The requirements and solutions for a good video call user experience for elderly people

Master of Science in Technology Thesis
University of Turku
Department of Computing
Software Engineering
2022
Lauri Laakkonen

UNIVERSITY OF TURKU
Department of Computing

Lauri Laakkonen: The requirements and solutions for a good video call user experience for elderly people

Master of Science in Technology Thesis, 57 p.
Software Engineering
April 2022

Socialising during Covid-19 pandemic has brought new challenges to tackle. Usage of technology and especially video calls are something which can bring a lot of new opportunities to handle the challenges. To be able to use video calls requires a targeted solution for elderly people and people who suffer from impairments brought on by ageing. Vooler is a custom made solution which is solving the issues that elderly people might face with more regular video call solutions. The thesis looks for answers about what current video calls are achieving and for what kind of users. There are also investigations about what kind of things have to be accounted for so elderly people can use video calls effectively and what sort of solutions can be used to handle difficulties that the users might face. Additionally accessibility has a big focus, since the probability of requiring some level of accessibility is high in elderly people. These are made into individual points and made as requirements in the thesis, so they can be used to evaluate individual systems. Finally the viability of solutions used by Zoom, Teams, Google Meet, Jitsi and Vooler are evaluated with the help of an interview based case study. The results of the study favour Vooler when it is evaluated for elderly people use, when considering basic use and accessibility.

Keywords: user experience, user experience evaluation, usability, usability evaluation, accessibility, elderly, video call

Contents

1. Introduction

The purpose of this research is to refine the UX product “Vooler”, which is a video call solution, similar to Zoom or Google meets. Vooler tries to provide the easiest to use and simplest solution implementation of video calls for elderly and special needs people to use. Additional thing to consider about Vooler is also that its target audience is elderly and technologically less experienced people, which can be boiled down to be special needs groups.These can be also categorised by disabilities they might suffer from, such as poor eyesight, poor motoric abilities, memory problems. It is also an important motivator that the people aged 80 and over should triple by 2060s [7], which means that the need for new approaches and standards will gain more importance for solutions which are focused on elderly people. Additionally the concept of “Ageing in place” in developed countries has emerged, which means the need for elderly people to be able to age in place as autonomously and safely as possible, while trying to relieve the burden of being supported by their carers [7]. This comes with its own difficulties, that would have to be evaluated in a way which enables efficient use of resources and prioritisation of issues to provide most value for the customer. This means that the needs of the customers when considering the UX have to be identified and should be made as the baseline of implementations, since if a simple solution can be used by someone inexperienced, it should be completely fine for someone who is more experienced.
These issues and research questions are to be answered within this thesis, in the following manner. First there will be discussion about the definition of UX and what it consists of. After that the thesis will explore the differences and different needs of various user types and group them in a manner which helps identify important UX factors for said users. When these factors have been identified, I will go through other video call solutions in order to evaluate how these have been addressed within them. Ultimately the goal is to improve Vooler, so in the case study the thesis will create and explore solutions for testing and improving UX within our project.

RQ1 How do current video call solutions compare when considering social interaction for elderly people?
RQ2 What are the UX requirements for social interaction through video calls, if the target audience is special needs groups and nursing staff?
RQ3 What UX elements have to be accounted for in video calls for basic social interaction, if the target audience is special needs groups and nursing staff?

2. User experience and usability

2.1. What is user experience?

User experience, or UX for short, can be defined as “person's perceptions and responses resulting from the use and/or anticipated use of a product, system or service”, which also “includes all the users' emotions, beliefs, preferences, perceptions, physical and psychological responses, behaviours and accomplishments that occur before, during and after use” according to ISO 9241-210 standard. [2] UX is also affected by things like brand image, functionality, presentation, system performance, interactive behaviour and assistive capabilities of the system. It is also affected by users' physical and internal state from prior experiences, skills, attitudes and personality and the context of use. [2] UX can also be defined as a momentary mostly evaluative feeling (good or bad) which comes with interacting with a product or service. UX therefore does not solely focus on product and the materials, which can also be thought as the technical implementation, but it also takes into account humans and feelings, and the fact that these feelings are subjective and therefore there is no “right answer”. [3] It is also important to note that usability is a big part of UX, so it is good to differentiate usability heuristics, so evaluating usability specifically is possible.

2.2. Usability

Usability heuristics is a good way to evaluate usability [1][18].
According to Jakob Nielsen there are ten heuristics, which are:

1. Visibility of system status
2. Match between system and the real world
3. User control and freedom
4. Consistency and standards
5. Error prevention
6. Recognition rather than recall
7. Flexibility and efficiency of use
8. Aesthetic and minimalist design
9. Help users recognize, diagnose, and recover from errors
10. Help and documentation

The heuristics are mostly a rule of thumb rather than being specific guidelines for measuring each specific case of usability [1]. These heuristics can be applied in many different ways and into many different things, but everything does not fit to every use. Here are examples of usage for these ten heuristics.

1. **Visibility of system status**

Users should be informed about the system status at all times and appropriate feedback should be given to the user within a reasonable timeframe.
For example, if something is being loaded for the user, it would improve the users experience if they knew that the system is loading something and even better if the status (for example the percentage of progress) is indicated if the loading time is longer than a few moments.

2. **Match between the system and the real world**

Systems should use terms and words that are understood by a more average user compared to technical users and technical terms. These should also follow real-world conventions, to make information appear in a logical and natural order.
For example icons should follow real life counterparts, such as email often has a “physical letter” as its icon, when in reality emails have very little to do with these physical letters, but it makes it very easy for the user to understand this connection between electric mail when it is connected to the real life mail. Another example is that terminology should follow real life conventions, such as “library” as a term for a collection of tv shows and movies or for games.

3. **User control and freedom**

Users might accidentally access menus and options that they did not intend to so they should have very clearly marked “emergency exit” buttons for these so they can leave unwanted states easily.
This could be a clearly marked exit button, and ability to exit menus with pressing outside of the menu. Having multiple ways to do these “undo” actions are not bad, but they should be intuitive and they should be simple to understand.

4. **Consistency and standards**

The built system should follow commonly used standards and conventions that are widely used, this means certain things are quickly if not instantly understood what they do.
This could be for example the number of unread messages in an email inbox. The number of messages functions to indicate unread messages.

5. **Error prevention**

Users should be protected by making things in a way that prevents errors from happening, for example disabling sending of messages without titles, or having confirmations for “dangerous” operations such as deleting items or doing something that is not easily recoverable or redoable. Also design should direct users to do the most likely action, for example sending an email message should be the primary function, but deleting / cancelling this action should be available, but less intrusive on the screen, since that is less likely to be used.

6. **Recognition rather than recall**

The memory load for the user should be minimised. All objects, actions and options should be visible for the user. Users should also not have to remember information from one part of the system to another, the needed information should always be readily available if the user needs it.

A good example of this could be the previews for fonts within word when you want to select from the list of fonts, the fonts usually have a text or the name of the font in the style of the font, so no testing or memory is needed to know what the font is like.

7. **Flexibility and efficiency of use**

There should be efficiency enablers for expert users and inexperienced users alike, but they should only enable and not hinder inexperienced users. These can be for example keybinds or certain features that have some sort of quick access in some less obvious manner.

8. **Aesthetic and minimalist design**

Less is more, dialogues and text boxes should not contain irrelevant or rarely needed information in normal use. Space and attention spans are limited, so the less these can be used, the better, since it makes the visibility of the system much better and makes it visually much more appealing as well. Visual layouts should also respect the principles of contrast, repetition, alignment and proximity.

For example if there are a lot of functionalities that need to be included, they should be categorised and put under certain features, so the navigation of these functionalities does not become a tedious and difficult task. Word would be a much harder tool, if all of the tools would be a big page with individual buttons for all of them.

9. Help users recognize, diagnose, and recover from errors

Error messages should be meaningful, and not only for technical people. They should clearly indicate what went wrong and it should be easy to understand what it is referring to and it should be constructive to the user about a solution for this error if one is needed. For example setting a password for an account might require specific characters to be used, but the user might use only lower case letters. The system should then show a meaningful error, that the user has not fulfilled these requirements that are set for these passwords, and there should be example of these different characters, eg. upper case letters are ABC and numbers are 123.

10. Help and documentation

Though it is better for the system to be understandable without any documentation, that is not always feasible. Therefore you might need documentation explaining certain features and giving basic and easy to understand instructions for the user, eg. step by step instructions. The help and documentation should be easy to find and search and this usually should be accompanied with harder functionalities of the system so the instructions for that are not hard to find and the more convoluted parts of the system is easier to understand.

3. Video call UX and accessibility requirements for elderly people

In this chapter, the thesis will explore how in video calls the basic flow of user experience is done, which in turn will answer the **RQ2** “What are the UX requirements for social interaction through video calls, if the target audience is special needs groups and nursing staff?”. These will have defined usability traits which can be noted as basic features of the system. It is also good to note the importance of case specific heuristics for successful implementation [20][21]. Without the case specific heuristics the evaluation of the product can be lacking. Additionally there will be evaluation of different groups with their special needs, who need specific accessibility accommodations, so they can effectively use the video calls for their needs. This evaluation is done by assessing the difficulties of their conditions and figuring out the limitations of the given accessibility issue. This is used to answer the **RQ3** “What UX elements have to be accounted for in video calls for basic social interaction, if the target audience is special needs groups and nursing staff?”. There is also a section where the motivations of doing these are discussed, whether creating a single unified solution for all users is sensible or somehow unviable. The tools required for the evaluation also are discussed in this chapter, for it should be discussed in relation to the product. Since traditional usability heuristics can not cover everything usability has to offer, tools have to be made which evaluate the solution more accurately [22].
With the basic usability requirements and the specific special needs user groups and their accessibility requirements, a system can be specified so that it supports the basic use and the more specific nuanced needs of special needs groups.

3.1 Video call usability in basic cases

This segment describes the minimum of how features and usability is designed in these systems, which are meant for average elderly users and the supporting people around them. The basic flow of the video calls does not include the challenges that some specific user groups within that group could face while using this kind of a system, but they are more specifically examined in section 3.3.

3.1.1 The purpose, functionalities and the challenges of a video call

Video calls in a social setting brings unique challenges to the providers of such services especially when it is targeted for elderly people. There are different tasks that users need to be able to perform in order to use these services properly. In certain parts of these

processes there can be, or for some people, it is required to have assisting personnel to perform these tasks. The main tasks required for a successful call are now chronologically ordered on when they would probably be encountered in a real call.
These will form the Usability Requirements, which will be later referred to as UR's.

FIGURE 3.1 Basic flow of a video call for elderly people

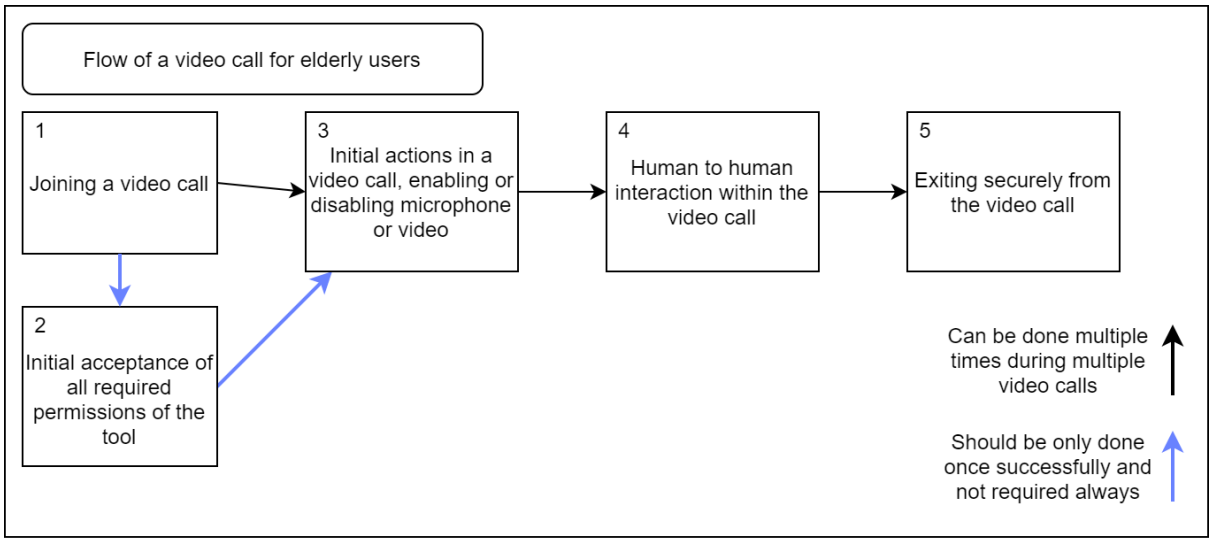
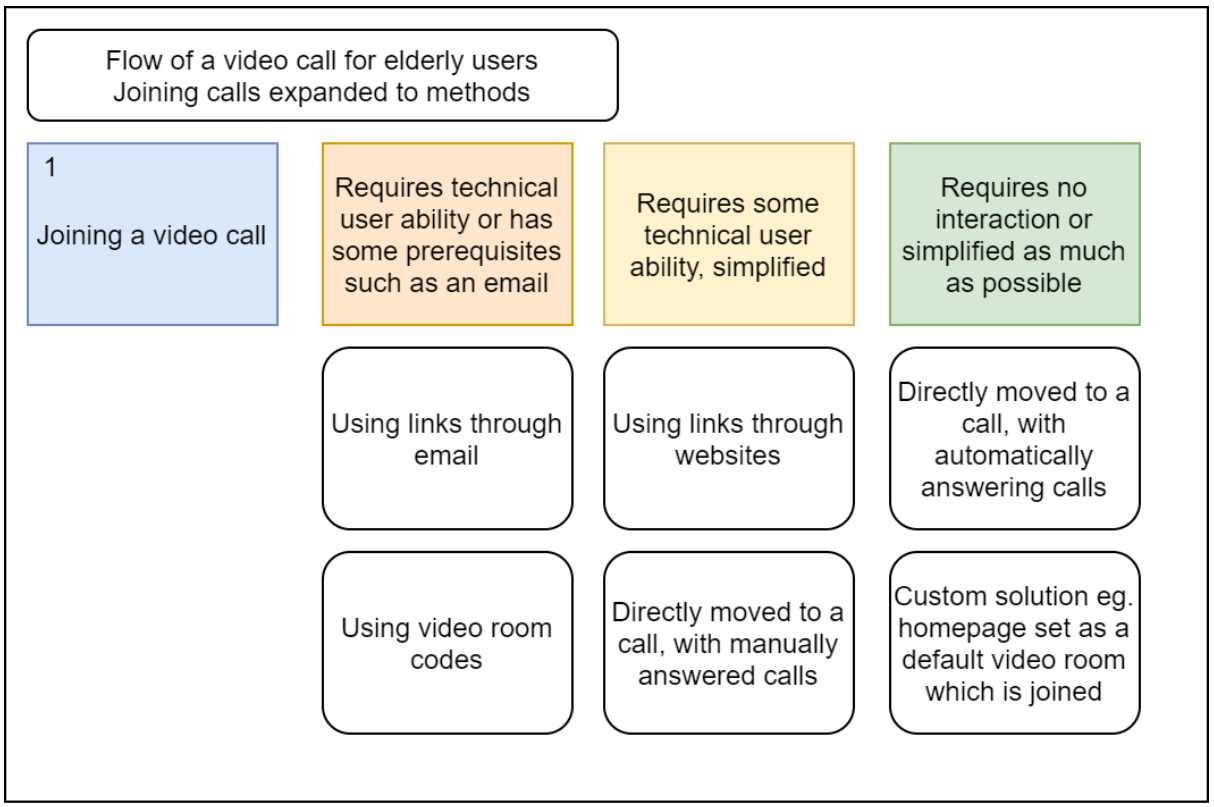


