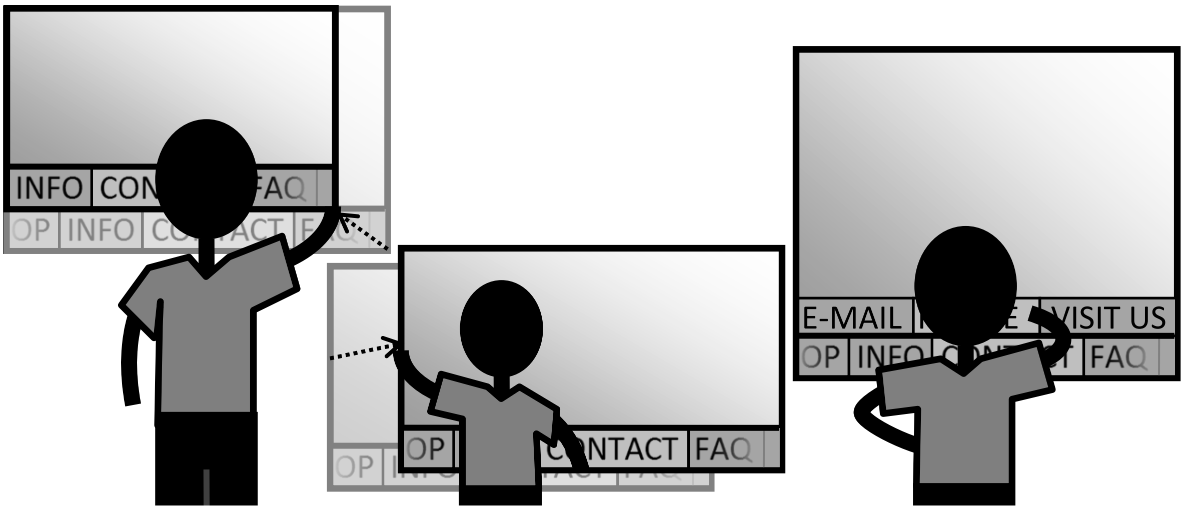
CHALMERS
UNIVERSITY OF TECHNOLOGY



Development of a Multiuser GUI for Large Touch Screens

A Design for a Varying Number of Simultaneous Users Handling Separate Applications

HELENA AHLSTRAND

Department of Product and Production Development
Division Design & Human Factors
CHALMERS UNIVERSITY OF TECHNOLOGY
Göteborg, Sweden, 2014

THESIS FOR THE DEGREE OF MASTER OF SCIENCE IN PRODUCT DEVELOPMENT

Development of a Multiuser GUI for Large Touch Screens

HELENA AHLSTRAND

Division Design & Human Factors
Chalmers University of Technology
Göteborg, Sweden, 2014

Development of a Multiuser GUI for Large Touch Screens

HELENA AHLSTRAND

Chalmers University of Technology

Division Design & Human Factors

SE-412 96 Gothenburg, Sweden

+ 46 (0)31-772 1000

Printed by Chalmers Reproservice

Gothenburg, Sweden 2014

ABSTRACT

Applications running on large touchscreens situated in public areas are often designed for either one user or a predefined number of simultaneous users. In cases of non-collaborative applications, i.e. when the designs are intended for one user only, people may have to wait for a current user to finish before they can start to interact with the screen. The initiator of this thesis - i4sense - requested a intuitive solution which allowed for a varying number of users to work simultaneously on the screen with separate applications. In this report, a design concept which solves said problem is being presented.

In order to facilitate the development work, the GUI was broken down into three cornerstones; the layout, the viewport and the menu. After a thorough literature review and a pre-study in which the most critical product characteristics were defined, a number of concepts for each cornerstone was developed. These were methodically evaluated using a number of decision tools where characteristics such as ease of use and adaptability were weighed according to their importance as defined in the pre-study. The basis for the evaluation of the viewport included the results from a user trial carried out at Lindholmen Science Park using mock-up designs of the top concepts.

The end result is a solution where users can open their own viewport by a single touch on an unoccupied area on the main screen. The viewports can be resized by touching and dragging the corners of the frame and moved around by the same gesture on the frame edges. In order to avoid interference between simultaneous users the viewports are not allowed to overlap. To ensure that the full area of the screen is readily available for oncoming users, viewports which are left idle for a given amount of time are automatically removed. Along the bottom of each viewport a toolbar displays the options of applications to the user.

The methods used to obtain the above results were found to be sufficient. It was however discovered that the methodical approach yielded different results than did the user trial concerning the viewport. This was not unanticipated as the methods had radically different focus. While the decision matrix mainly evaluated the suitability of the concepts in some specific cases rather than their combined impact on usability, the focus of the user trial was the participants’ experience of the interaction. Future work might therefore involve additional user trials covering the usability of the two remaining cornerstones.

ACKNOWLEDGEMENTS

I would like to express my very great appreciation to Fredrik Stål and all the colleagues at i4sense whom I had the privilege to work alongside during the writing of this thesis. Not only did they provide valuable knowledge and insight during the thesis work, they also made me feel very welcome and part of the team from the first day at the office.

The support provided by my mentor at Chalmers University of Technology, Lars-Ola Bligård, has been much appreciated and will not be forgotten. Many thanks also to my examiner Anna-Lisa Osvalder.

Furthermore, I like to thank everyone who participated in the user trial at Lindholmen Science Park, both the people whom I invited to the site and especially the people passing by who sacrificed their precious lunch breaks in order to help me out.

Finally I wish to acknowledge the help provided by my friends Anna Johansson and Andreas Robertsson, as well as my family, whose encouragements and endless proofreading kept me motivated even during the latest hours of writing this report.

TERMINOLOGY

This list consists of a number of abbreviations as well as definitions of words that will be used throughout this report.

Application / Main screen application	The *Main screen application* refers to the application that runs on the large touch screen itself, i.e the final concept solution that will be presented at the end of this thesis. This concept does however incorporate a solution for users to manage various applications inside their personal viewport. The latter are simply referred to as *applications*.
GUI	Graphical User Interface
Kiosk	A kiosk in this context refers to an interactive computer terminal of the kind that can be used to for example buy tickets. The size of the display is often about the same size as a computer screen and situated a few decimetres above waist level. Some kiosks use an attached keyboard and mouse, whereas some of them use touch screens.
Mock-up	In software development mock-ups can be used to illustrate how the final GUI will look, without having to develop all the functionality. The mock-ups can range from simple sketches to semi functional interfaces where single functions can be tested.
Multi touch (MT)	The term multi touch is used to describe a manipulation with both hands or with multiple fingers of one hand.
Single touch (ST)	The term single touch describes a manipulation with one finger.
Unity3D	The game engine used to create the applications for the touch screens applications is called Unity3D. It has an integrated development environment which supports code written in C# or Javascript, the former of which have been used during this thesis work. Unity3D and C# was used for compatibility with other software developed by i4sense.
Viewport	In computer graphics, the concept viewport has many definitions. In this report, a viewport will refer to the area related to one single user. This area contains everything that is meant specifically for that single user to see or to use, for example menus, pictures, movies, icons or background.

TABLE OF CONTENTS

1 INTRODUCTION

In an intense and competitive environment, new means of communication and innovative ways of conveying a message may be of great importance for a modern day company. The company i4sense focuses on creating large-scale interactive surfaces intended to achieve just that. Whether these interactive surfaces, in everyday language - touch screens, are intended for an exhibition or an information board, there are a number of basic requirements which applies to all of them. In the technology oriented reality of today, customers or users who come across new technology more or less expect to immediately understand how it works, and if that does not happen, interest is lost rather quickly. This means that these interactive surfaces, just as any other product, need to be both intuitive and intriguing for the user. Depending on where the interactive surface is situated and what it is intended for, users with varying experience will be using it. Even a first time user needs to be able to figure out how to use it just by trying it out and exploring its functions. One major issue regarding large touch screens is the amount of space that may be left unused when only one user is operating it.

At the moment, an easy way to allow for several users to use the same application is to divide the screen into a predefined number of work areas. A screen could for example hold two same size square areas with attached menus (Figure 4). This solution does however limit the use of space, since an expected number of simultaneous users needs to be predefined and parts of the screen may remain unused if this number is not reached. This space could instead have been used to enlarge someone else's workspace.

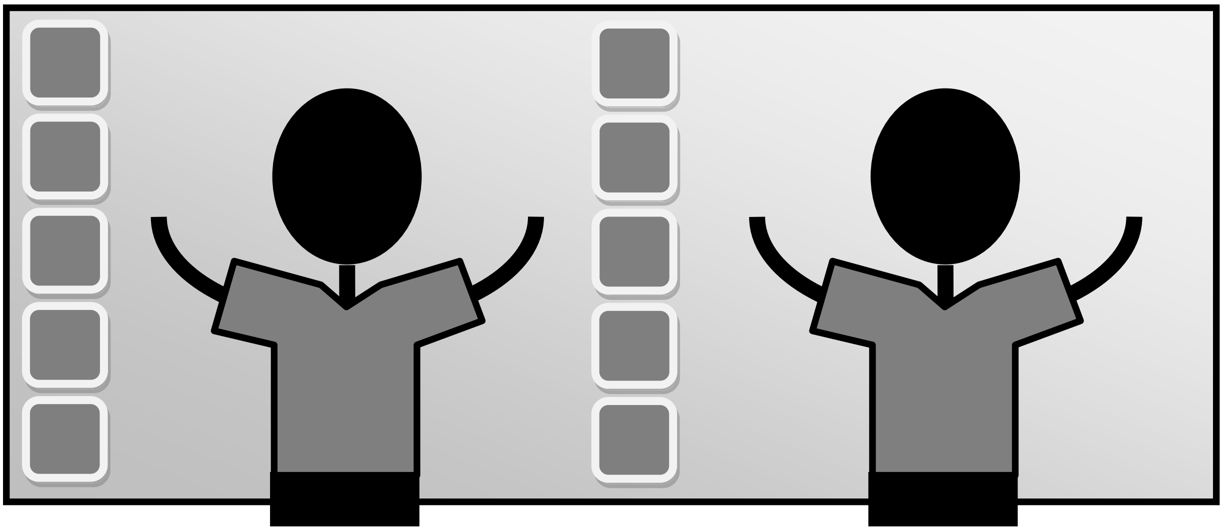


Figure 1 - A simple solution when sharing the screen between a predefined number of users, in this case two, is to divide the screen into two work areas.

1.1 PURPOSE

The purpose of this thesis is to present a design solution for publically situated large touch screens which will enable a varying number of users with no previous experience to work simultaneously on the screen with separate applications. The thesis is constrained to a number of demands set from the initiator of this thesis, i4sense, which are listed below.

- The main screen application must be able to handle a varying number of workspaces.
- The workspace shall be able to shift position as well as size.
- The solution shall provide means for switching between applications within ones workspace.

1.2 PROBLEM DEFINITION

The large touch screens produced by i4sense are often used by other companies and in turn their customers. The main use for the companies is to share information with their customers in a modern and interesting way and, needless to say, to promote the company. Since the screens are often placed at exhibitions, lobbies, information centres or public areas, very few of the users have interacted with the same specific application before. Both the interface and the attached media or information therefore needs to be suitable for first time users.

In most cases, the number of users that will be using the screen is not known in advance. When an undefined number of users want to use the screen simultaneously and this number changes continuously, the software needs to provide a possibility to adapt to these changes in an appropriate way. When deciding how this should be done, all users need to be taken into account. A solution that works well for the newcomer, i.e. a user who is about to start using the screen, may not be ultimate for his or her fellow users who are already using the screen.

Every workspace needs to have an attached menu in order to allow users to work independently at their own workspace rather than having to physically move to another part of the screen in order to access menu controls. A menu in this sense does not necessarily mean a list of options, but some means to facilitate for example the choice of media or browsing between different applications. It should also be easily accessible yet not be in the way. More specifically, the menu should not cover anything the user want to access, not obstruct the orientation in the media and not cause the workspace to be perceived as small or cluttered.

To prevent the program developers at i4sense from having to rewrite the program for different customers in order to meet their graphic profile, the design solution should give room for some changes in settings. These adjustments should be easily operated to give each main screen application a unique look, without compromising its function and intuitive features.

1.3 GOAL & LIMITATIONS

This thesis will result in a GUI concept including functional descriptions of the elements of the main screen application. The functional descriptions will include critical issues for operating the screen such as how to open, move, scale, stretch and close the workspace. Secondary issues such as thickness of borders etc., are up to the final stages of programming where specific needs of the customers are taken into account.

1.4 REPORT OUTLINE

An outline of the report structure is presented at the start of the following chapter.

2 PROCEDURE

The GUI can be broken down into three main development areas; the layout, the viewport and the menu. The layout refers to how the full screen should look when the application is running, how the viewports should be arranged on the screen and how they should work together. The viewport is the workspace that belongs to one user and can be manipulated by that user only. Each viewport will have an associated menu that can be used to switch between applications etc. In this thesis, these main areas are further on referred to as corner stones.

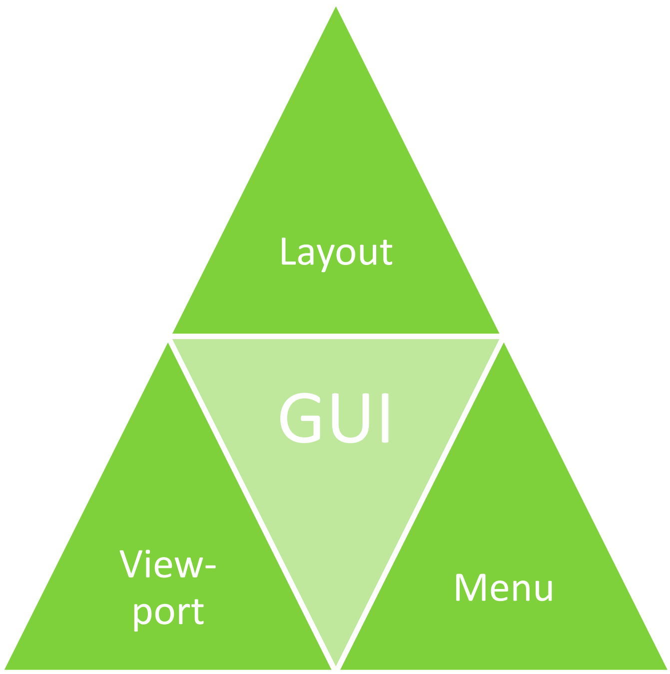


Figure 2 - The GUI consists of three corner stones; The layout, the viewport and the menu.

During software development, iterative processes are often used, e.g. the Spiral model (SDLC, 2011) and agile development (Drugovic, 2012), where feedback rather than planning, is used to control the development. However, due to limited resources, this master thesis did not allow for the development and testing of numerous functional prototypes. Instead, the development work was carried out in a comprehensive and sequential way which reduced the number of iterations. Mock-ups created in the same environment as the final solution, namely Unity3D, was used for concept testing and evaluation. This approach is more similar to new product development (NPD) rather than the above mentioned software development methods, and therefore methods recommended when working with NPD were used. The NPD process consists of four main phases; Idea generation, Concept Development, Product Development and Product testing. This thesis included work with the first two phases, and in order to structure the workflow, these two phases were divided into five steps suited for this project. Each corner stone went through these steps separately from each other. Prior to this, a literature review and a pre-study covering the full GUI was carried out. The concept refinement work was finalized by the integration of the corner stones into the final GUI.

Figure 3 below illustrates the general phases of the NPD process as well as the variant of the process, which was used in this thesis. The italic font marks the chapters following in this report.

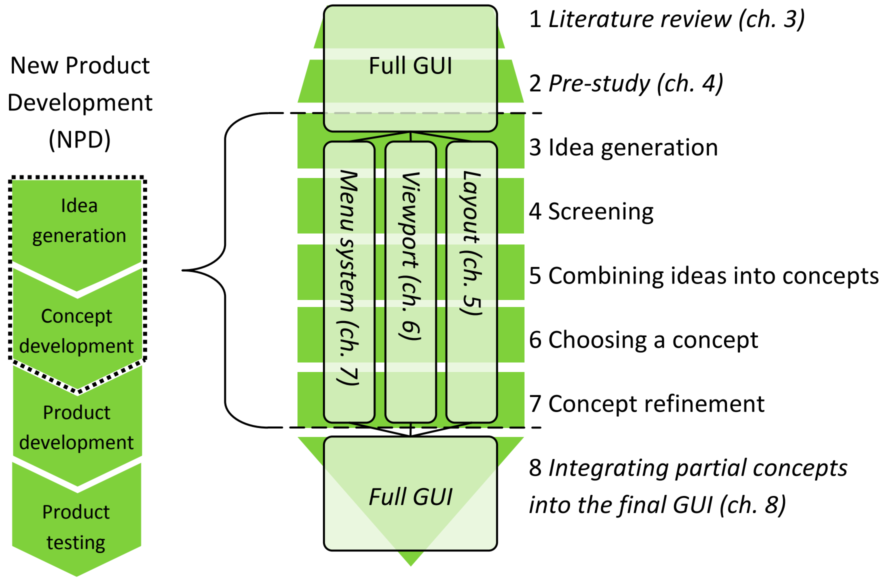


Figure 3 - The general phases of the New Product Development Process (left) and the eight step process used in this thesis (right). In the beginning of the process the GUI is considered as one element, in steps 3 to 7 the work is carried out in separate for the three corner stones, which are integrated again in the 8th and final step.

2.1 LITERATURE REVIEW

Prior to the development work an extensive literature review was carried out. Since the amount of published work on large-scale touch screens was found to be limited, the review included existing products, such as touch screens for single users, e.g. smart phones, kiosks, and interactive table tops. However, inspiration and knowledge about previous work with larger screens was achieved via a market research studying competitive products. Even though the focus of the thesis was on GUI design and cognitive ergonomics, physical ergonomics was briefly studied, as the physical load on the user shall not exceed existing recommendations.

2.2 PRE-STUDY

The pre-study aimed to serve as a basis during the rest of the thesis work. With the findings from the literature review in mind, a user profile, a Use Profile Table as well as a number of use cases were created.

When using a product, the users' mind-set and the way they approach problems are significantly influenced by the environment they are currently in (3.5.1). Since the touch screens can be used virtually anywhere, three different use case scenarios were created in order to gain a good insight regarding in which situations the final solution may be used. The use cases also take into account, which difficulties or distractions the users may encounter from the surroundings.

When creating a solution that is tailored for a specific user, a detailed user profile is of great importance. Even though this project aimed at a wide range of users, it was still necessary to specify the types of users which would actually be in contact with the product.

To decide which areas that should be in focus throughout the development work, a Use Profile Table was set up, and this in turn resulted in a list containing the focus areas ranked in order of importance. This ranking was used as a validation tool when assigning weights to the requirements used in the decision matrices (2.6).

Use Profile Table

One way to assess which areas in a HMI system to focus on during product development is to conduct a Use Profile Table. It takes into account how and where the product is used, by whom and how often.

The UPT contains a column with statements concerning the users experiences and relation to the product at hand. By selecting the statements that best correspond with the users profile, one can determine the importance of various product characteristics.

2.3 IDEA GENERATION

The main method for generating ideas was brainstorming. This process was carried out in turns, each with a different focus, the first being the GUI as a whole. The following set of brainstorming sessions focused on each of the three corner stones which ultimately were broken down even further, resulting in 12 sessions altogether (see Figure 4). The ideas generated were in different form, ranging from simple sentences or bullet lists to sketches with varying level of detail (see some examples in Appendix A).

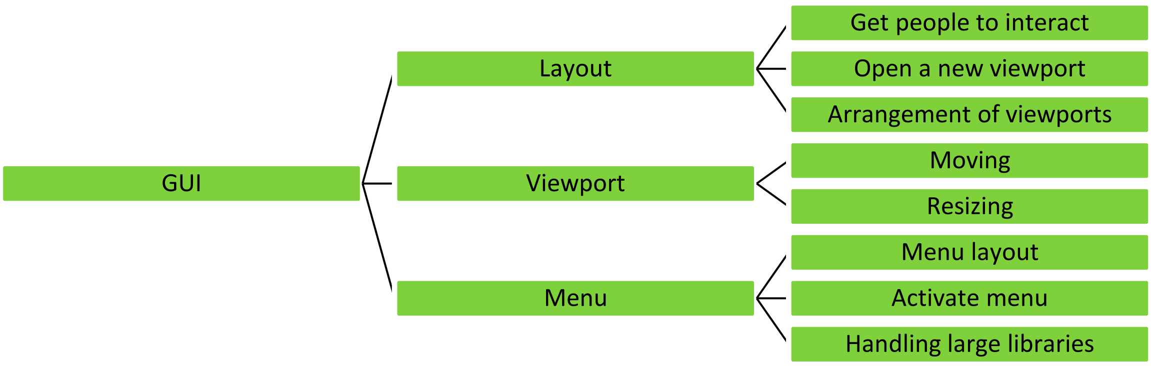


Figure 4 – The 12 brainstorm sessions (divided into three levels) illustrated in a tree diagram.

It was discovered that a majority of the ideas generated from the first general brainstorming session were of such character that they could easily be filed under one of the categories in the third set of sessions (rightmost column in Figure 4). All but a few ideas could be said to belong to one of the three corner stones. In other words, the solutions for each of the corner stones appeared to be rather independent of each other. For this reason, it was decided that the work could be divided between the corner stones and carried out separately without the risk of losing dependent ideas or concepts in the process.

2.4 SCREENING

The initial screening following the idea generation was a result of discussions with the programmers at i4sense. Their knowledge and previous experience aided the process of choosing which ideas to continue working with. All ideas generated during the brainstorming sessions were analysed and those that either seemed unrealistic, were perceived as too low-tech or unnecessarily advanced were discarded.

Combining the large amount of material and the fact that this was a fully verbal activity, documenting every argument and opinion that led to whether an idea was discarded or not would have been very time-consuming. However, ideas which initially seemed especially promising, but for some reason still did not make it through to the next phase, are accompanied by an explanation to the decision to discard them.

2.5 COMBINING IDEAS INTO CONCEPTS

As previously explained, from this phase on the work and development regarding the layout, the viewport and the menu was divided and carried out separately.

The ideas remaining from the idea generation and screening were combined into a number of concepts for each of these three corner stones using Morphological Matrices. These matrices have the ability to generate a large amount of concepts, though only a fraction of the possible outcomes made it through to the concept evaluation. The ones that did not were either too similar to the concepts that did, or they were immediately judged to be too impractical.

Morphological Matrix
A Morphological Matrix is a product development tool used to cross breed partial solutions to create a variation of complete solutions. The partial solutions are arranged in a table, where each row represents a different function of the product which said solutions aim to solve. By selecting one partial solution from each row and combining them into one complete solution many concept variants can be created.

2.6 CHOOSING A CONCEPT

To be able to evaluate the concepts generated by the Morphological Matrices, a list of criteria that the concepts should fulfill to the highest extent possible was set up. The criteria were assigned weight factors from 1 to 3 that related to how important they were for the success of the solution, 1 being not so important and 3 being very important. When creating the list of criteria and assigning the weights, findings from the literature review and the pre-study were used as a basis.

Decision Matrix
A Decision Matrix is a tool for evaluating solutions or concepts against a number of criteria in order to compare them to each other. This can be done in different ways. Either a grading system with numbers can be used, or the solution can simply receive a Yes or No depending on whether it fulfills the criteria or not. In more advanced matrices the criteria can be weighted depending on importance or complexity.

Both the criteria and their associated weight factors were inserted into a Decision Matrix where the concepts were graded on how well on a scale from 1 to 3 they fulfilled each criterion. A total score was calculated for each concept by multiplying the weight and the grade for each criterion and adding them together. Additionally, the percentage of the maximum score possible was calculated for each concept in order to get a better understanding of the results.

