User Interfaces - Coursework 3

Members

- Benjamin Bellavia el19bb
- Kirsty Hammond ll17k6h
- Sreegandh Devireddy sc19s2d
- Radu-Alexandru Giuclea sc19r2ag
- Thomas Leeds sc19tewl
- Ugochukwu Emeka Okeke sc19ujeo

Table of Contents

Market Segment

“Outdoor enthusiasts who want to explore and organise a large personal video library. This group collects massive quantities of video from action cameras (e.g. GoPro), video drones (e.g. DJI Mavic), and mobile phones (e.g. slow-motion footage from an iPhone). The video is typically from a variety of locations, is of different lengths (ranging from a "moving photo" of a few seconds to 3 hours of raw skiing footage), and activities (e.g. cycling, parascending, football, or skiing).”

PACT Analysis

Person

Primary users

The primary users of this software will be the outdoor enthusiasts who wish to aggregate their collected video into one library. They will want their videos to be organized in a way that makes it easy to locate specific footage or easy to browse through collected footage. It is not assured that the user will be particularly tech-savvy, so the UI should be simple and intuitive, with the option of additional capabilities for more advanced users (in toolbars for example). Therefore the program must provide options to categorize and group the imported footage via different metrics.

Secondary users

The secondary users of this software will likely be people who are not outdoor enthusiasts, but still wish to categorize large amounts of footage, e.g. filmmakers or hobbyists. This differs from the primary user group as the secondary users will likely require more in depth options for organizing footage, and perhaps limited video editing/exporting capabilities from the software, so footage can be prepared to be processed. These users will also likely want integration with other software or platforms they may want to use, for example video editing software, or being able to directly post videos.

Activity

Potential Activities

Browsing videos (UI for both large and small amounts of files)
Tagging or organizing videos into different categories (Manually)
Locating specific footage in the library
Tagging or organizing videos into different categories (Automatically)
Editing or preparing video for other uses
Posting or sharing video to social media

Temporal Aspects

The software should be designed to support infrequent use, with a low UI memory burden. Frequent, simple activities should be prioritised on the interface, with more complex/less frequent options relegated to menus or toolbars.

Cooperation

The software should support integration of sharing and export options, e.g. to allow other users to import a collection, or to export to a particular social media format.

Complexity

The software should be designed primarily for users with limited computer experience, and so the UI should be simple and intuitive to use, minimising the learning curve required to use the software effectively. The software will provide a suite of tools for the user to choose from, rather than a step-by-step process.

Safety-critical

The software should ensure a non-destructive workflow, by not overwriting or editing the users' original files without clearly making sure that is what the user wants. Furthermore, there should be frequent autosaves of edited media, as well as an "undo" feature to prevent destructive operations. Finally the software should not store or share any of the user's personal information without their permission.

Nature of content

The software should support multiple types of media, and be flexible in organizing files of different types. The software should allow the user to manipulate or edit this content, via a variety of tools.

Context

Target platform: Desktop

Large personal video libraries often require large storage devices, such as hard disks or cloud storage, and so are rarely stored locally on mobile devices.
While a mobile application would be the most appropriate for media captured on a mobile phone, this fails to include other sources such as drones or action cameras, which often require their data to be exported physically (e.g. by SD card or cables).
Transferring this data to a desktop application is often much easier than to a mobile one, due to the technological trend of minimising ports on mobile devices. Consequently a desktop application is likely to be the most convenient for the user.

Physical Environment

As a desktop application, the physical environment is most likely to be indoors, with direct access to the user's personal video library.

Social Environment

For use with a single user. Could potentially provide options for social media and sharing of the content (e.g. aspect ratios, formats, ect)

Technology

Hardware Inputs

The software should be designed for use with a keyboard and mouse or trackpad. To ensure a consistent user experience for laptop users, the software should minimise use of the numpad, and avoid the need for complex trackpad inputs (e.g. long distance dragging).

Hardware Outputs

The software should be designed for use with a screen. This should take into account a range of landscape display sizes from small laptops to large monitors.

Software Inputs

The software should be compatible with a wide range of video and image formats, as well as be able to read and interpret the relevant metadata.

Software Outputs

As a possible feature, the software could export data to an organised set of folders, or edit the metadata to include tags.

Processing

The software must be able to organise the photos by different metrics such as date, time or location. This requires the ability for the users to interact with the relevant metadata. Potentially more complex image recognition techniques could be explored, e.g. to group photos/videos by faces, recognise selfies, etc.

Persona

Ethan is a 30-year old ski instructor who is ambitious in achieving his goals. He is capable of using a computer, but is not very comfortable with doing so. He is motivated by constantly surpassing his own limits, seeking thrills both on and off the slopes, and impressing his friends with his abilities and courage. His core frustrations are not having enough money to travel to skiing hotspots that are far away, non-skiers criticizing his technique, and people who spend a lot of time glued to their phones rather than living in the moment.He enjoys having a drink with friends after a long day on the slopes, but often returns home early so he can spend time with his wife and their two cats. He likes skiing on uncharted routes and recording the runs on his GoPro, and sometimes asks his students to film him doing jumps. Despite generally avoiding spending time on his laptop, he uses it to store his skiing video, and upload compilations to his facebook page a couple times a month. He likes watching action movies on DVD, but otherwise prefers a good book to watching TV.He prefers renting high quality equipment despite higher prices, and gets his information on new equipment or routes to try out from the friends he's skiing with. According to him, his biggest achievement was "pulling off a backflip on camera at a ski ramp in the Alps".

Scenarios

1)

As a semi-professional, Ethan skis or snowboards on uncharted routes a couple times each week, using a Go-Pro attached to his helmet to record the experiences. Every Saturday evening, Ethan connects his camera to his personal laptop to transfer the media recorded during the week. Once the transfer is finished, he opens Tomeo and imports the newly created folder in the application. Ethan is not very experienced with computers, but he learned to use Tomeo quickly and it has been invaluable in organizing his ever-growing media library. He uses a feature located on the application's main screen to label each media item with the activity depicted in its content, then sorts them by activity and by date, and arranges them into folders within the application's drag-and-drop folder hierarchy editor. He exports the folders created this way to his central media folder by simply clicking a button. The entire interaction lasts no longer than 15 minutes, after which Ethan happily shuts down his laptop and goes to the kitchen, to help his wife with dinner.

