Design and implementation of a timeline for the Web

by

Tobias Jansson

LIU-IDA/LITH-EX-G--15/037--SE

2015-06-10
Examensarbete

Design and implementation of a timeline for the Web

av

Tobias Jansson

LIU-IDA/LITH-EX-G--15/037--SE

2015-06-10

Handledare: Klas Arvidsson
Examinator: Jonas Wallgren
Design and implementation of a timeline for the Web

Tobias Jansson
tobja258@student.liu.se

ABSTRACT
Visualizing time information using a timeline has been done for over 250 years. Using a timeline can assist users in making sense of time data which could otherwise be hard to understand. A timeline has been implemented at Ida Infront, and its purpose is to help managers understand how cases are progressing in a case management system created by Ida Infront. The timeline is designed using usability methods and has been evaluated with regards to its usability and how it interacts. Results show that there are a number of things that can increase usability and add meaningful interaction, such as using icons to collect similar data into one point and adding filtering capabilities so that users can find interesting data faster.

INTRODUCTION
Motivation
Visualizing data points spread through time using a timeline is a practice that can be traced as far back as 1765 A.D. [3]. A timeline can be employed to make sense of the time aspect of data, which could otherwise be hard to imagine.

However, visualizing temporal data means more than simply ordering them along an axis [11]. The white space (the blank space between data points) is also an important part of the visualization and may, for instance, be used to indicate long passages of time [8].

Making use of a timeline may also considerably improve the ease of which temporal data can be understood. For instance, when trying to interpret evidence information it was shown that presenting the evidence in chronological order impacted the jurors’ decisions [17] (original source: [19]).

The timeline that is to be implemented is intended to be a part of a case management system created by Ida Infront AB [13]. Its purpose is to help managers gain a better understanding of which employees has worked on a specific case. It will also be possible to see when there are standstills in the process, leading to a more efficient workflow.

While timelines designed for the Web already exists [26, 28, 6], there does not seem to be any timelines based upon user studies. Creating a timeline and applying prototyping and evaluation research thus seems like a good way to try to make the scientific world and the Web design world interweave.

Purpose
The purpose of this thesis is to implement a timeline that is to be used as a part of a case management system. Previous research made in the area of information visualization will be studied and applied while implementing the timeline.

Research questions
• How can a timeline be designed in order to maximize its usability?
  Usability in the sense that a user should easily be able to use the timeline to identify standstills in the process, as well as being able to discern who are involved in a particular case.

• What kind of interaction can be used in order to make the timeline more useful to the user?
  As cases can be ongoing for several years, changes will be spread across a large timespan. Being able to interactively manipulate the displayed time range in some way will be required as screens do not have the necessary space to show the entire timeline at once.

  Also, the case management system is able to automatically carry out tasks, thus many events might be created during a very short period of time. Letting the user control the timeline’s data density would allow the user to choose whether he/she wants the timeline to display the “big picture” or all possible details.

Limitations
Due to time constraints Web accessibility will not be considered when designing the timeline. Web accessibility refers to the ability for people with disabilities to use the Web [14].

Also due to the aforementioned time constraint and an extensive literature review, it will not be possible to carry out the desired number of iterations when designing the paper prototype and testing the implementation.

Because the work is done at Ida Infront whose customers are external, it is not possible to recruit real users for testing the timeline’s design. Instead employees at Ida Infront who have knowledge of how real users use the system will act in place of real users.

BACKGROUND
This thesis will (as mentioned earlier) be carried out in collaboration with Ida Infront AB.

The case management system, which the timeline is to be a part of, is targeted towards government agencies who are in need of a business-critical system that supports the individual agency’s processes and also keeps a journal of everything that is changed.

The intended users is employees at government agencies. They will all have knowledge of how to use the system, and also know that every action is logged with a timestamp. Therefore, the users should be able to understand what the timeline shows.
Frida Vestman at Ida Infront is the project’s stakeholder, and Christian Svedin at Ida Infront is the project’s supervisor. A requirement specification for the timeline can be found in Appendix 1.

