “Highlighting differences in files previews could be useful, if possible.”

Participant-04

Usability Test Notes

Scenario 1 – selected “Choose Directory” immediately

- Commented that he likes the “scan” button being disabled
- Since “Scan” disabled knew to select directory first
- “I would like to see the folder scanned on select instead of clicking scan.”
- “Just immediately search the directory on select”

- “I would like an option to Select All and nuke everything in a folder”
 - Usually downloads folder can re-download or have backed up somewhere else
- “I like the breaking up by color. It shows where each set ends nicely.”
- “Purple/pink threw me off a little, but overall striping works well.”
- “I like the file preview for a quick look.”
- “I would like to see last modified for each section of duplicates be a feature”

Scenario 2 – selected directory immediately again, faster this time

- Like clarity of buttons, “Tells you exactly what they do”
- Would like option to simply delete older files for each set of duplicates
- Used search feature to search for “resume”

During SUS

1. “Not something I would *need* to use often, but would once in a while”
2. “Not complex at all, easy, straightforward”
5. “I like the results at the top. They are helpful”, “Sort and search are nice features”

“I like the confirmation boxes when deleting”