The concept with the best result went on to the next phase, which for the layout and menu meant concept development. The viewport however was found to require a more in-depth evaluation, which was why a user trial was carried out.

2.6.1 CONCEPT TEST – USER TRIAL

Studying how the product or system is used is the prime source of information regarding human-machine activities (Bligård, 2011). Unfortunately, the extent of this thesis did not allow for all generated concepts to be studied this way. Nevertheless, since the viewport was considered to be the most critical part of the design, a concept test was carried out to complement the decision matrix in the process of deciding upon one final concept solution. The concepts were tested, evaluated and rated in a user trial.

Fully functioning virtual prototypes of the concepts were created in Unity3D. The prototypes were made to look and behave as similar as possible to the final product since that would make the conclusions drawn from the users' opinions more reliable. Ideally the concepts should be tested without the trial subjects knowing what they are evaluating, but since only the frame was possible to manipulate, it was obvious to the trial subjects that this was the focus.

User trial layout

The user trial took place at Lindholmen Science Park using a permanently installed touchscreen located in the ground floor lobby. The touchscreen measured 198 x 108 cm and was positioned 97 cm from the floor. It used an infrared grid in order to recognize and translate gestures on the screen. Although there were touchscreens accessible in more closed off areas where the trial could have been carried out in private with fewer distractions, the lobby was chosen to mimic the environment where the final application is most likely to run.

Since the trial subjects might be influenced by the previously tried concept(s) when trying the next one(s), the order in which they were tested was varied. Shifting the order intended to decrease learning effects.

Information given to the trial subjects

Before the trial subjects actually tried the concepts they were informed about the thesis work, the purpose of the trial and the limits of the prototypes. Thereafter they were faced with one of the concepts and asked to experiment with the functionality until they felt comfortable enough in how it functioned to verbally explain which gestures did what. Finally they had to place the concept window inside a rectangle in order to test its precision. This approach guaranteed that they had fully understood and tried the functions before they filled out the section of the form regarding that particular concept (Figure 5). An even numbered scale (six) was used to avoid indifference, i.e. the trial subjects could not choose the midpoint on the scale. This was repeated with all three concepts before each trial subject finally was asked to rank the concepts in order of preference. Notes on the interaction were taken during the time of the test.

	I do not agree					*I agree*
It's easy to understand <u>how</u> to use it	○	○	○	○	○	○
It's easy to use	○	○	○	○	○	○
It's fun and feels good to use	○	○	○	○	○	○
The precision is good	○	○	○	○	○	○
Comments:______						

Figure 5 - Question form used in the trials

Participants in the trial

About 50% of the trial subjects were people who had been invited to the site to participate in the trial. These participants had been hand-picked to represent different age groups and educational backgrounds. They also had varying experience with technology. The remaining part of the participants was people passing by who were asked on the spot and agreed to participate.

Data collection and extraction

The data collected from the trial was subjective and semi quantitative (Bligård, 2011). This data could be used to calculate the average grading of the concepts, both in total for each concept and for each of the four graded characteristics (Figure 5), as well as the average ranking. The data also revealed how many of the trial subjects who tried single touch versus multi touch gestures as a first action.

Since two of the concepts received very equal grading and ranking, the same calculations were done once more, this time using only the data from the concept that was tested first by each trial subject. This did however not generate any significantly different results, plus the sample size was questionably small.

There was also an attempt to single out the best concept by assigning weight coefficients to the different questions that corresponded to the importance of that particular characteristic. For example, ease of use was considered more important than precision. Unfortunately, this approach did not facilitate the choice of concepts since they appeared to be even more equal when the weights were included.

2.7 CONCEPT REFINEMENT

In this phase the final concept for each part of the design was refined and designed in detail. This included the design of the physical appearance of all corner stones as well as variations of secondary issues such as border thickness, viewport measurements etc. Though most of the partial concepts had already been comprehensively described earlier in the process, this phase aimed to put together the final solution with no ambiguities in the design description. The work also involved defining the algorithms for the function of the different corner stones, some more complex than others.

2.8 INTEGRATING CONCEPTS INTO THE FINAL GUI

As previously mentioned the layout, viewport and menu system were developed separately and functioned fairly independently from each other. The final phase of the development work meant integrating these three corner stones to create the final GUI for the main screen application. In addition to the integration of the cornerstones, a list of all variables which can be adjusted by the program designers was also compiled during this phase.

3 LITERATURE REVIEW

Before the idea generation and concept development, a thorough literature review was carried out. In this section a selection of the findings that have been of interest throughout this thesis and may be of interest for further development is presented.

3.1 INTERACTIVE ITEMS

Interactive items refer to buttons or controls that in some way respond to touch gestures. In the sub-chapters below recommendations for designing these interactive items are presented.

3.1.1 ICONS AND SYMBOLS

An *icon* is a graphic, which is supposed to resemble the physical object or action it represents. For example, the recycle bin on the computer desktop represents a physical recycle bin, i.e. the place to throw trash.

A *symbol* on the other hand, is a sign that is agreed by convention to stand for something, e.g. in computer context a floppy disk is used to represent the action of saving your work even though not many young people nowadays actually ever have used one or even know exactly what it is. Written text is also considered to fall into the symbol category. Since symbols can mean different things for different people, and can be read in other ways than intended by the designer, it is often better to use icons in a GUI when this is possible (Vile & Polovina, 2000).

3.1.2 BUTTONS

Touch screens allow the interface designers to mimic physical components and the way they work and respond to touch. While observing the visitors in the Berlin Museum of Natural History, Eva Hornecker found that users interacted with button-shaped items similar to the way they would have with a physical button, i.e. with quick and hard pushes and hits (Hornecker, 2008).

3.1.2.1 SIZE AND SPACING

When it comes to the button size on touch screens, larger size leads to better performance. The EIA Committee on Product Accessibility recommend a smallest button size of 19.05x19.05mm (Xia Jin, Plocher, & Kiff, 2007). There are different recommendations for the distance between buttons, while ISO standards suggests a distance of 5 mm, others propose that 10 mm between buttons is preferable (Xia Jin, Plocher, & Kiff, 2007). However, these numbers were concluded from experiments on kiosk displays, which are a lot smaller than the touch screens used in this thesis.

3.1.2.2 CLOSE AND ERASE CONTROLS

It should not be too easy for the users to accidentally hide or delete objects in an interaction, because doing so by accident can frustrate them and even cause them to leave the interaction. To avoid these mistakes, frequently used controls should not be placed too close to the delete controls. Likewise, if a gesture is defined, that gesture should not be too similar to other gestures used in the interaction (Peters, Gastrich, & Rome, 2011). The designers of the previously mentioned touch screen table top of the Berlin Museum of Natural History tried to solve this dilemma by requiring the close button to be pressed for a longer time than other buttons, but people did not understand this function and were rather annoyed (Hornecker, 2008).

3.1.2.3 POINTING

It is a common behaviour to point at the thing one talks about (Hornecker, 2008), but if the user points too close to or accidentally touches the screen, the screen will react to the touch event. Buttons etc. should therefore not invite the user to point at them. If possible, they can be made easier to describe in words, decorated with for example an "X" or an arrow, so that a user explaining to another user how to interact explains this verbally instead of by pointing.

3.1.3 MENUS

Interactive menus may be designed in any number of ways, though the most common designs can be divided into three main categories; folders, toolbars and drop-down menus.

3.1.3.1 FOLDERS

The most common operating systems on personal computers use folder solutions for browsing. The user is then presented with a number of icons or symbols representing either an object, application or a folder which can be used to reach another sub level (Figure 6).

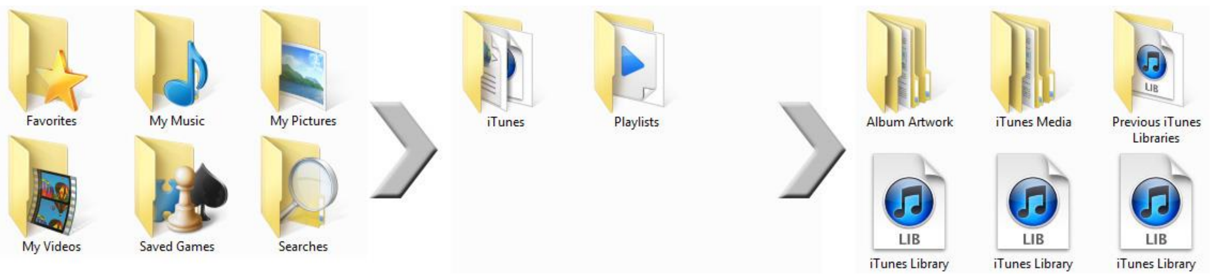


Figure 6 – Folder menu on a computer

This is a very effective way to sort a large amount of information in a way that is easy for the user to survey. It does however require the user to go back up the hierarchy to choose another application, and frequently used functions should not be buried under multiple layers of menus (Kaminani, 2011). Besides computers, this concept is also used in touch screen kiosks for buying tobacco etc. (Figure 7).

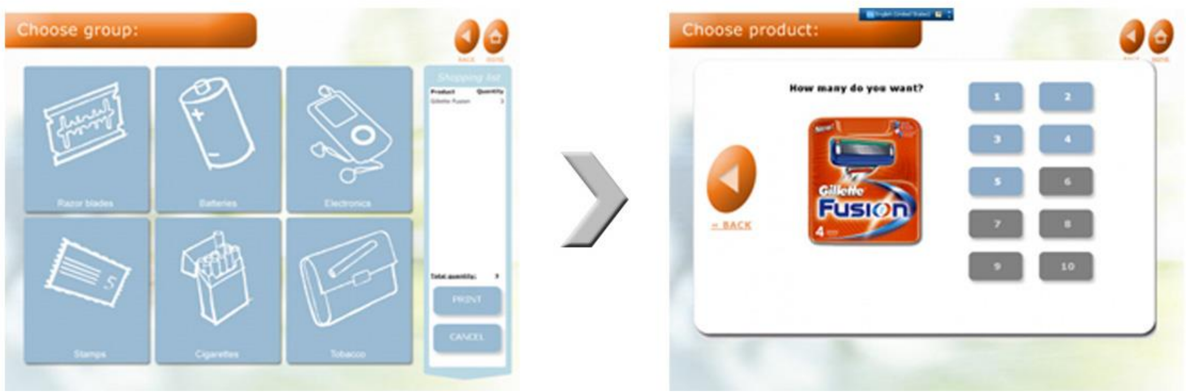


Figure 7 - Folder menu on a kiosk.

3.1.3.2 TOOLBARS

A second type of menu that is common in computers is the toolbar (Figure 8). It can be used with either graphic symbols or text to define content. A benefit with these is that they can easily be visible and reachable at all times without occupying too much space, unless they need additional rows for sub menus.

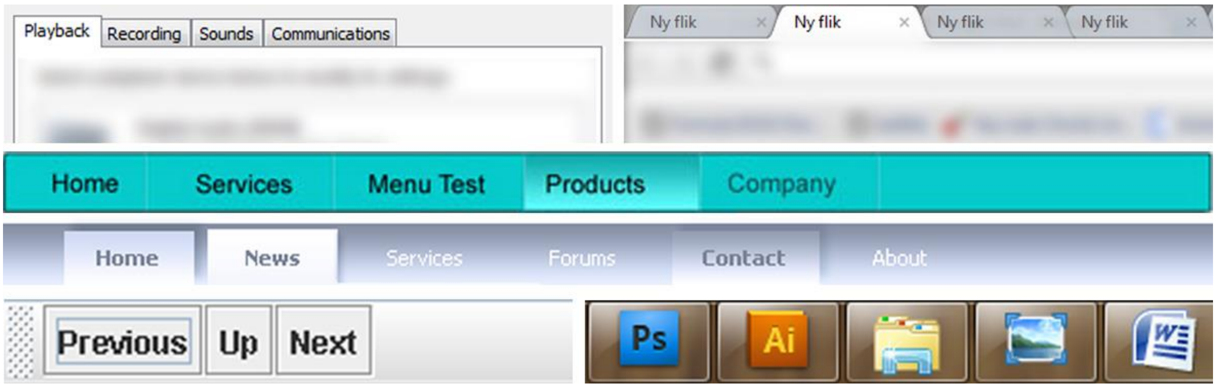


Figure 8 - Examples of toolbars on computers

3.1.3.3 DROP-DOWN MENUS

Another common type of menu is the drop-down menu (Figure 9) which can be called on by a click on an icon or similar. This solution only requires the one icon that is used to open the menus to be visible at all times. Just like the folder solution, drop-down menus can easily be used with sub categories.

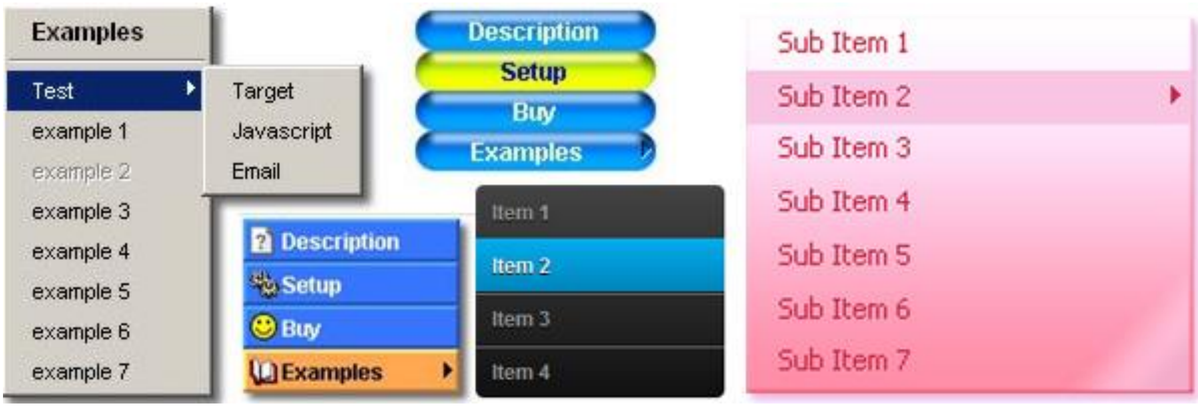


Figure 9 - Examples of drop-down menus on computers

3.2 USER GESTURES

Users may call on different functions by using different touch gestures on the screen. The gestures can range from simple clicks with one touch point to more advanced gestures, for example drawing a circle to open a menu (alai6666, 2007). During the usage the users learn continuously, and if they were to experience for example a long response time at some point, they will frame a mental model that will be applied to all other tasks (Kaminani, 2011).

3.2.1 PINCH/UNPINCH

A common technique on touch screens is to pinch or unpinch in order to zoom in and out or to shrink or stretch an object (Peters, Gastrich, & Rome, 2011). This can be done with one or two hands using one or zero stationary fingers.

Perceptive Pixel (Arcarpm, 2010) uses a technique where the viewport is scaled by zooming in and out using this pinch/unpinch-technique on the frame itself (Figure 10).

Jeff Han illustrates a variant of this technique where the user is required to use both hands to resize the viewport (alai6666, 2007). The viewport of this application does not have a visible border. Jeff Han who is demonstrating the functions is obviously familiar with the software, but it may be difficult for a newcomer to understand that this action is only possible in one particular area if is not indicated by any visual cues (Figure 10).

In both of the above cases the viewport itself is moved to another position by dragging and dropping using one finger.

Figure 10a - Pinch/unpinch technique used with one hand (left) and with two hands (right).

In application windows within the operating system Windows, a wide top border is used to move the application window around. The outmost frame sides, including the top and bottom, are used stretch and shrink the window, whereas the corners are used to scale, stretch and shrink them. A very similar mechanism is used on computers with the operating system Mac OS X. Examples of GUIs provided by these two leading operating systems can be seen in Figure 11 below. Since Windows market share is almost 92 % and nearly 7 % of the remaining share belongs to Mac OS X (Net Marketshare, 2012), it should be safe to say that most people who have used a personal computer are familiar with these techniques.

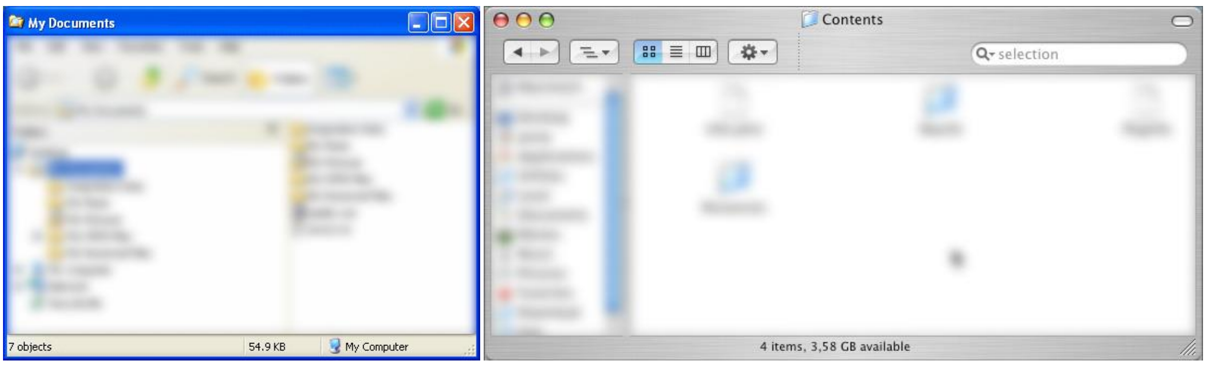


Figure 11 - Application windows in the operating systems Windows (left) and Max OS X (right).

The computer users are made aware of the existence of these functions by the use of different cues. For example, the mouse pointer transforms into an arrow when the user hovers it over the frame, indicating that a click and drag action is possible. However, this is not possible for a touch screen where only the touch actions are registered. Other cues used in these computer windows are the triangular symbols seen in the bottom right corner of the windows, which aim to invite the user to click and drag. Another symbol sometimes used to describe the action of moving something is the two crossed double pointed arrows (Figure 12).

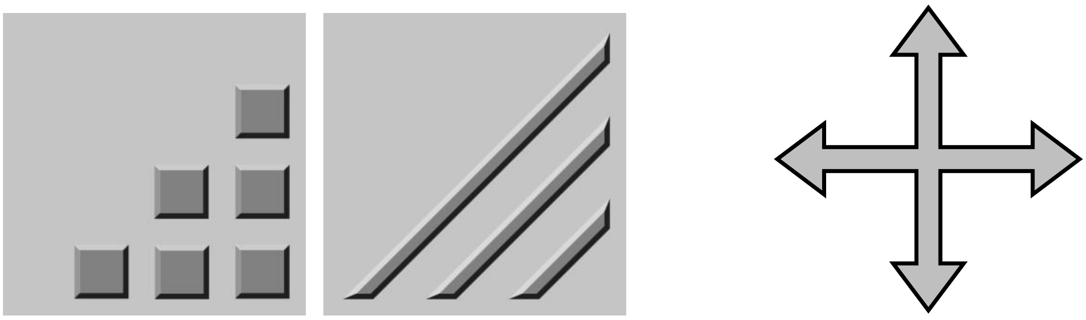


Figure 12 – Scaling symbols used in corners of windows on computers (left) and moving symbol used on computers (right).

3.2.2 SWEEPING

The interactive table top used at the previously mentioned Berlin Museum of Natural History used two different mechanisms for scrolling. Firstly, a technique similar to the one used in computers containing simple up and down arrows. Secondly, a direct dragging of the text itself by a sweeping gesture, a technique often used on smart phones. Although Hornecker did not conclude which mechanism worked better, she did discover that using both of them in one application seemed to cause a cognitive conflict. The arrow technique can be thought of as a viewing window that moves over the text, while the other technique resembles the physical action of dragging the text itself up and down like a sheet. Since the text on the table top did not have any visible border around it, it was quite difficult for the users to know where the active region allowing a sweeping motion started and ended. This caused users to sometimes drag their fingers beyond the sweep area without understanding why the scrolling suddenly stopped (Hornecker, 2008).

3.3 ERGONOMICS

Interaction on a vertically oriented touch screen should not require sustained contact, since that may force users to hold their hands out at arms' length for a longer period of time than would be comfortable (Symanski, Goldin, Palmer, Beckinger, Gilday, & Chase, 2008). Activities needed for interaction should be designed to avoid sustained contact so that the users can rest their arms periodically (Peters, Gastrich, & Rome, 2011) and have the elbows in an angle of about 90° (Boghard, et al., 2011).

3.3.1 OCCLUSION

Sometimes when using a product the users block visual information or controls with parts of their bodies, as shown in Figure 13. Part of the screen is blocked from view by the users thumb. This phenomenon is called occlusion (Sousa & Matsumoto, 2007). When working with touch screens, occlusion is most likely to be caused by the users' hands, arms or fingers. When designing a touch screen application it is therefore important to keep in mind that about 90% of the world population is right-handed (APA PsycNET, 2012).

Figure 13 – Occlusion occurring during use of a smartphone.

3.3.2 VISUAL PREFERENCES

When deciding what areas of a touch screen will be used for which purposes it is important to take human behaviour into account. People tend to prefer focusing on objects 15° below the horizontal line, with an angle of view of 30° (i.e. ±15°); though having to focus on objects up to 45° below the horizontal line is considered to be within an acceptable range. (Pheasant & Haselgrave, 2005)

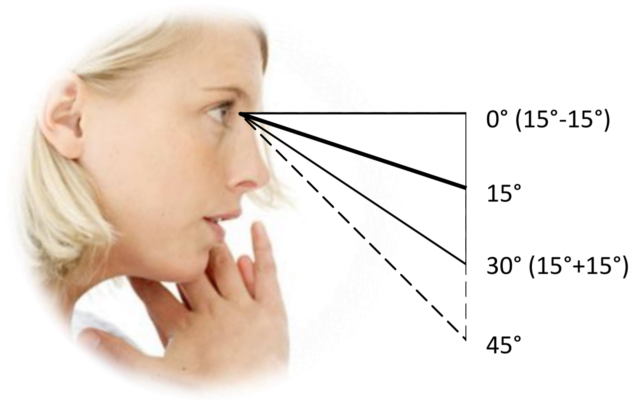


Figure 14 - Visual preferences

3.3.3 COLOUR AND CONTRASTS

The choice of colour is important in any design. As 8% of all men and 0.5% of all women are colour blind and since 99% of those suffer from red-green colour blindness (Flück, 2010) these colours are extra important to keep in mind when creating designs. If possible, the design should be redundant, meaning that colours are only used as an enhancement and not as the only existing cue (Boghard, et al., 2011). For example, in a traffic light green means go and red means stop, but since the lights are placed in a specific order, the driver only needs to know whether the top or bottom light is lit to determine the traffic light status, the colour is merely added to make it even clearer.

It is recommended by the Web Content Accessibility Guidelines Working Group that the visual presentation of text and images containing text on websites should have a minimum contrast ratio of at least 4.5:1 (Vanderheiden, Guarino Reid, Caldwell, & Lawton Henry, 2012). The recommended ratio for text in other contexts, for example printed on paper, may differ from this number, but a lit screen on a computer is more similar to the touch screens used in this thesis.

3.3.4 AESTHETICS

D. A. Norman argues that good aesthetics have the ability to make people better problem solvers. In pleasant and positive situations users feel more relaxed, which make them more tolerant to minor difficulties and more probable to overlook problems with the interface design. They also get more creative and flexible in finding solutions to whatever problem they are faced with. Furthermore, a negative perception of the presentation will require the user to focus more extensively on the information. This may result in better concentration, although it leads to a narrow tunnel vision, which may be good in some situations but not ultimate for problem solving (Norman, 2002).

3.4 INTERFERENCE BETWEEN USERS

It can be assumed that a product which is used by several users have different requirements than products intended for single users. The users may affect each other during the usage in various ways and different solutions can result in different levels of interference between users.