**Technical Background**

Ida Infront uses a wide variety of techniques. Some systems are built using Java Platform, Enterprise Edition (Java EE), and in particular Apache Wicket [1] is one framework that is used.

Apache Wicket (from now on called “Wicket”) is a Java framework that uses a component-based approach which makes it easy to extend with custom components.

**THEORY**

**Usability**

The International Organization for Standardization (ISO) standard 9241-11 [15] defines usability as follows: “Extent to which a product can be used by specified users to achieve specified goals with effectiveness, efficiency and satisfaction in a specified context of use.”

When designing with usability in mind, Gould et al. says that design guidelines can be helpful [12]. However, they also acknowledge that guidelines has two inherent drawbacks. First off, they are independent from the context of the system that is being designed. Secondly, they often rely on opinion rather than data.

Gould et al. also recommends three principles when doing design work [12]:

- **Focus early on users and tasks**
  
  Knowing the target audience and what they want to do with the system is important.

- **Empirical measurement**

  Users should carry out work using prototypes early in the process while noting how well the tasks can be performed as well as the users’ reactions.

- **Iterative design**

  When problems are found during tests of the prototype, they must be fixed. This should be done in an iterative cycle as pictured in Figure 1.

**Interactivity**

The ISO standard 9241-210 [16] defines an interactive system as such: “combination of hardware, software and/or services that receives input from, and communicates output to, users”

Research in the information visualization field tends to be about visualization (i.e. representation) while overlooking the area of interaction [29].

With this in mind, Yi et al. [29] has created a framework that aids discussion and evaluation of interactivity. The framework features seven categories as follows:

![Figure 1: An iterative process consisting of design, test and measure, and redesign recommended by Gould et al. [12]](image)

**Select**

The act of marking some data item(s) as interesting, as a way to keep track of it. After selecting one or several items, the visualization can indicate this in several different manners. One is to add an identifying label to the item, so that the user can easily find a particular data item again.

jHeatmap is an interactive heatmap that lets the user select rows, columns and individual cells. Selecting rows and columns will cause them to become highlighted in the heatmap, while selecting an individual cell will make an information box appear with more details about the value in that cell [9].

**Explore**

A way for the user to display another part of the data. One technique for exploring, which is employed by jHeatmap [9], is the ability to pan around the dataset, thus allowing the user to see different parts of the data.

Slingsby et al. has explored hierarchical layouts [22] and developed HiVE, which is a notation for describing how data is to be visualized in hierarchical layouts. The notation is divided in two parts: states and operators. One of the operators in HiVE that allows for exploration is oCut, which allows the user to show a different subset of the data by removing a level in the hierarchy.

**Reconfigure**

Reconfigure refers to the ability to see data from different perspectives. For example, the act of sorting is one technique that falls under this category. jHeatmap [9], for instance, allows the user to sort either a row or a column in an ascending or descending manner.
Another example is when relationships are visualized with nodes and edges. Allowing the user to drag nodes around means that the visualization is reconfigurable.

Elmqvist et al. [10] has researched using stacked scatterplots as a way to explore multidimensional data which is then visualized in 3D. The user is able to turn the visualization in all three dimensions, thus reconfiguring it. In Figure 4 (in Appendix 3) an example of the system can be found.

Yet another example comes from HiVE, where the state sOrder allows the user to reconfigure information by changing its order in the hierarchical layout.

**Encode**

When changing the data representation, the information displayed is said to be encoded. One example is when a user switches from a pie chart to a bar chart.

Another example comes from HiVE, where the operator oLayout allows the user to encode information by changing between different rectangular and non-rectangular displays.

**Abstract and Elaborate**

Abstract and elaborate refers to the user’s ability to influence the amount of detail displayed.

Bade et al. [2] has created a zoom mechanic for their line graph visualization where the details of the data can be increased as well as decreased. By allowing changes to the detail level, the user can decide themselves whether the data should be more abstract or elaborate, depending on the type of analysis they are currently carrying out.

**Filter**

A filter is a way for the user to hide data that is not relevant to the displayed visualization. For instance, allowing the user to move a slider in order to affect the range of the data which is retrieved from a data set.