FIGURE 3.2 Joining calls expanded with level of user ability required

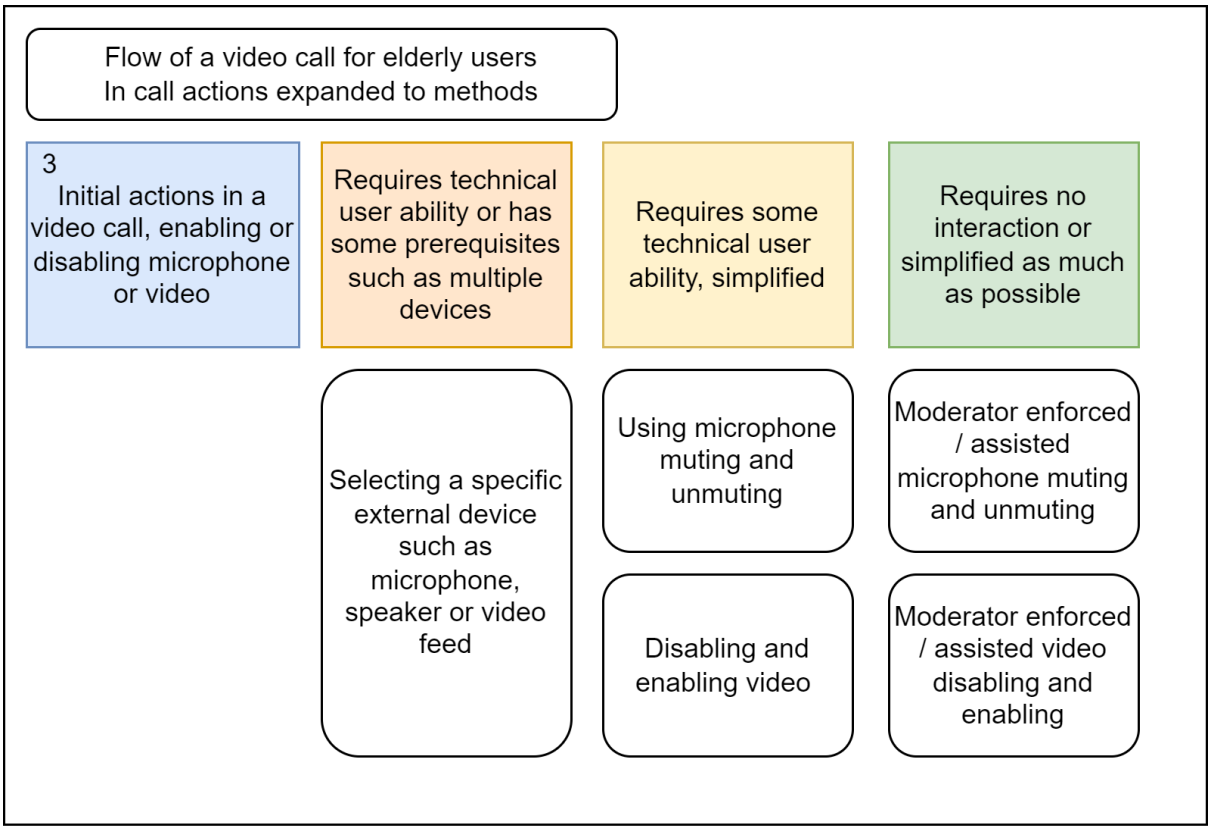


First thing is related to joining and creating video meetings. Users need to be able to join and direct themselves to the correct video meeting. Alternatively they need to be able to create a video meeting and communicate the required information to other users in order to get them to join the video meeting. This usually consists of directing the joining user to a page with a code, using a specific link or URL or using an app along with a code, which they can either directly use to join to enter the call. As shown in Figure 3.2, the relative difficulty of joining differs between the implementations. Solutions become easier for the end users to manage, the more effort is done for them. For example, doing simple links in email is a lot more difficult for novice users, when compared with a ready made system, which even answers for the other person. Sometimes though users prefer autonomy, so doing actions for them automatically can be unwanted in more self-sufficient users.

These joining options should be made as simple and straightforward as possible. It is also worth noting, that especially elderly people might not have their own email address, so using those services in order to send and receive links to video calls is not always viable for them, and alternative ways to circumvent this is sometimes needed.

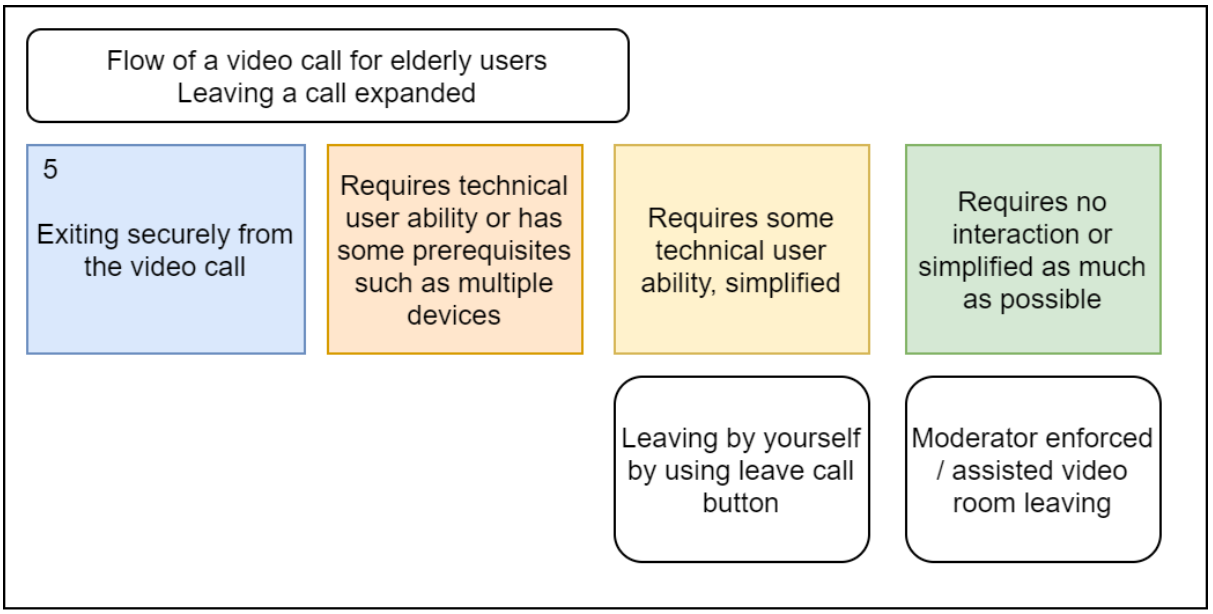
This will form the first UR, UR1, which requires multiple different routes to the same destination, in order to work well for as many users as possible.

FIGURE 3.3 Initial actions in a video call expanded with level of user ability required



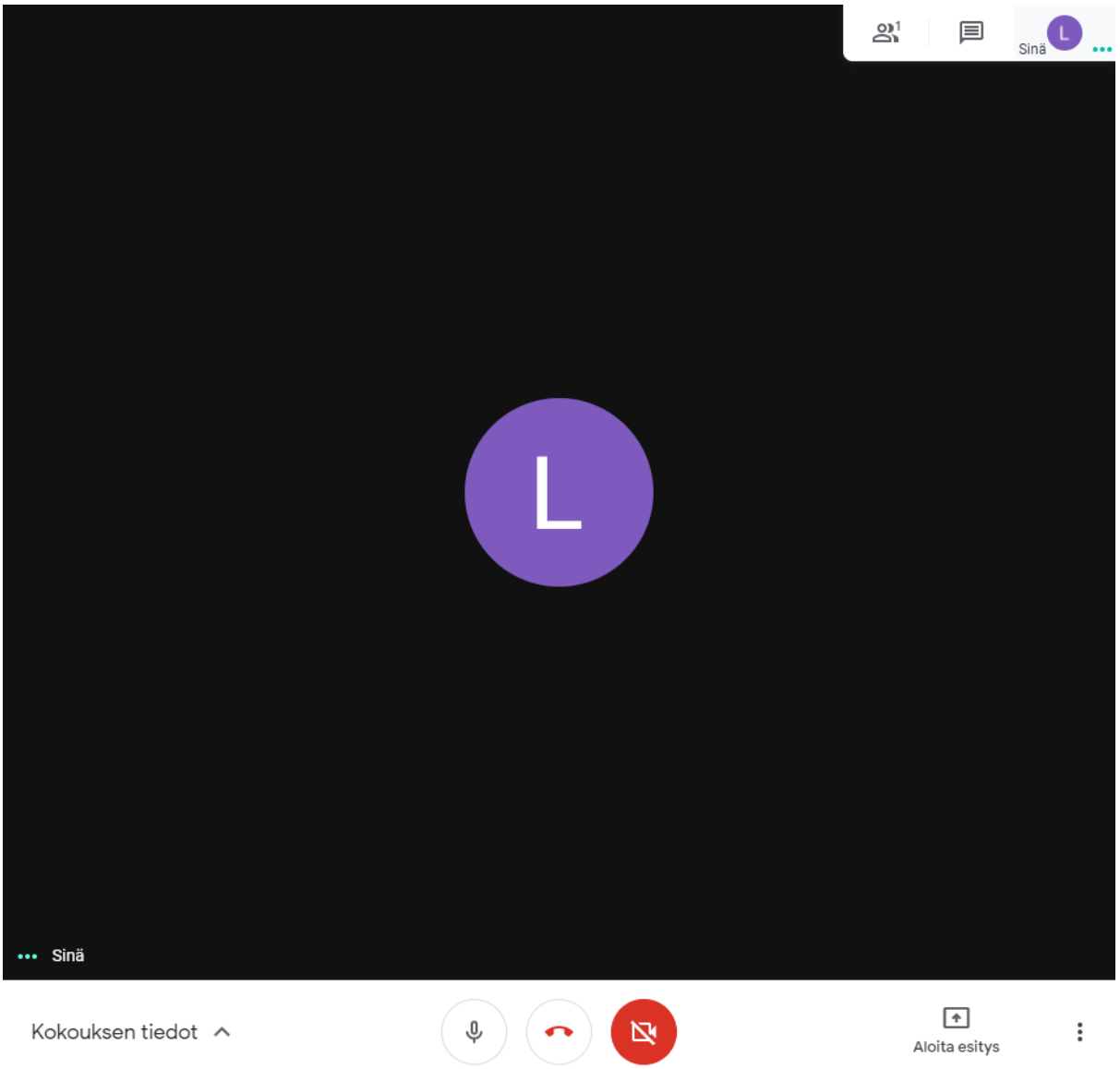
Once a user enters the video call, the user might be prompted to allow the device to use the camera and microphone of the device. Usually this is done once, if at all, depending on the program used to do these video calls. This is a critical part, since especially with browsers, if the initial permission is not given to the program, the following parts usually become much harder to resolve without any external help. These services usually take this into account, and it is usually detected somehow, and something is done to help resolve these issues. The operating system dictates what kind of prompt will be shown, but users can be prepared to deal with this popup, with explaining that this kind of a permission will be asked.
Inside the call, users might have to be able to control the states of their video camera and microphone on their own, depending on the meeting size. The bigger the meeting is, the more important this becomes, since having more microphones on can cause annoying background noise and even some echo, which makes the call much less enjoyable for the users. As shown in the Figure 3.3, these actions can be made for the user, but as explained before, user autonomy has to be taken into account when this should be done for them. People willing to learn and who do not like actions done for them can find the “easier” options intrusive.
Call controls are mostly buttons within the call UI which need to be understandable and the state of these features should be understandable to the user, so it is as user friendly as possible to use the system. For example the situation of being muted and trying to speak should happen as seldom as possible, which is made possible by conveying all of the needed information on the screen as clearly as possible.
Most important elements which the user can interact with in the calls are microphone, video, chat and end call buttons, additionally screen sharing is very important too when viewing content from a user’s screen is wanted. The reason these things are critical to video calls is that the main way to communicate in a video call is by using voice and video communications. This sometimes is further enhanced by using sharing options, where users can view for example screen shared content. Additionally chat is quite important too due to the fact that if some hardware device is failing, there is still a way to communicate between the users regardless of the state of the microphone or camera.
These controls include three Usability Requirements, UR2, Status of the system should always be easily identifiable. UR3, Visibility of elements should be made in such a sense that things that you need most should be most visible and things you need less can even be hidden. And UR4, System should be made to be as simple as possible, so it does not overwhelm the user.
Lastly the user needs to be able to end the call, so that everything gets wrapped up properly and there does not happen any forgotten camera or microphone situations if the call gets left open and unattended.

FIGURE 3.4 Leaving a video call expanded with level of user ability required



Call ending functions do not differ from normal call controls significantly, so they can be considered the same thing essentially, but one additional note about this is some sort of automation and error prevention system, so users do not leave the call unintentionally, and that they do not accidentally leave the call open. Figure 3.4 expands on this, there are mostly two ways for the user to be removed from the call intentionally, where the user makes the decision themselves or someone makes it for them, both usually very viable choices, since if the call is being ended, nobody has the need to leave themselves from the call.
This part includes UR5, which is error prevention. In short it is the design to make errors preventable on actions which are less used and dangerous in nature, which cause hard or irreversible outcomes.
A good example about implementation of this, is Google Meets. While the joining with a link is as good as expected, the code part is somewhat limited, just because of how random it is. Having many random characters in a row is very hard to communicate via speech for example, which can prove to be a difficulty factor for certain types of users. An example of a code google provides to join with is a 10 letter code, with two hyphens to divide the code abc-defg-hij.

FIGURE 3.5 Picture of Google meets main UI in a video call



As shown in Figure 3.5, Google Meets UI is quite simple, with deliberately tame colouring and elements. While this is good enough for regular users, this might not be good for elderly people with age related issues and difficulties with technology and other user groups with disabilities which makes it more necessary to have a solution with accessibility taken into account [13].

3.1.2 The system usability requirements of video calls

Usability Requirements (UR for short), are the requirements for the basic use of the software in a generalised setting with generalised people, which in this scenario is specifically elderly people and the supporting people around them. These can be derived from the previous example, which are based on the basic flow of the described situation. This is to create straightforward solutions for situations which can be interpreted in multiple manners. That

allows users to use the system in their own way to get satisfactory results no matter the way the user wants to use the system.

UR1. There should be multiple routes to enter a call within the system with different operating principles, to accommodate different limitations of the users.
UR2. The system should show clearly and without interaction the status of most important statuses in the call. These can be for example the microphone status, video sending status and connection status.
UR3. All important elements for basic use should always be visible and the function of the element should be as easy to understand as possible.
UR4. All solutions should favour simplicity over complexity and the system should be made as easy as possible for non-technical people and technically illiterate people to understand.
UR5. Error prevention measures, so issues are prevented from happening alltogheter.
UR6. Accessibility accommodations should be made, especially regarding issues which are commonly found in the target audience, but these can be made as options which can be enabled. These were not directly discussed in the previous example, but they are present for example in the choice of colour scheme in the system and additional options which support different kinds of use, such as automatic text literation from speech.

3.2 The motivation of grouping individuals in special needs groups

In order to categorise users to meaningful groups, it is important to consider what are the significant differences between each user group. While these qualities do not apply to everyone within a generalised group, the qualities can still be considered quite generalizable. It is also important to note that as it is mentioned in “User Sensitive Inclusive Design” [5], it is not a simple task to create a “universal” solution when thinking of these solutions. It is important to note, since users might have nuanced differences in how they optimally would use the given technology, which makes creation of a unified approach sometimes impossible. It is easier to approach this with a more inclusive approach, doing things with some user groups in mind rather than trying to make a “one size fits all” solution which does not work so well for some users [19]. Also users who are included in the process of creating interfaces feel a sense of ownership of the product, and end results are often better designed for the end users [19]. The inclusive approach means that usability is considered with all potential user groups in mind when designing the systems, which means they have to be designed with minority groups in mind, for example people with disabilities. This approach is required, since traditional User Centered Design does not encompass all of the characteristics with all of the possible user groups that might interact with the system [5]. Also something to note about the User Sensitive Incluse Design is that it might often require experimental techniques, and the nature of this design is to have it shareable with the mainstream researchers and product developers so it can be widely adopted.
While designing these User Sensitive Inclusive Solutions, it is good to consider that they will include a greater variety of user characteristics and functionality [6]. It is also worth acknowledging that sometimes creating specialised solutions for “able-bodied” users or in contrast “non-able-bodied” users might cause conflicts of interests in user experiences between solutions, so it is important to create personalisable and adaptive interfaces [6]. Also it might be needed to recruit “representative users” for these solutions. Additionally, there is the need to specify the exact characteristics and limitations of the user group [6].