3.4.1 USER CLASSIFICATIONS

Users are often classified based on their relationship to each other or the product. Buur and Winbur divided users into two categories; primary users, which are those who use the product for its primary purpose, and secondary users, who also use the product but not for what it is primarily intended for (Buur & Winbur, 1994). Janhager added two more user types to this classification; side-users, referring to people who are affected by the product but not using it, and co-users, who in some way cooperate with a primary or secondary user without actually using the product themselves (Janhager, 2003).

3.4.2 LAYERS ON TOUCH SCREENS INTENDED FOR MULTIPLE USERS

There are a number of solutions on the market today for handling multiple objects such as photos, videos and textboxes on a touch screen. In most of these solutions the objects are sorted like layers, much like the windows on a computer. There are basically three different kinds of common layer concepts.

The layer order may be pre-defined, meaning that an object in the background cannot be brought to the front or vice versa (Figure 15). To reach objects in the background the user will have to move the objects, which are in front, to another location (patcon96, 2007).

Figure 15 - Multi-touch application with a static layer order, made by Obscura (patcon96, 2007)

A second solution that has been used, by for example divITSystems (Figure 16), keeps the most recently touched or manipulated object in front. Therefore, objects which have not been used for some time will eventually end up in the background (divIT, 2012).

Figure 16 - Multi-touch application where the most recently manipulated image is placed on top, made by divITSystems.

A variant of the previously mentioned solution is one used by MultiTouch Ltd where the object that has most recently been opened stays on top (multitouchfi, 2010). This of course requires that the objects can be opened by the users and is not in full size from the start, which is the case with some applications. In the MultiTouch Ltd application (Figure 17), several miniatures of the pictures are floating on a row in the background, and require the user to tap them to open. There are also applications by others creators where miniatures or icons float around randomly, for example Obscura (Figure 15).

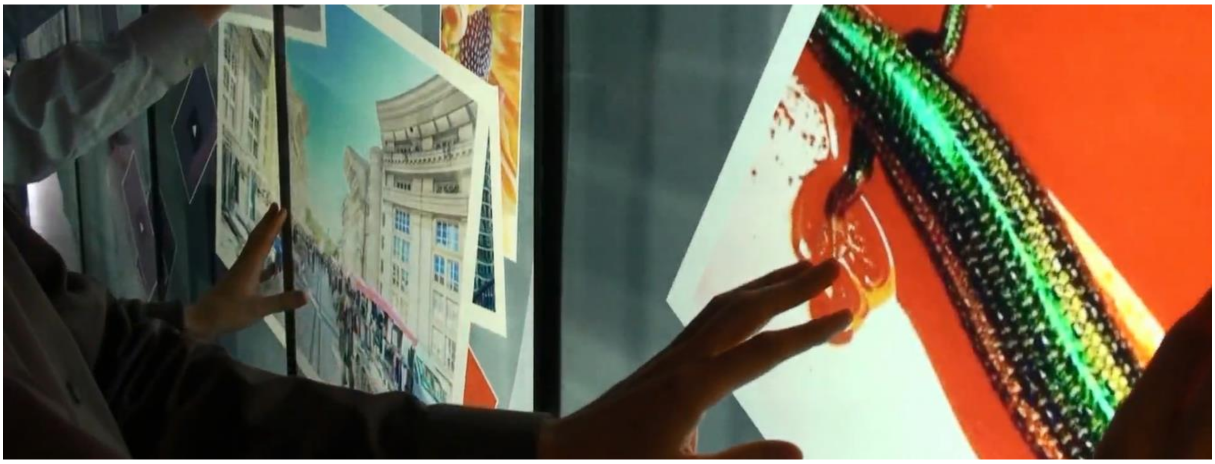


Figure 17 - Multi-touch application where the most recently opened image stays on top, made by MultiTouch Ltd.

All these solutions using layers require a mutual respect between users since they have the possibility to invade each other's space. This can be done primarily by covering an object that someone else is working with, with their own object. Interference between users' workspaces is a common problem with multi touch screens that are used by several simultaneous users (Hinrichs & Carpendale, 2010). This problem can be diminished by a design that clearly defines the workspaces and does not allow for these workspaces to be positioned on top of each other (Peters, Gastrich, & Rome, 2011).

3.5 INTERACTION DESIGN DILEMMAS

In the following sub-chapters additional design dilemmas are presented and discussed which could not be categorized into any of the previous chapters.

3.5.1 PUBLIC ARENAS

In public arenas, users are confronted with multi touch screen systems without any previous knowledge of its function. In a setting where objects compete for attention and where the user is free to interact or not, the system has to survive on its own (Hornecker, 2008). A low threshold for interaction is essential and the first ten seconds of usage need to provide an incentive to continue (Hornecker & Stifter, 2006).

Hinrichs and Carpendale observed that strangers working on a multi touch screen in a public space interacted with each other while faced with conflicts or problems. They also often tried to help each other troubleshoot interactions, though not all users seemed to want a social experience while working at the multi touch screen (Hinrichs & Carpendale, 2010).

An advantage with public arenas is that they allow for over-the-shoulder learning, meaning that users who are not comfortable with trying new things in public can observe from afar (Brignull, Izadi, Fitzpatrick, Rogers, & Rodden, 2004). Seeing people using the system do not only teach other users how to use it, it also makes them aware that the content is interactive. At the Berlin Museum of Natural History, not everyone who passed by realized this, and sometimes it was discovered by accident. It may however be more common to accidently realize this on touch screen table tops since people tend to lean over it to see better or slide their hands over it when passing by. Also, many of the visitors were children who have less hesitation in touching objects (Hornecker, 2008).

Another thing that differentiates younger users from adults is that children need much less guidance in order to start interaction (Lepicard & Vigouroux, 2010). Adults tend to be quite cautious and often begin interaction only after watching other people first, while young adults approach interaction with the purpose of figuring it out, perhaps because they are more used to that kind of technology (Peters, Gastrich, & Rome, 2011).

3.5.2 INVITING PEOPLE TO INTERACT

There is a risk that multi touch screens will be taken for static video displays. As mentioned above, not everyone in the German museum realized that the content was interactive, since there was not always somebody else there who used it. In an attempt to show users the interaction possibilities of the table they used pointing hand icons next to some of the items, which were meant to invite the users to touch the item in question (Hornecker, 2008).

Some suggests the use of animations to draw attention to interface controls, in particular pulsing, colour cycling and fading, since these animations do not change the location of the controls on the screen (Symanski, Goldin, Palmer, Beckinger, Gilday, & Chase, 2008). The designers of the Plasma Poster at FX Palo Alto Laboratory also worked a lot with animations, choice of display colour and font size to get the screen to stand out in the local lighting conditions and to invite for both observation and interaction (Churchill, Nelson, & Denoue, 2003).

Another solution proposed is to use an avatar-based help system to assist users with onscreen interaction (Vanacken, Demeure, Luyten, & Coninx, 2008). In this solution the avatar would appear either when a user aimed to complete a task but failed repeatedly, or when the system had been interactive for a certain period of time.

3.5.3 USE GRADUAL ENGAGEMENT

When confronted with an interactive touch screen, users are most likely to only use one finger in their first attempt to manipulate an object (Jacucci, et al., 2010). Therefore, multi touch gestures should only be required when the users have had a chance to familiarize themselves with the system.

3.5.4 ENSURE ONSCREEN CHANGES ARE A RESULT OF USER INTERACTION

All changes on the screen should be a direct result of user interaction (Moscovich, 2007). An object that moves or changes appearance without input from the user can be very confusing. This is especially important to consider when multiple users work close to another without collaborating.

4 PRE-STUDY

In order to gain more information applicable to the scope of this thesis, a pre-study was carried out. The study focused on defining use-cases, user profiles and also concludes a prioritization between product characteristics. These definitions are used as a basis throughout the following concept development work.

4.1 USE CASES

To get a better understanding of the environment in which these touch screens and accommodating main screen applications can be used, three possible scenarios are described below.

Shop-windows - These kinds of applications often aim to entertain and at the same time promote the company. The users may not be that interested in the content, but rather playing with the application for the sole reason to be entertained until the bus or their friend arrives. Since there is no pressure to perform, functions may be discovered ”by accident” rather than by conscious problem solving. (Figure 18)

Figure 18 - A touch screen located in a shop-window (i4sense).

Fairs - Whether it is a fair for newly launched products or job opportunities, fairs are often characterized by a large number of people in motion. These people are in most cases interested in the content of the application rather than the application itself and the amusement of using it. Because of the large amount of people, users can have the possibility of learning how the application works by first observing other users. Nevertheless, the feeling and knowledge that they are being observed can make some users uncomfortable if they cannot figure out how it works right away. (Figure 19)

Museums – In a museum there are a lot of other objects competing for the visitors' attention. The users are of varying ages, including both children and seniors, and do not necessarily have any particular technical skills. The relaxed environment invites the users to feel their way with the application and, just like the shop-window scenario above, discover functions by accident. (Figure 19)

Figure 19 – Large scale touch screens in different environments (MultitouchSupplier), (Lawrence).

4.2 USER PROFILES

Primary users: This product is meant to be used by adult men and women of various nationalities. All users should be literate and have at least moderate English skills. They are not expected to have any previous experience with the software and are therefore considered to be novice users. Most of the users are assumed to have some basic computer skills, at the very least to the extent that he or she is familiar with common symbols used in a computer environment. The design is not intended for people with severely impaired vision or for people with loss of sensation in fingers. Though the solution should be adjusted to suit adult people of normal height, it is beneficial that people with shorter reach, for example children or people in wheelchairs, may also be able to use it.

Secondary users: The secondary users are the programmers at i4sense who are responsible for maintaining and updating the software. These users are mainly 20-35 year old Swedish men with an education in engineering. They use the program Unity3D in their daily work and can therefore be thought of as expert users. Anyone watching the screen without physically interacting with it is also considered to be secondary users.

Side-users: This product is often located in public areas, which makes people in the vicinity side-users. The product should be of as little inconvenience as possible for these people, who may just be passing by without any initial interest in the product.

Co-user: In order to be defined as a co-user for this product a person would have to be either at a different location communicating with the primary user on the phone or some other media or next to the primary user but without being able to actually see the screen. Should the user see the screen, he or she would be defined as a secondary user, as mentioned above.

4.3 IMPORTANT PRODUCT CHARACTERISTICS

A Use Profile Table was conducted in order to find which areas to focus on during the development, i.e. which characteristics are especially important for this specific product (Janhager, 2003). When applying the user profile defined above on this Use Profile Table one can determine the importance of the various characteristics of the product. For each row of Table 1 , the most applicable choice, for example "Newcomer", is chosen and this corresponds to a level of high importance within the adjacent category, in this case "Easy to use". If the level "Specialist" had been chosen, the level of importance for "Easy to use" would have been very low.

Table 1 - A Use Profile Table with the degree of performance for this product marked in bold font.

	Categories	Degree of performance	Extent of importance of the product
Use experience	*Length of use and education*	**Newcomer** Experienced Specialist	▽ Easy to use
	Frequency of use	**Rare** Occasional Frequent	△ Physical ergonomics △ Stress factor
Influence on and responsibility of use	*Influence on the choice of product*	**No influence** Some influence Great influence	▽ Adaptability
	Influence on the use situation	**No influence** Some influence Great influence	▽ Physical ergonomics ▽ Confidence
	Responsibility in use	**No responsibility** Some responsibility Great responsibility	△ Reliability △ Confidence
Emotional relationship to the product	*Ownership*	**Use of general product** Use of rented product Use of owned product	▽ Easy to use △ Characteristics ▽ Adaptability
	Social aspects	Of little importance **Of some importance** Of great importance	△ Aesthetics/Sense △ Characteristics
	Mental influence of product	User with no mental influence **User with some mental influence** User with great mental influence	△ Semantics △ Aesthetics/sense
Degree of interaction with the product	*Cognitive interaction*	No cognitive interaction Some cognitive interaction **Great cognitive interaction**	△ Semantics
	Physical interaction	No physical interaction **Some physical interaction** Great physical interaction	△ Physical ergonomics

In order to translate the UPT triangles to results usable for this thesis, the triangles are translated into numbers; the wide end = 3, middle of triangle = 2 and the sharp end = 1. For every instance a certain aspect appears, a grade is received. These numbers are summarized in Table 2, which also contain an average calculation and its priority level for each aspect. As can be seen, the top priorities when designing this product is to make sure that it has a high adaptability and that it is easy to use. A good use of semantics is also essential.

Table 2- Calculation of the priority levels for the characteristics of the product.

Aspect	Total score	Average	Priority
Easy to use	3+3 =6	3	1
Physical ergonomics	1+3+2=6	2	3
Stress factor	1=1	1	5
Adaptability	3+3=6	3	1
Confidence	3+1=4	2	3
Characteristics	1+2=3	1.5	4
Aesthetics/Sense	2+2=4	2	3
Semantics	2+3=5	2.5	2

Priority	Aspect
1	Easy to use
	Adaptability
2	Semantics
3	Physical ergonomics
	Confidence
	Aesthetics/Sense
4	Characteristics
5	Stress factor

5 LAYOUT

The layout is the arrangement of all the different parts of the application, i.e. what it looks like when it is running on the touch screen. There were two main problems to solve when designing the layout; how to open new viewports and how the viewports should be placed and interact with each other. Being able to open a viewport is a prerequisite for the following interaction, but the interaction between viewports is an ongoing event, which affects the users throughout the usage. An unsuccessful solution may cause the users to leave the interaction prematurely. Initially, solutions for inviting people to interact was also considered, meaning that the design should have some element which made passers-by want to stop and interact. However, it was soon realized that trying to solve this dilemma took more time than justifiable.

5.1 IDEA GENERATION

A large number of ideas for the layout and associated components were generated through brainstorming. They were the result of four brainstorming sessions with increasing level of detail (Figure 20). However, as mentioned above, getting people to interact was found to be a too complicated task, which led to the decision to exclude it from this thesis. The general ideas for the other focus areas are presented below.

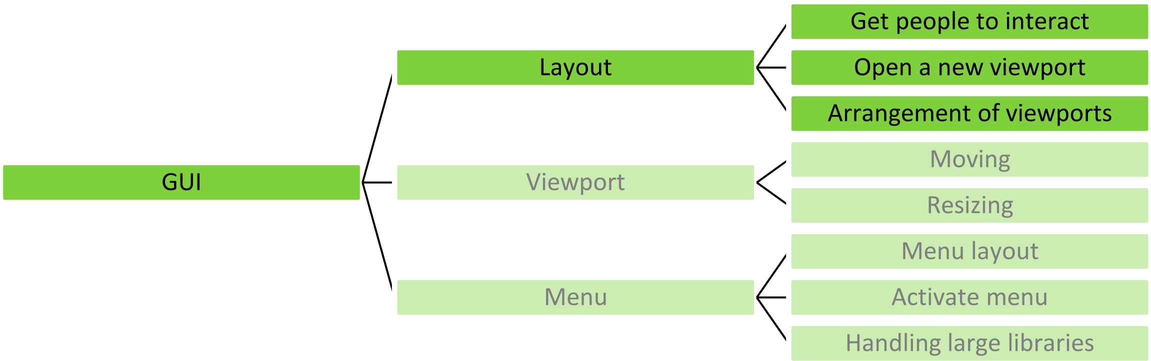


Figure 20 - The brainstorming sessions for the layout.

5.1.1 OPEN A NEW VIEWPORT

New viewports can either be opened by clicking or touching an item or by a specific touch gesture. The former obviously requires the presence of an item to touch. These can in turn behave in different ways; either moving around, being stationary, disappearing after being touched, or as an icon (small representation of the real viewport) being enlarged to full size. If icons are used, they may be arranged in a structured way, for example along one of the borders or in a random way (Figure 21).

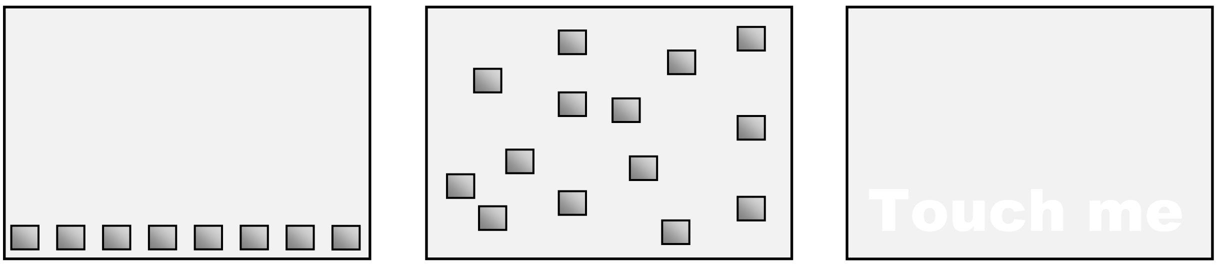


Figure 21 – Icons arranged along the border (left), scattered on the screen (middle) or a text inviting the users to touch the screen (right)

If the users are required to perform a specific touch gesture in order to open a new viewport, this touch gesture needs to be very simple. The users should not be expected to perform advanced gestures as a very first action of the interaction. One benefit with this solution is that it does not require any item (Figure 21), making the visual design look less cluttered and easier to adapt.

5.1.2 ARRANGEMENT OF VIEWPORTS

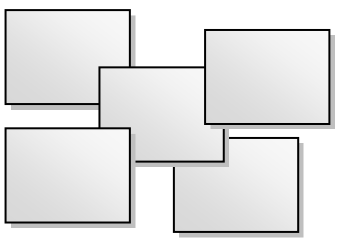
Three variants for the arrangement of the viewports and their interaction towards each other were considered. They could either be arranged as layers (Figure 22a), colliding boxes (Figure 22b) or in a grid (Figure 22c). Below is a short description of each variant, including some of their advantages and disadvantages.

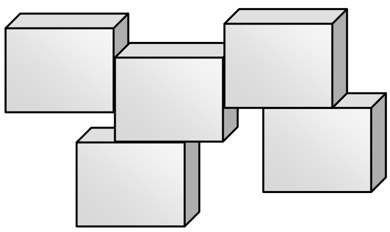
Layers - In the layer arrangement the viewports can be thought of as papers scattered on a table (Figure 22a). They can be placed on top of each other and cover each other completely or partially when they are moved around or resized. One benefit with this solution is that it provides a very natural feel when every motion is followed through since nothing is blocking them. On the other hand, this increases the risk of disturbing other users.

The order of the layers can be decided in a number of ways. For example, the most recently manipulated or the most recently opened viewport can be placed on top. The order may also be decided by the size of the viewport. If the smallest viewport is placed on top of a larger one it will only cover parts of it.

Colliding boxes - An alternative solution for the arrangement of the viewports is to have them work like three-dimensional boxes that will collide if they get too close to each other (Figure 22b). The viewports do not necessarily have to look like boxes as long as they behave as such, or the boxes could be viewed with an orthographic projection. To alert people that two boxes/viewports are colliding, this event needs to be recognized, for example via a visual signal, since an auditory signal would not be suitable for a multi touch application.

Grid - The viewports could also be arranged according to a predefined grid (Figure 22c). A benefit with this solution is that every user is allowed an equal amount of space, and there is no need or risk for them to intrude on each other's work spaces. This design is however not practical for situations when the number of users changes often since the viewports would have to change both size and place every time a user initiates or leaves the interaction.





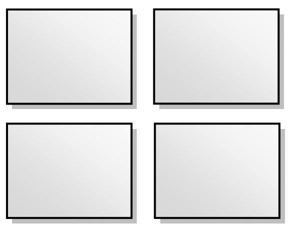


Figure 22a – Viewports arranged as layers

Figure 19b – Viewports arranged as boxes

Figure 19c – Viewports arranged in a grid

5.2 SCREENING

All of the generated ideas were assumed to be able to contribute to the following combination of ideas to create a full concept. Therefore, no ideas were discarded during the screening phase.

5.3 COMBINING IDEAS INTO CONCEPTS

A morphological matrix was used to combine the ideas from above into a manageable number of concepts. The morphological matrix included various solutions for both focus areas, i.e. how the viewports should be arranged and how new viewports should be opened. This specific matrix allows for 20 unique combinations, many of which would be extremely similar to each other. Since the issue of how the viewports should be arranged and interact towards each other was established to be the most critical, one concept with each of the solutions for this was chosen to be sketched into concepts and evaluated further.

Table 3 - Morphological matrix showing how the three concepts were generated. The bold font indicates that there are different variants of the named solution, all of which are written in the cells directly below. I.e. it is not possible to move from a box containing bold text to a box which is not positioned right beneath it.

Viewport arrangement	**Layers**			**Colliding boxes**	**Grid**
Layer arrangement	Most recently opened	Most recently manipulated	Smallest in size	None	
Means for opening new viewports	**Touchable icon**			**Specific touch gesture**	
Location of icon/ type of gesture	Floating around in background	Around the edges of the screen		MT	ST

Concept 1 Concept 2 Concept 3

Concept 1 – Layer based with floating icons

The viewports are arranged as layers where the most recently opened viewport stays on top (Figure 23). Since viewports are allowed to overlap, the user friendliness is strongly influenced by the users. When the design does not restrict users from intruding on each other's space, the interaction between users is based on mutual respect.

Viewports are opened by touching one of the icons floating in the background. The user cannot decide exactly where the viewport opens, since that is decided by where on the floating path the icon is situated when activated by the user. This may however be solved by adding the possibility for the users to click and drag the icons to the position where they want it to open. These gestures may be advanced for new users but since this is only an additional feature it is not considered a disadvantage that effects the concept in a negative way. The users should never have to wait for a floating icon to appear within reach, so the number of icons should be much higher than the number of possible open viewports

.

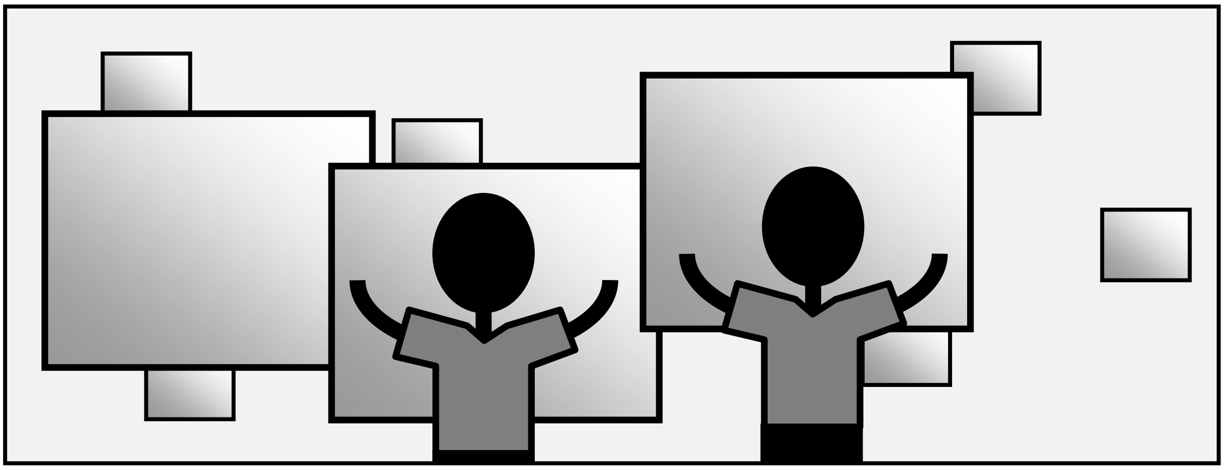


Figure 23 – Layer arrangement with the most recently opened viewport on top and touchable icons floating in background.