2)

After finding out from a friend about a new online skiing trend called the "360 challenge", Ethan decides it's an ideal opportunity to demonstrate his skills to his friend network. He knows he has plenty of videos of him doing the stunt required by the challenge, so later that day he opens Tomeo on his laptop and imports his central media folder into the application with two clicks. He then clicks the filters option, and adds the labels "stunt", "360", and "skiing". Within a couple seconds, he is presented with a list of 4 videos matching those labels. Ethan watches each of them, and sees a stunt he particularly likes in one of them. Since it's a longer video that includes not only that stunt, but also some performed by some of his students, he uses the "Trim" function to change the beginning and end time of the video. He knows this will only change the application's version of his video, and that his file is in fact still intact. To make the video look even better, he sets it to play in slow motion by setting its playback speed to 0.5. Once he's happy with the video he's seeing, Ethan easily locates the "Export to Social Media" option and selects Facebook as the platform to post on. This takes him to his web browser, where he only needs to input a description, then click Post. Wishing to see his close friends' reactions face to face first, he closes his laptop, not awaiting any notifications.

Development Cycles:

1) Basic layout and functionality

Prototype

Cycle goals

The main goal of this cycle is to reorganize the Tomeo application provided in order to give the user more control over the application status. In particular, we will focus on designing the main layout, adding functional video controls, and displaying the available media in a more intuitive, logically organized manner. The cycle doesn't aim to add great detail, but rather establish the central design and layout components of the interface.

Prototyping Technique

The low-fidelity paper sketch prototyping technique has been chosen for this cycle.

Technique motivation

A paper sketch prototype was deemed appropriate for the first design cycle, as it is easy to change and improve upon, and helps set a general direction for the design process.
Firstly, since a sketch prototype requires little time to produce or change, emotional attachment to the output of prototyping is low, encouraging criticism and improvement. This would help the team have an open and productive discussion, analyzing whether the proposed design matches the requirements established in the PACT analysis, and what could be done to improve the interface.

Secondly, the sketch prototyping method is beneficial since it makes producing a detailed design difficult, thus ensuring the team's focus remains on the main aspects of layout. Starting the process at a high level of abstraction, then adding details progressively makes it easier to ensure a proposed layout is feasible and likely to succeed before committing large amounts of resources to developing it.

Moreover, the ease of producing a sketch prototype has the added benefit of being able to explore multiple design alternatives, and choose the one that best suits our requirements. Brainstorming should be done in the first cycle, to make sure the best ideas will be the ones being improved upon in the following cycles. In that regard, the team's intention is to have each member produce their own sketch, then select for evaluation the one that best matches requirements and the cycle's goals.

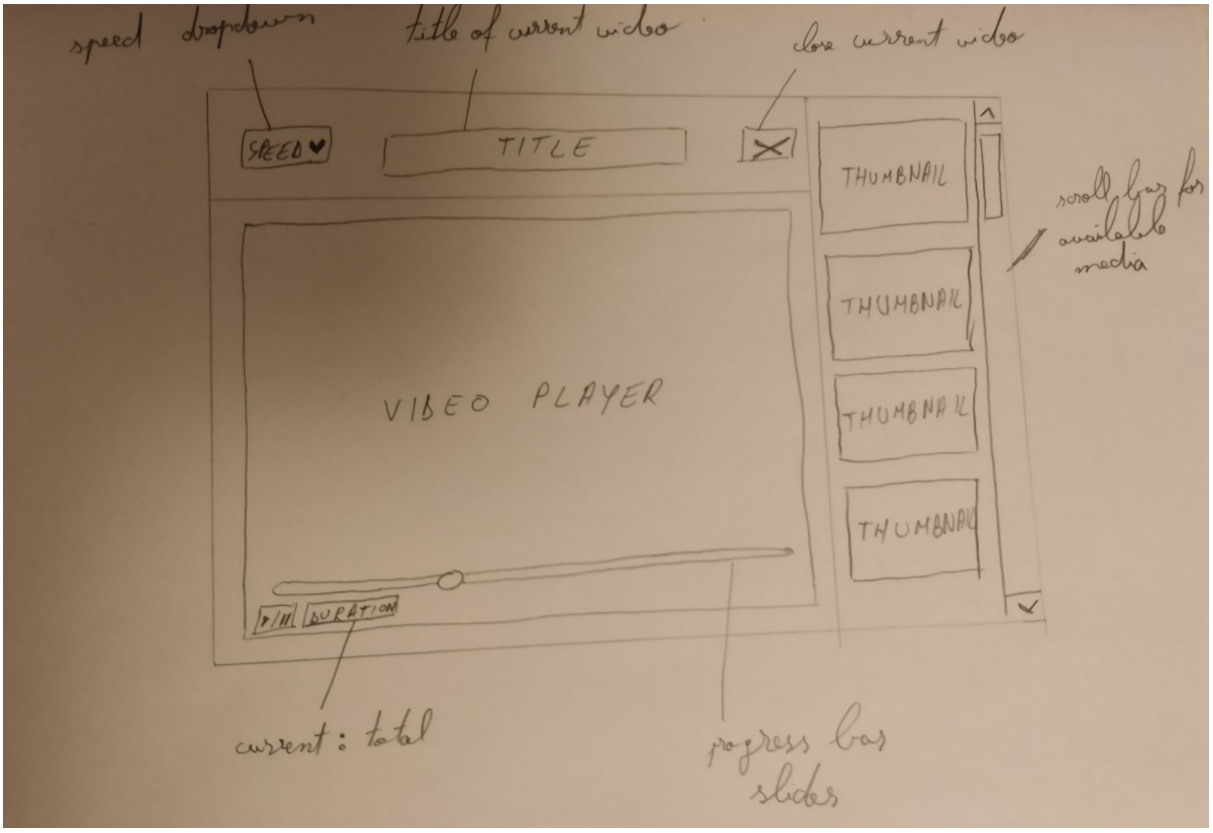
Design motivation

The prototype's design reflects the cycle's goals by identifying and laying out the central elements of the interface. The video player has been identified as the central element of the interface, and thus takes up the most space on the screen, and is located on the left side, where most western users focus their attention when first opening an application. Video controls are placed at the bottom of the video player as is common in other media software, and enable the user to play or pause the media, observe its current progress, and navigate it using the WYSIWYG method of dragging a slider. In the upper part of the main screen, the user has the option to set the playback speed of the currently playing video, observe its title, or close the video that's currently open in the player. The available media not currently playing is displayed on the right side of the screen, in order not to draw attention away from the video being played, while still being easily discoverable. A scrollbar ensures that many media items can be

searched through and viewed at an appropriate size, and clicking an item in the list would open it in the video player, thus giving the user control over which media they wish to view at any time.