One important factor between users for effective technology usage is how familiar they are with new technologies [4]. For example the study “The use of technology by the elderly” done in 2010 conducted that within elderly age groups (65-85+) some technologies are not used by a significant percentage, such as mobile phones were used by 54,1% of females and 81,3% by males. This means that a significant portion of these people have no experience at all with mobile devices, which might lead to difficulties when using said

devices. To deal with this, elderly people can be taught to use these technologies. This allows us to identify one clear category to divide users by, users who are experienced with technology, and users who are not. Having previous technical experience allows users to be more technologically literate, being able to use certain features which might not be apparent for users who have not previously used said technology. Also overall intuition is better with people who have been using technology, since often old working ways of doing things usually are not changed if there is no reason to, so previous skills with other tools might prove useful later. Even though the elderly user might not be optimal in their technology use it is still a very good tool to improve health, independence and feeling of safety [27]

One majority group of users that can be considered into the special needs group is ageing and elderly people. This makes it very sensinsible to group them into a single or multiple groups with certain probable characteristics that might impair their technology use. There are various things that get affected by ageing and the changes are without exception negative in their nature. One of these traits is vision, which loses its acuity, the colour perception fades and the susceptibility to glare also worsened [8]. Hearing is often also affected by having the hearing range to drop down from the higher end, therefore causing hardship when trying to hear high-pitch sounds and often hearing overall suffers [8]. This means that elderly people literally perceive technology in their own unique way compared to younger users. There is also motor control which changes for the worse, making fine motor control and coordination more difficult. On top of those issues, medical conditions such as arthritis can change the ability to interact with any technology physically for the elderly people [8]. These can cause issues if they are not carefully considered within the design of the technology. But it has to be also noted that user specific design is important, due to the fact that all elderly people are very unique in their physique and nature [24]. The uniqueness is also grounds for creation of tools which are specified on the user level, so the user is always getting personally designed solutions based on their needs [25][26]. The overall cognitive ability is also affected, ageing has been associated with general slowing down of the cognitive processes, decrease in the capacity of memory, decrease in the attentional control and difficulty in goal maintenance [8]. The aforementioned conditions can slow down the users performance and might result in a greater number of mistakes when older people interact with technology, especially with technology, which is not designed with their capabilities in mind [8].

In order to break down these needs the users need to be divided into certain groups that have the characteristics of certain problem situations that groups with a certain disability / challenge are facing. Also, in order to fully grasp the difficulties that certain one might face, you have to take into account the use cases that they will be acting in when using any activity, which in this case is the scope of social interaction through video calls.

3.3 Groupings by impairments of users

Within this section, the objective is to specify different special needs groups. The issues underlying their conditions might be different, but categorically they can be put into the same group, since the issue which causes problems might be different, but the solutions are the same. These range from medical conditions to general technological inexperience.
Lastly the groups are then categorised as accessibility requirements, which later on can be referenced directly, since the explanations are given within this section.

3.3.1 Vision impaired users

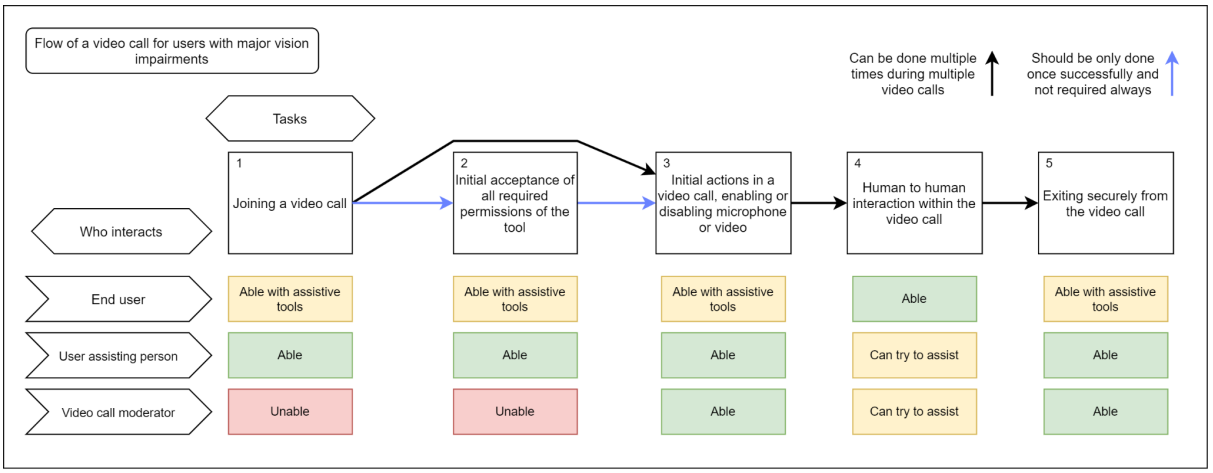
There are multiple groups in regards to vision. So they have to be divided into subgroups even with the umbrella group of vision impaired users. Vision impairments can include, but are not limited to complete blindness, partial blindness, poor eyesight, poor contrast vision. Additionally these issues do not have to come alone, so these issues can cause even hard base level situations for these users. These vision impairments, will later be referred to as AR1 as an umbrella concept for vision impairments.

Blind users

This user group consists of people who cannot see at all, or their vision is very poor (legally blind). Being completely blind usually leads to users using assistive technology in order to manage browsing websites or other services. A legally blind person might still have vision in their eyes, but they have very poor eyesight, which can also lead to the use of these blind assisting tools or requiring considerable accommodations in the software side to make the usage viable. It is important that these users have a text alternative available for them when implementing features. For example buttons, icons which are relevant for the user and images should have text alternatives, for example having aria-labels attached to said elements, which allows screen readers to be able to decipher the purpose of the element [10]. For example an icon for microphone muting, should have the corresponding label attached to it, so a user can inspect said element with its descriptive text and decide if they want to use it or move to another element. Also it is important to have everything necessary accessible by using the tabulator key, since these assistive technologies use the same principle in order to navigate the page [10][29]. Content should be also available in multiple formats, since it can be problematic if users cannot listen to or cannot have a text equivalent of a video for example [13]. Most blindness related issues that might occur can be mostly

covered by having the WCAG accessibility level of AA. Blindness will also be referred to as AR1.1, which is major vision impairments and/or blindness.

FIGURE 3.6 Flow of a video call for elderly people with major vision impairments

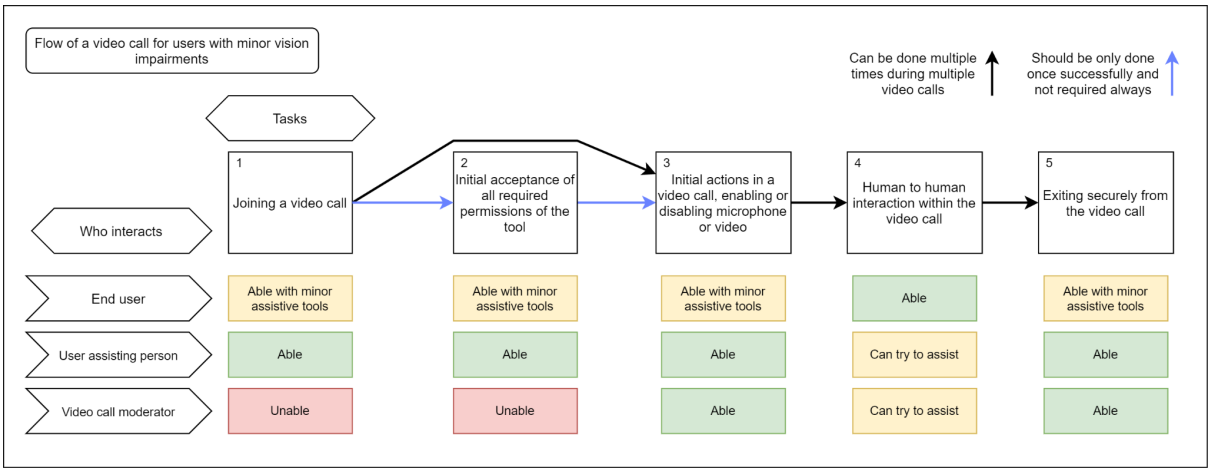


As shown in Figure 3.6, major assistance is required. In order to use the interface in a reasonable manner either a skilled user with blind interface experience or an assisting person is required. This means, for example, support for tabulator scrolling is required in order to switch between elements and to use the interface. Only in the part where human to human interactions are done, the user can realistically rely on themselves [29]. Assistance to the end user can be given by either the assisting person at any time and within the video room done by the room moderator. In this case, it shows that it would be ideal that the end user would not be relied upon as an actor, but rather the responsibility of managing the video call should be transferred to others if possible.

Low vision users

People with difficulties related to their vision. This includes poor acuity, which means vision that is not sharp, tunnel vision so you can only see the middle of the visual field, central field loss, which means the ability to only see the edges of your visual field, and clouded vision [13]. In order to make it easier for low vision users to use the system, the system should support as much zooming or scaling of the content as reasonably possible. To achieve the WCAG level of AA related to scaling the content, at least text should be able to be scaled up to 200% without loss of functionality [10]. This kind of impairment can either be classified as AR1.1 for major vision impairments or AR1.2 for minor vision impairments, depending on the severity of how poor the user's vision is.

FIGURE 3.7 Flow of a video call for elderly people with minor vision impairments

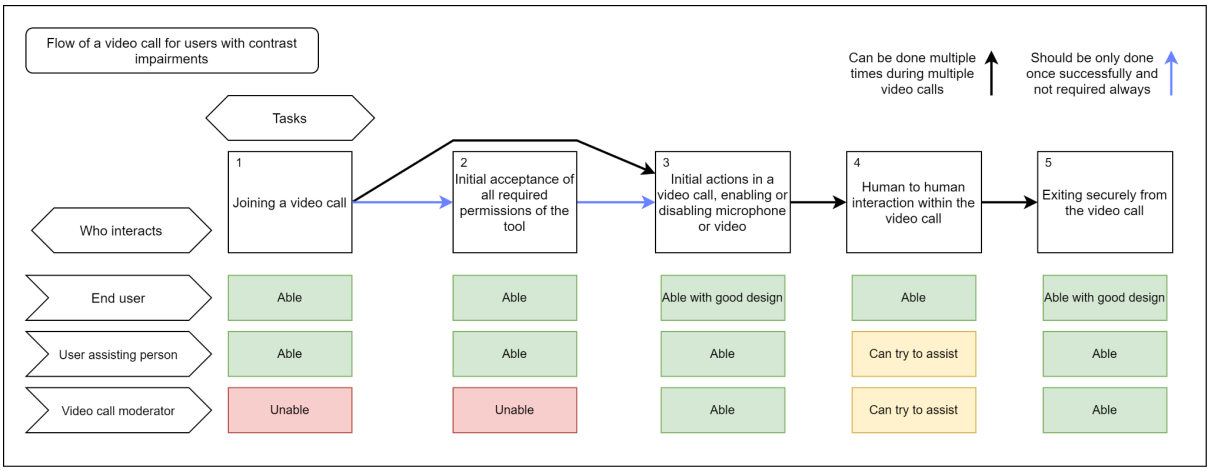


Shown in Figure 3.7, as an end user with slight issues with their vision uses the system, there is a lot more leniency in how solutions can be done. The biggest requirement these kinds of users bring is that the content on the screen is either able to be scaled bigger or a specific setting in order to deal with the lesser able vision can be compensated for. Assistance might not be required by the personnel, but it might still be valuable for the user as reassurance.

Contrast vision impaired users

Users with poor contrast vision, for example due to medical conditions such as cataract or glaucoma, can have a wide range of impairment. For example people with cataract in its early stages have a slightly harder time detecting letters with poor contrast compared to an average person [11]. This issue is amplified with the use of smaller letters, so contrast becomes even more important in those cases [11]. These effects only become worse when these types of conditions get worse. WCAG or Web Content Accessibility Guidelines [10] has guidelines in order to achieve certain levels of accessibility determined by it.
To calculate relative contrast, WCAG uses the formula of “(L1 + 0.05) / (L2 + 0.05), where L1 is the relative luminance of the lighter of the colours, and L2 is the relative luminance of the darker of the colours.” [10]. Regarding contrast, WCAG suggests that normal text should have at least a contrast ratio of 4.5:1 and large text the contrast ratio of 3:1 in order to reach the second best AA level of accessibility. To reach the best AAA level, text should have a contrast ratio of 7:1 and large text the ratio of 4.5 [10]. Users with poor contrast vision fall under the AR1.3 category, which is users with impaired contrast visibility.

FIGURE 3.8 Flow of a video call for elderly people with contrast vision impairments



As it is shown in Figure 3.8, contrast impaired users should have less difficulty in actually viewing the content on the screen, but differentiating elements, especially ones which are close in colour with each other and placed next to each other can cause issues in usability. This should already be designed within the system by default, that the colouring is as user friendly as possible. Colour schemes can be also expanded to support more specific or severe types of colour blindnesses for example as settings. Support from other people should not be necessary, but it can always be a good thing to have it as a possibility.

3.3.2 Cognitive, learning and neurological disabilities

These impairments are mostly neurological, behavioural and mental health related. This means these can affect the users ability to hear, speak, move, see and how they understand information. These issues can, but do not have to affect the intelligence of the person [13]. People with these sort of impairments can be gathered to the group AR2, people with cognitive based impairments.

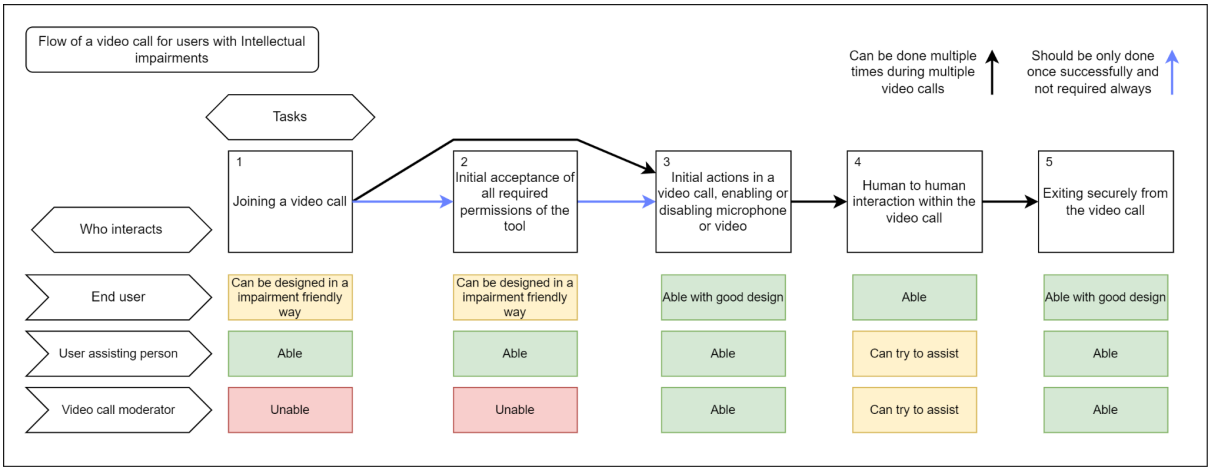
Intellectual disabilities

Also called “Learning disabilities” in European countries, which includes for example Down syndrome. These generally involve impaired intelligence, slower rate of learning or difficulties comprehending complex concepts. There are many different causes of intellectual disabilities, where Down syndrome is one of them [13]. Ultimately it is hard to have a single evaluation of severity with these issues due to the nature of the issue being a wide range of effect on the person [23].

While these are hard to pin down to a single medical condition, the effects are usually very similar. Complexity of the system becomes difficult to handle and reduces the accessibility

and UX suffers. Also having complex wording and sentences that are hard to read and uncommon words that are difficult to comprehend can cause issues to these users [13]. Also use of visualizations such as images, graphs or other illustrations help to contextualize the content [13]. Users with these sort of impairments can be referred to with the AR2.1, Intellectual impairment or a learning disability requirements.

FIGURE 3.9 Flow of a video call for users with intellectual impairments



As visualised in Figure 3.9, intellectual disabilities may range heavily within the group itself, but tasks most complicated might prove to be too difficult for these users. For example the convoluted joining and permissions giving comes usually from the system used, for example browser and the mobile device OS, which are out of control for the developers. Only thing that can be accounted for is the introduction of the permission, explaining how to proceed and possibly providing some assistive pictures or such.

Memory impairments

These impairments include limited ability to recall things from short-term memory, lack of long-term memory or limited capability to recall language. For example, dementia is one cause of these memory impairments [13]. Similarly to Intellectual disabilities, but additionally memory impaired users have hard time handling complex system structures, so the developed software should favour simple low UI depth solutions. AR2.2 Memory impairment or a memory affecting disease, is what can be used to refer to users with this kind of impairments.