Concept 2 – Colliding boxes opened by single touch

With the viewports acting like boxes that can collide with each other the users will not be able to cover others viewports with their own and in that way interrupt or disturb each other's usage (Figure 24). However, the collision needs to be visualized to make it clear to the users what happened if the viewport stops to respond to active touch gestures because it has been blocked. There are no items that aid the opening of new viewports, instead there needs to be a hint somewhere, telling the user to simply click the screen in order to open a viewport, either in text or for example a pointing hand icon.

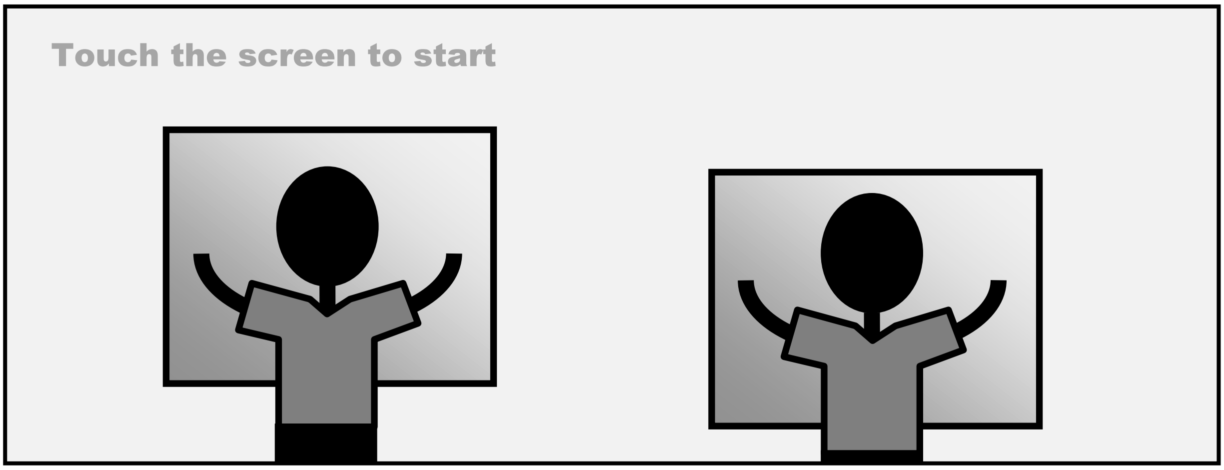


Figure 24 – Viewports arranged as colliding boxes opened by single touch on screen.

Concept 3 – Grid with a predefined number or viewports

This solution allows for the users or an operator to select the number of displayed viewports with a touch button. If for example four people are using the screen, the no. 4 touch button will divide the screen into four equally sized viewports. An advantage with this solution is the utilization of the display area. The obvious drawback is that the users have very little control over their own viewport, since it is locked to a grid and cannot be moved or resized, and whenever the number of viewports changes, the current users´ viewports will be moved and resized to fit the grid.

Depending on where the screen is situated, it might be better to place the buttons along the lower border instead, so that they are easier to reach for children or people in wheelchairs. An alternative to the numbered touch buttons could be to only have two buttons, one with a plus sign and the other with a minus sign, which would increase or decrease the number of viewports.

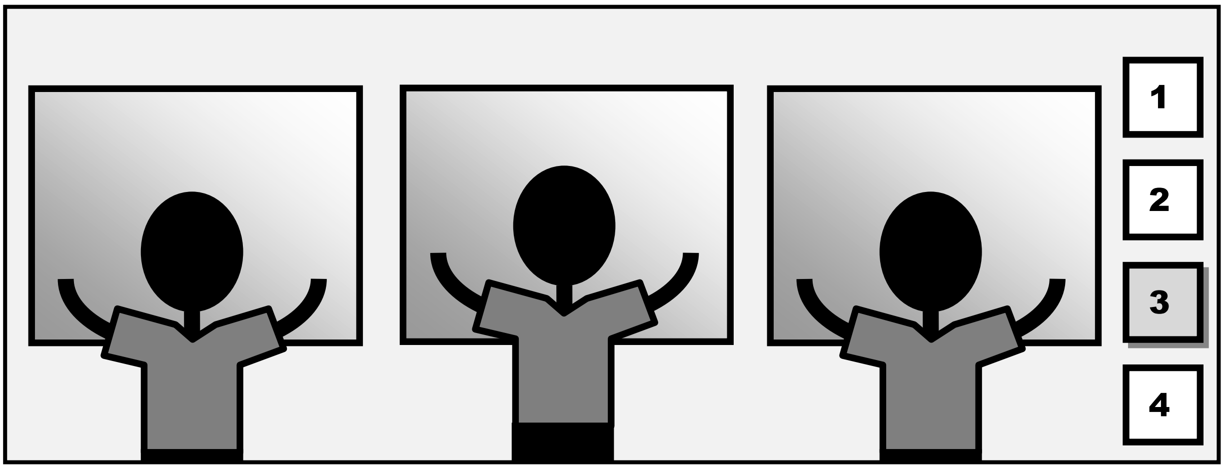


Figure 25 - Viewports arranged in a grid which is called on by icons arranged along one side of the screen.

5.4 CHOOSING A CONCEPT

In order to evaluate these three concepts and decide which one to move forward with, a list of weighted criteria was set up in accordance with the findings from the literature review and the pre-study. The criteria which follow are accompanied by a short explanation and the weight factor within parentheses:

- *Easy to understand that simultaneous use is possible* (w=3) – If this is not clear to the users the whole idea with the product is lost. This is therefore a very essential criterion, which is way it received the highest weight possible. (4.3)
- *Easy to understand how to use the screen simultaneously* (w=3) – Knowing that simultaneous use is possible is of no use if one does not understand how to accomplish it. This goes hand in hand with the above criterion and was weighted accordingly. (4.3)
- *Should not instil a feeling that one is intruding on another's space or vice versa* (w=2) – Users should be able to work in their viewport without interference from fellow users. Intrusion does not necessarily hinder the functionality, though the users may not enjoy the usage as much if disturbed by fellow users. A medium weight was therefore appropriate.
- *Current users should not be disturbed when new users join the interaction* (w=2) – Any changes to one user's viewport should be a consequence of his or her own actions, not somebody else's. The weight factor was based on the same reasoning as the above criterion. (3.5.4)
- *Efficient use of space* (w=1) – The solution should allow for as much of the screen as possible to be used. As this criterion does not have a direct impact on the functionality of the product, nor the active users' experience, it received the minimum weight.
- *Easy to open a new viewport* (w=3) – Opening a viewport is the first thing a new user does, meaning that the gesture required to do this needs to be very simple. This action is a prerequisite for the following interaction and it was therefore weighted as highly important. (3.5.3)
- *Adaptable look* (w=3) – The look of the layout should be easy to customize. This is supported by the Use Profile Table, which revealed that adaptability was one of the top priorities for this product; hence it too received the highest possible weight. (4.3)

Table 4- Decision Matrix for evaluating the three layout concepts.

Criteria	Weight	Concept 1 Grade	Concept 1 Total	Concept 2 Grade	Concept 2 Total	Concept 3 Grade	Concept 3 Total
Easy to understand that simultaneous use is possible	3	3	9	3	9	3	9
Easy to understand how to use the screen simultaneously	3	3	9	3	9	2	6
Should not instil a feeling that one is intruding on another's space	2	1	2	3	6	3	6
Current users should not be disturbed when new users join the interaction	2	2	4	3	6	1	2
Efficient use of space	1	3	3	1	1	3	3
Easy to open a new viewport	3	3	9	3	9	3	9
Adaptable look	3	1	3	3	9	2	6
Total score			**39**		**49**		**41**
Percentage of maximum score			**76,5%**		**96,1%**		**80,4%**

The results from the Decision matrix indicated that Concept 2 was better at satisfying these criteria than the other two. Concept 2 not only got the best results among the three, it received full scores for all criteria except for the one which was weighted as least important. It was therefore decided that Concept 2 should proceed to the next phase of the product development process.

5.5 CONCEPT REFINEMENT

The chosen concept had to be refined and fully developed; especially the solution for opening new viewports was formulated in a way that gave room for development. It had previously, in the concept phase, been decided that this should be done with a simple touch gesture on the screen background. What was left to determine was where the viewport should open in relation to the touch point, what measurements it should have and how it should be avoided that a new viewport opens too close to an existing one.

5.5.1 OPENING AND MANIPULATING VIEWPORTS

As previously mentioned a new viewport is opened by touching the background on the screen. When the viewport opens it will assume the width X_0 and the height Y_0 and have its centre at the first registered touch point; i.e. if the user touches the screen with five fingers, still only one viewport will open. Centring the viewport at the first touch point assures that it is situated within a reachable distance for the user. The measurements X_0 and Y_0 will be decided by the application designer, since different measurements are suitable for different applications. The user can then resize the viewport and the measurements will then be denoted by X and Y.

A large rectangle (a in Figure 26) behind the viewport blocks the background and prevents new viewports to open too close to existing ones. It assumes the size of the viewport it belongs to (X and Y) plus 1.2 times the starting size X_0 and Y_0 to ensure that new viewports will not touch the existing ones but still be able to open relatively near. When a user touches this rectangle a text box will appear on that spot with a suitable message of the application designers' choice which tells the user that a new viewport cannot be opened there (see Figure 26).

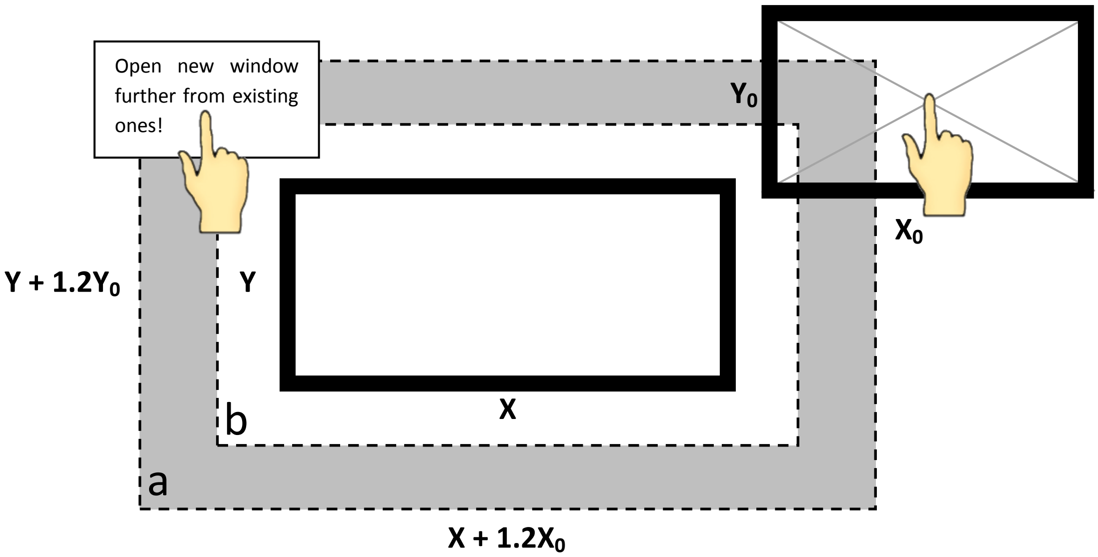


Figure 26 – Illustration of the underlying functionality of the viewport.

The smaller rectangle (b in Figure 26) primarily aims to give room for user errors, i.e. if users accidentally touch outside the viewport when actually aiming for the viewport itself. Nothing will happen if the user touches this rectangle, it is mainly there to shield the larger rectangle behind it. The size of this rectangle in relation to the viewport depends on the environment in which the application will be used, since different margins may be suitable for different target groups. It should however be of the form $X + \alpha X_0$, where the designer decides α so that the margin of error does not change with the width and height of the viewports (X and Y), but stays constant.

Both of the rectangles could either be completely invisible or have some kind of design that indicates that this particular area belongs to the already open viewport, for example a ripple or partial transparency. However, having this as a standard largely decreases the customization possibilities.

5.5.2 VIEWPORT ARRANGEMENT AND INTERACTION

As a result of not allowing the viewports to be placed on top of each other, there will be occasions when two or more viewports collide with each other, for example when a user tries to enlarge or move a viewport into an area that is occupied by another viewport. A collision of this sort has to be visualized in some way so that the user understands why the viewport suddenly stops to follow the user's commands. If the user tries to scale or move the viewport in a direction that is blocked, for example by another viewport, the part of the viewport that is blocked should flash a couple of times, alerting the user that the viewport has collided with something. For example, if a user tries to move a viewport to the right and it is blocked by someone else's viewport, the right side of the viewport that was moving can start to flash in red, or any colour and/or transparency decided by the designers. It is important that the stationary viewport should not react, since any changes to it should be a consequence of its own user's actions only.

It should also be possible to set an active region where the users can move around and start up viewports. That way, a company name, logotype and contact info can be kept outside the active region and therefore not risk being covered. The inactive region may also contain hints or tips on how the application works.

Figure 27 – Example of how a Main Screen Application including company name, logo etc. may look in final embodiment.

6 VIEWPORT

The viewport refers to the work area that belongs to and can be manipulated by one user. It includes the application, some kind of menu and means for changing its size and location. The latter manipulation, i.e. resizing and moving the viewport, was the focus of this development work. The menu was designed separately which is documented in chapter 7.

6.1 IDEA GENERATION

With inspiration from the literature review many solutions for manipulating the viewport were generated. Ideas for the design of the frame were also considered, as well as cues for the interaction. These ideas were generated through all levels of the complete idea generation process pertaining to the viewport, marked in Figure 28.

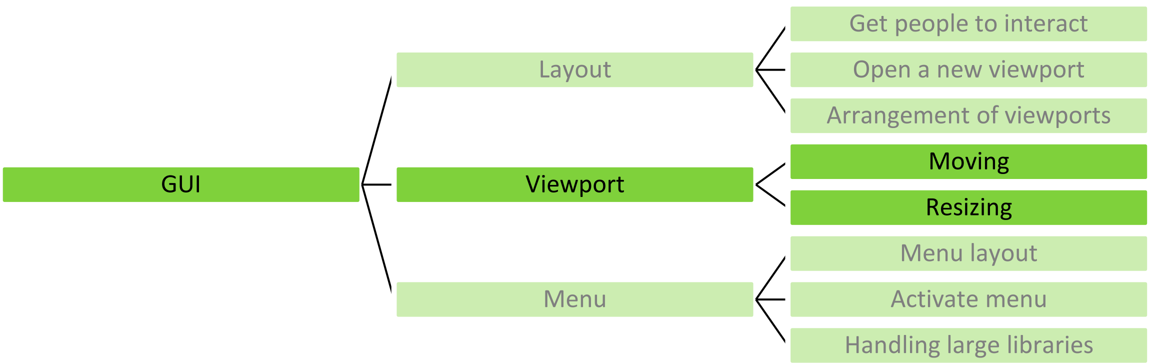


Figure 28 - The brainstorming sessions for the viewport.

6.1.1 MOVING AND RESIZING

In this report scaling refers to proportionally resizing both the width and the height of the viewport, whereas stretching will affect only one of the two.

Using edges and corners

For some applications it should be possible to both scale and stretch the viewport. One way to do this is to use the edges for stretching the viewport in one direction at a time, while the corners are used to stretch the two adjacent sides simultaneously, in other words scale the whole viewport (Figure 29a). What happens to the application inside the viewport for different actions should be possible for the designer to decide since different functions are suitable for different applications.

If the application does not support zooming, the restrictions on the frame can be set so that it can only be scaled with a fixed aspect ratio and with the media locked to the frame at all times (Figure 29b). Both the corners and the closest halves of the adjacent edges will then be used to scale the viewport.

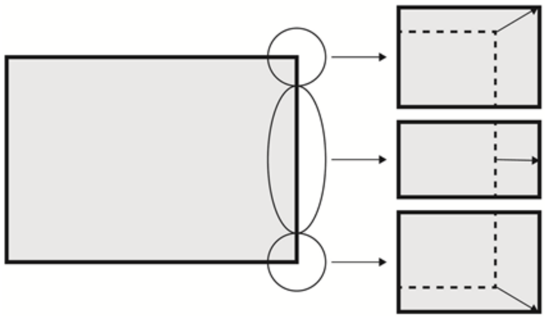


Figure 29a - Scaling without retaining the aspect ratio

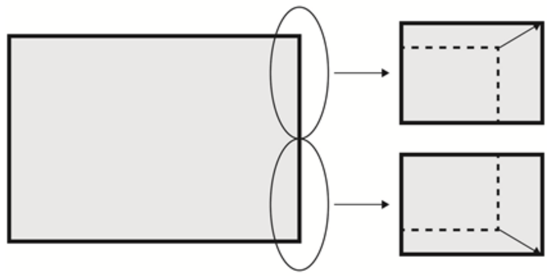


Figure 25b – Scaling with maintained aspect ratio

Pinch and unpinch for scaling

In many modern touch screen applications, pinch and unpinch gestures are used to dynamically zoom in and out. This technique applied on a viewport can work in three different ways; however all of them would have the following in common:

1. No matter if two fingers from the same hand or one finger from each hand is used, the scaling should occur around the centre point that forms between the fingers (Figure 30). The centre point is updated continuously as the fingers are moving.

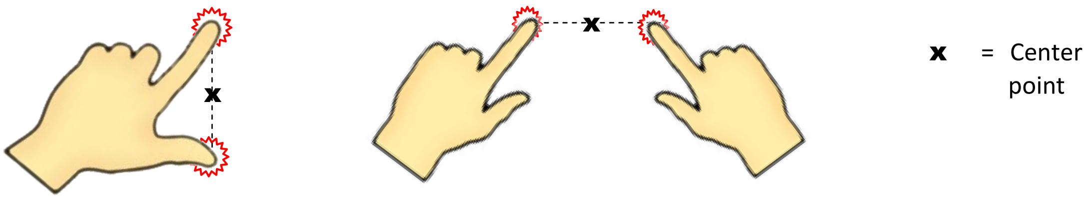


Figure 30 - When two or more fingers are used a centre point is formed between them. Any scaling or stretching should occur around this point.

2. When scaling the viewport its size changes at the same rate as the distance between the fingers (Figure 31). For example, if the distance between the fingers is doubled, the size of the viewport will double ($X_1/X_0 = 2$).

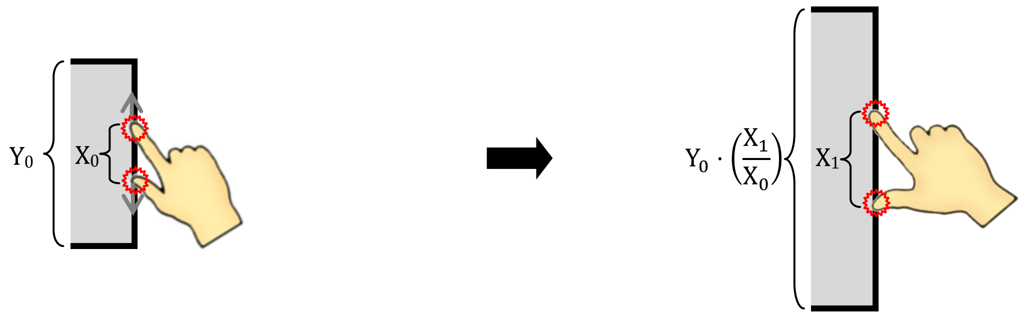


Figure 31 - The relationship between the distance between the fingers and the viewport measurement is of such sort that if the distance between the fingers changes by a certain percentage, the length of the edge in question changes with the same percentage.

3. Combining the two rules above will assure that the fingers remain on the same position relative to the frame during the resizing (Figure 32).

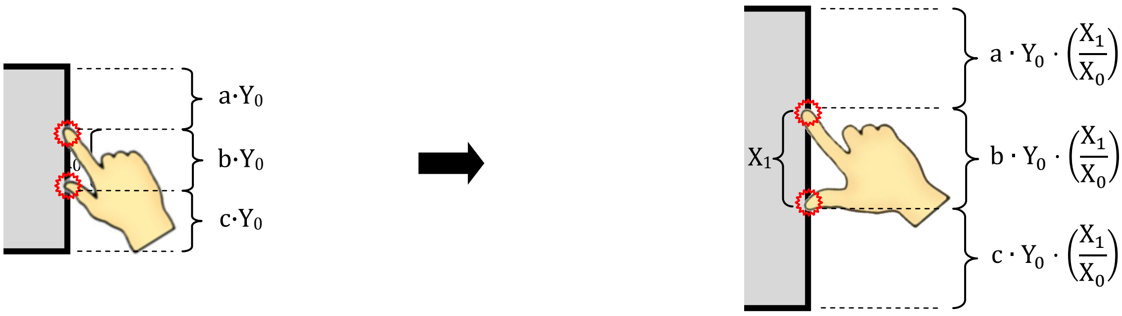


Figure 32 - The fingers will remain on the same relative position on the frame during the resizing.

Three different variants of pinch and unpinch solutions are presented below.

Basic – Scaling with locked aspect ratio can be done by pinch/unpinch gestures on the frame. This can either be done using two fingers on the same hand, e.g. thumb and index finger, or one finger from each hand (Figure 33). The former would be suited for scaling on one edge and the latter for scaling with the fingers on two opposite edges. Since this solution requires two fingers, additional touch points which are registered after the first two will be ignored. This may cause confusion if a user who intends to scale the viewport places multiple fingers of each hand on opposite edges of the frame, and the first two registered touch points are two fingers from the same hand, i.e. on the same edge. This gesture will still result in a scaling of the viewport, but at a different rate than the user probably expects.

In the basic solution all multi touch gestures on the frame will scale the viewport, but it also needs to be possible to move it. This may be done in two different ways – the first being a single touch anywhere on the frame, which will eliminate the possibility to stretch the viewport using gestures on the frame. The second solution is to move the viewport by a single touch on a certain area, e.g. a header or a button, which maintains the option of using single touch on the edges to stretch the viewport.

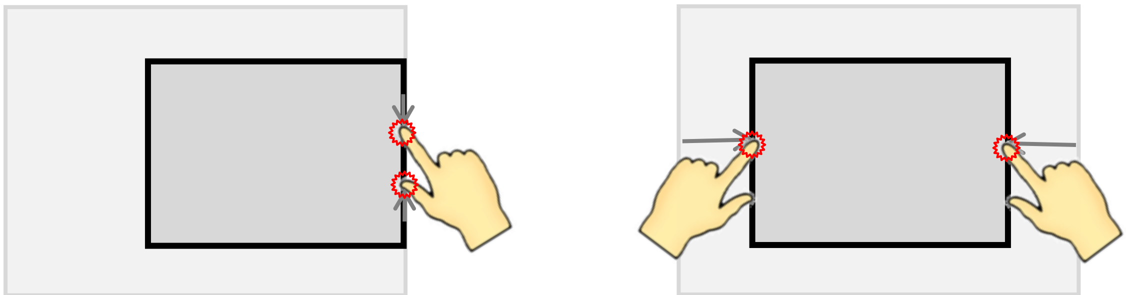


Figure 33 - Scaling with pinch/unpinch with two fingers of the same hand on one edge (left) and one finger of each hand on opposite edges (right).

Based on number of touch points – Most of the solutions presented in this report require the viewport to have some kind of frame that can be manipulated. The main reason to have a frame is to distinguish gestures which are meant for the viewport itself from others meant for the application inside it. One way to eliminate the need of a frame is to use a cluster of fingers to manipulate the viewport and single fingers to zoom and navigate in the media inside the frame. For example, resizing the viewport can be done by using the pinch/unpinch technique with two or more fingers per hand while the same gesture used with only one finger per hand would allow the user to zoom in and out in the application (a and b in Figure 34). Likewise, the user could move the viewport by sweeping with several fingers of one hand and in the same manner move the application content with one finger (c and d in Figure 34).

This solution would however require a rather complex algorithm to cluster the fingers of one hand together. One dilemma is to decide whether two closely positioned fingers should move the viewport or if it should trigger the zoom function in the application.