Overall, the interface is logically organized as it is split into two groups of items: those related to the media currently being displayed, and those related to the other available media. The design makes it easy for the user to observe the current state of the application, and control it through intuitive, familiar methods such as clicking buttons, scrolling through a list and dragging sliders. The design makes it convenient for the user to play or view media in a simple, intuitive and efficient way through methods which are easy enough for a user with little to no experience with a computer to perform.

Evidence of Design



Evaluation

Evaluation Technique

A Cognitive Walkthrough has been selected as the evaluation technique for this cycle.

Technique Motivation

It was decided that a cognitive walkthrough would be the most appropriate evaluation technique for this cycle since it would ensure our design meets key requirements and is appropriate for the target user group.

A cognitive walkthrough is effective for evaluating an initial design, as it takes into account who our primary users would be and what they might use our software for. Having to explicitly lay out assumptions made about the users and context of the use case encourages the team to abandon possible biases and try to visualize the design from an user's perspective, rather than that of a user interface designer. Furthermore, needing to choose a specific task and break interaction down into basic steps should help identify potential logical flaws in the way our design was thought out. As an added benefit of thinking about possible tasks users might want to accomplish, the team might also identify features not currently existing in our prototype that could be found useful by members of targeted user groups.

Going through the interaction steps slowly and viewing them from the perspective of the user in their assumed context we can discover assumptions we had subconsciously made about interaction that may not hold in all cases, as well as previously unnoticed design flaws. This is particularly helpful in the first cycle of development, as it significantly lowers the risk of investing resources into developing an interface that performs poorly with the primary user group or does not satisfy requirements.

Evaluation Outcome

The Cognitive Walkthrough highlighted issues with the layout of functional elements on the screen, as well as the need for additional components in the interface. As the designers of the prototype, we had made certain assumptions during the design process that the evaluation has shown to be untrue.

One of the assumptions made during design was that since it's a dropdown menu, the "speed" option should be in the top left corner of our layout. This has been proven untrue by the evaluation, since a user would most likely seek controls related to video playback in the bottom part of the media player, and therefore have some trouble figuring out what the correct action would be. Another subconscious assumption made while prototyping was that the user would be using a mouse when interacting with the interface, and that was challenged during evaluation by thinking about how a task could be carried out using a trackpad. Indeed, clicking a moving slider and dragging it might prove troublesome for users with limited dexterity, or unfamiliar with using a trackpad.

Additionally, while discussing what implications being at a friend's house would have on the user and the interaction, it was pointed out that users might need to lower or mute the volume in order to not disturb other people in the same room, therefore adding new elements to the interface was suggested. However, the team also paid attention to positive aspects revealed by the walkthrough, such as the similarity between player controls in our interface and in other media playing software, or the benefits of organizing the available media in a scroll area. It was decided that these features should remain in our design and be improved upon in implementation and in the following cycles.

Overall, the Cognitive Walkthrough helped us recognize aspects that needed improving in our prototype, as well as elements of design that benefit interaction and therefore should remain central to the interface and be further developed in the upcoming stages. Overall, the cognitive walkthrough assisted in modifying parts of the prototype and elements of the design to a more user friendly interactive platform by looking from a user standpoint.

Evidence of Evaluation

It has been assumed that the imagined user would be interacting with the implementation of our proposed design. Additionally, the following assumptions have been made about the conditions of the use case:

User	Ethan, a winter sports enthusiast with basic computer skills
Context	At a friend's house, in a hurry, using a laptop with a trackpad
Task	View a specific part of a long video depicting ski stunts, in slow motion

The results of the evaluation were the following:

Step	Current Action?	Perform?	Associate & Interpret?
1. Scroll down in the video list	✓ Can see a list of videos in the right side of the screen	✓ Familiar with scroll bars	✓ Can see the video thumbnails and scroll bar moving on the screen as expected
2. Click on desired video	✗ Thumbnail may not be enough to recognize the correct video if there are many similar ones	✓ Familiar with the concept of clicking an item to interact with it	✓ Can see the clicked video title on top of the screen, and the video being played
3. Drag progress slider to navigate to the appropriate part of the video	✓ Likely familiar with the process of dragging a slider to navigate a video, thanks to having previously used media playing software such as YouTube	✓ Current time in the video is displayed making navigation easy ✗ Dragging a slider may be difficult while using a trackpad	✓ Timer is updated, and the video is in the desired state
4. Click the "Speed" Combo Box	✗ Might be looking for the option in the area with the rest of the video controls	✓ Will likely click on item to interact with it	✓ A dropdown menu is shown, containing possible playback speeds
5. Click the "0.5x" option	✓ Likely to know the number refers to the playback speed, as the option is labelled appropriately ✓ Likely to know a value below one will reduce playback speed	✓ Will likely click the option to select it	✓ Will see the current playback speed multiplier in the top left of the screen ✓ Video will play at the desired speed

Code

Video

https://drive.google.com/file/d/1T_1I_GsebGoTxLWDqVEepz2UlNc_blok/view?usp=sharing

Tag to repository

https://gitlab.com/Psydonic/comp2811/-/releases/DevCycle1

Changes from Prototype

There were several changes made to this prototype when the code was written. First, the title of the video was removed, as not only did it take up too much space above the video player window, it also made the layout look less clean. As well as this, the video controls such as the player bar and the pause/play buttons were moved to be underneath the video rather than overlaying it, as they could obscure a small part of the video. The speed box was also moved to be next to the video controls to keep the controls all in the same place, in order to cause the least amount of confusion. A volume and mute control was also added, so that the user would be able to modify the volume of a video.

2) Layout for more complex functionality

Prototype

Cycle goals

The primary goal of this cycle is designing a layout that provides access to functionality for organizing a video library, in addition to the functionality for selecting and playing videos that was implemented in the previous cycle. Possibilities include updating the interface with a “tags” feature that facilitates sorting and searching and a “metadata” section to display more information about the currently selected video. Organizing the layout to accommodate such features will be the main focus of the cycle, but simple functionality will also be implemented. Implementation of more advanced functionality lies within the scope of the following cycle.