FIGURE 3.10 Flow of a video call for users with memory impairments

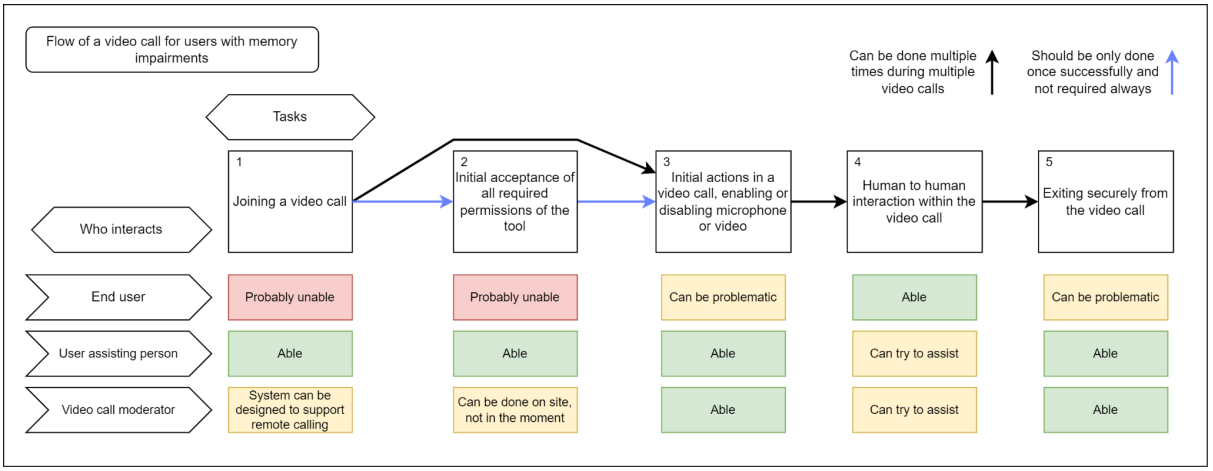


Figure 3.10 displaying the effects of memory impairments. Depending on the severity of the case, certain things can be difficult to do, but for example within elderly people memory problems usually mean that the solution almost has to work automatically itself. If user agency is required in those cases it means that the product is usually impractical. When dealing with less severe cases of autonomous people, they can be classified similarly to people with learning disabilities, where the system should be as simple as possible.

3.3.3 Motor disabilities and impairments

Also known as physical disabilities, which includes limitations and weakness of muscle control. This can also include involuntary movements, tremors, poor coordination and paralysis. There can also be joint disorders, for example arthritis, pain which limits movement, limbs that are missing and limited ability to sense touch [13]. While it is hard to explicitly say which issues are minor or moderate since they usually depend on the severity of each issue, the major issues are easier to identify with the inability to use certain parts of the body to interact with the conventional interface. The things that have to be considered for

these kinds of impairments can be things such as having complete keyboard support for your system, since the same principles which are used for keyboards usually can be extended to be used for specialised tools which these users can use [13]. Additionally having short time limits to respond and complete tasks can be detrimental since the users with this kind of issues might not be as dexterous as some other users could be [13]. Navigation should also be made in a uniform predictable way, since complications with such elements can create barriers of use for certain people who cannot navigate these systems the regular way. Also having all content with alternatives of text will also help these users, since text content inside pictures or in other pages can be inaccessible for these people [13]. Users with general physical impairments, can be referred to with AR3, physical impairments.

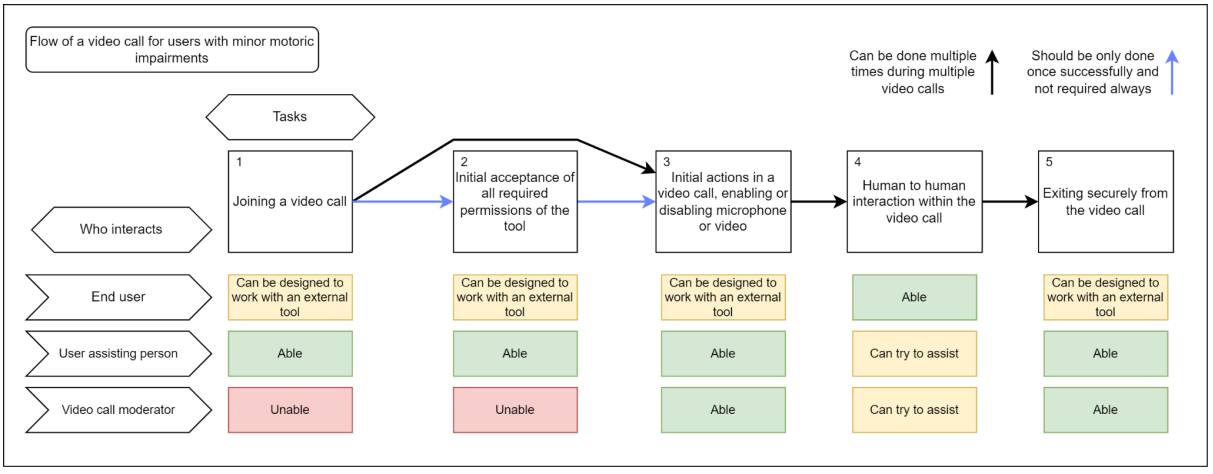
Rheumatism

This can refer to Arthritis, which generally causes degeneration, inflammation and damage to joints [13]. Alternatively rheumatism can also refer to bone and joint pain or even Fibromyalgia or other causes for soft tissue or muscle pain [13]. These people fit the AR3.1, Users with minor motoric impairment.

Reduced dexterity and muscular dystrophy

These refer to the aging process doing negative effects to the body, such as loss of ability with hand-eye coordination. Additionally this can also mean that the person might suffer from progressive weakness and muscle degeneration [13]. These users also fit the category AR 3.1.

FIGURE 3.11 Flow of a video call for users with minor motoric impairments



As shown in Figure 3.11, issues with motorics which do not completely disable the usage of hands might prove issues with extended usage of the motoric functions, but thankfully the video call does not require much micromanagement of the system to be useful for these people. But if an assistive device is needed, there should be support for that to be used. These can in reality mean tabulator controls for the page as well as clear and simple designs in the page.

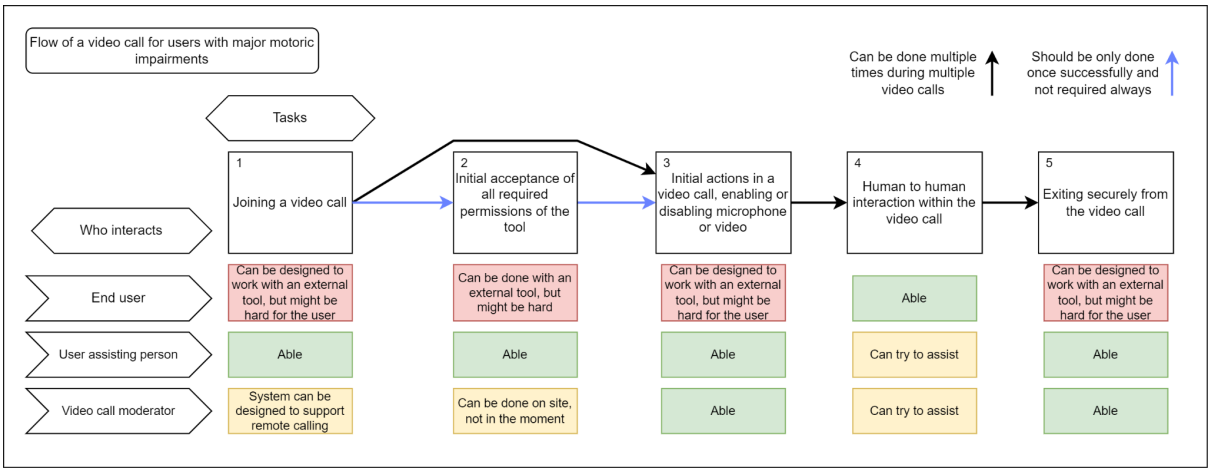
Tremor and spasms

This problem is usually recognizable by its involuntary nature of movements or muscle contractions. These can even be continuous or rhythmic muscle contractions [13].
These are users with AR3.2, Moderate motoric impairments.

Quadriplegia or other major motor impairments

Quadriplegia or amputations can create such barriers to use that the system cannot be used in the original intended way and has to be navigated with an external tool or device in order to accommodate the user's needs [13]. These can be classified as AR3.3 Major motoric impairment.

FIGURE 3.12 Flow of a video call for users with major motoric impairments



Visualised in Figure 3.12, major impairments cause users to have a generally difficult time in any motoric maneuver, therefore it would be a good idea to actually make the system automatic in the end users perspective, moving all of the responsibilities of the video call for the managing staff. This allows difficult tasks for the end user to be simple tasks for the trained staff.

3.3.4 Lack of technological literacy skills

There are a few common factors related to technological literacy, or rather technological illiteracy among especially elderly people. As a generic term, these can be identified as AR4, weak technological literacy skills.

Anxiety which kills the desire to explore

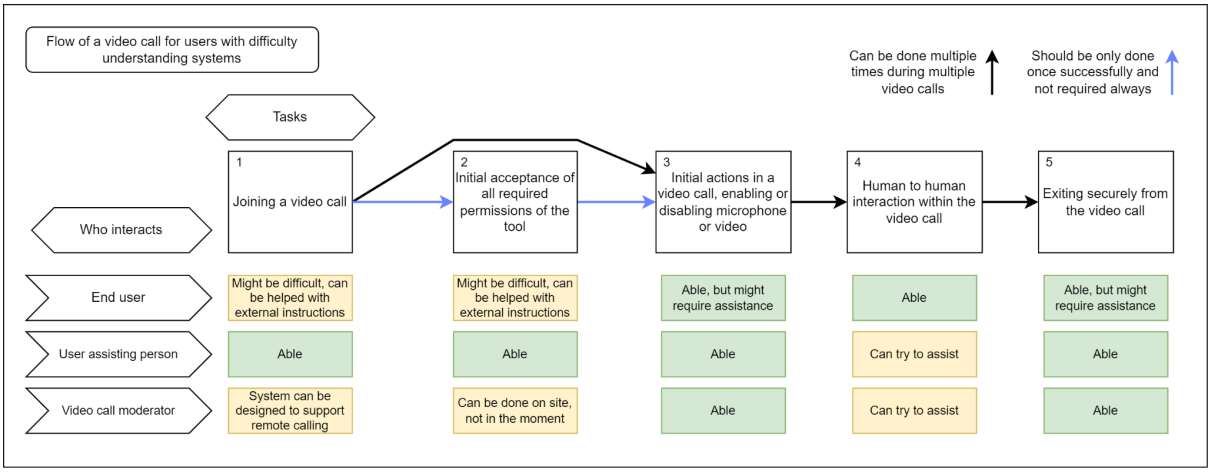
Users might feel overwhelmed by the feeling of anxiety, since it is a common concern that something that they might do can essentially break their experience. Routines might become anxiety inducing if there is a fear of things going “off script” [14]. This becomes especially evident when users are required to transfer their skills between devices such as a desktop and a laptop or moving to use different kinds of mobile devices, like smartphones from two different manufacturers [14]. There is also the fear of dangerous online interactions and since people who are not technologically very literate, that is why they do not have a basis of understanding how internet threats work, so these users do not have a model for how to minimise the threats that online might impose, therefore everything can be a possible risk factor [14]. The users are usually also very afraid of admitting that something is not working for them, because it is hard to differentiate between a self caused situation and something the system has done wrong itself, or better known as a bug. Also for these users it is hard to explain the situation they are in, so asking for help becomes much harder for them which furthers the probability that the user is content with staying stuck rather than asking for help [14]. And due to this, these users are reluctant to explore the software product, which removes partly the ability to learn the whole system, since these users are content in staying in their confines. This issue is also further causing problems for these users, since more and more these new devices and systems reward exploration, since there is less commonality across interfaces [14]. In order to improve the elders' efficiency to handle tasks, the user interfaces should be very verbal in order to explain better what everything is and means in a way that is understandable for the elderly person [32].

There is also the lack of knowledge of how technology is built with standards, which are not so obvious to complete novices. These for example can include desktop computers clicking, double clicking, scrolling, clicking and holding, and so forth. These become even more ambiguous for mobile devices with touch screens, which have even more small unique maneuvers which might allow you to navigate the system further. Knowledge of these hidden standards is quite absent and can cause even more confusion, anxiety and the feeling of not being in control if these were to be activated by accident by an inexperienced user [14].

Furthermore, these are hard to learn and hard to determine when and where to apply for example mobile gestures.

The shift in the direction of minimalistic “clean” interfaces can also cause problematic things for users with little skill, and unwillingness to explore, since for example icons might replace text explanations of features, and you might even need to hover over a feature to possibly see a text explanation of the feature, if the system is well built. Additionally, since mobile design prioritises the users to be able to see the most important elements by default in the neutral state, or the state where you are in the software without any interactions. That might lead the user to never question the state of the system, since everything they usually need is in hands reach, but they never realise not so apparent actions such as scrolling or gestures, which might lead to more features or for example scroll your menus content [14]. In order to improve readability of the system, it can be useful to reduce the number of features in available at a time, interface should not hide any features, larger components help readability, avoidance of computer terms and usage of descriptive texts are helpful [33]. The ability to read a situation is identified as being literate, therefore if the user is unable to do that in a system, it should be identified as an AR4.1 system literacy issue.

FIGURE 3.13 Flow of a video call for users with with difficulty understanding systems



As it can be shown in the Figure 3.13, the users without technological skills can use the system, but it builds trust and confidence if the system can also be supported by outside help such as instructions or some sort of tutorial. This allows people to not fly in blind and have an easier time and safer more fulfilling experiences. Target audience of elderly people might have difficulties with the system if there is not a paper version of the instructions available.

Systems have become too abstract

The move towards mobile devices and things like cloud-based services and storage abstracts the activities into general “services”, which causes problems with technically illiterate people, since they prefer to work in their “scripted” way, with their specific device and their own trusted technique, where things change as little as possible. So when similar features or meanings come in different contexts, they might confuse these users even further.

This causes these users to not be able to transfer their already learnt skills in other systems to a new context. The new environments just cause the users to become lost [14].

Especially elderly people are used to learning things in a specific way, where they learn a “script” where they remember the layout of interfaces. So a new device with self-explanatory design can be hard for users with little patience for teaching themselves to learn. Especially if they have an already familiar and comfortable existing system.

Changes in interface causes people to be unhappy, and especially due to the previously explained reasons this is even more so the case in elders.

Technically less competent people do not use the same services with different devices since they associate them by device. The PC might be the weather checking tool, the tablet might be used to check the news and the phone is used to listen to the radio. These kinds of users are prone to thinking that the same functionality is different on different devices, even though it might be the same application and the same interface [14][31]. Literacy of the system can be improved with a systematic messaging system, which allows users to have a clearer and more expected way to navigate the system [34].

Overall, the issue is that even though for technically literate people, these conventions might be useful and easy, that might not be the case for all users. So instead of familiar looking controls making users feel comfortable, every new environment feels like a completely new world with its own intricacies to learn and a big chore to tackle. If the system is too hard to comprehend on an abstract level, the user is then suffering from an AR4.2, system abstraction issue.

FIGURE 3.14 Flow of a video call for users with with difficulty understanding system abstraction

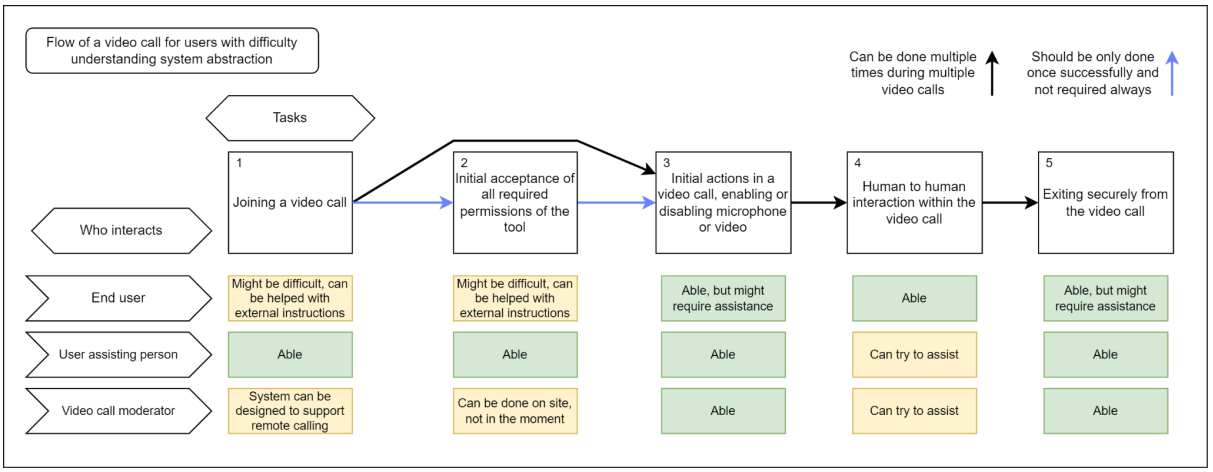


Figure 3.14 shows when it comes to understanding the system in an abstract level, it is not per se needed for the end user, but it might help using the system in a more non-scripted way, where user is able to make their own decisions about flows of things rather than having to do everything in a manner which is always the same. These might create issues which can not be solved in any other way than the end user following a specific script to follow instructions correctly, in those cases the use cases should be made as simple for these people as possible, so the ability to do something wrong is minimised for the end user.