In jHeatmap [9] it is also possible to filter data. The user is able to hide rows as well as columns which the user might find distracting or not interesting.

**Connect**

Lastly, the connect category encompasses interaction techniques that lets the user find data related to other data. An example is if we consider the visualization with edges and nodes again. One technique that would fall under the connect category, is that when the user hovers over a node all the neighboring nodes are highlighted.

Elmqvist et al. [10], in their study of multidimensional scatterplot matrixes, shows how the values in the matrix connects to each other by drawing lines between the scatterplots.

It should be pointed out that this categorization does not cover all kinds of interaction techniques that are applicable to information visualization. For instance, an undo/redo technique might be used so that the user, for example, could undo a previously applied filter. However, because this technique and other kinds of general techniques are not unique to information visualization, they were not categorized by Yi et al. [29].

**Prototyping**

When prototyping, low-fidelity prototypes are advantageous as their cost is low and they can be quick to develop. Furthermore, they also allow for rapid iteration and makes it easy to evaluate multiple design prototypes in parallel [21].

Exploring multiple design prototypes in parallel has been studied by Baekker et al. [27]. Their study found that using multiple prototypes when evaluating design can help the designers get more critical comments from test users, since they will not feel that they may cause disappointment when criticizing a particular design. Evaluating multiple prototypes at once may also cause the test users to disregard comments on smaller details of the prototypes, and instead focus on the bigger picture.

Carolyn Snyder’s book *Paper Prototyping* [23] details the prototyping method that was followed.

**Paper Prototyping**

Prototyping is split into four phases, with an optional iteration phase, as outlined here.

1. **Target audience and goals**
   
   An initial meeting is required where the target audience is decided, and the goals of the system are determined.

2. **Tasks that should be achievable**
   
   All tasks that the users should be able to carry out using the system needs to be detailed.

   In order to consider a task complete, the following information is needed:
   
   - The task’s goal.
   - Input, for example: usernames, passwords, server names, etc.
   - Assumptions.
     
     One example is when testing how users react to signing up with a username that already exists. An assumption would be that the username that each user will try to register with during testing is already taken.
   - All steps needed in order to complete the task.
   - The amount of time it would require an expert to carry out the task.
   - Instructions that will be given to each user when evaluating the task.
   - Notes, such as instructions for how to do test the task, what to look out for, and so on.

3. **Creating a prototype**
   
   Before creating a prototype, a list has to be compiled containing all pages and major elements which is needed to complete the previously defined tasks. Based on this list, a paper prototype needs to be created.

4. **Performing a usability study**
   
   With the prototype finished, performing a usability test with test users remains. The test users will ideally be real
users, i.e. the ones who will use the system once it is finished.

During the study, each user will be asked to complete the previously created tasks. Notes will be taken regarding how each user behaves, how they complete each task and what they think about during the study.

5. Iteration
If time permits, the prototype should be updated according to the findings found when testing. The prototype may be tested again in order to see if more issues arise.

Evaluation

Heuristic Evaluation
Heuristic Evaluation [18] is an informal method of evaluating a user interface. Carrying out a heuristic evaluation means that one or several person(s) look(s) at the interface and come(s) up with an opinion on whether the interface is good or bad. In order to make the evaluation more organized, a set of rules should be used to make the evaluation more consistent. Nielsen et al. [18] suggests a set of nine usability heuristics as the base for this kind of evaluation, which is found in Table 1.

Results show that using just a single person for this type of evaluation is not reliable. The results improve considerably if several people carry out the evaluation independent of each other [18].

Cognitive Walkthrough
Cognitive walkthrough [20] is a design evaluation methodology that introduces tools to assess a system’s usability as well as identifying causes to usability issues early in the design process. Its purpose is to evaluate how well users can perform certain tasks with no or a very limited set of instructions.

In order to perform a cognitive walkthrough, the following information is necessary:

- A description of the design of the user interface.
- A scenario for one or several task(s).
- Assumptions about the target audience as well as the context.
- The steps needed by a potential user to accomplish the task(s) while evaluating.