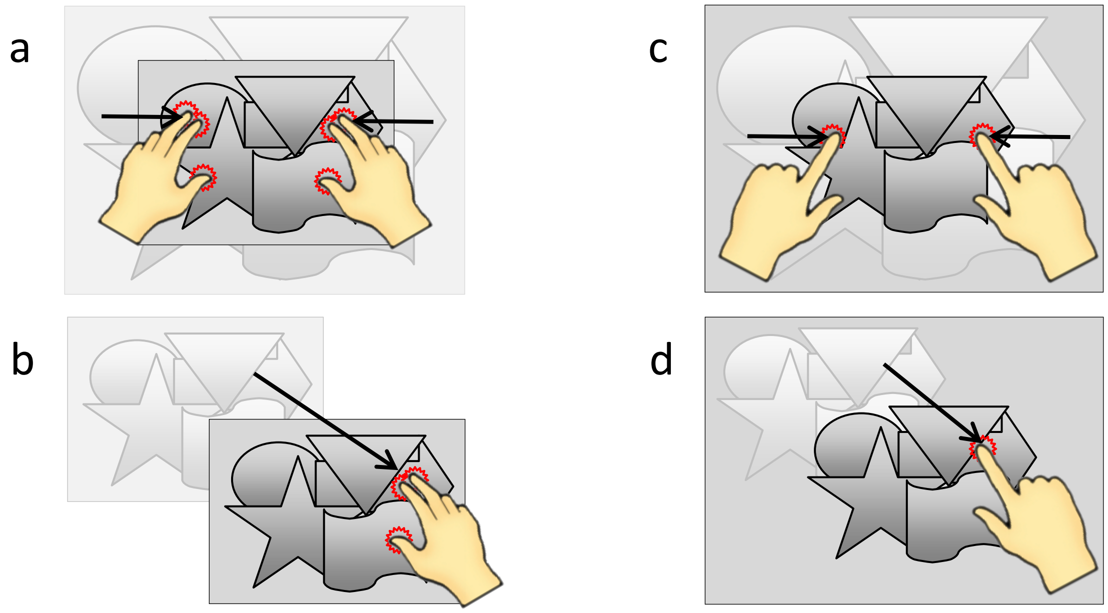


Figure 34 – A cluster of fingers is used to resize (a) and move (b) the viewport while single fingers are used to zoom (c) and navigate (d) in the application inside the frame.

Direction based – One way to modify the above solution so that it also supports stretching by multi touch is to differentiate between different angles between the fingers and a horizontal line. The possible gestures can be thought of as different parts of a circle divided into eight equally sized circular sectors (Figure 35). For example, if the fingers are moving in opposite directions on a vertical or horizontal line (± 22.5°) the viewport will be stretched, and vice versa for pinching (a and b in Figure 35). Scaling will occur when the fingers are placed diagonally (c and d in Figure 35).

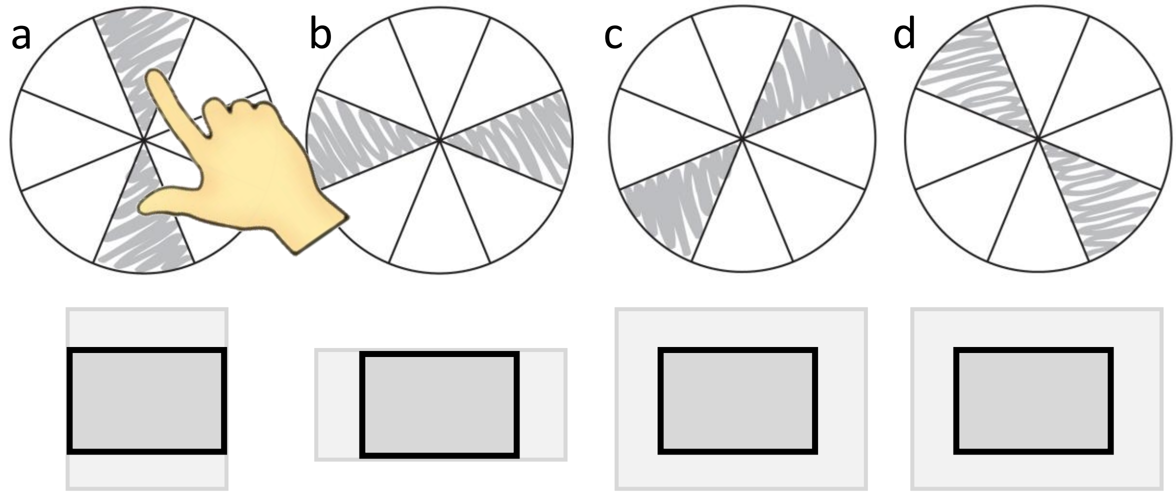


Figure 35 – Moving the fingers on a vertical (a) or horizontal (b) line will stretch the viewport, while moving them diagonally (c and d) will scale the viewport.

Buttons

An easier way to resize the viewport is by using buttons, which will shrink or enlarge it a prescribed amount with just one click. The buttons need to be positioned so that it is clear that they refer to the viewport and not to the application inside the frame, e.g. on the frame itself (Figure 36). Having prescribed size increments limits the solution in two ways; only certain sizes are allowed and the speed with which the user is able to resize the viewport is limited. More increments would make the viewport more conformable, but it would require the user to click more times to accomplish the desired size and vice versa. It is also important that the buttons position on the screen does not change with each click, which will restrict the directions in which it can stretch This solution requires that the resizing is centred around the buttons (to the right in Figure 36).

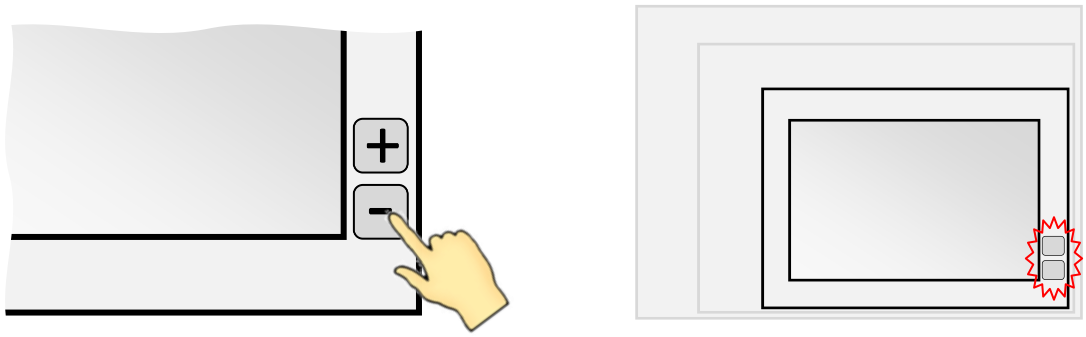


Figure 36 - Using buttons to shrink and enlarge the viewport, e.g. with the symbols + and – (left). The scaling needs to be centred around the buttons (right).

Slider

Another variant would be to replace the buttons in the above solution with a slider. The restrictions in which direction the viewport can expand would be the same as if using buttons. However, with a slider the size of the viewport could change continuously without requiring pre-defined step sizes and multiple clicks for greater changes in size.

6.1.2 ADDITIONAL FEATURES

Arrows

Arrows can be used to indicate a scaling and/or stretching function. However, arrows for this purpose need to be designed very carefully or they might be confused with other functions, such as scrolling. The user may think that a new image, movie clip or likewise will appear by clicking on the arrow, or that the arrows can be used to navigate in the application. Arrows such as ←, → and ↔ all have a base of the same width, meaning that they are symmetrical and its centred position does not change when the symbol changes. The arrows ← and → can be thought of being the arrow ↔ with parts of it hidden. Arrows such as <, > and <> do not possess this quality, meaning that the position of the arrow either has to change as the symbol change, or the arrows < and > will not be centred (Figure 37).

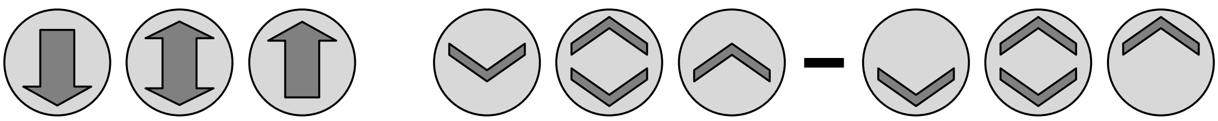


Figure 37 - Illustration of two different arrow designs.

Arrows can be used as cues to whether it is possible to stretch or scale the viewport further in a certain direction (Figure 38). They also show the user that the functions of scaling and stretching are actually available. They do however suggest to the user that it is only possible to resize the viewport by dragging on and perhaps around the arrows, which may not be the case.

Figure 38 - Arrows used to show the users in which directions scaling and stretching is possible.

Different aspect ratio for the viewport and the application

As previously mentioned, some of the applications that will run inside the viewports will require a locked aspect ratio. However, this does not necessarily mean that the viewport needs to maintain the same aspect ratio at all time. For example, an application with the aspect ratio 1:1, i.e. square, can stay the same when extending the height or the width of the viewport. Instead of the application covering the whole viewport there will then be an unused space between the edges of the application and the edges of the viewport (Figure 39). This is similar to when a movie is played on a device with different aspect ratio than the recording device. This solution would mean that the viewport would function the same no matter if the application has a locked aspect ratio or not. However, since having different ratios on the application and the viewport will result in unused space this solution will not be very space efficient.

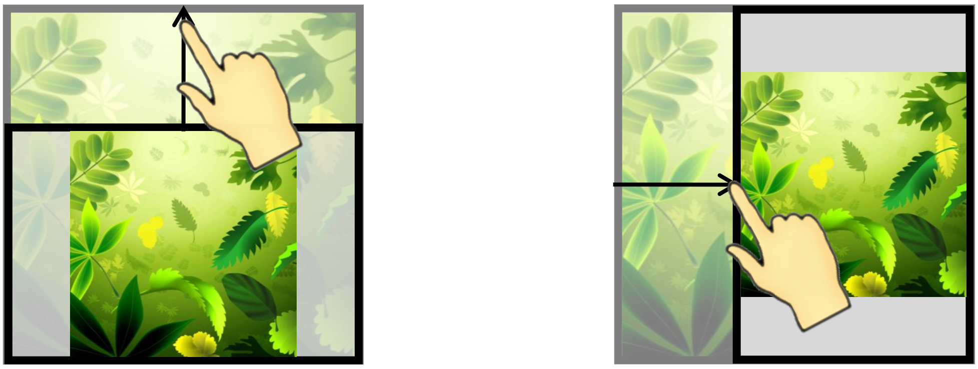


Figure 39 - A varying aspect ratio on the viewport combined with a locked aspect ratio on the application will result in unused space between the edges of the application and the edges of the viewport.

Frame design

Many of the ideas for manipulating the viewport require a frame, meaning that the design of the frame also should be taken into consideration. However, as mentioned in the limitations of this thesis the aspects such as border width, corner radius, frame colour and transparency will be up to the designer for each specific company to decide.

6.2 SCREENING

Only two of the above presented ideas remained after the screening process. As can be seen in the bullet lists below, only the ideas for moving and resizing were evaluated. The other ideas were additional features which can be used when found suitable by the application designers, regardless of which viewport concept is chosen.

- ✓ Using edges and corners
- ✓ Basic pinch and unpinch

The following ideas were discarded:

- ✗ Based on number of touch points
- ✗ Direction based
- ✗ Buttons
- ✗ Slider

The solution where scaling was *based on the number of touch points* seemed promising, though after much consideration it was decided that even though this solution was both innovative and had a high-tech feeling, it was not suitable for the user profile. The user profile clearly states that the product should be designed for first time users and a function like this one may be difficult to discover just by experimenting with the application.

Since it was difficult to visualize the usability of the *direction based* solution, a working mock-up was created in Unity3D and tested on a tablet computer by the employees at i4sense. Regardless of their previous experience with similar software, it took a while before they were using the software without being surprised when the direction of the stretching and scaling changed. Even then, they found the change to be very sudden and difficult to foresee.

Using *buttons* or a *slider* to resize the viewport was considered to be unlikely to impress and intrigue the users. Integrating them into the design would also restrict the adaptability, a property which in the pre-study was found to be one of the most important for this product.

6.3 COMBINING INTO CONCEPTS

When combining the various ideas into complete concepts using the morphological matrix, the ideas were identified as being too different from each other preventing any grouping of the ideas as had been done in previous morphological matrices. Based on this, another layout (Table 5) was used where the different parts of the viewport are listed in the rightmost column and the actions were listed as column headings. Three different concepts were created and each corresponding solution and action was marked with the concept number to which it pertains, e.g. Concept 1-3. A more in-depth description of each concept is found below.

Table 5- Morphological Matrix showing how the three viewport concepts were generated.

Action / Part	Move	Scale	Stretch
Corner		Concept 1, Concept 2	Concept 1, Concept 2
Edge	Concept 1	Concept 2	Concept 2
Icon			
Header	Concept 2		
Frame (MT)		Concept 3	
Frame (ST)	Concept 3		
Not an option			Concept 3

Concept 1 – Moving by edges and resizing by corners

Corners are used to scale or stretch the frame while the borders are used to move the frame around (Figure 40). This solution does not differentiate between multi touch and single touch since it only registers which part of the frame that is activated, which makes it very stable. This solution works best when allowing a varying aspect ratio, since a locked aspect ratio on the viewport would require the user to drag the corner in a specific direction or his or her fingers would lose contact with the corner.

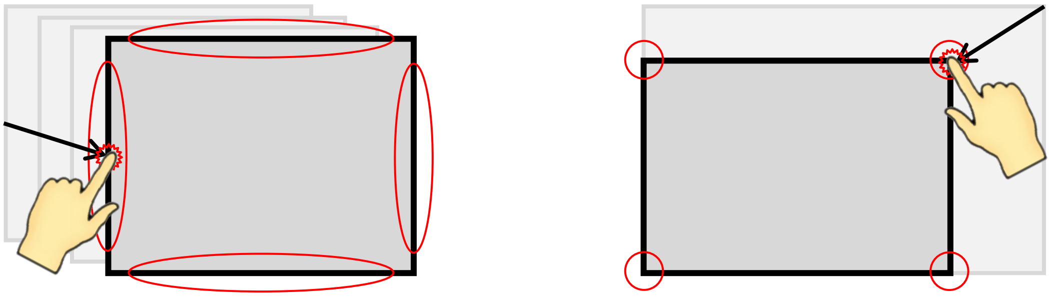


Figure 40 - Illustration of how the viewport in concept 1 is moved (left) and scaled or stretched (right).

Concept 2 – Moving by header and resizing by corners and edges

The frame consists of borders and a header. The header is used to move the viewport while the borders and corners are used to stretch and/or scale it, depending on whether stretching is allowed or not (Figure 41). If the aspect ratio is locked, dragging one border will also cause the adjacent borders to change in order to adapt to scale. This concept is similar to windows on the desktop, which are used by many popular operating systems for personal computers. This will allow many users to recognize the functionality. However, this solution will pose a risk of the user not being able to reach the header once they've opened the viewport. The viewport will be opened with the centre in the initial touch point and this may cause the header to be too high up.

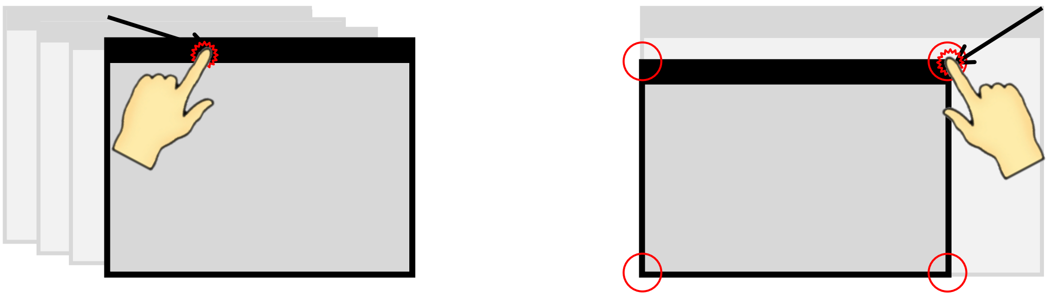


Figure 41 - Illustration of how the viewport in concept 2 is moved (left) and scaled or stretched (right)

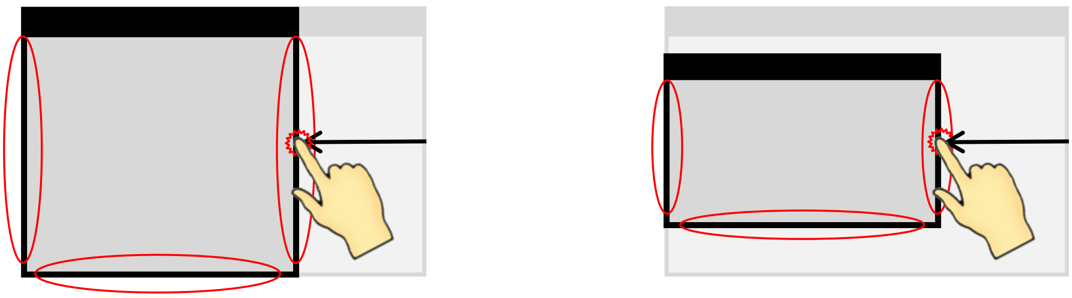


Figure 42 - The right, left and bottom borders can also be used to stretch (left) or scale (right) the viewport.

Concept 3 – Moving by single touch and resizing by multi touch

This concept utilizes multi touch versus single touch to differentiate between scaling and moving the frame. In other words, the use of one finger anywhere on the frame will cause it to move, while two or more fingers moving towards or away from each other will shrink or enlarge the frame respectively. If more than two fingers are placed on the frame only the first two will be registered. This solution does not support any means for stretching the viewport. One benefit with a solution where both moving and scaling can be done by one hand anywhere on the frame is that the users never need to reach for a specific part of the frame. Consequently, it will always be possible to avoid occlusion. A disadvantage is that the users, before they learn how the viewport functions, may try to move the viewport with multiple fingers of one hand and thereby trigger the scaling function.

Figure 43 - Illustration of how the viewport in concept 3 is moved (left) and scaled (right).

6.4 CHOOSING A CONCEPT

The existing demands for the viewports, which were satisfied by all three concepts, were that they should be possible to move and resize. The criteria that the concepts were graded against are listed below and accompanied by an explanation and the weight factor within parentheses.

- *Possible to change aspect ratio* (w=3) – Whether the aspect ratio should be fixed or not should be decided by the application designer and not by a restriction in the design itself. This criterion was graded high because adaptability is highly prioritized. (4.3)
- *Work well with fixed aspect ratio* (w=2) – The gestures should not feel less intuitive when used together with a fixed aspect ratio. This criterion received a medium weight as this does not apply to all designs.
- *Possible to move higher than one can reach* (w=1) – There may be occasions when the user wants to move or enlarge the frame so that it is larger or higher up than he or she can reach. A possible scenario is for example if one person standing next to the screen wants to show or present something to other people or a crowd standing further away from the screen. This is however not the main purpose of most of the screen applications. During individual usage, the top border of the viewport should be at approximately the same height as the user’s eyes for maximum comfort. Moving the viewports further up will cause them to flex their neck backwards in an uncomfortable position. (3.3.2)
- *Possible to scale larger than one can reach* (w=1) – See explanation for previous criterion. (3.3.2)
- *Always opens at a reachable height* (w=3) – Since the viewport will open with the first touch point as its centre, there is a risk that the user cannot reach part of the viewport if they start out from a high point or the initial viewport measurements are very large. If they cannot reach the parts needed to manipulate the viewport some of the most important functions will be inaccessible. This is therefore a very important criterion.

- *Efficient use of space* (w=2) – The visible area of the application should be as large as possible compared to the total size of the viewport, i.e. if the viewport has a frame or buttons, they should not take up space at the cost of the application view. As this criterion does not have a direct impact on the functionality of the product, although it may have a negative effect on the users' overall experience, it received a medium weight.
- *Adaptable look* (w=3) – The main design should be possible to use in different applications. It should therefore be possible to make the design look unique only by small modifications. (4.3)

Table 6 - Decision Matrix for evaluating the three viewport concepts.

		Concept 1		Concept 2		Concept 3	
Criteria	**Weight**	**Grade**	**Total**	**Grade**	**Total**	**Grade**	**Total**
Possible to change aspect ratio	3	3	9	3	9	1	3
Work well with locked aspect ratio	2	1	2	2	4	3	6
Possible to move higher than one can reach	1	3	3	1	1	3	3
Possible to scale larger than one can reach	1	1	1	1	1	3	3
Always opens at a reachable height	3	3	9	1	3	3	9
Efficient use of space	2	3	6	2	4	3	6
Adaptable look	3	3	9	2	6	3	9
Total score			**39**		**28**		**39**
Percentage of maximum score			**86.7%**		**62.2%**		**86.7%**

Unfortunately, two of the concepts received the same score in the Decision Matrix, meaning that the evaluation against the table of requirements was not sufficient in deciding on one best concept to continue working with. Furthermore, even though the above criteria may distinguish which concept has more extra features they cannot be used to estimate how easy one concept is to understand and use compared to the other. Since these are very crucial factors they were considered well worth to evaluate further. It was therefore decided that the viewport concepts should not only be evaluated against theory but also in a physical user trial before any final choice of concept should be made.

6.4.1 CONCEPT TEST

A main screen application with functional mock-ups of the three concepts were created in Unity3D and tested by 18 trial subjects on a touch screen located in the lobby of Lindholmen Science Park. This main screen application included three numbered buttons, which were used to switch between the concepts. A fourth button was used to open a rectangle where the trial subject should try and place the viewport, in order to test its precision. However, the trial subjects did not use these buttons as they were controlled by the trial-leader. All concepts contained a non-functional screenshot from the i4sense website and had a red close button in the upper right corner (Figure 44). The trial layout is described in detail in section 2.6.1 and the data collected in the trial can be found in Appendix B.

Figure 44 – One of the trial subjects is exploring the functionalities of a concept at the trial. The numbered buttons in the upper right corner of the touch screen was used to switch between the concepts that were being tested.

The trial subjects were asked to grade each concept on the following four criteria on a six-grade scale.

- *It is easy to understand how to use it* – Are the functionalities easy to comprehend?
- *It is easy to use* – Are the gestures needed to manipulate the viewport easy to perform?
- *It is fun and feels good to use* – Does the usage instil a good feeling?
- *The precision is good* – Is it easy to resize the viewport to specific measurements?

The average grading on all criteria as well as the total average for each concept are compiled in Figure 45. The y-axis ranges from 4.00 to 6.00 since all grading results fell within this range. Both concept 1 and concept 2 consistently had numeric results above 5.0. Concept 3 got slightly lower results than the other two, especially on the criteria “Easy to use” and “Good precision”. Concept 1 had the highest grading on three of the four criteria giving it an average grade of 5.33. Concept 2 was close behind concept 1 on three criteria and higher graded on one criterion, resulting in an average grading of 5.32. Concept 3 had the lowest grading on three of the tested criteria as well as the total average, which were 4.97.