Lag can delay adopting new things

Lag, for example in rural areas can cause systems to behave less than optimally. This in part causes slower adoption of new technologies. Slower or even non-existent internet can be a reason for unwillingness to try to stay in touch with current technology, since you can be quickly left behind, especially if you are elderly. It is important to remember that these kinds of people might use, by default, technologies that have worked for them, which might be older equipment that they have found to work in the past.

Elderly people can view technology as things that should work now and should work the same in the years to come, which makes them quite reluctant to change their equipment to something new or different. Also technology might be viewed as luxury or a displeasing necessity of the modern world [14].

Vocabulary can also cause issues with people who are not accustomed to technical or terms related to technology. For example, elders might not know what a browser means, they might rather understand that to be just the “internet” and not the tool to view the internet.

There is also the risk that when people lag behind in acquiring new technologies, their skills and knowledge can become outdated or even obsolete when it comes to new technology. To reduce the effect of this phenomenon elderly users should be introduced to technology as soon as possible, which allows them to adapt more easily [28][31]. When the problem lies in the overall experience of the user and they do not have any backbone of knowledge on how modern systems usually work, they can be categorised by AR4.3, inexperience of modern practises.

3.3.5 The accessibility requirements of video calls for elderly people and the supporting people around them

In order to create requirements based on different special needs user groups, they will be referenced with accessibility requirements (AR). They are based on the groupings described in the previous chapter. These are based on Web for all user accessibility stories and the various accessibility requirements proposed by different standards, such as WCAG (Web Content Accessibility Guidelines), UAAG (User Agent Accessibility Guidelines) and ATAG (Authoring Tool Accessibility Guidelines) [13].

AR1 Vision based impairments means that the users ability to see with their eyes has been hindered in some manner. This requires visual and or other accommodations in order to make the systems more suitable for these users.
AR1.1 Major vision impairment or blindness. Grouping of users with blindness to very impaired visibility, which can be classified as legally blind. While these two can be very different in how they operate a system, similar methods of helping overcome issues they face can be offered.
AR1.2 Minor vision impairment or low vision. Users with small hindrance in vision, which is still a big enough factor to be considered something worth taking into account. These changes usually require scaleable content, where users can either zoom the content in, or enhance the size of the elements in the UI.
AR1.3 Impaired contrast visibility. Users with certain medical conditions which causes the user to have a hard time differentiating between colours or knowing where a certain colour ends where the other colour starts. This is somewhat comparable with certain types of colour blindnesses.

AR2 Cognitive based impairments. Users with hindrance in their use of their cognitive abilities, where they can have a harder time learning, remembering or overall understanding the system.

AR2.1 Intellectual impairment or a learning disability, which means users have less mental capacity to learn or understand the system, where this makes navigating and understanding the system a much harder task than normal.
AR2.2 Memory impairment or a memory affecting disease. This means users can not rely on their memory in order to effectively use the system. Therefore having as little to remember at a time as possible is important.

AR3 Physical impairments, which covers users with a certain degree of loss of physical abilities or other factors which cause difficulty in usage of the human body itself.
AR3.1 Minor motoric impairment, eg. reduced hand dexterity or rheumatism. Users with small issues with their physical abilities, which might deter the users from having long enduring sessions where they have to use their body actively to navigate systems
AR3.2 Moderate motoric impairment, eg. tremors or spasm. Group where there is a clear factor in manipulating the UI, clicking and hand precision overall is greatly reduced and certain tasks might become very hard or even impossible without accommodations.
AR3.3 Major motoric impairment, eg. quadriplegia. Users with the complete loss of function in their critical body parts, which are used to navigate the systems under normal conditions. These require heavy accommodations, where for example mouth or some other body part can be used for navigation and interaction.

AR4 Weak technological literacy skills. Inexperienced users, especially elderly people, suffer from the gap of where they are technologically and where modern technology currently is. This occurs in many ways, with usually inability to use a system comfortably as the result.
AR4.1 System literacy issue. Users with the inability to “read” the system in a way where they would understand the status of it, how to navigate it, or if there are some common practices which they would be able to use in their favour.
AR4.2 System abstraction issue. Users with issues in grasping how the system overall behaves in a practical sense for them. For example being able to grasp the concept of cloud services, where a user can access the same things from multiple endpoint devices.
AR4.3 Inexperience of modern practises. Group of people, who do not have the practice and depth of knowledge to understand and use transferable knowledge from system to system. For example the search function and the magnifying glass icon related to it, might always present itself as a unique challenge for these users.

4. Evaluating the purpose of current video call solutions and effects the their UX

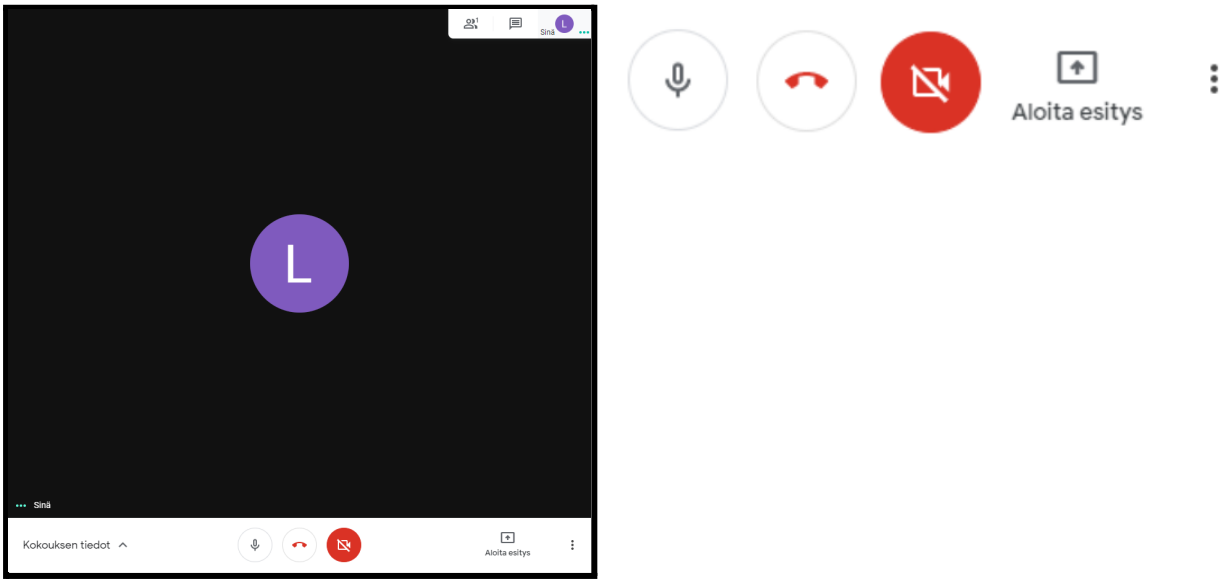
To evaluate video call solutions and their strengths and weaknesses considering UX, it is important to evaluate what are the most relevant parts of the user experience. It is also important to note, that these preferences are different from user to user, and there is no one single right answer. This chapter will answer the **RQ1** “How do current video call solutions compare when considering social interaction for elderly people?”
For example, one important thing about video call solutions is the **Visibility of system status**. It is relevant for the system to communicate to the user about what state they are in, especially regarding their status of output. Video calls can send video and audio, so it should be quickly recognizable to the user, which mode they are in, sending or not sending. This is quite important to easily visualise if they can be heard or seen, since both of these parts are important for video communications. It is also important to minimise the risk of accidental output, showing or leaking unintended audio, so status of that should be easily available for users at all times. Features of the video calls will be only briefly mentioned, but they will be expanded within chapter 4 later on.

4.1 Video call solutions compared in this work

To review current video call solutions, there should be a list of solutions to compare each other to. In this work, the video call solutions which will be compared will be Zoom, Microsoft Teams, Google Meets, Jitsi Meet and Vooler. Zoom, Teams and Meets being market leaders, Jitsi being an open source free to use video call solution and Vooler, which this work is centred on improving. Vooler is based on the Jitsi open source code and in an enhanced version of the same core solution.

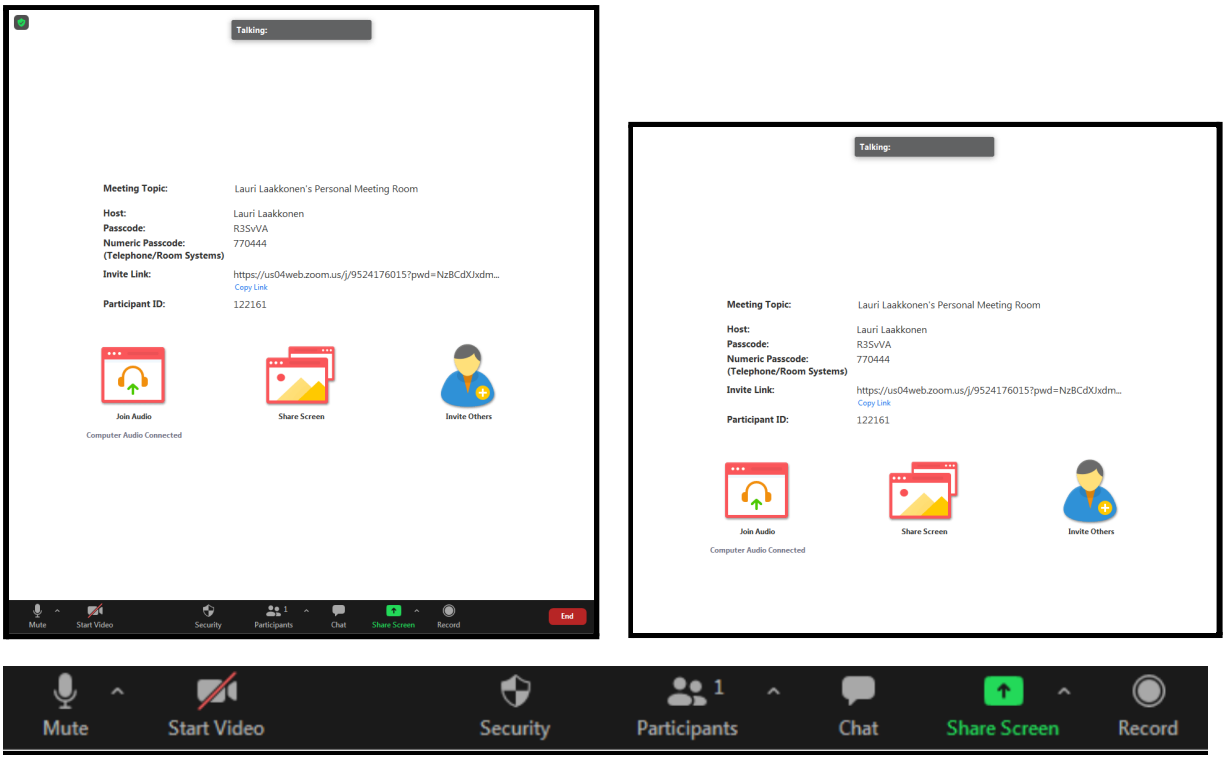
In this chapter we will take a quick look into each of these systems.

*FIGURE 4.1 Visualisation of **google meet's** call UI*



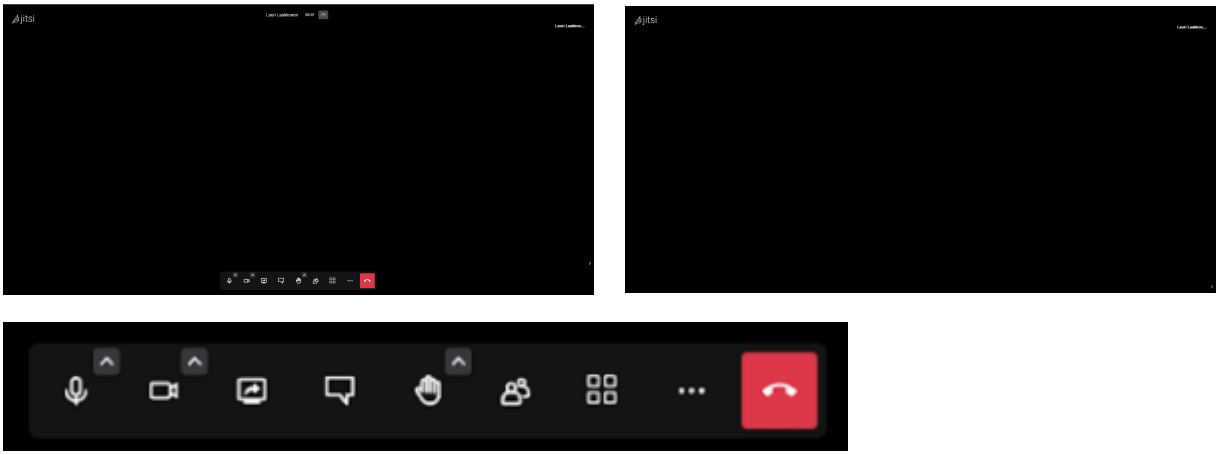
As shown in Figure 4.1 Google meets has a simple design and it can be classified as a modern tool for modern people. Quick and very simple solution from Google, which offers a basic video call experience. Good simple usage of colours and icons. Default user is assumed to have a google account.

*FIGURE 4.2 Visualisation of elements in **Zoom's call UI***



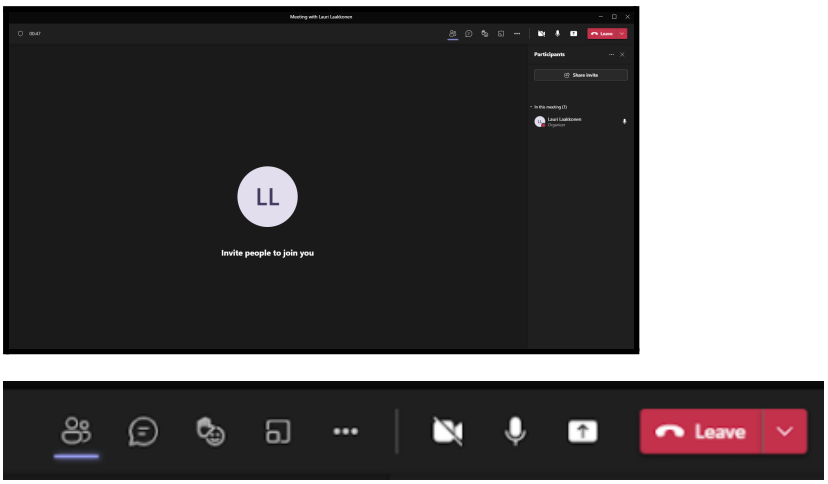
Zoom video calls are more information filled, which can seem somewhat less sleek, but overall it is still a very good design for the same use as google meet. Figure 4.2 shows the expanded design of the Zoom UI, which has many small features which are useful for many users, though not always needed by all.

*FIGURE 4.3 Visualisation of elements in **Jitsi's call UI***



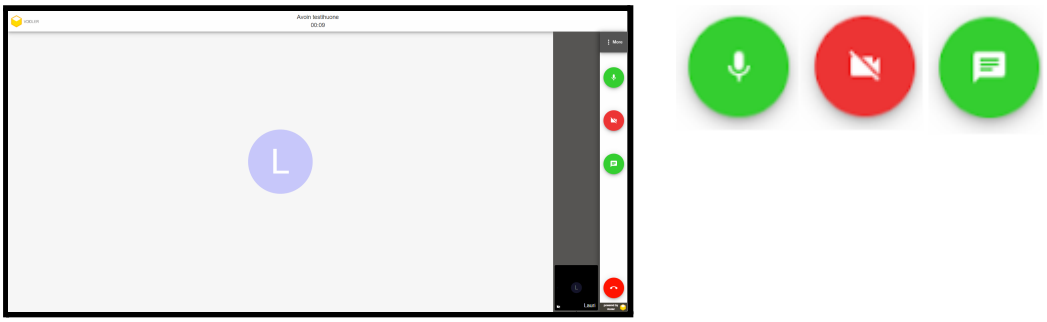
Jitsi, a middle ground between Google Meet and Zoom, where style goes over functionality. Buttons have icons and expanded explanations if you hover over the functions. Also worth noting, Jitsi is an open source project, by their respective community. Figure 4.3 shows that the extent of features is quite large for Jitsi and some could even say its overloaded with features.