Techniques and software used

The Wireframe prototyping technique has been chosen for this cycle, and Google Docs’ Drawing tool has been used to produce the prototype. A Paper Sketch prototype and another Wireframe have been produced initially, and have been merged into a single Wireframe, which was then evaluated.

Technique motivation

Wireframe prototyping using software tools has been chosen as a technique since it is similar to paper sketch prototyping in flexibility, while making it easier to visualize how our layout would look when implemented. Furthermore, using software allows changes to be made to the design quickly and easily during discussion, with elements being resized/removed/altered, without having to start a new sketch. Using google drawings also allowed us to work collaboratively on the prototype in real time, which improved the speed of design.

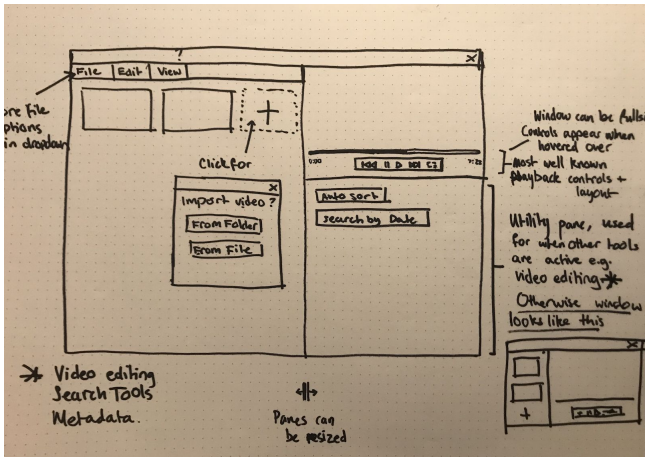
Design motivation

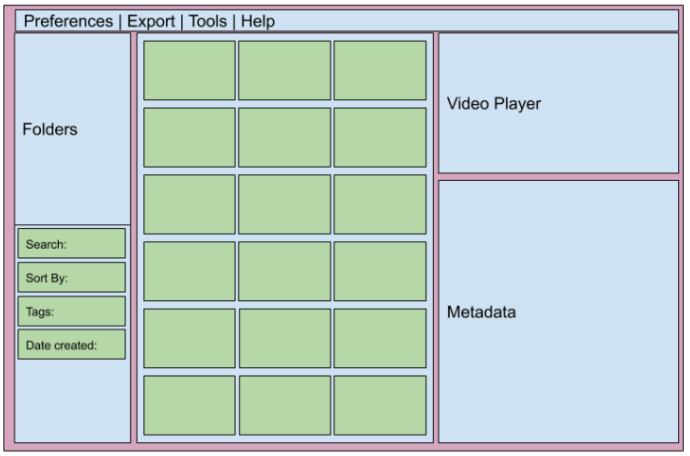
We decided to create two candidate prototypes, in order to obtain a range of ideas, that would then be merged into one as a result of a preliminary evaluation. In the first two candidate prototypes It was concluded that the video player in designs 1 and 4 was too small for its relatively high priority in the interface, while in 2 and 3 it was too large, obscuring space for other windows. Consequently we settled on a medium size that can be seen below. This still evidently communicates its importance in the interface without taking up too much space for other windows.

We determined that a grid-like viewer would be the most intuitive for users to find certain media by thumbnail (which would require sorting features), but also recognised that a folder hierarchy could be useful in order to navigate through sub folders. Finally we agreed that it would also be useful to the user to display additional information about the selected media, such as its metadata and tags. Due to the nature of the videos being landscape, we decide to place the video grid on the bottom in order to allow them to tile horizontally

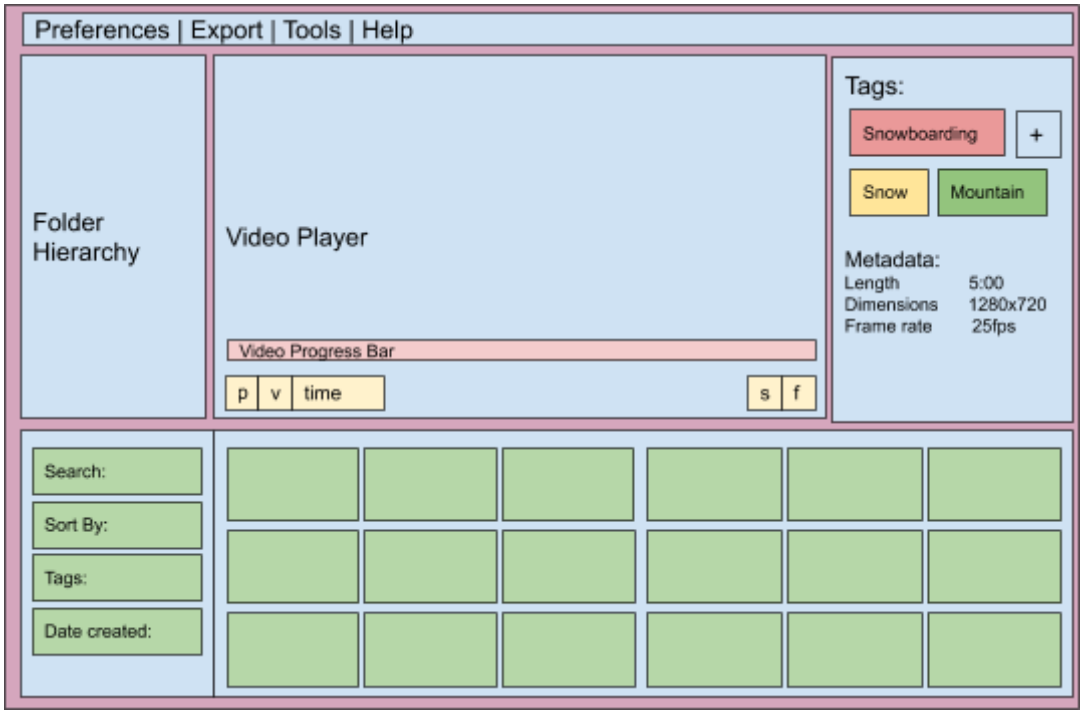
Evidence of Design

Initially, two candidate prototypes were considered:





Following discussion, the options were merged into one wireframe prototype, which was then evaluated:



Evaluation

Evaluation Technique

Heuristic Evaluation has been used as the evaluation technique for this cycle.