Performing a cognitive walkthrough involves several steps. First, the designer presents the proposed design to a group of people. This group individually evaluates the design based on a set of criteria which is tailored to the design.

To evaluate, the evaluators goes through the listed steps while considering the interface’s behavior and the way it effects the user. If a step seems like it could be hard for the target audience to execute, the claim must be supported by empirical data, a theoretical argument or relevant experience. The same goes for steps which the reviewer deem to be easy to complete considering the target audience.

Streamlined Cognitive Walkthrough
Streamlined Cognitive Walkthrough [24] is a variant of the previously discussed Cognitive Walkthrough. The following steps are carried out when doing a streamlined cognitive walkthrough. The steps have been shortened for brevity’s sake.

- Target audience, tasks, and the user interface
  First off, the target audience must be defined. All tasks that the users should be able to accomplish using the system need to be produced. Additionally, a description or implementation of the system is required.

- Explaining the procedure
  Before beginning the walkthrough, the following things need to be explained to the team who participate in the walkthrough.
  - The goal of doing the walkthrough.
  - What will be done and what will not be done during the walkthrough.
  - How the walkthrough will be carried out.

- Carrying out the walkthrough
  For each task, go through all steps needed to complete the task. At each step, the following two questions need to be answered:
  - Does it seem like the user knows what to do in this step?
  - Consider that the user takes the right step, will he/she know that he/she is one step closer to the goal?

  For each step, the following types of notes needs to be taken.
  - If there does not seem to be anything wrong with a step, no note will be taken and the walkthrough continues on to the next step.
If the design assumes that the user have certain knowledge that he/she in reality do not possess, the problem should be noted along with the knowledge required by the user to complete the step.

In case someone points out a design flaw, attempts should be done to formulate it as a problem that the user may have, which should then be noted along with the design flaw.

Design ideas may arise, and they should be acknowledged and noted.

Likewise, if important functionality is missing, that should also be acknowledged and noted.

• Revision

Once the walkthrough is completed, the system should be revised in order to fix the issues that were found.

METHOD

Paper Prototyping

In order to create a prototype of the timeline, the method outlined in the section Paper Prototyping is followed, as it is cheap to create prototypes using paper.

Tasks will be defined as part of the prototyping, and based on them a paper prototype of the timeline will be created. As the timeline will consist of a single view and just a few components, there may be no need to compile a list of screens.

Upon completion of the paper prototype a usability study will be conducted where each user will be asked to complete the previously created tasks using the paper prototype while notes are taken simultaneously. The notes that will be taken encompasses several areas, including user thoughts, observations on the user’s actions, and inferences made during the study.

After concluding the usability study, another iteration of the paper prototype may be created and evaluated again. Once the evaluation does not yield more important feedback, the paper prototype will be implemented for Web browsers.

Implementation of the prototype

The implementation will be done as close as possible to the paper prototype, and any deviation will be noted. In order to implement the timeline so that Web browsers can display it, D3.js is used. By using D3.js, it is possible to create custom visualizations that allows developers to freely decide on how the data should appear on-screen.

Wicket is used to implement the back-end, as Ida Infront uses it when developing systems which meant that their employees would be able to continue working on the timeline upon conclusion of this thesis.

Evaluating the implementation

As discussed in the section Limitations, there are no users of the system available. Instead, employees at Ida Infront who have knowledge of how users use the system will act in place of real users. Therefore, to evaluate the implemented timeline, a streamlined cognitive walkthrough (which is discussed in the section Streamlined Cognitive Walkthrough) will be conducted.

Since the intended users and all tasks that should be achievable will already be defined when creating the paper prototype, there will be no need to redefine them for the streamlined cognitive walkthrough. Also, since the timeline will be implemented at this stage, there will be no need to create a description of the user interface.

A meeting is to be held where the walkthrough of the implementation is carried out. Starting off the meeting, several things will need to be explained. First off, the goal in doing the walkthrough. Secondly, what will be done (walking through the steps of each task) and what will not be done (designing) during the walkthrough. And last off, how the walkthrough itself will be done.