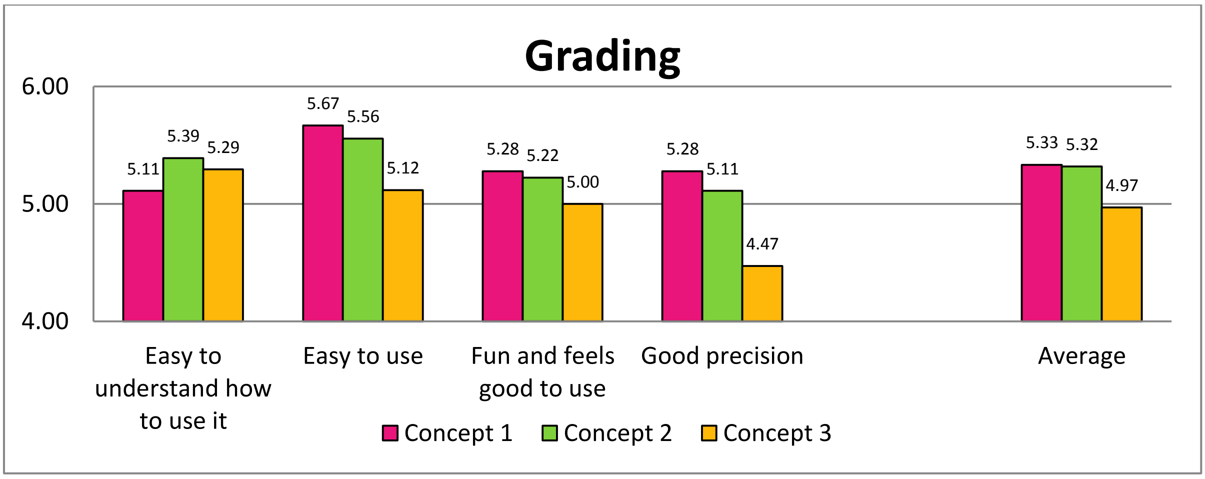


Figure 45 – Grading results from the concept trial showing that concept 1 has the highest average grading followed by concept 2.

Finally, the trial subjects were asked to rank the concepts in order of preference. As can be seen in Figure 46, 56 % of the trial subjects preferred concept 1, while 28 % ranked it as their least favoured solution, making its average ranking 1.72. Concept 2 was ranked first by 28 % and second by 72 %, also giving it an average ranking of 1.72. Concept 3 was ranked last by 72 % of the trial subjects. The average ranking of concept 3 was 2.4.

During the trial it was observed that most of the trial subjects initially tried single touch gestures when faced with a new concept. Only concept 3 required multi touch and even here almost 90 % of the users first tried single touch when experimenting with this concept.

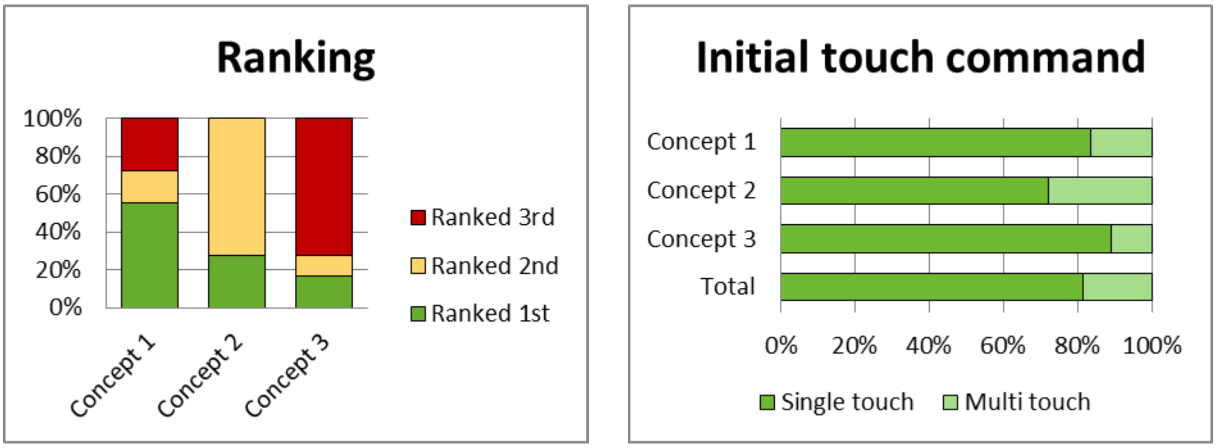


Figure 46 The user subjects ranking of the concept (left) and their first tried touch command (right).

Though uncommon, a few times some trial subject accidentally closed the viewport when trying to grab the upper right corner. Therefore the close button was moved from its original position, which was on the top border next to the right corner, a bit further to the left (Figure 47). The literature review had already revealed that controls for closing should not be placed too close to other frequently used controls, though it did not suggest an appropriate distance. However, the distance used in the prototype was apparently too small.

Figure 47 - Close button initial position (left) and final position (right)

Another observation was that some trial subjects attempted to double-click (double-touch) on the frame and in these scenarios did not get a response since there was no available function for this command. However, when experiencing this they were not surprised or seemed disturbed by this but continued on using the screen as intended.

6.4.2 FINAL CHOICE OF CONCEPT

Unfortunately, neither the decision matrix nor the concept test gave a conclusive result. In the user trial a larger number of the trial subjects ranked concept 1 as their first choice than those who chose concept 2. However, it is important to note that even though a larger amount of people ranked concept 1 as their preferred concept, not one person ranked concept 2 as their least preferred concept. Their numeric average was exactly the same (1.72). Concept 3 on the other hand was ranked as the least preferred solution by 72% of the users and only as the most preferred solution by 16% of the trial subjects. Based on this, concept 3 was discarded for continued development.

As both concept 1 and concept 2 have very similar results during the trial it was important to study the decision matrix once again. One of the main factors as to why concept 2 scored much lower than concept 1 was the fact that the header might not be within a reachable height for the user. This was an important factor, which could not be tested during trial as the viewport itself was opened by the trial instructor and not by the users themselves. Should this feature have been tested, it is possible that concept 2 would have received both lower grades as well as rankings. In addition, the highest part of the screen used for the test trial was only 210 cm from the floor, meaning that even if the reachability feature had been part of the trial the maximum height to which the header could be positioned would still have been within a reasonable reachability.

In an attempt to differentiate the two concepts further, an analysis of probable programming issues was considered. When analysing the solutions of concept 1 it was found that it required a smaller number of algorithms to function. It needs one algorithm for each corner and one for the rest of the frame, in total five different algorithms; comparing this to concept 2 which required one algorithm for each corner, one for the header and one for each of the remaining borders, i.e. eight different ones. Concept 1 would due to these facts be easier to develop as well as have fewer potential risks for error.

Due to the reasoning above, concept 1 was chosen as the better concept to put forward through the development phases.

6.5 CONCEPT REFINEMENT

The measurements for the viewports, meaning the maximum and minimum allowed size as well as the thickness of the surrounding frame and its corner radius may be set as a ratio to the screen size rather than an absolute number. That way, the main screen application will be easily transferrable between screens of different sizes. For example, the maximum allowed height could be 90% of the screen height etc.

In order to make the look of the viewport compatible with various applications, it should be possible to change the colour and the transparency of the frame as well as adding shadows (Figure 48).

The viewport can be set to fade away if the viewport has been idle for some time. The exact amount of time depends on the application since an application with a lot of text or a movie clip may be used even though no physical interaction is taking place for a long time.

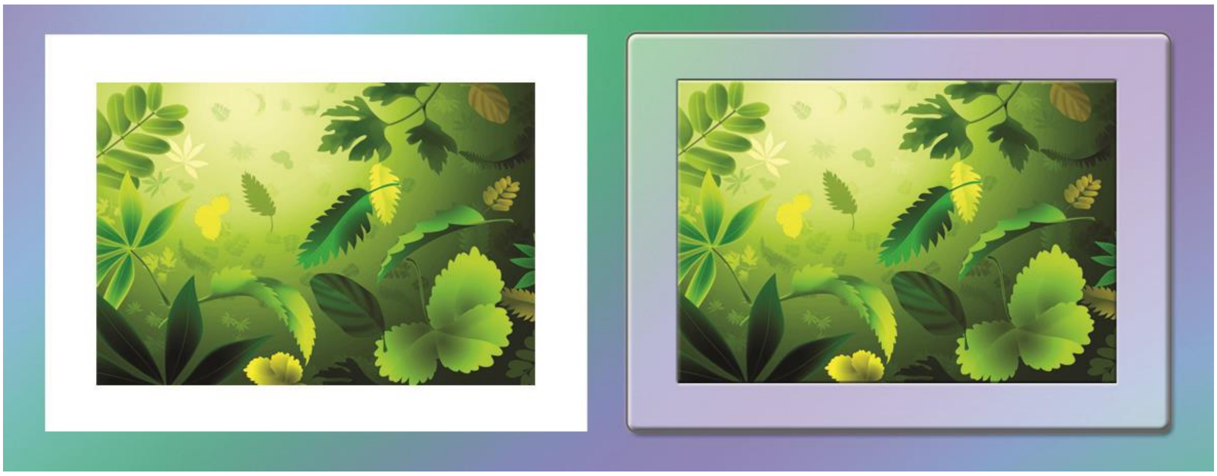


Figure 48 - A white frame without rounded corners (left) and a frame with rounded corners and 50 % transparency (right).

7 MENUS

Every viewport needs to have a menu that can be used to switch between applications.

7.1 IDEA GENERATION

Unlike the previous brainstorming sessions regarding the layout and the viewport, the work concerning the menu was mostly about creating own adaptions of already existing solutions. As mentioned in the literature study all kinds of electronic devices with screens, e.g. TVs, computers, cell phones, mp3-players have menu systems in various forms, which have been studied. One major difference between personal computer menus and menus on touch screens is that the possibility of hovering over a certain text or button does not exist in the latter. Figure 49 shows the brainstorming areas related to the menu corner stone.

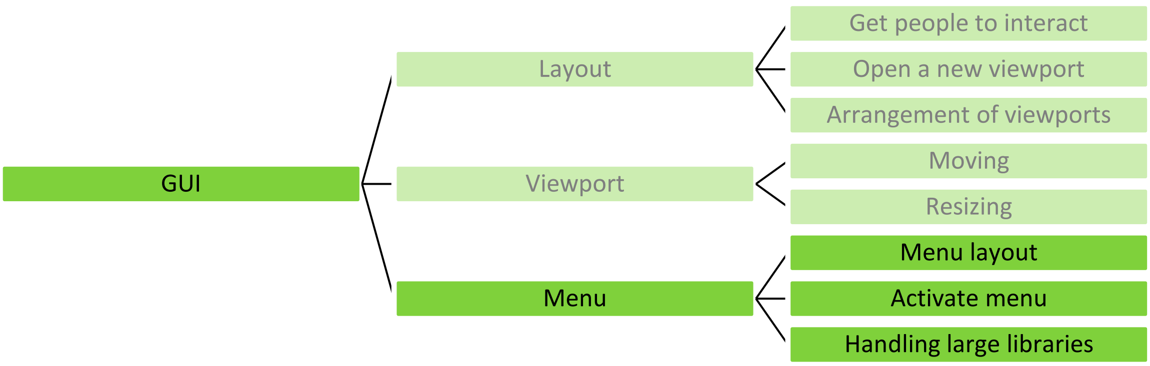


Figure 49 - The brainstorming sessions for the menu.

7.1.1 MENU LAYOUT

Toolbar – The toolbar menu can be visible at all times and placed on top of the application or between the application and the viewport border. The toolbar works well with both text and icons (Figure 50).

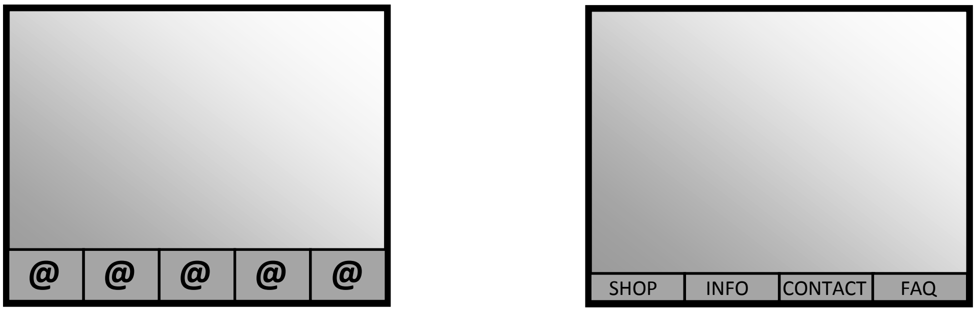


Figure 50 - Toolbar menu layout.

Folder – The folder menu cannot be visible at all times without covering the application, meaning that it would have to be activated in order to be used, for example by a button (Figure 51). Icons are better suited than text in this type of layout though text can be added as a complement to the icons, as can be seen on personal computers. One benefit with the folder menu is that it works very well with subfolders.

Dropdown – This type of menu is typically used with text. Having a button that opens the menu prevents it from occupying too much space when it is not used (Figure 51). Dropdown menus also have the ability to handle large libraries with subcategories.

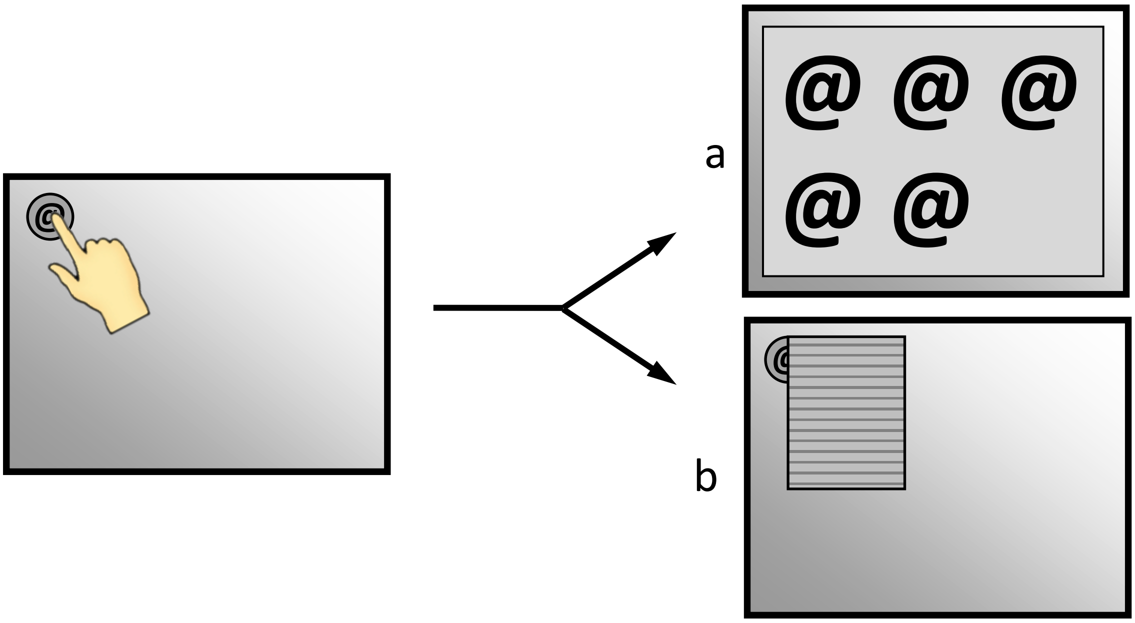


Figure 51 – Folder (a) and dropdown (b) menu layout.

7.1.2 ACTIVATE MENU

Visible at all time – Tabs or toolbars are typically placed alongside an edge, enabling them to stay visible at all time without covering the application.

Partly hidden – One way to indicate that a menu exists without having it fully visible is to use a small tab or a button to activate the menu.

Hidden – A menu can also be completely hidden and called on by a specific hand gesture, for example by using one or two fingers to draw a circle or double click on the screen.

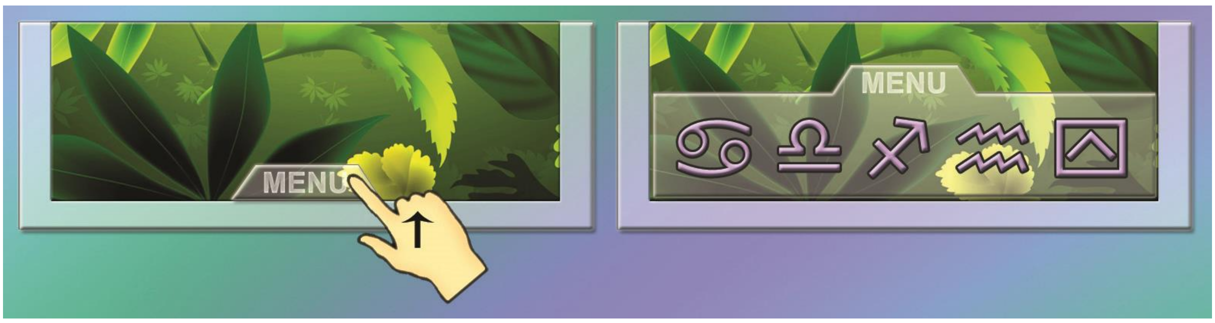


Figure 52 - Partly hidden menu to be dragged upward in order to view the various menu options.

7.1.3 HANDLING LARGE LIBRARIES

As mentioned above, folder and dropdown menus both work well with large libraries and subcategories. In the folder solution, the number of subfolders is not limited by the size of the viewport. When opening a folder, the content of said folder comes into view as it replaces the view of the previous level. A dropdown menu can also handle relatively large libraries, though the number of hierarchy levels is restricted mainly by the width of the viewport (Figure 53). Toolbars in personal computers typically only use one hierarchy level since all options are placed on a row.

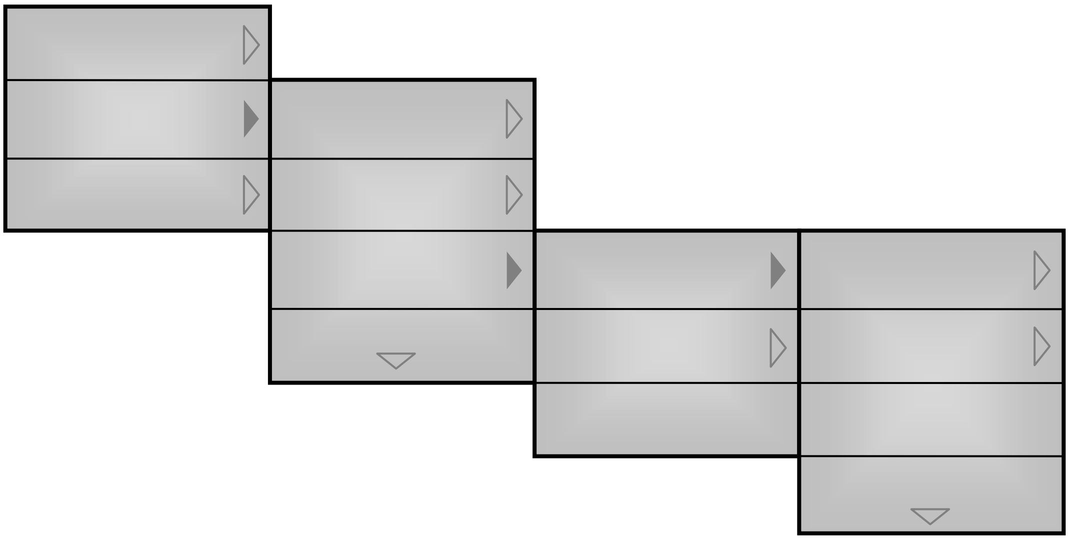


Figure 53 - Demonstration of a dropdown menu with four hierarchy levels.

7.2 SCREENING

The ideas generated for the menu were all solutions which had been tried and tested by others and were known to work well with various products. It was therefore unnecessary to discuss their functionality and general usability during the screening process. Instead, a list of requirements was set up in order to decide which menu layout was best suited for this specific product. Different variants of the layout chosen could then be designed and evaluated further in the next phase.

The criteria that the three different layouts, i.e toolbars, folders and dropdown menus, were evaluated against are listed below.

- *Not occupy space from the application when not used* – The menu should not occupy too much space and preferably cover as little as possible of the application when it is not used.
- *Easy to detect the menu* – If the menu is always visible the users get a good overview of the alternatives.
- *Work well with both text and icons* – The design should look good independently of whether text or icons are used to symbolize the applications.
- *Few steps necessary to reach the desired application* – The user should not be required to perform an excessive amount of touch events to access the application.
- *Easy to access the menu* – Complicated gestures or too small buttons will decrease the accessibility.
- *Handling large libraries* – The menu should be able to handle as large libraries as possible as well as subcategories.

Some of the above criteria are dependent, meaning that they have a great influence on each other. For example, a menu that is easy to detect will have to be visible and will therefore inevitably occupy some space. It was therefore extra important to carefully consider the weight coefficients used in the decision matrix below (Table 7).

Table 7 - Decision Matrix for evaluating the menu layout.

		Folder		Toolbar		Drop down	
Criteria	**Weight**	**Grade**	**Total**	**Grade**	**Total**	**Grade**	**Total**
Not steal space from the application when not used	2	3	6	1	2	3	6
Easy to detect the menu	3	1	3	3	9	1	3
Work well with both text and icons	3	2	6	3	9	1	3
Few steps necessary to reach desired application	2	2	4	3	6	2	4
Easy to access the menu	2	2	4	3	6	2	4
Handling large libraries	2	3	6	1	2	2	4
Total score			**29**		**34**		**24**
Percentage of maximum score			**69,0%**		**81,0%**		**57,1%**

The screening process eliminated both a folder solution and the use of drop-down menus. One of the criteria was that the menu should be easy to detect and in contrast to this the purpose of a drop-down menu is that the content is hidden until being called for. A folder menu cannot be visible at all times since it would steel a lot of space from the application if it was. The remaining type of menu was the toolbar solution since these could be used efficiently with text, symbols or icons as well as being visible at all times without taking up too much space.

7.3 COMBINING IDEAS INTO CONCEPTS

As mentioned in chapter 3.1.3 toolbars may come in different designs, one resembling buttons on a row and one being tabs resembling paper archive dividers. These can in turn be arranged vertically or horizontally and along different borders of the viewport. Another alternative would be to place the toolbar between the border of the viewport and the application inside it.

These variants were inserted in a morphological matrix and combined into three different concepts (Table 8).

Table 8 - Morphological Matrix showing how the three menu concepts were generated.

Type of toolbar	Tabs	On a row
Arrangement of icons	Vertically arranged	Horizontally arranged
Location	On top of application	Between border and application

Concept 1 Concept 2 Concept 3

Concept 1 – Floating on top

The first concept has the symbols vertically arranged and floating on top of the application (Figure 54). The symbols themselves are the only visible part of the menu system since there are no tabs or boxes surrounding them, they are simply spread out evenly along the left or right side of the viewport.

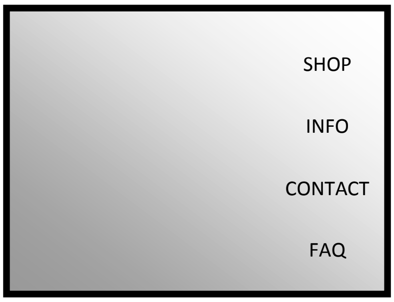
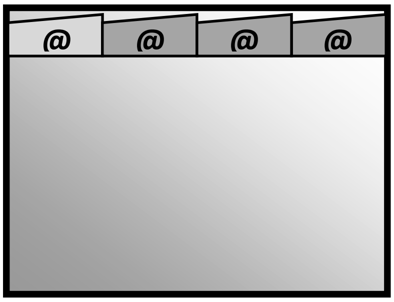


Figure 54 – The menu in concept 1 consist of symbols vertically arrange along one side and floating on top of the application.

Concept 2 - Tabs

This concept consists of horizontally arranged tabs situated between the application and the frame border (Figure 55). Symbols are located inside these tabs, which may work better with text than other symbols because of their rectangular form. Widening them to make space for e.g icons, will evidently take up more space at the cost of the application. The active tab can be highlighted to distinguish it from the inactive ones.