Technique Justification

Heuristic Evaluation was chosen as the evaluation technique for this cycle since it provides a fast and effective way to assess the usability of the prototype. This is appropriate in the early stages of development, as it helps expose design problems early in the process.
Since the technique doesn't involve users, evaluation can happen quickly and communication can happen effectively between the evaluating expert and the design team. Not requiring resources external to the team, an expert inspection is simpler and more efficient than user-based evaluation. The chosen technique also allows for more flexibility than other alternatives, since no specific tasks need to be set, and evaluation can be focused on the interface as a whole rather than on specific aspects of the user experience.

Heuristic Evaluation provides an extensive set of results, and ranks them by gravity, indicating to the design team what their focus should be. Being based around usability principles, this helps offer the team a clear direction in the early phases.
In summary, Heuristic Evaluation has been chosen in this cycle because it's a flexible method, providing clear and comprehensive results that highlight the areas of our design that need to be improved.

Evaluation Outcome

The evaluation has revealed certain problems with the designed prototype, and ranked them by gravity (low - medium - high). This helped the team orient the design effort in the right direction early in the development process.

Nielsen's 10 general principles for interaction design have been used as heuristics, in order to test the prototype against well-established standards. The evaluation was carried out by our team in two stages, and revealed both positive and negative aspects regarding how our design matched the chosen heuristics. The first stage was where most positive aspects were noticed, likely because of the team's bias, but also where most key problems with the prototype were identified. The second stage of evaluation predominantly revealed lower priority issues, but also identified high priority issues that weren't immediately obvious, such as the folder hierarchy and the video preview grid not being grouped together in any way despite representing different ways of depicting the same files. Ranking the problems by importance allowed the team to focus on the key aspects that need improving and not get distracted by the minor issues that are either easy to fix or have no significant impact on the user experience.

The analysis produced a list of possible usability issues that the team will use to improve the prototype both during the coding stage of this cycle and in the following design cycles.

Evidence of evaluation

The following table contains observations made during the design and evaluation process, some of which have been addressed and corrected. Positive aspects have been highlighted in green.

Observation	Nielsen's Principle	Importance
Users can see the way their media is currently organized in the folder hierarchy, and change the organization through WYSIWYG methods like drag and drop	1, 2, 3	High
Users have control over which parts of videos are rendered, at what speed and at what volume	3	High
Menu in upper part of the screen organized intuitively	2, 4	Medium
The folder navigation component is on the left side of the screen, as is common in other applications	4, 6	Low
Despite showing the same items in different ways, the folder hierarchy and the video preview grid are not grouped	3, 7, 8	Medium
"Viewer" pane does not display names of videos	1	High
Video controls taking up a large portion of the video player	8	Low
Panes don't allow their resizing by the user	3	Medium
Users are not able to use the keyboard to interact with the video renderer	7	Medium
If the user decides not to use the tags feature, the pane would unnecessarily take up screen space	8	Medium
Users don't have the option to remove panes	3	Medium
Metadata may be confusing or unnecessary to inexperienced users	8	Low
"Tags" and "Date created" widgets in the bottom left take up screen space despite being options in the "Sort by" dropdown	8	High
Tags and Metadata are grouped in the same box, despite being different logical parts of the interface	2	Medium
The main element of the interface, the video player, is surrounded on three sides by panes that may draw attention away from it	8	Low

Code:

Video

https://drive.google.com/file/d/1V2knaZG3X_CvZ1TGsoQyT23peDk_QvbH/view?usp=sharing

Tag to repository

https://gitlab.com/Psydonic/comp2811/-/releases#DevCycle2

Changes from Prototype

The layout has been changed to reduce its complexity and minimise space consumption. The folder hierarchy has been moved into a tab widget, in order to take up less space, which was deemed appropriate because the folder hierarchy is not a high priority interface element. Furthermore the preview grid has been moved to the right side of the screen, in order group it with the folder hierarchy. The sorting menu has been moved to the video preview widget, in order to reduce the size it takes up, as well as to follow the established interface style. The tags and metadata widgets have been separated as advised in the evaluation, and the play progress bar has been moved down in order to allow more space for the video.

3) Implementation of functionality and interaction

3rd cycle- **dropdowns** added to the main window(import vds, open/close files/folders). **Organize and sort videos** based on metadata.

Prototype

Cycle Goals

The main goal of this cycle is to implement functionality of the user interface and enhance interaction. The interface should let the user navigate through videos more easily, which could be made possible by sorting videos based on their metadata. In addition, the dropdown items on top of the interface should let the users import videos and open folders and files from their device. Another primary point of focus for this cycle is obtaining feedback from primary users regarding our layout and interactive components and adapting the design accordingly. Aesthetic features are outside the scope of this cycle, and will be addressed in the fourth cycle.

Technique Chosen

Native Prototype (High Fidelity - Using QtDesigner)

Technique Motivation

A high-fidelity prototyping approach has been chosen for this cycle in order to obtain relevant user feedback on the appearance and proposed functionality of our interface. Native prototyping provides us with the ability to analyze the interactions between potential users and the prototype, improving our design based on the results.

Testing an interface with potential users should be done late enough in the design process that the team has an idea of what layout and interactions would be relevant to test, but early enough that substantial changes can still be made based on the results. High-fidelity prototyping covers not only the user interface aspects such as layout and aesthetics, but also user experience aspects, allowing observation of interaction and user behaviour. After using a low-fidelity prototype to orient our design in an appropriate direction in the first cycle, a high-fidelity prototype will help us get a clearer idea of how our application should behave. In order to identify issues with the available functions, their placement and behaviour, a horizontal approach has been identified to be the most useful approach for this cycle.

Native prototyping has been chosen as a particular technique, to make the interface as similar as possible to how the real product would look at this stage, as well as highlight possible implementation issues. It is expected that it will be possible to obtain qualitative data by observing users interacting with the prototype, as well as by collecting individual feedback. Since the design team is familiar with Qt, developing such a prototype was decided to be the fastest and most effective high-fidelity prototyping technique to adopt in this cycle.