When carrying out the walkthrough, all steps of each task (which will be created while working on the paper prototype) will be gone through, and for each step two questions needs to be answered.

The first question is: “Does it seem like the user knows what to do in this step?”. And the second question is: “Consider that the user takes the right step, will he/she know that they are one step closer to the goal?”.

When walking through each step, notes will need to be taken. Refer to the previously mentioned section Streamlined Cognitive Walkthrough for more details.

Once the walkthrough is complete, the implementation should be revised in order to fix the issues that were found.

RESULTS

Paper Prototyping

Target audience and goals

The target audience was determined to be users of the case management system, which typically are employees at government agencies.

Goals of the timeline:

• Letting a user quickly gain an overview of cases.

• Have the ability to display most events from the case management system.

• Be able to display important events in a more prominent fashion.

Tasks that should be achievable

Three tasks were created, and they are mentioned briefly here. They can be found in full detail in Appendix 2.

• Which documents has the status ’incoming’?

The goal with this task is to find all documents which are marked as ’incoming’.

• What happened?

The goal with doing this task is to determine what happened with a case during a specific week.
• **Acquire an overview of a case**

  The user is tasked with gaining an impression on how a case has progressed.

**Creating a prototype**

Since the timeline will be a sole page, there was no need to list the required pieces for the paper prototype.

The paper prototype has been photographed and one view of it may be found in Figure 2.

**Usability study**

A usability study of the timeline paper prototype was conducted with employees at Ida Infront and during the study notes were taken of how the prototype was used to accomplish each task.

All three tasks were possible to perform without any issues being noted. One common thought was that showing the timeline day-by-day seemed unnecessary, since only the days when something actually occurred were of interest.

After the usability study a list of important findings were compiled, which may be found in Table 2.

**Iteration**

No iteration of the paper prototype was done, however all findings from the usability study were applied when implementing the timeline.

**Implementation of the prototype**

With the technical background section in mind, the timeline was implemented as a Wicket Panel. This made integration with the case management system easy to achieve, as a new version of the system is currently in development in Wicket.

Simple Java classes (i.e. the classes only contained fields with data types native to Java, also called POJO) were used to model all data in Wicket.

- Long gaps between events on the timeline made it hard to overview. Because of this, The timeline will be non-linear and the length of the gap between events will let the user know approximately how much time has passed. This also entails the use of a different format when displaying the date.

- Arrows are used to indicate incoming and outbound documents. In the prototype they point left and right, respectively, but having them point downwards and upwards would be more intuitive.

- Icons indicating events alternates sides on the prototype. In order to appear in a more consistent manner, they will all be placed on one side of the timeline.

- The zoom controls were deemed unnecessary in the initial implementation and was therefore removed.

Table 2: Important findings from the usability study.

When the front end requested data from Wicket, a JavaScript Object Notation [5] (JSON) representation of the server’s data was returned.

Data-Driven Documents [4] (D3) was used to create the visual part of the timeline. D3.js is a JavaScript library which can be used to combine HyperText Markup Language (HTML) with Cascading Style Sheets (CSS) and Scalable Vector Graphics (SVG) to visualize data.

D3.js generated HTML and SVG markup as well as inline CSS declarations. The implementation before conducting a streamlined cognitive walkthrough may be found in Figure 3b.

One major change made in the implementation compared to the paper prototype was displaying the timeline vertically instead of horizontally. This change was accomplished by rotating the timeline 90° clockwise along its Z-axis, thus preserving the look and general behavior of the timeline. The rotation was done so that it would work better on smaller screens, thanks to close contact with the project’s stakeholder and supervisor when implementing the paper prototype.

One requirement from the requirement specification was missed when prototyping, which is the ability to select a date range to display. As the project’s technical supervisor wanted to have this functionality, a decision was made that it should be added directly to the implementation.

One of the requirements from the requirement specification (which is found in Appendix 1) states that “The timeline shall fit the graphical profile of iipax for web solutions”. To accomplish this in the implementation, suitable colors were picked from Ida Infront’s graphical profile (which is not available to the general public).