*FIGURE 4.4 Visualisation of elements in **Teams**' call UI*



Teams offers similar experience to google meet, and is often used in professional settings due to having built in calendar tool and chat channel support. Figure 4.4 shows that similarly to Jitsi, the UI is made in a simplified manner, which can be somewhat difficult for users to use when the number of features is increased.

*FIGURE 4.5 Visualisation of elements in **Vooler's** call UI*



Vooler is the system which tries to target audiences of eldery and people less capable with technology. This is to support services and content which is reliant on video calls and to provide that to elderly people. As shown in the Figure 4.5, the content of the video call features is very limited and contains only the most important things on the end user UI. Vooler is based on the Jitsi open source project, but improves on it by providing systems around it to enable different new ways to use video calls. For example different ways to join a call itself. This was also partially done in response to Covid-19's new challenges that were brought to especially elderly people with increased safety precautions and social distancing recommendations [30].

Vooler aims to fix fundamental issues with other solutions. It is as simple for basic users as possible. The colouring scheme and other designs are made in a new user friendly manner, for example colours are welcomingly white and buttons have clear status with colouring. It also tries to tackle issues with the hardest part of using this technology for unskilled users, joining the call. Currently the default choice of link and room codes are there, but there is also the function to do direct calls for the end user. This direct call can be configured so that the call automatically is answered, after which the video call can be used without the end user's interaction with any technology.

Objective of Vooler is to offer a video call for users, which could replace live meetings, especially those, which are not possible due to Covid. It is also a tool for nurses, to be able to do more in a more efficient manner. Meetings which do not have to be made in person, can be significantly sped up by just connecting remotely, which eliminates travelling and switches that to connecting.

4.2 Comparing the solutions

Comparing the solutions will be done with Jakob Nielsen's 10 usability heuristics and by using the Chapter 3 Usability requirements and Accessibility requirements. The solutions will also be compared with each other. The scope of elderly people and special needs groups will be taken into account as well, to understand if the solutions have support for them or if the purpose of the solutions is mostly for generic, non-aided use and if this has effect on the general user experience of the system. Evaluation is done by the author, using heuristics and requirements which are determined in chapters 2 and 3.

FIGURE 4.6 Format of table comparison in this work

Scaling of following figures	
Rating	Example text description
Black, nonexistent	Not implemented
Bright red, bad	**Low viability**
Dark red/brown, ok	**Some viability**
Dark green, good	**Viable**
Bright green, best	**Very viable**

Figure 4.6 will be used as a scale for the quality of the solution. These will be after important segments in chapter 4 as pictures to allow easier visualisation of the solutions.

4.3 Nielsen's heuristics of usability in video call solutions

As discussed in section 2.2, Jakob Nielsen's ten heuristics can be used for general evaluation of usability in software. These will be evaluated individually for each system, but having focus on certain fields is also logical, since not all heuristics are as important for certain systems as others. To understand the discussion below, refer back to section 2.2.

And check each corresponding heuristic. All relevant heuristics are also opened up as a figure after each heuristic discussion. The evaluation of solutions has been done by the author of this work.

1. Visibility of system status

Each system has quite comparable system status information outside of the video call solution, each following quite standardised protocols for page changes and such. Inside the video call Zoom and jitsi work similarly, good UI which is hidden when user does not interact with the system, which in turn makes it less desirable. Interaction in this context means moving around a cursor or tapping a mobile device's screen. Meet, Teams and Vooler have a constant UI which does not require interaction to use its core features. This is important since two statuses which the user is always using are their microphone and camera, if either one of them is not behaving as they are expecting, it can lead to uncomfortable use.

FIGURE 4.7 Visibility of system status

Visibility of system status					
	Zoom	Teams	Google Meet	Jitsi	Vooler
Core functionalities outside of video calls	**Works well using industry standards**	**Works well using industry standards**	**Works well using industry standards**	**Works well using industry standards**	**Works well using industry standards**
Core functionalities inside video calls	**Works excellently, has good design for all users**	**Status is shown as an icon**	**Status is shown as an icon and color**	**Status is shown as an icon**	**Status is shown as an icon and color**
Video call requires interaction to see core features	**Requires interaction to see the core featues**	**Always shows core features without interaction**	**Always shows core features without interaction**	**Requires interaction to see the core featues**	**Always shows core features without interaction**

2. Match between system and the real world

Video call solutions all use a quite similar style in their selection of icons and phrases. Only major difference between Teams, Meet, Zoom, Vooler and Jitsi is that the meeting room code can be customised in Vooler and Jitsi. This allows use of codes which are normally incomprehensible for users to be used in a more natural manner. For example if the meeting room is held in a code or link called “MEETINGROOM”, it is much clearer what needs to be typed and can be even instructed over a telephone call easily.

FIGURE 4.8 Match between system and the real world

Match between the system and the real world					
	Zoom	Teams	Google Meet	Jitsi	Vooler
Use of icons in and out of video calls	**Works well using industry standards**	**Works well using industry standards**	**Works well using industry standards**	**Works well using industry standards**	**Works well using industry standards**
Use of technical terms and their understandability	**Works well using industry standards**	**Works well using industry standards**	**Works well using industry standards**	**Works well using industry standards**	**Works well using industry standards**
Ability to customize content to make it more understandable	**Content is not customizable**	**Content is not customizable**	**Content is not customizable**	**Allows the customization of links which are used to join calls**	**Allows the customization of links and codes which are used to join calls**

3. User control and freedom

User control and freedom is important for users who are not reliant on others for basic use. Importance of this is mainly professional or similar usage. The solutions offer very similar experience on professional use, so comparing yields very similar end results.

4. Consistency and standards

Consistency and standards are also very well utilised in all solutions. This means there are very little meaningful differences between all solutions. Similar features can be found in all solutions, the only major difference is the additional features Vooler introduced in order to cater to specific user groups.

5. Error prevention

Error prevention can be categorised in two significant ways for video calls, if the error is prevented in advance by doing an additional confirmation for dangerous actions and if the user can recover from said dangerous actions quickly. Zoom and Vooler has the best with both things considered. Meet and Teams have no initial error prevention, which can in some contexts be considered a good thing, but they do have a recovery mechanism for erroneous dangerous actions. And Jitsi has weakest error prevention from all, for example leaving a call is a single button action which can be only recovered from by joining the call again from step 1 of the call joining process.

FIGURE 4.9 Error prevention

Error prevention measures					
	Zoom	Teams	Google Meet	Jitsi	Vooler
State of error prevention	**Well error prevented, dangerous actions have to be confirmed**	**All dangerous actions do not require confirmation, but they can be recovered from**	**All dangerous actions do not require confirmation, but they can be recovered from**	**Weak error prevention, weak ability to recover from errors**	**Well error prevented, dangerous actions have to be confirmed and recovered from**

6. Recognition rather than recall

This heuristic can be considered in a few ways. Whether or not all of the core functions are always visible, so that there is no need to recall the user interface at any point, if it is always showing for the user. Second thing to consider is the explanation of each feature and button. Zoom and Jitsi have an UI which hides in use, but Teams, Meet and Vooler does not. Zoom is the only solution which has constant text explanations for all features where others have a hover or hold on tablet explanations for buttons.

FIGURE 4.10 Recognition rather than recall

Visibility of system status					
	Zoom	Teams	Google Meet	Jitsi	Vooler
Video call requires interaction to see core features	**Requires interaction to see the core featues**	**Always shows core features without interaction**	**Always shows core features without interaction**	**Requires interaction to see the core featues**	**Always shows core features without interaction**
Functionality description text	**Always shown below the button**	**Requires hovering cursor over feature to see text**	**Requires hovering cursor over feature to see text**	**Requires hovering cursor over feature to see text**	**Requires hovering cursor over feature to see text**

7. Flexibility and efficiency of use

The solutions all offer ways to speed up use, but while Zoom, Teams and Meets use quite complicated two or three button keybinds, Vooler and Jitsi use single key binds. This can be seen as a bad or a good thing. If the quick commands are too simple, they can be accidentally triggered, which can cause unwanted issues. While complicated keybinds make the use of the system necessarily hard. Vooler counters this issue by disabling end users from being able to use keybinds, which can be a restriction on user freedom, but it is done to

simplify the experience. Outside of the video calls, the efficiency factors are quite comparable between the systems.

FIGURE 4.11 Flexibility and efficiency of use

Flexibility and efficiency of use					
	Zoom	Teams	Google Meet	Jitsi	Vooler
Complexity of keybinds	**Multi key binds, harder to learn and use**	**Multi key binds, harder to learn and use**	**Multi key binds, harder to learn and use**	**Very simple single key binds**	**Very simple single key binds**
Error possibility with keybinds	**Very unlikely**	**Very unlikely**	**Very unlikely**	**Easy to do unintentional inputs and cause errors**	**Protected from end users, but users with special access can cause errors for themselves**

8. Aesthetic and minimalist design

As with other heuristics which follow industry standards these also follow industry standards quite well. Designs are done in a simple manner, excess is cut off and overall all designs in their own unique ways have minimum required visual clutter and text contents to make the system work fine. All in all, all solutions are comparable in aesthetic and minimalist design.

9. Help users recognize, diagnose, and recover from errors

Error recognizing, diagnosing and recovery is something which Google Meet handles very well. Error situations are autodetected and correction measures are given on screen immediately when an issue is detected. Teams and Zoom have less errors overall due to their application nature. In case of errors they perform manageably, but they are not as good as Meet. Jitsi and Vooler have the least comprehensive error prevention, issues are not always error coded, so they are handled as unexpected errors.

FIGURE 4.12 Understandability of errors which users recognize, diagnose, and recover from

Help users recognize, diagnose, and recover from errors					
	Zoom	Teams	Google Meet	Jitsi	Vooler
Quality of error recognition, diagnosing and recovery	**Good quality**	**Good quality**	**High quality**	**Average quality**	**Average quality**

10. Help and documentation

Similarly to heuristic 9, the systems follow a very similar pattern in help and documentation. Google does the highest quality documentation with good compactness and information quality. Zoom has multiple portals with confusing layout, but it still contains mostly all important information. Teams has a long list of features which has certain simplicity to it, but still it can be hard to navigate. Jitsi does not offer very good help documentation, since it is largely incomplete, but the quality of the contents is good. Vooler has both text and video content in the help and documentation, with quality of the contents being decent.

FIGURE 4.13 Quality of help and documentation

Help users recognize, diagnose, and recover from errors					
	Zoom	Teams	Google Meet	Jitsi	Vooler
Quality of help and documentation	**Good quality**	**Good quality**	**High quality**	**Partially missing, good quality**	**Good quality**

4.4 Usability requirements in the solutions

Usability requirements or UR's for short are going to be evaluated next. Note that the requirements are evaluated with the scope of well rounded usage for elderly people in video calls. That means that the original target audience might not be elderly people for the solution, except for Vooler, which has the main target audience of elderly people. These solutions will also be discussed in general, so it should also be evaluated on "what is the role of current video call solutions when considering social interaction".

UR1. Routes to a call and accommodations or limitations of users

Zoom has multiple solutions to have participants join a call. Users mainly use links to join, which directs them to use the desktop application, web client or the mobile application on mobile devices. Users can also use a random 9 to 11 digit code which can be used in application or web client, the digit is entered and the user is directed to the correct video room. There is also a room invite message system, where if you are online, you can receive direct invitations to meetings. This can be used on applications only.
With mobile devices with calling functionality, you can also directly call a zoom call if the user knows what the teleconferencing number is. There is also a H.323/SIP connection with IP addresses, which is extremely complicated considering the scope of the work, so it will not be elaborated further, but it allows users to join a zoom call using the zoom app and correct IP address.

Teams uses three methods of joining, it has support for mainly links, which can be embedded to the teams app or calendar events. They can be opened with Windows app, browser or Teams mobile app. It is quite limited in the department of different joining methods, but its main audience is professional use where it is acceptable to such.

Meets is similar to Zoom in its joining features. It has basic link support with compatibility with Google calendar, Gmail and other Google services. The link gives you access to the meeting room where the room owner either lets you in or keeps you out.
Additionally you can use google meets codes, which are random 10 character codes, which are inserted in the web client or mobile application in order to join a meeting.
There is also phone dialling support for devices which can do phone calls.

Jitsi has the same link and phone call features as google and zoom, but contrary to those solutions, it does not have a “code” joining feature, but rather you can determine the meeting room link, which can serve the same purpose. For example if you have a meeting with an understandable code, then it can be easily communicated to another person, so it serves a similar purpose, but the solution is possibly even better.

Vooler features solutions which are similar, but enhanced versions of other solutions. Links can be used similar to other systems. These links can be embedded to calendar events and such. As with Jitsi there is a customizable meeting room code part of the link, which can be

made custom fit for any meeting. This same code though can be used in several places where it can be entered to join a room.
Additionally the phone calls are replaced by an application calling solution, which connects users by using groups, where they can contact each other using phone call type connections, which do not require any sort of mobile phone number. The calls are done using a contact list where you should only see the people who are relevant for you and calls are done by using just a click of the button. Receiving calls can be done either by manually accepting the call or depending on the client by automatically being answered the call for them.
For ease of access to a recurring meeting in a specific room, you can have a custom made room joining button in the mobile application, which directs the user to a same room, where for example scheduled events can happen on a daily basis. This removes a uncertainty factor about where the user should be going.
There is no phone number call support for Vooler.

FIGURE 4.14 Routes to a call and accommodations or limitations of users

Viability of the joining method for elderly people and special needs groups					
	Zoom	Teams	Google Meet	Jitsi	Vooler
Links	Low importance	Low importance	Low importance	Low importance	Low importance
Built in calendar events	Some importance	Some importance	Some importance	Some importance	Some importance
Code Joining	Some importance	Some importance	Some importance	Important	Important
Mobile phone number calling	Some importance	Not implemented	Some importance	Some importance	Not implemented
Direct app calls with auto answer	Not implemented	Not implemented	Not implemented	Not implemented	Very important

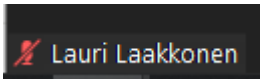
Figure 4.14 shows the system's solutions ranging from no viability to very viable. Additionally if there is no implementation it is shown as not implemented. Links and calendar events are rated lower on viability due to understanding of systems related to said methods, electronic calendars and emails are not prevalent with elderly people [14]. Codes are rated differently based on if the code itself can be customised in order to make it understandable language for the user. For example code MEMORYCAFE is much easier to comprehend and understand than code XHY-JKSD-LMK. Phone calls are usually quite viable in the sense that most people have a phone, but viability as a whole is reduced since the numbers that are

used to call are usually complex and calling with a voice only call defeats the purpose of using a video call.

UR2. Visibility of user statuses in the call UI

Main statuses in the call that the user should be aware of are microphone status (can you be heard), video status (can you be seen) and connection status (can you interact at all in the room).

FIGURE 4.15 Visibility of user's statuses in Zoom

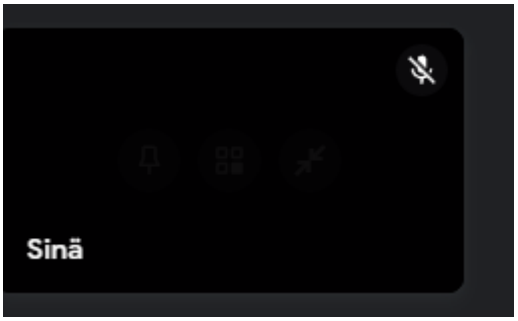


Zoom offers limited information about users status in a call. As shown in Figure 4.15, the only thing that can be determined from the user is their microphone status, which is also shown as a red icon, which is easy to see. The status of video can be also deduced by seeing if the user is showing any video. Problem with not explicitly showing video status is that if there is some issue with the video source, it can not be determined if the camera itself is off. Network status is not visible for users, it can only be observed with the quality of sound and video.

FIGURE 4.16 Visibility of user's statuses in Teams

Teams shows a very simple UI for statuses. As Figure 4.16 shows, similarly to zoom, only microphone state is shown, the status is only shown if the user who is observing these statuses hovers over the Teams app window with a cursor, otherwise the information is hidden. Network status is not shown in Teams.