Design Motivation

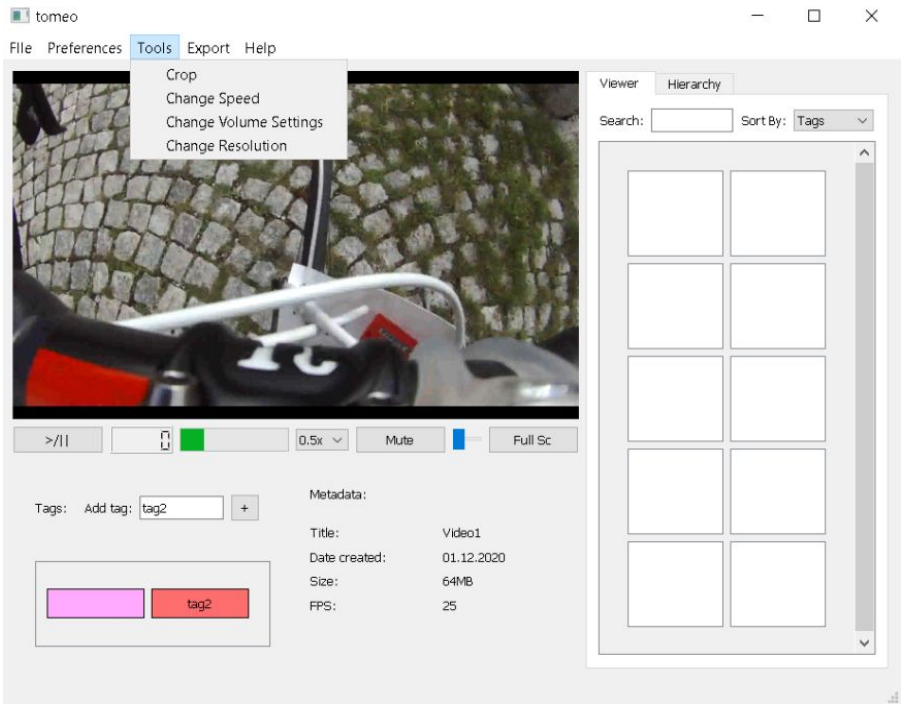
The design of the prototype for this cycle follows up on the prototype and implementation produced in cycle 1, aiming to improve the layout of certain interface components and add ways of accessing more functionality.

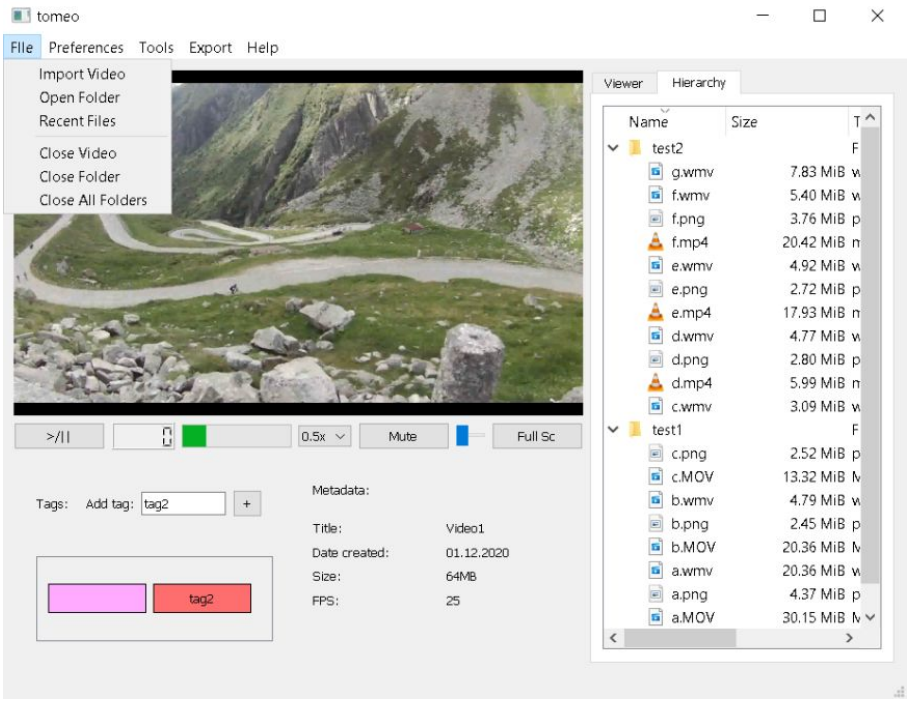
To improve the interface’s similarity to common applications that users may be accustomed to, dropdown menus have familiar titles, and contain functions users can expect to find under each menu.

The placement of the tags and metadata components below the media player component serves to indicate by grouping that the displayed tags and metadata are related to the file being rendered currently, rather than an arbitrary file.

Boxes have been added around these components to clearly define their boundaries, and ensure they are perceived as two conceptually different parts of our design. In the “viewer” component we have added searching and sorting functionality, with the folder currently open in our application as their intended scope. These would serve to ease navigation of a large folder of similar items, and enable the organizing of information in a clear fashion by offering the option to sort by size, date, as well as by user-added tags. This ability to add tags to files and use them to navigate their library would enable users to easily organize their library by attaching to files information that is both meaningful to the user and usable by the application as a searching and sorting criterion.

Evidence of Design





Evaluation

Technique chosen:

Usability Test in a Controlled Setting

Technique Justification

Usability testing was chosen as the evaluation technique chosen for a variety of reasons, first and foremost being that getting feedback from users that are most aligned with the target audience of the program, and then acting on it will be one of the most effective ways of making the program more suited for our target audience. The users interviewed can also help give us specific feedback on things we may have overlooked in the program design, or parts of the program design that may be unnecessary or hard to find/use. Another reason usability testing was chosen is that we can also ensure that the features in the prototype currently have been implemented correctly, as a user may attempt to perform an action with the program that is unanticipated, which would need to be fixed as it could cause confusion on the user's part, or an error in the program.

Evaluation Outcomes

The usability test has revealed both positive and negative aspects of our design, offering us meaningful insights into how our primary users perceive the layout and interactive elements of our interface. This will help us solve issues with the layout of our interface, optimize interaction and improve functionality, while also giving us a starting point for improving aesthetics in the next cycle.

Four users of varying experience with using computers have been selected to participate in the study, which consisted of two parts. In the first part, the users were observed and recorded while completing a series of tasks using our interface, which enabled us to identify which tasks a user would find difficult and what mistakes they're likely to make. Participants were asked to think out loud as they carry out the tasks, which impacted the accuracy of measurements regarding performance time, but added the benefit of gaining insight into the users' thought process as they interact with our interface.