**Evaluating the implementation**

**Target audience, tasks, and the user interface**

The target audience and the tasks were reused from the paper prototyping stage. As the implementation existed, there was no need to create a description of the user interface and instead the implementation could be used directly.

**Walkthrough results**

When walking through each task (in Appendix 2), notes were taken. All steps worked as intended — therefore no notes were taken regarding that part. However, while walking through the implementation some suggestions came up.

These suggestions have been compiled into a list of issues and possible improvements has been composed and can be found in Table 3.

**Revision**

No revision was carried out.

**Interaction categories**

With regard to Yi et al.’s framework (which was discussed in the section Interactivity), the timeline falls into the following categories:
• Once a user picks a date range where not all event types from the original data are present, the corresponding filter buttons are not removed.
• Filtering by event type does not hide days where no event occurs.
• Being able to group events by week, month and year would be a nice addition.
• Animations could be used when filters are applied. For example, event icons could fade away when filtered, and when a user selects a date range it could slide into the new representation.

Table 3: A list of issues and possible improvements from the streamlined cognitive walkthrough.

• Explore
The timeline does not show all data immediately, instead it allows the user to explore by scrolling which will reveal events which occurred earlier or later depending on which direction the user scrolls.

• Abstract/Elaborate
When the timeline is displayed, there are only icons that represents every different event type. By hovering an icon, the user is able to make the information more elaborate as details are displayed.

• Filter
Along the top of the timeline there are two different controls which allows the user to filter in two different ways. The first set of controls allows the user to select which event type(s) to display. The second set of controls allows the user to select the date range to display.

DISCUSSION

Results
In the beginning of this report, it is mentioned that there are a lot of timelines for the Web available, but none that seem to build on user studies. Despite the fact that it was impossible to recruit real users for testing the timeline, there are some conclusions that can be drawn from the results.

By using icons to differentiate between different kinds of events, it is easy to gain an overview of the timeline and identify interesting data. Letting users filter by event type and a

Figure 2: One view of the paper prototype.
date range makes it easy to find data considered interesting by the users. As the timeline is displayed vertically, there are no disruptions to the ordinary scroll flow of web pages. All other timelines mentioned in the beginning share a common trait; they are displayed horizontally. Since they are displayed horizontally they are harder to use on narrow screens. Using tooltips to display more data worked. One drawback with the current implementation of the tooltips however, is that it may not be immediately obvious that it is possible to hover over the icons. Another drawback is that no information other than the event type and the date it occurred is visible instantly. Having more data displayed up front could make it easier to quickly gather data, but displaying too much data at once may have the opposite effect.

The timeline was rotated in order to accommodate smaller screens. In April of 2015 at least 20% of visitors who visited sites monitored by StatCounter had a screen whose resolution along the X-axis was 800 pixels or less [25]. By displaying the timeline vertically, width became a non-factor. In order to see how cramped the timeline could appear when not rotated, an early implementation where the timeline is oriented horizontally can be found in Figure 3a.

Another positive aspect was that scrolling worked naturally; when the timeline was displayed horizontally the scroll wheel action had to be overridden in order to provide scrolling. That approach had a major drawback: scroll events does not occur continuously but rather periodically (when using D3.js’ zoom behavior, the event fires at most every 50ms [7]), which in practice meant that no matter how fast the user scrolled, the timeline would only scroll a set amount. As the timeline is now displayed vertically, the user is able to scroll at any pace and in any way their web browser is capable of.

However, one drawback with vertical display is that on larger screens a lot of space appears unused. One way of increasing screen utilization could be to display the timeline horizontally when enough horizontal room is available, and have it display vertically when the screen is too narrow.

Method

If it would have been possible to carry out more iterations, the streamlined cognitive walkthrough may have become redundant as more issues could have been found while evaluating the paper prototype.
When doing the streamlined cognitive walkthrough problems arose that may have been found by redesigning the paper prototype and testing it again. Iterating like this may have revealed that a vertical timeline was more desirable and that the date range controls were missing.