FIGURE 4.17 Visibility of user's statuses in Meets



Meets as well keeps it simple, Figure 4.17 shows that UI keeps only tabs on the microphone. Camera and network status are not shown in the UI.

FIGURE 4.18 Visibility of user's statuses in Jitsi

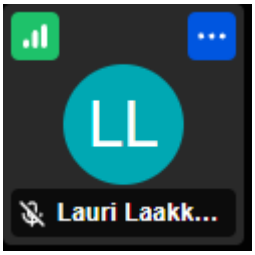
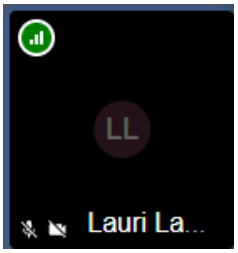


Figure 4.18 shows what jitsi has on hovering. Microphone status is always shown for users, but if the user frame is hovered with a cursor, it also shows network status, which can be clicked and extended for further information. The information includes connection delay, lost packages and other quite extensive information. The network status icon is also shown without hovering if it is less than optimal.

FIGURE 4.19 Visibility of user's statuses in Vooler



Vooler behaves similarly to Jitsi. As shown in Figure 4.19 Vooler has microphone and camera status constantly shown. Additionally as in Jitsi, network status can be seen when hovering over a user's frame or if the connection quality is less than optimal. The content of the additional information in the network button is the same as in Jitsi.

FIGURE 4.20 Visibility of statuses in the call UI for call interaction elements

Visibility of statuses in the call UI for call interaction elements					
	Zoom	Teams	Google Meet	Jitsi	Vooler
Buttons shown	**Microphone**	**Microphone**	**Microphone**	**Microphone**	**Microphone and camera**
Use of illustrative colors	**Yes**	**No**	**No**	**Little**	**Little**
Is network status shown	**No**	**No**	**No**	**Yes**	**Yes**

UR3. Constant visibility and understandability of all basic call functions

To evaluate the call interaction elements, which are the main focus once connection between one or multiple people has been done in a call, it is important for the user to understand their status in the call. The status can be simplified down to a couple of points, such as microphone status, video status and connection status. System status visibility in these solutions can be categorised into two main groups, the solutions with all main elements visible constantly, and those which require interaction, for example movement of the mouse to see. As Figure 4.3 and 4.5 show, Zoom and Jitsi are both not prioritising the visibility of elements at all times, creating a sleeker design and requiring interaction from the user to show microphone, video and other additional buttons on the screen. Comparing this to Figure 4.1, 4.6 or 4.7 for Meets, Teams and Vooler respectively, where all main elements are always visible, allowing users to navigate the system without having to worry about things changing unexpectedly.

Statuses are also designed some with colours in mind and some less so, for example Meets, Vooler and Zoom use colours to determine microphone and video status, which is advantageous for less experienced and less situationally aware users. Common convention is to have red as disabled and depending on the system, something neutral or encouraging as enabled. Jitsi and Teams have a more business appropriate approach of having neutral colours and relies on users who are accustomed with systems overall.

If compared with each other, and considering that the target audience is elderly people and special needs users, Vooler seems like a favourite, with Meets being a close second, Teams coming third with showing statuses but forgoing colouring. Zoom and Jitsi are similar with the hidden UI, but Zoom is slightly better with use of colours in its UI.

FIGURE 4.21 Constant visibility and understandability of all basic call functions

Constant visibility and understandability of all basic call functions					
	Zoom	Teams	Google Meet	Jitsi	Vooler
Buttons shown without interaction	No	Yes	Yes	No	Yes
Use of illustrative colors	Little	No	Some	Little	A lot

UR4. Simplicity of the system, ease of use for non-technical people

To measure simplicity of the system and ease of use, it has to be determined what causes the system to be simple and what makes it complex. Learnability and UI complexity are some factors that come into play when quantifying simplicity. For elders the most important factors are that joining a call is done in such a way that it is easy to understand, easy to learn or can be done for them. Within a call, the same logic applies to call functionalities, buttons should be easy to understand and clear for the user to use or the functionalities should be able to be used for them.

These can be put into two distinct categories, assisted solutions and non-assisted solutions. An obvious difference between the two categories is that the assisted solutions are doing things in such a way where users do not necessarily have to do anything to interact with the call, the actions can be done for them rather than by them.

All applications have moderation tools which can be considered assisted tools, so microphones or videos can be disabled with assistance, but only Vooler offers assisted features outside of that. Vooler allows calling and call answering by the person calling the end user by taking advantage of auto accepting calls. Additionally in call, certain features can be used to remotely enable the microphone, so the user can take part in group calls, where sound ambiance is considered important.

It is also very helpful that the UI does not change dynamically, for example with moving cursor over the application screen, which does happen in Zoom or Jitsi, but not in Meets, Teams or Vooler. For tablets these options are hidden unless the user interacts with the screen, if the user does not touch the screen, the options are hidden as per default. A specific point for Zoom is that it utilises text underneath the buttons, which displays clearly what clicking the said button does, such as “mute” mutes the microphone. Other solutions use an explanation text which only shows when you hover on a button.

FIGURE 4.22 Simplicity of the system, ease of use for non-technical people

Simplicity of the system, ease of use for non-technical people					
	Zoom	Teams	Google Meet	Jitsi	Vooler
Moderation features (things that can be done remotely)	Basic moderation (disabling mic/cam)	Basic moderation (disabling mic/cam)	Basic moderation (disabling mic/cam)	Basic moderation (disabling mic/cam)	Advanced moderation (enable or disable device)
Element visibility	Buttons shown on interaction	Buttons always shown	Buttons always shown	Buttons shown on interaction	Buttons always shown
Text explanations of buttons	Text under button	Shown on hover	Shown on hover	Shown on hover	Shown on hover
Automation for call receiving, auto answering	No	No	No	No	Yes

UR5. Error prevention measures

Error prevention can be mostly evaluated for features where operations can cause permanent effects which can be unwanted. For example microphone muting is not something that is error prevented, since usually doing the same action again can recover from that action. But features such as exiting the call are more difficult to recover from usually.

In Zoom it is clear that some thought for error prevention has been done, doing dangerous operations such as leaving the call is not possible to do without confirmation. Teams is more designed for professional use, so there is more lax error prevention, assuming users know what they are doing. Non-moderator users leaving a call have a rejoining option, which is a good measure to help recover from a problematic situation, but still overall there is not much more than that in error prevention. This is very similar to Teams in Meets, rejoining is possible, but the error prevention is lax.

Jitsi has the worst error prevention allowing users to do leaving and other actions instantly without easy recovery from the situation.

Vooler on the other hand tries to do things as safely as reasonably possible, leaving is protected and the user should know that they are indeed leaving the call and not clicking the button by accident. There are also features that allow an authorised person to do actions for the user, in order to simplify the experience further, which prevents end user based errors from happening altogether, but errors can still happen from moderation side.

FIGURE 4.23 Error prevention measures

Error prevention measures					
	Zoom	Teams	Google Meet	Jitsi	Vooler
End user error prevention	**Good error prevention**	**Decent error prevention**	**Decent error prevention**	**Weak error prevention**	**Good error prevention**
Moderator tool error prevention	**Good error prevention**	**Decent error prevention**	**Good error prevention**	**Decent error prevention**	**Buttons always shown**

UR6. Accessibility accommodations

The accessibility is considered as a general concept in UR6, but AR points will delve into individual issues in further detail.

Accessibility for the three big industry giants: Zoom, Meet and Teams, is easy to evaluate looking at their respective technical documents for accessibility [15][16][17]. Regarding Jitsi and Vooler, they also have accessibility accommodations, but are not as well documented so they have to be individually evaluated.
Keyboard and non-keyboard navigation usage is supported on all platforms. Using tabulator buttons to navigate is possible with all solutions. Keybindings are also supported, but Jitsi, Vooler and Meet have the easier to use ones due to being single key binds rather than the desktop apps using multi key binds.
Screen readers are also supported on all platforms, which supports users with low vision or blindness.
Zoom, Meet and Teams all have certain accessibility functionalities that can be used.
They also offer automatic live captioning, though Teams only for US english.
Additionally content scaling is supported in all platforms. Zoom and Teams uses specific settings for text size and content scaling, Meet uses in browser scaling and zooming, Jitsi has also web browser scaling and in Vooler uses the same as well.
Vooler though has especially designed for this use, and instead of relying only on users being self-reliant, it allows for other parties to help them achieve the goals of the video call, with for example remotely changing modes within the call.

FIGURE 4.24 Accessibility accommodations

Accessibility accommodations					
	Zoom	Teams	Google Meet	Jitsi	Vooler
Keybinds and keyboard navigation	**Full support**	**Full support**	**Full support**	**Full support**	**Full support**
Screen reader support	**Full support**	**Full support**	**Full support**	**Full support**	**Full support**
Live captioning	**Full support**	**US English only**	**Full support**	**Not supported**	**Not supported**
Content scaling	**Specific setting in app**	**Specific setting in app**	**Works with browser content scaling**	**Works with browser content scaling**	**Works with browser content scaling**
View mode support	**Handled by moderator**	**Handled by moderator**	**Handled by moderator**	**Handled by end user**	**Handled by moderator**
Designed for remote use	**No**	**No**	**No**	**No**	**Yes**

4.5 Accessibility requirements in the solutions

Accessibility requirements or AR's are next in evaluation. Note that the requirements are evaluated with the scope of well rounded usage for elderly people in video calls.

Zoom follows the WCAG 2.1 AA standard, which should mean all of the following issues should have some consideration behind them in the system. There is no documentation about other solutions following the same or anything comparable standard in their design. Vooler on the other hand has had thought put into these issues.

AR1 Vision based impairments

AR1.1 Major vision impairment or blindness mainly requires the system to support screen reader support. All of the systems employ support for this. Quality of this support is hard to compare well, but considering system simplicity and coverage of this feature is key. All in all, all of the systems have good coverage of screen reader support, both for end users and regular professional users.
AR1.2 Minor vision impairment or low vision can be mostly handled with content scaling features, which is also something all of the solutions offer. WCAG 2.1 AA requires 100% to 200% scaling capability, which all of the solutions do support. Notably application solutions also support this on desktop with keybinds similar to browsers.
AR1.3 Impaired contrast visibility is supported in Meet and Zoom with external contrast support, for example Meet uses an extension and Zoom can use system contrast settings. Teams is the only one with a built in high contrast mode. Jitsi has a clear lack of contrast support. Vooler on the other hand has contrast designed from the start in such a way that it is usable by most from the starting position without any defined extra mode to support people with difficulties with contrast, in other words it does not need a high contrast mode.

FIGURE 4.25 Accessibility accommodations for vision based impairments

Support for vision based impairments					
	Zoom	Teams	Google Meet	Jitsi	Vooler
Major vision impairment or blindness	**Well supported**	**Well supported**	**Well supported**	**Well supported**	**Well supported**
Minor vision impairment or low vision	**Well supported**	**Well supported**	**Well supported**	**Well supported**	**Well supported**
Impaired contrast visibility	**Supported with system a contrast setting**	**Well supported**	**Supported with a browser extension**	Not implemented	**Designed to work with contrast issues**

AR2 Cognition impairments

AR2.1 Intellectual impairment or a learning disability causes such issues that the system should favour the simplest solutions possible, with least amount of complexity. Tasks of the

video call are relatively simple, but confusion might come from multiple places at the same time and unfamiliarity with a system and new techniques for the user. Complexity levels are mostly the same, but only Vooler has a specific solution to create assisted and extra simplified solutions for joining a video call, using UI and possibly even a device, which is only designed for Vooler calls and joining a specific room. When the process is reduced to two to zero steps, usage becomes much more feasible.

AR2.2 Memory impairment or a memory affecting disease is a step further in the difficulty of the same problem as AR2.1, things which might be hard to learn, become essentially unlearnable, when concepts are not kept in permanent memory. This means the solution should support usage, which does not require the user to interact with the system at all. The solution should only focus on explaining what the device where the solution is used for. This is what Vooler tries to do with a few features. Automatic answering to calls enables remote usage with end users who are not capable of joining and answering calls on their own. Additionally there are explanations about the purpose of the device in neutral waiting states such as screen savers in order to make it easier to remind the end user what it is used for. Vooler has all the minimum requirements for usage for memory impaired users, but it can always be enhanced further, for example usage of pictures of other people rather than just names is preferred. Additionally all instances of showing your own face, essentially mirroring your face, should be disabled, since memory impaired users have a hard time adjusting to their ageing body.

FIGURE 4.26 Accessibility accommodations for cognitive based impairments

Support for cognitive based impairments					
	Zoom	Teams	Google Meet	Jitsi	Vooler
Complexity of use for self reliant end users with memory impairments	**High complexity**	**High complexity**	**High complexity**	**High complexity**	**High complexity**
Complexity of use for external supported end users with memory impairments	**Not supported**	**Not supported**	**Not supported**	**Not supported**	**Users do not have to interact with the system to use it**

AR3 Physical impairments

AR3.1 Minor motoric impairment, AR3.2 Moderate motoric impairment and AR3.3 Major motoric impairment can be considered as a group, since the difficulty level of the issue is hard to confine into one category, since usage of external tools and similar techniques is present in all solutions. Main difficulty for these kinds of users is the usage of tools. Some

people manage without extra tools, but have limited dexterity and might suffer from imprecise actions. Content scaling helps with imprecise actions, which is done in all systems. Users might also use tools which utilise the ability to switch between elements, similarly how tab selects between different elements. This is also present in all systems, there is no specific action that could not be done without a mouse. Only exception is the external management of all actions, which is only able to be done in Vooler with regards to remotely accessing video rooms and such. This is required only in the most severe cases with possibly some other issues also making the system so difficult that the user must be externally managed by the end user's nurse or someone with a similar role.

FIGURE 4.27 Accessibility accommodations for physical impairments

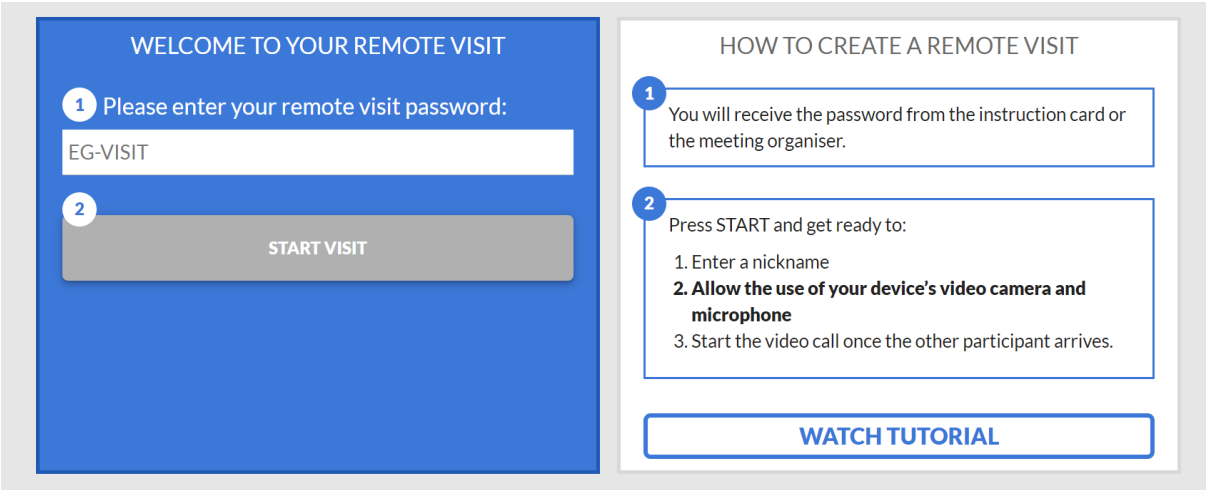
Support for physical impairments					
	Zoom	Teams	Google Meet	Jitsi	Vooler
Support for external tools	**Well supported**	**Well supported**	**Well supported**	**Well supported**	**Well supported**
Content scaling	**Well supported**	**Well supported**	**Well supported**	**Well supported**	**Well supported**
Designed for remote use (allows user to not interact with the device itself	Not designed for remote use	Not designed for remote use	Not designed for remote use	Not designed for remote use	**Designed for remote use**

AR4 Weak technological literacy skills

AR4.1 System literacy and AR4.2 system abstraction are similar issues to AR2.1 and AR2.2, the simpler the solution the better it is. While end users can be accommodated with remote management tools, the professional user needs to learn the system too. In order to manage these issues with the system, the tasks within the system should be easy to process. It also helps that tasks are made into palatable sized smaller tasks. For the industry tools Zoom, Teams and Meets the tasks are quite simple, meetings can be scheduled or instantly started. They are quite self explanatory, but can be made easier to handle with tutorials in format of video or text. Jitsi as a standalone video meeting software does not have any other built-in functionality except instant meetings, which is understandable for its purpose, but it poses the hardest to understand as a system.
Vooler on the other hand tackles the system issue with trying to implement guided process way of thinking. As shown in Figure 4.21, tasks are done on the left side and explanations of the task are on the right. Tasks are split into smaller objectives, which after doing all of them should yield the wanted result. This is expanded to most things users can do in their user

interfaces to simplify the system for the users. This should result in tasks that are done in hand holding fashion that have very easy work flows, which should offer the easiest to understand experience outside of a video tutorial or someone else showing how the task should be done.