The second part of the evaluation was a short interview that aimed to obtain qualitative data about the subjective experience of participants with our prototype. This part produced relevant information about how the interface may be perceived by people in the target user group. Analyzing results of the first part of the evaluation led us to rethink the layout of components in the main screen of the interface, as well as the way functions are organized in the dropdown menus. The second part revealed that while most users found our tags feature useful, they thought it wasn't immediately obvious how to use it, prompting us to think about ways we could present it better. It has also been found that users with little experience using computers thought the overall interface was "easy to use" and "intuitive", though participants' possible bias towards giving positive feedback has been accounted for in drawing conclusions from this data.

Evidence of evaluation

The usability test has produced a number of videos and recordings that the team has analyzed in order to improve our interface. Examples of videos of interactions are provided here, in order of relevance:

- https://youtu.be/QKKh0Pfm-Mo
- https://youtu.be/yQY1CcLf46I
- https://youtu.be/Bk91P1oDepI
- https://youtu.be/wNSwBXfoDpw

Code:

Video

https://drive.google.com/file/d/12iomO6FNVVuPO44-P5czEiW6u2haBiLF/view?usp=sharing

Tag to repository

https://gitlab.com/Psydonic/comp2811/-/releases/DevCycle3

Changes from Prototype

In response to user feedback, the layout of the interface has been rearranged. The tab menu has been split, as users found it difficult to locate the folder hierarchy, which they looked for on the left side of the screen. Furthermore import and export header feature have been moved to a file dropdown, as users felt this was more in keeping with other programs

4) Aesthetic aspects

4th- aesthetics. **Color options**, **themes**, **change layout** a bit if required to make the player look pretty.

Prototype

Cycle goals

This cycle will focus on the aesthetic aspects of the interface, such as the appearance of design components, color options and themes, and minor changes to component layout meant to make it more visually appealing. To ensure the final product will be seen as beautiful by users, we will aim to get feedback on various possible themes and layouts, and make the decision on which ones to include based on users' preferences.

Design Technique and Software Used

For this cycle, a high-fidelity visual design prototype has been created using Photoshop, with different color schemes.

Technique Motivation

A visual design prototype has been deemed the most appropriate for this cycle, as the main focus will be on aesthetic aspects such as color schemes, fonts and component styles, which are best demonstrated with software such as Photoshop.

Since interaction aspects have been evaluated in the previous cycle, the details that have yet to be decided are related to the aspects of the interface aesthetics. A high-fidelity prototyping approach is required in the final stage of development, as it enables the design team to obtain very accurate user feedback regarding aesthetics, and ensure the final product will be well-received by the target audience. Since only minor changes to the interface's layout are considered, multiple versions of the same prototype can be produced quickly, to demonstrate various combinations of color schemes, fonts and styles for different components such as buttons and dropdown menus. These versions can then be evaluated through quantitative and qualitative methods such as questionnaires or A/B testing to identify the best choices to include in the product that ensure users are presented with an interface they find visually pleasing.

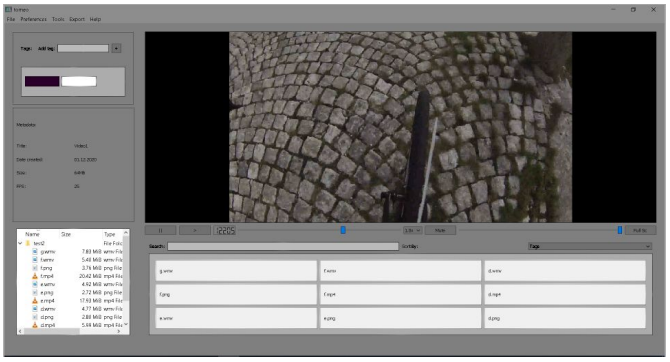
Design Motivation

We decided to create several possible coloured layouts for the aesthetic design of this cycle, and ask for user feedback to see which ones people would be most likely to use. We came up with 6 designs, including the default plain design, for a total of 2 light modes, 2 dark modes, 1 greyscale and 1 colourful designs. This was to ensure that we had a varied range of possible options for users to decide between.

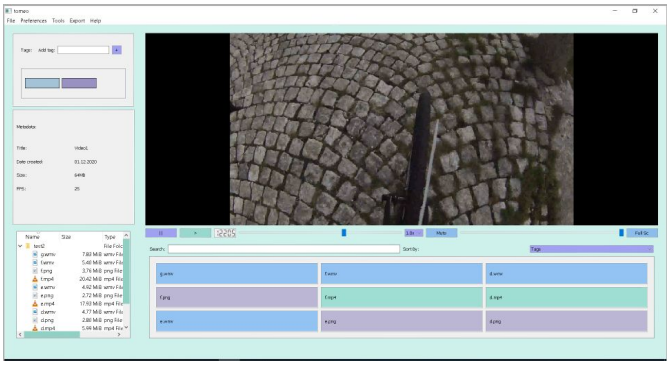
The grayscale mode was mostly intended to be a neutral theme that would not cause strain to the eyes when using the program for a long time, but was also intended for users that may have difficulty distinguishing between different colour hues. Both orange and blue dark modes were intended to be elegant and pleasing to the eye, and without any especially bright colours so that they would be appealing to most audiences. Additionally, the dark colour scheme is less likely to strain the users eyes when working late at night or in a poorly lit environment. This is the same for the light mode themes, turquoise and plain, where in the turquoise theme the colours were picked to be bright and appealing to the user, but without being too obtrusive, and the plain theme was left as is to give the user an option of using the program without the distraction of any colours. The colourful theme was intended for users that

may want a more exciting theme, but also for users that may become overwhelmed by the functionality in the program, and so it colour codes some aspects of the window. For example, the areas that correspond to tag control are pink, and the areas that correspond to folder and file management are light orange-yellow. In all of these themes, the brightest or most contrasting colours are placed on areas of interest in the program, such as the buttons, in order to draw the user's eye.