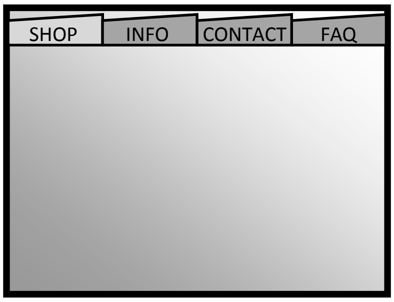
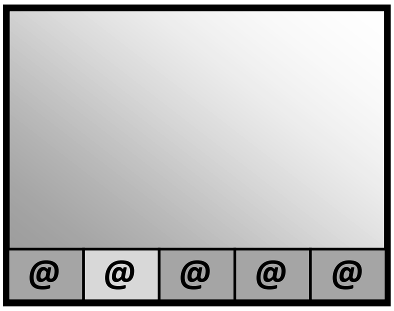


Figure 55 – In concept 2 the menu is made up by a row of tabs.

Concept 3 - Toolbar

In concept 3 the symbols are horizontally arranged in a row on a toolbar between the application and the frame border (Figure 56). As in concept 2, changing the appearance of the background on which the symbol is situated can highlight the active selection.



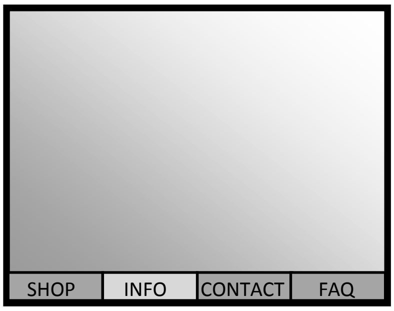


Figure 56 - Concept 3 has a toolbar situated along the bottom border.

7.4 CHOOSING A CONCEPT

Three concepts were evaluated based on the criteria below, many of which are the same as in the screening phase when the type of menus were evaluated.

- *Not steal space from the application* (w=2) – The menus should not occupy too much space inside the viewport borders as this space will be shared with the application.
- *Not cover the application* (w=3) – The menu shall preferably cover as little as possible if nothing of the application.
- *Work well with both texts and icons* (w=3) – The design should look good independently of whether text or icons are used to symbolize the applications.
- *Avoid occlusion* (w=1) – When used, the user should not cover essential parts of the menu or application from view.
- *Adaptable look* (w=3) – It should be easy to modify the appearance of the menu to fit different customers.

Table 9- Decision Matrix for evaluating the menu concepts.

		Concept 1		Concept 2		Concept 3	
Criteria	**Weight**	**Grade**	**Total**	**Grade**	**Total**	**Grade**	**Total**
Not steal space from the application	2	3	6	2	4	2	4
Not cover the application	3	1	3	3	9	3	9
Work well with both text and icons	3	2	6	2	6	3	9
Avoid occlusion	1	2	2	2	2	3	3
Adaptable look	3	3	9	2	6	3	9
Total score			**26**		**27**		**34**
Percentage of maximum score			**72,2%**		**75%**		**94,4%**

Concept 3 was chosen to move on to the next phase of the product development work as it received the highest score by a significant margin (Table 9).

7.5 CONCEPT REFINEMENT

The content of the menu will be decided by the program designers in consultation with their customers. Companies often have their own graphic profile containing specific colours schemes, fonts, logos etc. For this reason no such recommendations will be presented as part of the menu design solution.

As the viewports width can be varied, and the menu is placed along the lower border of the viewport, the menu will evidently also vary in width. In cases when there are too many menu options to fit to the width of the viewport, there must be some means for scrolling along the row of options. A sweeping gesture or buttons on each side of the menu would accomplish this equally well. However, a sweeping solution would eliminate the need for more objects in the viewport, as buttons would take up some space as well as making the viewport look more cluttered. Furthermore, a sweeping gesture would require less touch events since no step size needs to be defined, making the process of reaching options further down the row much faster. For these reasons the sweeping gesture was incorporated into the toolbar.

Several different solutions for indicating the scrolling function were considered. Many of them, for example arrows or scrollbars, would require more symbols or objects being added to the design, again making it more cluttered and less adaptable. This was avoided by using a design where the left and right side of the menu is faded if it contains more options in that direction. If only one side is faded, it means that the end of the row has been reached (Figure 57). It should also be possible to have the menu options on a loop, but this is an additional feature which the program designers can choose to utilize when they find it suitable.

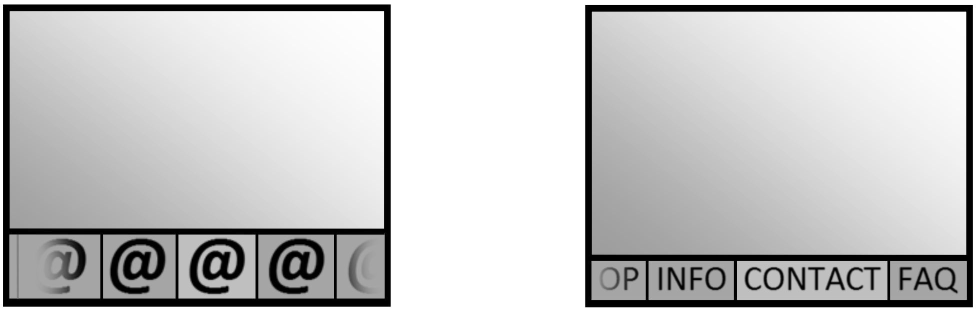


Figure 57 - The sides of the toolbar is faded to indicate that more options is available. In the rightmost figure the end of the row has been reached, hence it is not faded on that side.

A toolbar solution in its original form does not work well with large libraries including subcategories. Even though this was not the most critical criterion it would of course be preferable to have it as an option. With inspiration from the dropdown menus a similar solution for handling subcategories in the toolbar menu was designed. As a user chooses an option with subcategories a new toolbar, i.e. a row of options, appears above the first one, and so on (Figure 58). These rows will be placed on top of the application inside the viewport, so that the application measurements will not be affected when new rows are added. When the final choice is made and the menu is no longer used, the subcategory rows will fade away and reveal the full application.

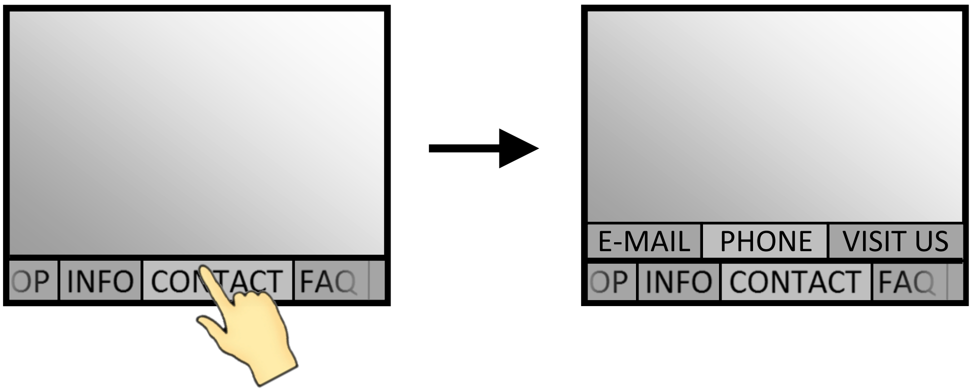


Figure 58 - The menu used with subcategories.

8 INTEGRATING THE PARTIAL CONCEPTS

Separating the development of the corner stones from each other was made possible by the clear interfaces between them. Having these clear interfaces also facilitated the process of integrating them into one final GUI. Figure 59 shows the full GUI with the final concepts for the layout (a), the viewport (b) and the menu (c). A more in-depth description of each of the corner stones can be found in section 5.5, 6.5 and 7.5 respectively.

The main screen in the final solution consists of an active and an inactive region. The latter is meant for information which needs to be visible at all time, such as company logos etc. The active region is where the user interaction occurs. Here users can open their own viewports by a single touch on an unoccupied area, though no closer than half the viewports opening width and height away from other viewports and the region border; as the viewport opens with the touch point as its centre this restriction is a necessity to prevent it from opening on an already occupied area.

The viewports can be resized by touching and dragging the corners of the frame. Dragging the corner towards the centre of the viewport will cause it to shrink while moving in the opposite direction will enlarge it. The same touch and drag gesture can be used on the edges of the frame to move it around. If a movement or enlargement is hindered by another viewport or the region border, it is interrupted. The colliding part of the viewport will flash a few times to indicate where the collision took place.

Along the bottom of each viewport a toolbar menu displays the options of applications or pages of the same application to the user. If all options do not fit the width of the viewport the user will have the possibility to scroll in the menu by a sweeping motion on the toolbar. Faded ends indicate in which directions scrolling is possible. When navigating in the menu, any sublevels will appear above the main menu and on top of the application. These will fold back when idle to prevent them from covering the application inside the viewport.

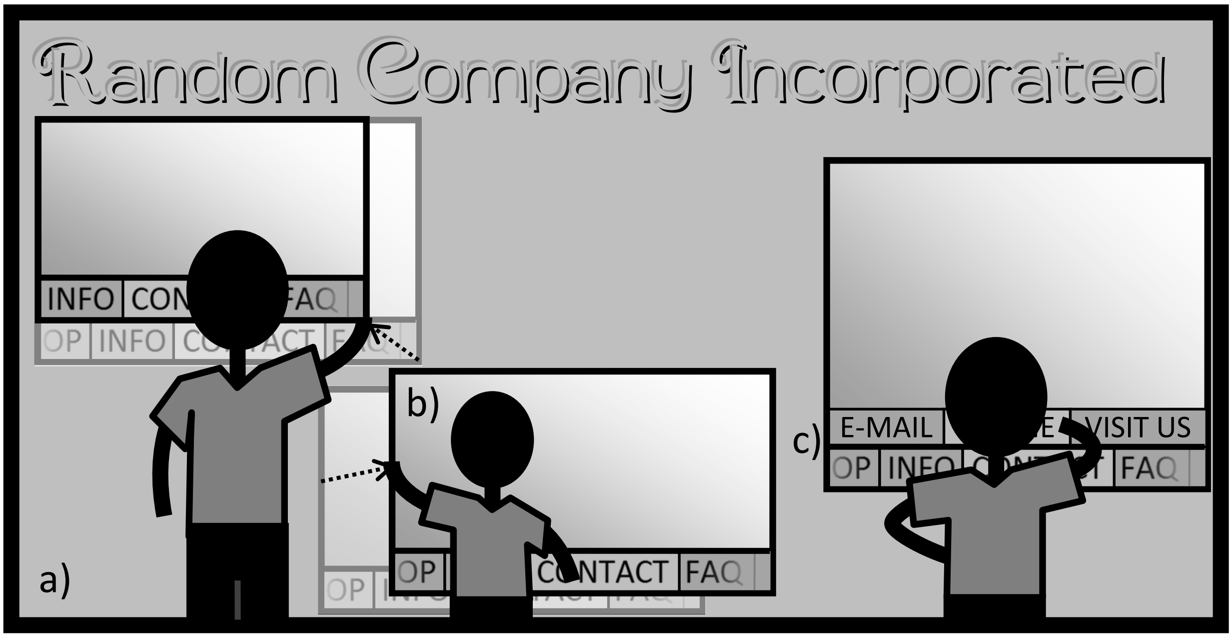


Figure 59 - The final GUI including the solutions for the layout, the viewport and the menu.

8.1 VARIABLE PARAMETERS

Throughout the development work emphasis has been placed on the possibility to adapt the GUI, for example to accommodate the graphic profile of a company or organization. As such, several of the design parameters can be modified to tailor specific needs of a variety of customers. The following list summarizes said parameters for each of the cornerstones.

- Layout (5.5)
    - Outline of the active region, i.e. where the viewports can be opened and handled
    - Starting height and width of the viewport as well as the rectangles behind it which is shielding the background close to the viewport
    - Whether or not to have a message pop up if a user tries to open a viewport to close to another one, the content of that message, and the measurements on the rectangular area where touch gestures will trigger this message to appear
    - The flashing parameters, i.e. colour, transparency and intensity, of the visual cue for when a viewport collides with another viewport or reaches the end of the active region on the screen
- Viewport (6.5)
    - Frame colour and opacity
    - Border thickness, both the visible frame and the thickness of the area which responds to touches
    - Corner radius
    - Time before the viewport fades away when not used
    - Whether or not to have a ripple
- Menu (7.5)
    - Colour and opacity of the main menu toolbar as well as the subcategory toolbars
    - Height/thickness of the menu
    - Whether to use icons, text or other symbols in the menu and the format of these
    - Whether to have the menu options in a loop

This list focuses on variable parameters which have been previously discussed. Further possibilities for personalization of the GUI may emerge after implementation, for example more specific details such as the response times of buttons.

9 DISCUSSION

9.1 FULFILMENT OF PROJECT AIM

The following section will discuss the three demands formulated in the thesis purpose (1.1) and whether these have been fulfilled. Thereafter, the three product characteristics that were deemed most important in the Use Profile Table (4.3) will be reflected upon.

9.1.1 PURPOSE FULFILLMENT

The following three demands, each relates to one of the corner stones of the GUI.

The main screen application shall allow for a varying number of workspaces

This demand relates to the layout corner stone, i.e. how the viewports (workspaces) are arranged on the screen. In this solution the viewports mimic boxes seen from above, which will collide with each other when they get close. New users can open their own viewport, given there is enough space, when they have approached the screen to join the interaction. When users leave the screen they have the option of closing the viewport, leaving room for other users, or the viewport will automatically fade away after a given time. With these properties the above demand can be considered met.

The workspace shall be able to shift position as well as size

The second demand was intentionally formulated in a way which did not determine whether or not the users themselves should be able to vary the size of the viewport and its position on the screen. This was an additional advantage which the final concept design provided. Likewise, a varying size does not necessarily mean that the aspect ratio of the viewport needs to be variable, though this solution did allow for stretching both the width and height. The viewport is surrounded by a frame where the corners are used to scale or stretch it whereas the borders are used to move the viewport around. Hence, the solution is independent of the number of touch points. This is a stable solution which fulfils the above demand.

The solution shall provide means for switching between applications within ones workspace

The third and final demand states that the solution should include means for browsing between applications, i.e. a menu of some sort. The final menu concept is an adaptation of the toolbar menus which can be found on personal computers. It does provide a solution for subcategories which enables it to handle large libraries of applications. The subcategory function was initially not a part of this specific concept, and the absence of this feature lowered its score in the Decision Matrix. Had it been included, this concept would have won with an even larger margin, which further suggests that this was the best solution for the menu.

9.1.2 IMPORTANT PRODUCT CHARACTERISTICS

The Use Profile Table (4.3) identified “Easy to use” and “Adaptability” as the most critical characteristics of this specific product.

Easy to use

It has been found difficult to determine how easy a product is to use without having users test is. The user trial only tested the three concepts for the viewport, and even then the results were far from conclusive. On average, two of the tested concepts received very equal grading from the trial subjects. The chosen concept did however receive a slightly higher grading than the other two on the criteria “Easy to use” (6.4.1). More precisely, the chosen concept received a grading of 5.67 on a scale from one to six. Setting aside the comparison between the concepts, this grade alone can be said to fulfil the criteria “Easy to use” quite well.

The concepts for the layout and the menu were never physically tested. Instead they were all evaluated in Decision Matrices, where they were graded on how well they fulfilled a number of weighted criteria. Both the grades and the weights were set by the author of this thesis. Though they were based on findings from the literature review as well as the results from the Use Profile Table, they are still estimations and can therefore not be verified to be accurate. In order to validate whether these functions are easy to use they will have to be more thoroughly tested.

Adaptability

Since the design should be possible to modify in order to suit different customers as well as users, adaptability was of great importance. This characteristic was therefore not only included in all Decision Matrices used throughout this thesis, it also had the highest possible weight in every one of these matrices. Furthermore, the adaptability of the design was largely considered during the concept refining phase for all three corner stones. All parameters which can be varied in order to modify the appearance of the main screen application, without altering the basic functions, are listed as “Variable parameters” in 8.1.

9.2 PROCEDURE USED IN PROJECT

During the thesis work, the project aim shifted from delivering a final product including the software to an increased focus on the development of a detailed concept with room for customization. In this section the tools used during the development work are reviewed and the decision to divide the GUI into three cornerstones is discussed.

9.2.1 DIVIDING THE GUI INTO THREE CORNER STONES

The decision to divide the development of the cornerstones was made at the end of the initial brainstorming phase. It was decided due to the fact that nearly all of the generated ideas could be said to belong to one of the corner stones. There was in fact only one exception, an idea which involved the menu being integrated into the design of the viewport frame. Having those designs so strongly linked to each other would have complicated any adjustments to the design, and with adaptability being one of the most important product characteristics, that would be a serious disadvantage.

A potential risk with this method is that it can result in a sub-optimized product, i.e. the focus on the complete and finished product may be lost if it has been divided into too small sections, in this case, cornerstones. As the three cornerstones in this thesis were found to be more or less completely independent this might not be a large issue. Also, since this product should allow for certain adjustments a modular solution is more preferable than a fused, though perhaps fully optimized, design.

9.2.2 TOOLS

The following tools, arranged in the order they appear in the report, were used during this thesis.

Brainstorming

The brainstorming method was used in the first phase of the development, i.e. the idea generation. The following three phases aimed to decide the best suited solution for each corner stone. Had any of the ideas generated in the brainstorming sessions not been adequate they would have been discarded later on in the process. Hence, the possible weakness of the brainstorming method was not the risk of generating useless ideas, but rather the risk of failing to generate ideas which could have been useful. Brainstorming is often used in groups which most likely results in more ideas than when performed by only one individual. If more contributors would have been present for the brainstorming sessions conducted during this thesis, the outcome might have been completely different. It is also possible that the literature review, which was carried out prior to development work and the idea generations, had both a positive and a negative influence on the brainstorming sessions. Positive in the sense that some of the ideas generated were inspired by the existing products that had just been studied, though negative since studying these similar products may have compromised the creativity by narrowing the vision of possible solutions.

Screening

As explained in section 2.4, the screening process was a fully verbal activity where each of the ideas generated during the brainstorming sessions were discussed. During these discussions the author of the thesis and one or more of the employees at i4sense was present. It might have been possible to evaluate the ideas using a more systematic tool, like the decision matrix was used during the screening phase of the menu development in order to decide which type of menu to use. However, since these ideas often were solution to partial functions, like sketches of a menu design or a response to a certain touch command, it would be quite difficult to evaluate them against each other. Additionally, evaluation such large amount of ideas using a systematic tool would most certainly be very time consuming.

Decision Matrix

Decision matrices are a common evaluation tool in product development. By evaluating partial functions separately an overview of how promising one solution is compared to others can be gained. In this thesis, very few of the functions included in these matrices were measurable, meaning that the grades they were given were based on estimations done by the author. Furthermore, which functions to include, as well as their assigned weight factors, was also decided by the author. Even though these were based on findings from the literature review and the pre-study, the interpretation of these findings can differ between individuals. The same can be said about the number of functions included as well as the range of both the weight- and grading scale.

Morphological matrix

The Morphological Matrices have the ability to combine partial solutions into a large amount of concepts. The three concepts chosen from each matrix was picked to represent as many of the inserted ideas as possible. Nevertheless, some of the partial solutions never made it through the morphological matrices, even though they had been considered to be good ideas in the preceding screening phase.

However, it was possible to make adjustments to the final concept later on in the concept refinement phase and in that way incorporate these partial solutions. For example, the final concept for the menu originally did not include a solution for handling subcategories. This was added in the concept refinement phase and inspired by a solution initially intended for another menu concept.

Use Profile Table

The Use Profile Table (UPT) was used to find out which product characteristics were most important to consider when developing this particular product. The calculations which led to the prioritized list which have been used throughout the thesis work were produced during the project. They are not a standard part of the UPT but were developed in order to enable quantification of the importance of the varying product characteristics.

User trial

Since the table of requirements was not sufficient to choose one of the three remaining frame concepts, a user trial was carried out.

Selection of test users

One benefit with carrying out the user tests in Lindholmen Science Park was the easy access to trial subjects. Though almost half of the test group was invited to represent a variety of target groups, the remaining half had very similar backgrounds; especially their education, line of work and technical skills were in the same areas. Furthermore, most of the trial subjects were asked to participate in the trial after they themselves had approached the touch screen in order to try it out, indicating that they also had a general interest in technology. But even though the solution was supposed to be designed to suit a wide range of users, as stated in the user profile, it is possible that the people who approached the screen at Lindholmen better represent the people who are likely to try out touch screens at other places as well.

In total, 18 persons participated in the trial. Though a higher number of participants would have been desired, this was determined to be enough, considering that the choice of final concept was not solely dependent on the results from the trial, but also backed up from previous findings. It probably would have been possible to get more of the people passing by to participate, but that would have made the test group even more homogeneous.

Trial layout

As mentioned in the method sections, the ideal case would be if the trial subject could try out the concept without knowing which part was being the focus of the test, but without any working application inside the frame this was difficult to do. It was considered that the task could be formulated in a way that did not reveal all the functions that were being tested, for example by asking the trial subject to try and manipulate the frame instead of asking them to move and resize it. However, this would only have any impact on the first concept tried and different prerequisites for the concepts would make the results less reliable.

To avoid learning effects the order in which the concepts were tested varied between trial subjects. Another way to do this could have been to use a baseline concept which the users could play around with in order to familiarize themselves with the touch screen. This approach would however had required the development of another mock-up.

One difference between the prototypes and the final solution was that the interactive application that should run inside the viewport was replaced by a static image displaying a web page. Therefore, if a trial subject performed a certain touch gesture inside the viewport it would not respond. It is possible that having static viewport content affected the users experience and in turn how they perceived and graded the interaction. The only way to be sure of the impact would be to test the prototypes with a running application, see Future work below.

9.3 FUTURE WORK

As was noted during the development of the viewport, physical user trials and tools such as decision matrices does not always reach the same conclusion. It therefore would have been interesting to test the concepts for the other corner stones as well, to see how the result from those user trials would influence the choice of final concept. Furthermore, two of the viewport concepts achieved similar grades from the participants in the corresponding user trial. A more detailed test of the partial functions of those concepts could aid the development of an additional combined concept, incorporating the most promising partial solutions from both said concepts.

It would also be interesting to investigate the effect different testing environments may have on the users' opinions of the concepts. The *Use cases* described in the Pre-study (4.1) could be used as inspiration to find suitable environments. Furthermore, it is very difficult to estimate the effect of having an application running inside the viewport during usage. Physically testing this could reveal aspects of the interaction that has not yet been considered.

During the final presentation of this thesis the risk of physical collision between left-handed people standing to the left of right-handed people was brought up, i.e. if they both use their preferred hand for interacting with the screen they may poke each other with their elbows. However, when manipulating the viewport no fine motor skills are required, meaning that most users should be able to perform all actions with either hand. User test can nevertheless be carried out to investigate this possible risk further.

Additional ideas

- Two of the trial subjects tried to double-click on the frame. Perhaps this was due to the viewport reminding them of the windows used on personal computers, where in some cases double-clicking is used to automatically maximize the window. Depending on where and for what the screen is to be used, it could be an idea to implement a variant of this function. However, for the use cases in this report such a function could prove to bother other users interacting with the screen.
- Since there is no strong incentive for a user to close the viewport after completing the interaction, it would be convenient to include a function which does this automatically. In addition to the function which allow the viewport to fade after a certain time of inactivity, with the use of a motion detection technology the system can sense when users distance themselves from the screen and consequently close the viewport.

9.4 LEARNINGS

As this thesis work progressed, the main lessons learned were the following:

- *There is limited research regarding large touch screens*

During the literature review it became apparent that the amount of published works and research on large touch screens was rather limited, indicating that the research topic had not fully matured at the time of writing.

- *The possible touch commands are restricted by existing norms*

Even though touch interfaces is a relatively modern technology, user expectations have already been set by current interface design practices. Since this interface was primarily intended for first time users, it was especially important not to deviate too far from these standards.