FIGURE 4.28 Vierailu.fi, Vooler's joining platform for visiting relatives showing guided process solution



AR4.3 Inexperience of modern practices provides a unique challenge, features or functionalities can be missed if the user does not know or understand some conventions that are widely used in many systems. For example zooming content can be very difficult without external help or knowledge about such features in a web client or app. This means features designed for users with very limited technical knowledge should be done in such a way that they can be accessed without them being "hidden features". For example keybinds should be visible for all users in some screen explaining how they work and what they do. Features that users might need and are not clear without explanation should have some explanation close to the said feature. Guided processes might help to do this, but this can not be utilised well for example in a video call, where screen space is limited and attention is already divided in the current task at hand.

Vooler has most purposefully made support for this kind of accessibility, it has guided processes and has multiple conventions to support end users and the users who are managing them. Zoom, Teams and Google Meet are more traditional in their design. Users need to learn the system, which requires possibly some external documentation or tutorials to use effectively. Jitsi has the poorest support for this, but it also has least depth as a tool, since it offers only video calls.

FIGURE 4.29 Accessibility accommodations for weak technological literacy skills

Support for weak technological literacy skills					
	Zoom	Teams	Google Meet	Jitsi	Vooler
Simplicity of the system	**Decently simple to use**	**Decently simple to use**	**Decently simple to use**	**Somewhat simple but poorly guided**	**System designed for multiple ways of use with extra features for simplicity**
Features which support technologically inexperienced users	**Good UI design with help texts on buttons**	**Good design with no extra accommodations**	**Good design with no extra accommodations**	**Good design with no extra accommodations**	**Users do not have to interact with the system to use it**
Features to support end user managing use (Eg. Nurse)	**Assumes some technical literacy skills**	**Assumes some technical literacy skills**	**Assumes some technical literacy skills**	**Assumes high technical literacy skills**	**Specific process design for inexperienced users**

5. Case study

This chapter discusses the question RQ1, how do current video call solutions compare when considering social interaction for elderly people. This chapter uses the combination of given characteristics of chapter 4 and combines that with usage of advanced users of this field as interview subjects.

5.1. Motivation of the study

Chapter 5 case study was conducted using the frame given from the sections 4.4 and 4.5, which then were extended to more comprehensive interviews. The reason to gather more information was to back up the claims made in sections 4.4 and 4.5, which had evaluations of each system's implementations of solutions. This means, the case study should reflect the quality of solutions in chapter 4 and the other way around.

5.2 Methodology of the study

Case study was done by interviewing 5 technology students, which gave results based on their views of the systems and the questions asked. The questions were based on the criteria laid out on chapter 3 and 4, where the interview questions were made as neutral as possible, but with the focus being consideration of the focus group, which is elderly people and special needs groups.
Interviews were made using screen sharing and a voice call. Additionally the content being shared was a mix of systems themselves and supporting content such as pictures to quicker show all necessary information before interviewees graded the solution.
Grading in the interview was done by grading solutions from 1 to 5, with 0.5 intervals, where higher is better. Interviewed people were allowed to grade the same score for all solutions if they felt equal.

Interview questions were the following,
How would you rate the systems:

- The routes to a call and accommodations of limitations of users (UR1)
- Visibility of statuses in the call UI for call interaction elements (UR2)
- Constant visibility and understandability of all basic call functions (UR3)
- Simplicity of the system, ease of use for non-technical people (UR4)
- Error prevention measures (UR5)
- Accessibility accommodations (UR6)

How would you rate the systems support for given accessibility requirements:

- Vision based impairments (AR1)
- Cognitive based impairments (AR2)
- Physical impairments (AR3)
- Weak technological literacy skills (AR4)

5.3 Results of the study

Results of the interview based case study have a clear trend. Since the focus of the interview and study overall is the usage of video calls for elderly people and special needs groups, the focus becomes simplification and taking responsibility away from the end user.

FIGURE 5.1 Interview results of the systems with five technology students visualised as scores

	Zoom	Teams	Meet	Jitsi	Vooler
UR1	3,5	2,8	2,6	2,8	4,7
UR2	2,9	2,4	2,4	4,2	4,2
UR3	2,9	3,9	3,9	2,7	4,9
UR4	2,8	2,5	2,4	2,4	4,7
UR5	4,1	4	3,5	2,2	4,5
UR6	3,9	3,6	3,6	2,5	5
AR1	3,9	3,5	3,3	2,1	4,3
AR2	2,3	2,2	2,3	2	4,9
AR3	3,5	3,5	3,5	3,3	5
AR4	3,7	3,4	3,4	3,4	4,6
Total	3,35	3,18	3,09	2,76	4,68

This explains, as shown in Figure 5.1, the scores heavily favour Vooler. This can be attributed to the fact that Vooler is the only solution, which specifically tries to solve issues where the end user has the least amount of ability, least amount of experience and highest amount of factors which makes general use difficult.
Zoom, Teams, Meet and Jitsi overall have good quality solutions for their given target audience, which is usually somewhat capable users who do not need any special help. On the other hand Vooler does its best to give a minimum amount of work for the end user, so as little as possible can go wrong. This means when comparing the solutions, Vooler has a clear advantage when for example joining, microphone and camera management are discussed, since the complication of doing each said task can be transferred to a more capable and potentially more experienced user of the system. The best example of this is call joining, which can be done with auto accepting the call in Vooler. No other system can do this due to the solutions not being considered for mainly nursing use, but Vooler which can be used for remote care has to have this feature in order to not be reliant on the answering side. When the technology performs as it is designed to do, it is hard to beat it with the other solutions.
This means accessibility, usability and all other features which the end user does not directly need to touch can be remotely handled for them, which allows almost all users in any state to use the said video calls. Though it is good to note that accessibility is quite well done for all solutions for individual users, when regarding basic accessibility.

5.4 Discussion and critique of the study

The results given in Figure 5.1 are promising. Though there has to be made a point that Vooler was only known by name before for the interviewed users and the target audience was clear even before the interviews. The conductor of the interview is also working for the said company which is developing Vooler, which might create bias. Additionally just hearing and seeing how the software should work can cause positive bias for Vooler due to the fact that real use issues cannot happen in controlled environments.
Additionally software such as Zoom and Teams had the most amount of pre-interview experience, which had caused some prenotions of the softwares being bad in some manner, which additionally could cause negative bias towards said solutions.
The core usage of the system is also a key factor to understanding why Vooler is outperforming other solutions. It is intended to solely fix current issues faced by elderly

people and nursing people with technology which could in ideal scenarios be helpful for them, which the other solutions clearly are not taking into consideration.

6. Conclusions

As a conclusion for the work, research questions stated in the introduction were answered in the following way. RQ1 Current mainstream video call solutions are not good enough when considering social interaction for elderly people. The current solutions are targeted for people who can use computers comfortably and can manage a video call session which is more in line with a business meeting. Social interaction assumes that users have a certain level of ability to use systems, which is not very suitable for eldery or technologically less adept users. Furthermore Chapter 5 includes an interview, which is conducted from already established important usability and accessibility requirements found in chapter 3, which RQ2 and RQ3 cover. Results indicate that Vooler, a video meeting platform built for eldery people and nursing staff, has a clear usability and user experience advantage when compared to more generic solutions. This is due to the fact that Vooler uses solutions which are tailor made to fix issues which the users might generally face in their technology use, which include processes which are mostly automatised or handled by another person for them. RQ2 asks the question, what the user needs to have a successful video call, especially for special need groups and nursing staff, which often work in collaboration. It boils down to a few points. The users need multiple ways to enter a call, the user interface should prefer elements which are always visible, elements used for the call's basic functionalities should be the only ones always visible, others should be hidden. Solution should favour simplicity over complexity, error prevention should be a priority and accessibility should be made a priority. RQ3 asks what UX elements have to be accounted for social interaction, if the target audience consists of special needs groups and nursing staff. These include vision based impairments, cognitive based impairments, physical impairments and weak technological literacy skills. These need to be considered in order to attain true accessibility in a system, which "normal" design does not necessarily cover.

Main takeaway from this work is that it is important to identify key features which are needed for successful implementation having a target audience in mind. For the Vooler project, remote usage and automation from end user perspective are key features which create the most value for the target users. To identify these key features, innovation and user feedback is needed. This can be achieved quite well using collaborative workshopping methods. Once some number of solutions have been created they can be then evaluated against each other

using experts of the field of any kind, which can range from end users to nursing staff, which can even include other people who understand end user needs.

Discussion and limitations of the work

This work was done as a side project while working on Vooler as a QA tester. This means closeness to the software itself can cause bias, but it also ensures complete understanding in the system itself. Additionally improvement for this work could be that someone would compare actual remote care solutions with each other and not just video call solutions, which have different target audiences.

References

[1] Nielsen, J. (2005). Ten usability heuristics, last accessed March 2022 https://www.nngroup.com/articles/ten-usability-heuristics/

[2] Quiñones, D., Rusu, C., & Rusu, V. (2018). A methodology to develop usability/user experience heuristics. *Computer standards & interfaces*, *59*, 109-129.

[3] Hassenzahl, M. (2008, September). User experience (UX) towards an experiential perspective on product quality. In *Proceedings of the 20th Conference on l'Interaction Homme-Machine* (pp. 11-15).

[4] Roupa, Z., Nikas, M., Gerasimou, E., Zafeiri, V., Giasyrani, L., Kazitori, E., & Sotiropoulou, P. (2010). The use of technology by the elderly. *Health Science Journal*, *4*(2), 118.

[5] Newell, A. F., Gregor, P., Morgan, M., Pullin, G., & Macaulay, C. (2011). User-sensitive inclusive design. *Universal Access in the Information Society*, *10*(3), 235-243.

[6] Gregor, P., Newell, A. F., & Zajicek, M. (2002, July). Designing for dynamic diversity: interfaces for older people. In *Proceedings of the fifth international ACM conference on Assistive technologies* (pp. 151-156).

[7] Bouakaz, S., Vacher, M., Chaumon, M. E. B., Aman, F., Bekkadja, S., Portet, F., ... & Chevalier, T. (2014). CIRDO: Smart companion for helping elderly to live at home for longer. *Irbm*, *35*(2), 100-108.

[8] Charness, N., & Boot, W. R. (2009). Aging and information technology use: Potential and barriers. *Current Directions in Psychological Science*, *18*(5), 253-258.

[9] Das, M., Gergle, D., & Piper, A. M. (2019). " It doesn't win you friends" Understanding Accessibility in Collaborative Writing for People with Vision Impairments. *Proceedings of the ACM on Human-Computer Interaction*, *3*(CSCW), 1-26.

[10] Caldwell, B., Cooper, M., Reid, L. G., Vanderheiden, G., Chisholm, W., Slatin, J., & White, J. (2008). Web content accessibility guidelines (WCAG) 2.0. *WWW Consortium (W3C)*, *290*, 1-34.

[11] Elliott, D. B., & Situ, P. (1998). Visual acuity versus letter contrast sensitivity in early cataract. *Vision Research*, *38*(13), 2047-2052.

[12] Power, C., Freire, A., Petrie, H., & Swallow, D. (2012, May). Guidelines are only half of the story: accessibility problems encountered by blind users on the web. In *Proceedings of the SIGCHI conference on human factors in computing systems* (pp. 433-442).

[13] Shadi Abou Zahra (15 May 2017). Diverse abilities and barriers, last accessed March 2022 https://www.w3.org/WAI/people-use-web/abilities-barriers/

[14] Kumar, S., Ureel, L. C., King, H., & Wallace, C. (2013, May). Lessons from our elders: identifying obstacles to digital literacy through direct engagement. In *Proceedings of the 6th international conference on pervasive technologies related to assistive environments* (pp. 1-8).

[15] Google Meet help, Google Meet accessibility, last accessed March 2022 https://support.google.com/meet/answer/7313544?hl=en

[16] Zoom, Zoom is for Everyone “Accessibility is at the heart of what we do.”, last accessed March 2022 https://explore.zoom.us/en/accessibility/

[17] Microsort Support, Accessibility overview of Microsoft Teams, last accessed March 2022 https://support.microsoft.com/en-us/office/accessibility-overview-of-microsoft-teams-2d4009e7-1300-4766-87e8-7a217496c3d5

[18] Nielsen, J. (1994, April). Enhancing the explanatory power of usability heuristics. In Proceedings of the SIGCHI conference on Human Factors in Computing Systems (pp. 152-158).

[19] Abras, C., Maloney-Krichmar, D., & Preece, J. (2004). User-centered design. Bainbridge, W. Encyclopedia of Human-Computer Interaction. Thousand Oaks: Sage Publications, 37(4), 445-456.

[20] Quiñones, D., & Rusu, C. (2017). How to develop usability heuristics: A systematic literature review. Computer standards & interfaces, 53, 89-122.
[21] Alonso-Ríos, D., Mosqueira-Rey, E., & Moret-Bonillo, V. (2018). A systematic and generalizable approach to the heuristic evaluation of user interfaces. International Journal of Human–Computer Interaction, 34(12), 1169-1182.
[22] Vermeeren, A. P., Law, E. L. C., Roto, V., Obrist, M., Hoonhout, J., & Väänänen-Vainio-Mattila, K. (2010, October). User experience evaluation methods: current state and development needs. In Proceedings of the 6th Nordic conference on human-computer interaction: Extending boundaries (pp. 521-530).
[23] Frank Lopresti, E., Mihailidis, A., & Kirsch, N. (2004). Assistive technology for cognitive rehabilitation: State of the art. Neuropsychological rehabilitation, 14(1-2), 5-39.
[24] Gregor, P., Newell, A. F., & Zajicek, M. (2002, July). Designing for dynamic diversity: interfaces for older people. In Proceedings of the fifth international ACM conference on Assistive technologies (pp. 151-156).
[25] Wobbrock, J. O., Kane, S. K., Gajos, K. Z., Harada, S., & Froehlich, J. (2011). Ability-based design: Concept, principles and examples. ACM Transactions on Accessible Computing (TACCESS), 3(3), 1-27.
[26] Newell, A. F., & Gregor, P. (2002). Design for older and disabled people–where do we go from here?. Universal Access in the Information Society, 2(1), 3-7.
[27] Lee, C., & Coughlin, J. F. (2015). PERSPECTIVE: Older adults' adoption of technology: an integrated approach to identifying determinants and barriers. Journal of Product Innovation Management, 32(5), 747-759.
[28] Eisma, R., Dickinson, A., Goodman, J., Syme, A., Tiwari, L., & Newell, A. F. (2004). Early user involvement in the development of information technology-related products for older people. Universal Access in the Information Society, 3(2), 131-140.
[29] Murphy, E., Kuber, R., McAllister, G., Strain, P., & Yu, W. (2008). An empirical investigation into the difficulties experienced by visually impaired Internet users. Universal Access in the Information Society, 7(1), 79-91.
[30] Martins Van Jaarsveld, G. (2020). The effects of COVID-19 among the elderly population: a case for closing the digital divide. Frontiers in psychiatry, 1211.
[31] Van De Watering, M. (2005). The impact of computer technology on the elderly. Last accessed April 2022. https://www.marekvandewatering.com/texts/HCI_Essay_Marek_van_de_Watering.pdf
[32] Wirtz, S., Jakobs, E. M., & Ziefle, M. (2009, August). Age-specific usability issues of software interfaces. In Proceedings of the IEA (Vol. 17).
[33] Darejeh, A., & Singh, D. (2013). A review on user interface design principles to increase software usability for users with less computer literacy. Journal of computer science, 9(11), 1443.
[34] Zajicek, M. (2004). Successful and available: interface design exemplars for older users. Interacting with computers, 16(3), 411-430.