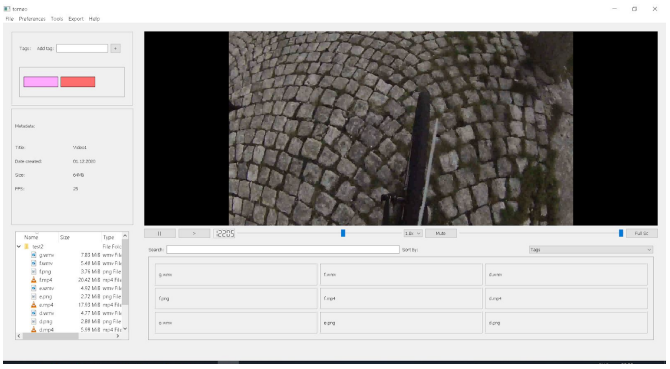
Evidence of Design



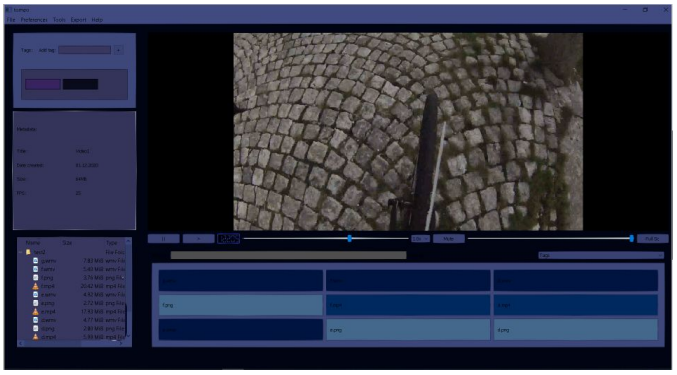
Greyscale



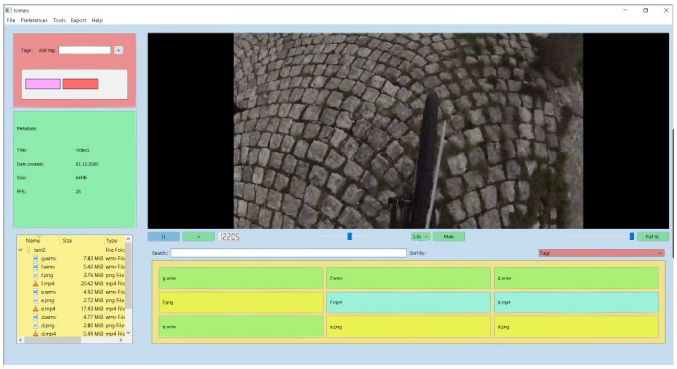
Turquoise (LM)



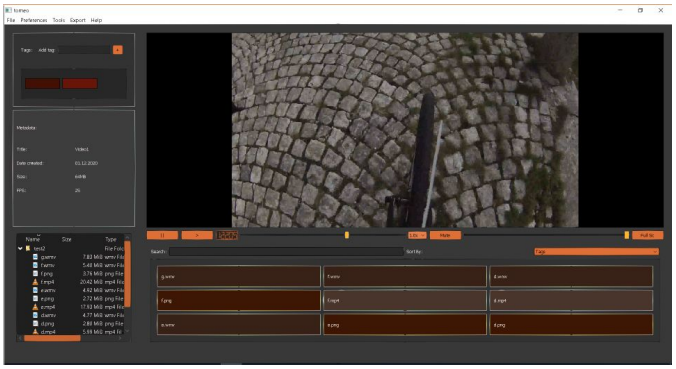
Plain (LM)



Blue (DM)



Colourful



Orange (DM)

Evaluation

Evaluation Technique

The evaluation technique we used was an online Questionnaire.

Technique Motivation

The controlled study using a prototype was decided on because it allowed us to do A B testing on prototype themes we have developed. This will then influence what themes are implemented into the design.

As our design layout has already been finalised it further demonstrated that we should use an A B testing methodology. This is because the only difference between the presented images was that the theme and it would be the most useful method of evaluation as we could get real-world data on what users like concerning the themes that will be available in the application.

There is also an opportunity for the tester to give general feedback on the prototypes and then specific feedback about what they liked about their favourite prototype and what they disliked about their least favourite prototype, this information can then further inform our development.

Evaluation Outcomes

From the questionnaire, we found that for the majority of people the Coloured Theme was their fifth or sixth choice which is pretty atrocious so the Coloured Theme will not be implemented into the design as it is so universally disliked.

It was also found that the orange theme was the most likely, being the first choice for the majority of people, this will not be made the default theme due to the fact that we want people to try the software out stock and if they decide they want to change themes they can do it easily. But this information will mean that we put it at the top of the themes so it is easy to select when users want to change over to it.

All of the other themes performed fairly consistently in the middle with not many people loving or hating them which is good. These will all be implemented into the design so the user can have the freedom to customise the user experience to any of these themes.

Evidence of Evaluation

Data collected using this form: https://forms.gle/P8w4HSpCa9e387ME9

Number of picks for:	Plain Theme	Coloured Theme	Blue Theme	Grey Theme	Orange Theme	Turquoise Theme
First Choice	1	1	1	0	5	1
Second Choice	1	0	0	4	2	2
Third Choice	3	1	3	1	1	0
Fourth Choice	3	0	3	1	0	2
Fifth Choice	1	3	1	2	0	2
Sixth Choice	0	4	1	1	1	2
Rankings Overall	4.833333333	7.166666667	5.5	5.166666667	3	5.833333333

Code:

Video

https://drive.google.com/file/d/12t0UQM_rkZnIg_5P4odiUyGtn-IhH1qd/view?usp=sharing

Tag to repository

https://gitlab.com/Psydonic/comp2811/-/releases/DevCycle4

Changes from Prototype

There were various technical issues when implementing this cycle, that meant that certain features could not be implemented. For example, there are issues with changing between themes that could not be fixed due to time constraints. Furthermore, the themes are not exact replicas of the prototype, however, they hold the main concepts, with dark and light themes available, with the preferred orange highlights.

Ethics

Focus group testing

All the participants have been thoroughly informed about the details of the study they were invited to take part in. Before agreeing to participate, they were presented the Participant Information Sheet and the Research Privacy Notice, and were asked to fill out the Participant Consent Form (see attached pdf below). Furthermore, to comply with COVID-19 regulations, all participants were selected from the lead researcher's accommodation, in order to be able to conduct the study in person.
https://drive.google.com/file/d/1auTVMZi2fvbfIFV715S8TVJGrnBm9TBE/view?usp=sharing

Survey

https://drive.google.com/file/d/1IYZnKCNS3waQz0U7645r-lB5_elHjxYl/view?usp=sharing