- *Different evaluation methodologies yield different results*

During concept evaluation it became apparent that the methodical approach using decision matrices to evaluate sub functions yielded a different result than did the user trial. While the decision matrices mainly evaluated the suitability of the concepts in some specific cases rather than their combined impact on usability, the focus of the user trial was the participants' experience of the interaction. Altogether, it appears advantageous to combine both methods when designing interfaces, as usability is a major component in successful interaction design.

10 REFERENCES

alai6666. (2007, May 16). *YouTube*. Retrieved February 27, 2012, from Jeff Han's 8 ft. Multi-Touch Display Wall: http://www.youtube.com/watch?v=JfFwgPuEdSk

APA PsycNET. (2012). Retrieved March 5, 2012, from American Psychological Association: http://psycnet.apa.org/journals/bul/84/3/385/

Arcarpm. (2010, April 28). *YouTube*. Retrieved February 27, 2012, from Perceptive Pixel Multi-Touch System: http://www.youtube.com/watch?v=SOki2nY8cvs

Bligård, L.-O. (2011). *Utvecklingsprocessen ur ett människa-maskinperspektiv.* Göteborg: Chalmers Tekniska Högskola.

Boghard, M., Karlsson, S., Lovén, E., Mikaelsson, L.-Å., Mårtensson, L., Osvalder, A.-L., et al. (2011). *Arbete och teknik på människans villkor* (2:1 ed.). Stockholm: Prevent.

Brignull, H., Izadi, S., Fitzpatrick, G., Rogers, Y., & Rodden, T. (2004). The introduction of a shared interactive surface into a communal space. (pp. 49-58). New York, New York: ACM Press.

Buur, J., & Winbur, J. (1994). *MMI. Design, Man-Machine Interface.* Copenhagen, Denmark: Dansk Design Center .

Churchill, E. F., Nelson, L., & Denoue, L. (2003). *Multimedia Fliers: Information Sharing With Digital Community Bulletin Boards.* Amsterdam, The Netherlands: Wolter Kluwer Academic Publishers.

divIT. (2012). Retrieved March 3, 2012, from http://www.divit.hr/showroom.html

Drugovic, A. (2012). *Agile Development*. Retrieved March 15, 2012, from http://www.agiledevelopment.org/

Flück, D. (2010, February 23). *Colblindor*. Retrieved March 3, 2012, from Color Blind Essentials: http://www.colblindor.com/2010/02/23/color-blind-essentials/

Hinrichs, U., & Carpendale, S. (2010). *Interactive Tables in the Wild: Visitor Experiences with Multi-Touch Tables in the Arctic Exhibit at the Vancouver Aquarium.* Canada: InnoVis Group.

Hornecker, E. (2008). "I don't understand it either, but it is cool" - Visitor Interactions with a Multi-Touch Table in a Museum. *IEEE International Workshop on Horizontal Interactive Human Computersystem (TABLETOP)* (pp. 113-120). IEEE Conference Publications.

Hornecker, E., & Stifter, M. (2006). Learning from Interactive Museum Installations About Interation Design for Public Settings. *OZCHI '06 Proceedings of the 18th Australia conference on Computer-Human Interaction: Design: Activities, Artefacts and Environments* (pp. 135-142). Sidney: ACM New York.

i4sense. (n.d.). Retrieved May 2, 2012, from i4sense: www.i4sense.com

Jacucci, G., Morrison, A., Richard, G., Kleimola, J., Peltonen, P., Parisi, L., et al. (2010). Worlds of Information: Designing for Engagement at a Public Multi-touch Display. *CHI 2010: Public Displays* (pp. 2267-2276). Atlanta: ACM.

Janhager, J. (2003). Classification of users - due to their relation to the product. *International Conference on Engineering Design.* Stockholm, Sweden.

Kaminani, S. (2011). Human Computer Interaction Issues with Touch Screen Interfaces in the Flight Deck. *IEEE/AIAA 30th Digital Avionics Systems Conference (DASC)* (pp. 6B4-1 - 6B4-7). IEEE Conference Publications.

Lawrence, I. (n.d.). Retrieved May 2, 2012, from Ida Lawrence Portfolio: portfolio.idalawrence.com

Lepicard, G., & Vigouroux, N. (2010). Influence of age and interaction complexity on touch screen. *12th IEEE International Conference on e-Health Networking Applications and Services (Healthcom)* (pp. 246-253). IEEE Conference Publications.

Moscovich, T. (2007). Principles and Applications of Multi-touch Interaction. *Ph.D. Dissertation*. Rhode Island: Brown University.

multitouchfi. (2010, May 20). *YouTube*. Retrieved March 3, 2012, from Great Wall of MultiTouch: http://www.youtube.com/watch?v=H4_pupFR10Q&feature=related

MultitouchSupplier. (n.d.). Retrieved May 2, 2012, from Multitouch Supplier: www.multitouchsupplier.com

Net Marketshare. (2012, February). Retrieved March 5, 2012, from http://marketshare.hitslink.com/operating-system-market-share.aspx?qprid=10&qpcustomd=0&qptimeframe=M&qpsp=157&qpnp=1

Norman, D. A. (2002). Emotion & Design: Attractive things work better. *Interactions Magazine*, pp. 36-42.

patcon96. (2007, June 8). *YouTube*. Retrieved March 3, 2012, from Obscura HP Multi-Touch Video Wall WSJ D5 HP conference: http://www.youtube.com/watch?v=UwoAxSvYCzk

Peters, H., Gastrich, R., & Rome, L. (2011). *Recommendations for Multi-Touch Screen Design.* San Diego, CA: San Diego State University.

Pheasant, S., & Haselgrave, C. (2005). *Bodyspace - Anthropometry, Ergonomics and Design.* London: Taylor & Francis.

SDLC. (2011, December 22). Retrieved March 15, 2012, from Spiral model: http://www.sdlc.ws/spiral-model/

Sousa, C., & Matsumoto, M. (2007). Study on Fluent Interaction with Multi-Touch in TRaditional GUI Environments. *TENCON 2007 - 2007 IEEE Region 10 Conference* (pp. 1-4). Taipei: IEEE Conference Publications.

Symanski, R., Goldin, M., Palmer, N., Beckinger, R., Gilday, J., & Chase, T. (2008). Command and Control in a Multitouch Environment. *Proceedings of the Army Science Conference (26th).* Fort Monmouth, New Jersey.

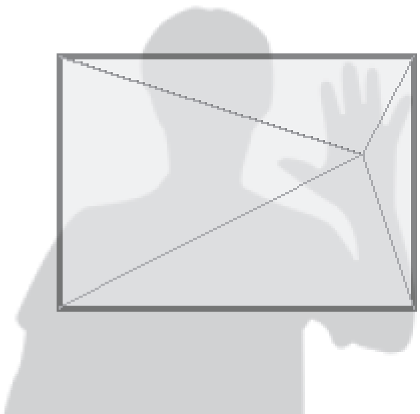
Vanacken, D., Demeure, A., Luyten, K., & Coninx, K. (2008). Ghosts in the interface: Meta-user Interface Visualizations as Guides for Multi-touch Interaction. *Horizontal Interactive Human Computer Systems*, (pp. 87-90).

Vanderheiden, G., Guarino Reid, L., Caldwell, B., & Lawton Henry, S. (2012, January 3). *W3C Web Accessibility initiative*. Retrieved February 27, 2012, from How to Meet WCAG 2.0: http://www.w3.org/WAI/WCAG20/quickref/20120103/#meaning

Vile, A., & Polovina, S. (2000). *Making accessible web sites with semiotics.* Retrieved February 27, 2012, from http://homepages.gold.ac.uk/polovina/publications/Semiotics4AWDshortpaper.PDF

Xia Jin, Z., Plocher, T., & Kiff, L. (2007). Touch Screen User Interfaces for Older Adults: Button Size and Spacing. In *Lecture Notes in Computer Science* (pp. 933-941). Springer Berlin / Heidelberg.

APPENDIX A



Open a viewport by touching a free space on the screen. If there's enough available space a "confirm" question and button can be presented to the user first (to avoid accidental opening of viewports). Otherwise an error message can appear explaining that not enough space is available and suggest that the user tries to open a viewports somewhere else.

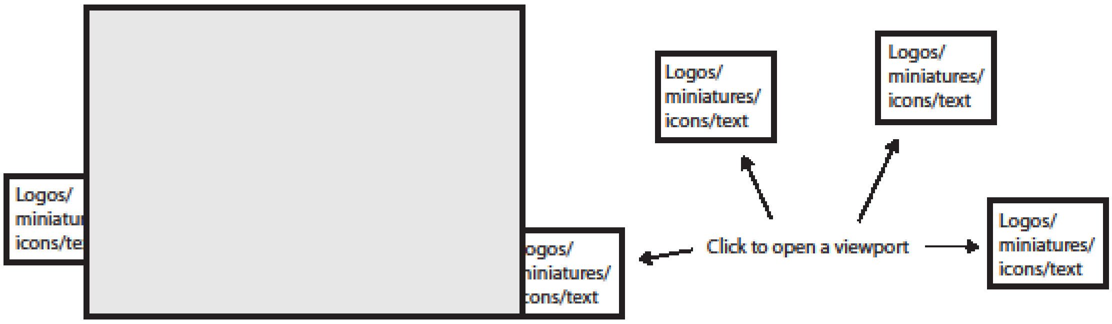
An avatar illustrating the interactive possibilities can walk around and open new viewports (not working ones) so that users understnad how it is done.

One drawback is that the avatar needs to be in the background so it doesn't disturb interacting users. This also means that the avatar will eventually be covered by viewports and it's purpose, i.e showing people how to open new viewports, will be lost.

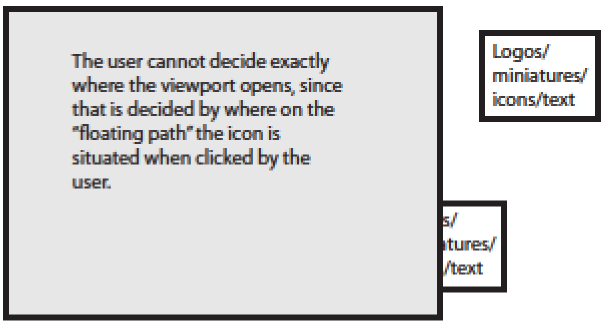
The user can decide exactly where the viewport opens, apart from the restrictions set by screen size and other open viewports.

No screen space is lost to icons or miniatures.

There should be more floating icons than possible open viewports, a user who have already understood how to open a viewport should not have to wait for an icon to be reachable. Hence there need to be an abundance of icons.



Moving items catch our glance easier, but do they invite to interact or touch?



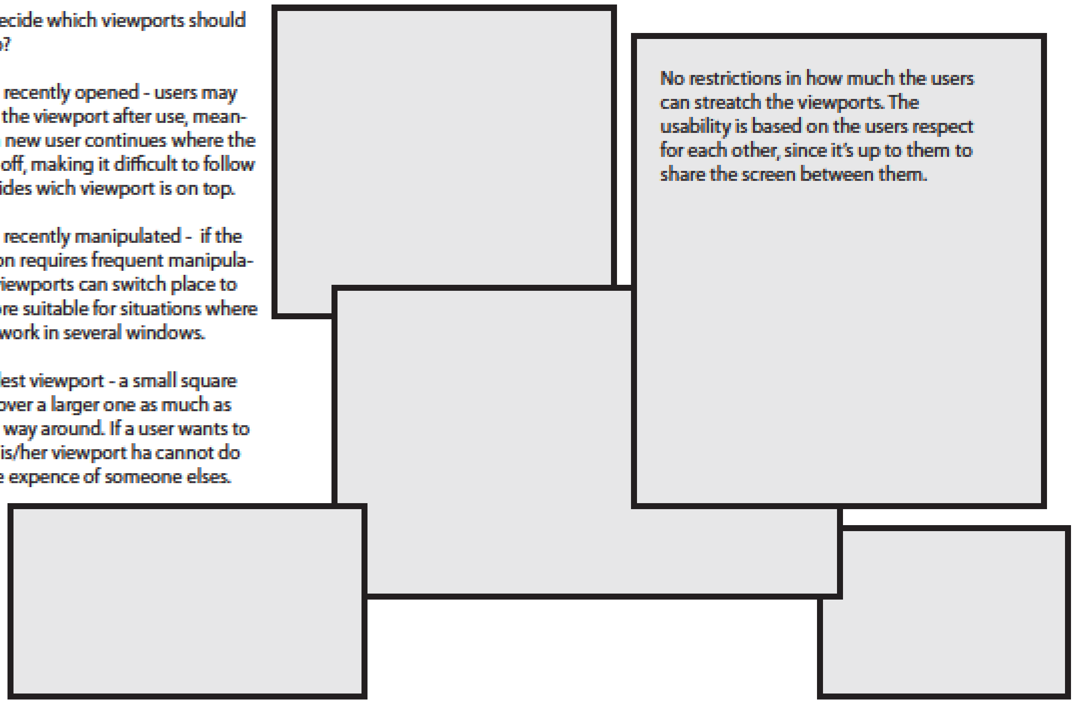
The user cannot decide exactly where the viewport opens, since that is decided by where on the "floating path" the icon is situated when clicked by the user.

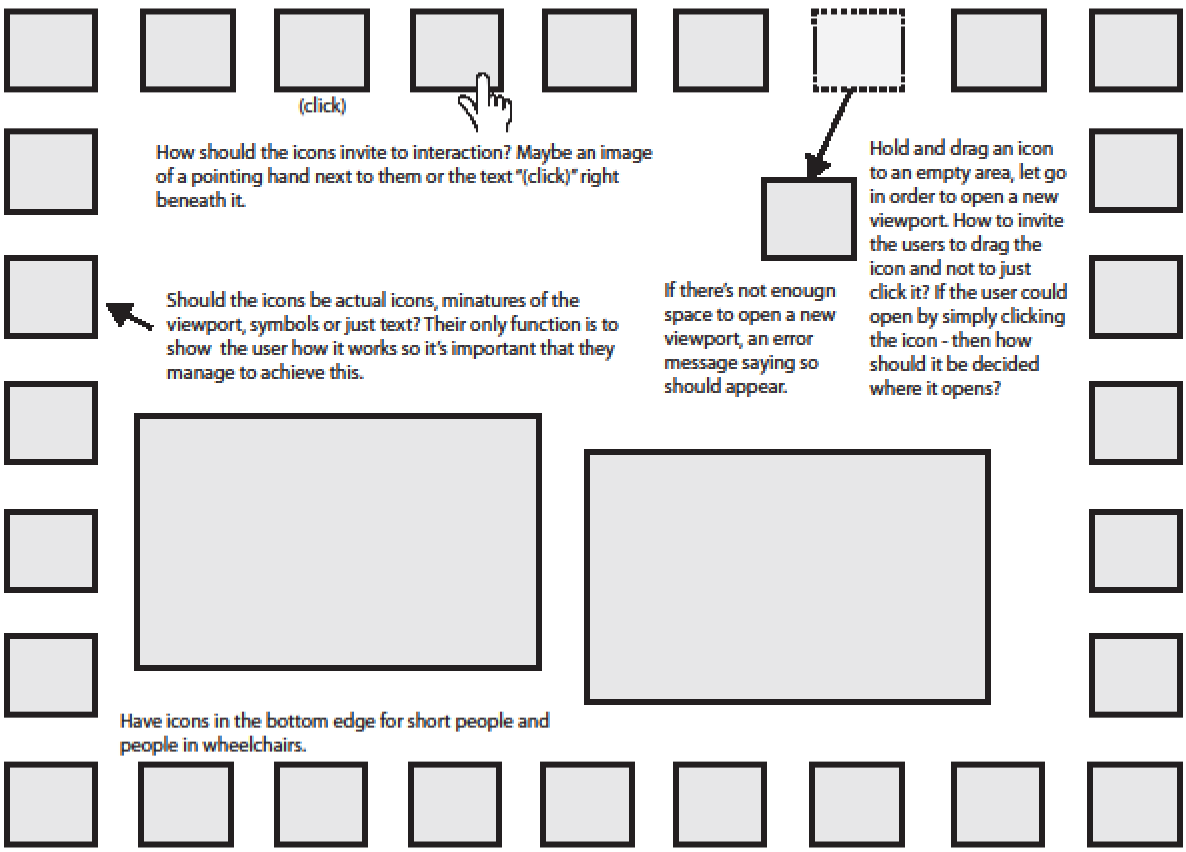
How to decide which viewports should be on top?

The most recently opened - users may not close the viewport after use, meaning that a new user continues where the old takes off, making it difficult to follow what decides wich viewport is on top.

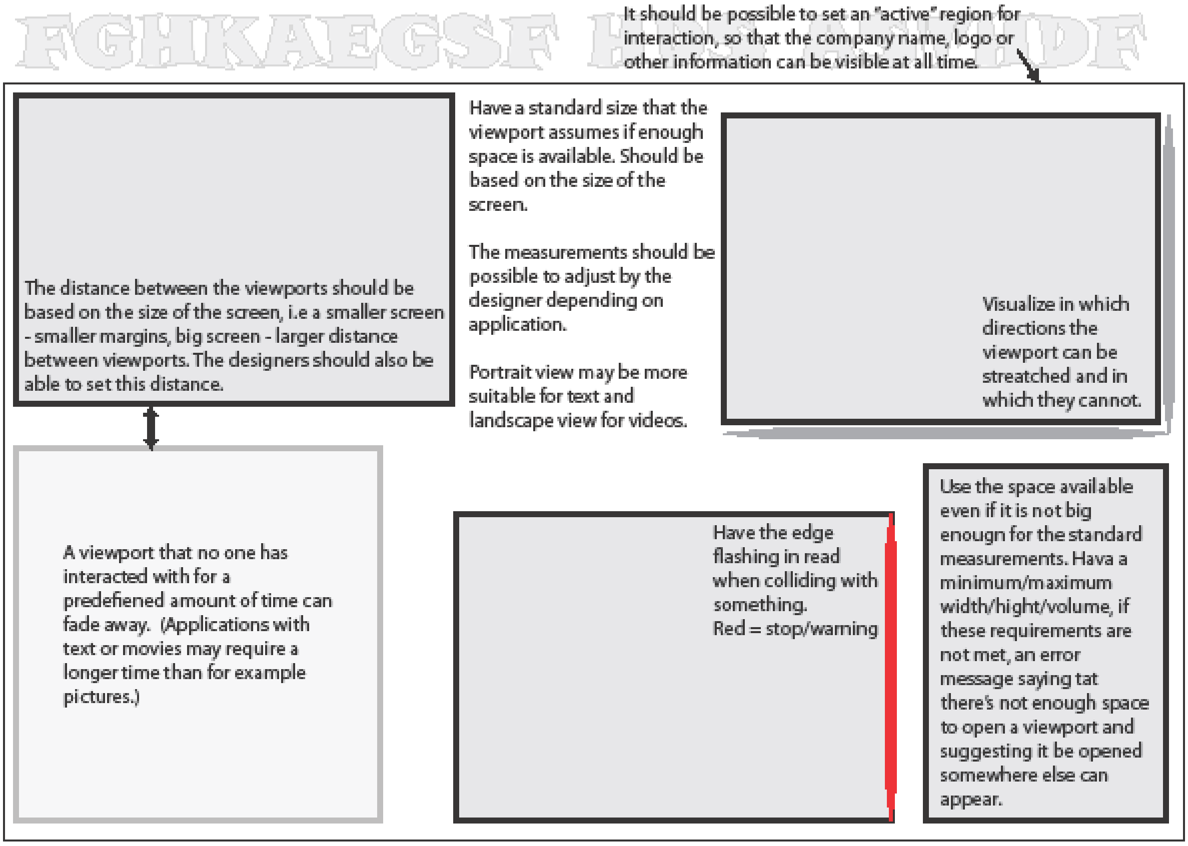
The most recently manipulated - if the application requires frequent manipulation the viewports can switch place to often. More suitable for situations where one user work in several windows.

The smallest viewport - a small square doesn't cover a larger one as much as the other way around. If a user wants to enlarge his/her viewport ha cannot do this at the expence of someone elses.






Chose how many viewports that should be available "on site".
If the number of viewports is changed during interaction the users may be annoyed, because their interaction is affected by someone elses operation.
1
2
3
4
5
6


It should be possible to set an "active" region for interaction, so that the company name, logo or other information can be visible at all time.
The distance between the viewports should be based on the size of the screen, i.e a smaller screen - smaller margins, big screen - larger distance between viewports. The designers should also be able to set this distance.
Have a standard size that the viewport assumes if enough space is available. Should be based on the size of the screen.
The measurements should be possible to adjust by the designer depending on application.
Portrait view may be more suitable for text and landscape view for videos.
Visualize in which directions the viewport can be streatched and in which they cannot.
A viewport that no one has interacted with for a predefiened amount of time can fade away. (Applications with text or movies may require a longer time than for example pictures.)
Have the edge flashing in read when colliding with something.
Red = stop/warning
Use the space available even if it is not big enough for the standard measurements. Hava a minimum/maximum width/hight/volume, if these requirements are not met, an error message saying tat there's not enough space to open a viewport and suggesting it be opened somewhere else can appear.

APPENDIX B

Concept 1	2	3	1	1	1	3	2	2	1	3	1	3	1	1	1	1	1	3
Easy to understand how to use it	4	5	6	5	4	5	6	5	5	4	6	5	6	6	5	5	5	5
Easy to use	6	5	6	6	5	5	6	6	6	5	6	5	6	6	6	6	6	5
Fun and feels good to use	6	4	6	6	5	3	4	5	6	6	6	4	5	6	6	6	5	6
Good precision	6	4	6	5	5	6	3	4	5	6	6	6	4	6	6	6	5	6
Total	22	18	24	22	19	19	19	20	22	21	24	20	21	24	23	23	21	22
SingleTouch on first try		✓	✓	✓		✓	✓		✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
MultiTouch on first try	✓				✓			✓										

Concept 2	1	2	2	2	2	2	1	1	2	1	2	1	2	2	2	2	2	2
Easy to understand how to use it	5	4	6	6	5	4	6	6	6	6	5	6	6	5	5	6	5	5
Easy to use	6	4	6	6	4	4	6	6	6	6	6	6	6	6	6	6	5	5
Fun and feels good to use	5	5	6	5	4	5	5	5	5	6	6	6	4	6	6	4	5	6
Good precision	6	5	6	5	4	6	4	4	4	5	6	6	3	6	5	6	5	6
Total	22	18	24	22	17	19	21	21	21	23	23	24	19	23	22	22	20	22
SingleTouch on first try	✓	✓	✓	✓		✓				✓		✓	✓	✓	✓	✓	✓	✓
MultiTouch on first try					✓		✓	✓	✓		✓							

Concept 3	3	1	3	3	3	1	3	3	3	2	3	2	3	3	3	3	3	1
Easy to understand how to use it	6	5	5	5	3	6		5	6	6	4	6	5	5	5	6	6	6
Easy to use	6	5	5	5	3	4		3	6	6	5	6	4	5	6	6	6	6
Fun and feels good to use	6	6	5	4	3	4		4	6	6	4	6	3	5	6	5	6	6
Good precision	5	5	5	4	2	4		5	5	5	5	6	2	4	6	2	5	6
Total	23	21	20	18	11	18	0	17	23	23	18	24	14	19	23	19	23	24
SingleTouch on first try			✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
MultiTouch on first try	✓	✓																

Trial Subject																		
First tried concept no:	3	3	2	2	1	3	1	1	1	2	2	3	1	2	2	3	3	1
Next tried concept no:	1	2	3	1	3	2	2	2	3	1	3	2	2	3	1	1	1	3
Last tried concepts no:	2	1	1	3	2	1	3	3	2	3	1	1	3	1	3	2	2	2