Cognitive walkthrough was chosen over heuristic evaluation for several reasons. First off, heuristic evaluation relies, as the name implies, on heuristics. Therefore those who carry out the heuristic evaluation first has to come up with a good set of heuristics. Secondly, doing a heuristic evaluation necessitates people who are trained in user interface design, as experience plays a big role in determining whether the chosen heuristics seem to be fulfilled or not.

In contrast a cognitive walkthrough can be carried out by just about anyone. Obviously the results (in the form of design ideas and suggestions on what could be done about certain steps) will differ depending on how much experience those who carry out the walkthrough have, but at the very minimum anyone should be able to determine whether the tasks are possible to carry out or not.

One interesting aspect that could have been measured is the time it took to complete each task. Doing this for each iteration (had there been any) during the usability study could have provided insight into whether the design actually improved with regards to time. There are at least two possible hindrances in doing this. First off, results would differ between each person participating in the study. Secondly, having the same persons repeat the test for each iteration would diminish the value greatly; since they remember how the task was carried out previously, it would take a much shorter amount of time for them to finish the tasks a second time. Finding a sufficient amount of test users to avoid the above mentioned hindrance may be an issue in itself.

Replicability
All results from the study are qualitative by its nature, which is a impediment with regards to replicability.

A problem with this is that it is possible that different conclusions would be reached and that designs would differ. With enough iteration, designs may converge but this would certainly depend upon the domain in which the timeline is supposed to be used.

The method itself did result in a design that worked well, and usability issues as well as different types of interaction that could be added in were found. By repeating the described method, it is likely that a good design will surface.

Reliability
Reliability is obviously influenced by the above; if the author of this report repeated the procedure, chances are pretty high that the same results would be reached. However, considering the qualitative nature of the method used, the same results would maybe not be obtained if someone else repeated the same method.

Validity
Concerning validity, the method used produces an implementation which is then evaluated. A paper prototype is produced which is then evaluated with a focus on usability. Because the evaluation is centered around defined tasks and is carried out in a structured manner, the paper prototype that is produced has been tested with regards to usability. Multiple types of interaction was also found when during the creation and evaluation of the paper prototype. By testing the implementation of the paper prototype, it was possible to find out whether the paper prototype actually worked as well as it did on paper.

Source criticism
Many sources referenced in this thesis are relatively new, while some are old. Despite being old, some of the concepts from the old sources are still relevant today.

One such example is Gould et al.’s article Designing for usability: key principles and what designers think [12] written in 1985. Their three principles of design can be found (not literally, but figuratively) in later works, such as Carolyn Snyder’s Paper Prototyping [23] which was released in 2003.

Polson et al.’s [20] and Rick Spencer’s [24] articles on cognitive walkthrough describe a method which is still usable and feels relevant today.

Societal implications
The case management system is today in use in several government agencies in Sweden. While the timeline’s primary purpose is to help managers gain an overview of cases, it could also save time and money. As the timeline displays how much time has elapsed between events, money and time could potentially be saved due to processes becoming more efficient in government agencies. For instance, it may be possible to see that at one certain point in all cases nothing happens for a while. By identifying situations like these, an investigation could be launched to determine if it is possible to cut the time needed for the process and therefore save money.

CONCLUSIONS
A timeline has been implemented as a part of a case management system at Ida Infront. By doing a usability study of the paper prototype, an initial implementation of the timeline could be developed. One of the issues that was noted during the usability study was the fact that no events occurred on a lot of the days. In order to solve this issue, events were instead displayed in a non-linear fashion, and the amount of white space between the events indicated how much time had passed between them. After evaluating the timeline by doing a streamlined cognitive walkthrough, several ideas that could aid usability and add interaction were noted. One such interaction was the ability to group events by weeks, months and years.

Future work
- Investigate which frameworks might be suitable for visualizing a timeline
- Test a design where the timeline is responsive, meaning that on smaller screens it will appear vertically while on larger screens it will appear horizontally.
REFERENCES


APPENDIX 1

Requirement specification
This list has been translated from Swedish but it should be accurate.

- The timeline shall display what has happened in a case in the case management system
- The timeline shall consist of events shown in chronological order
- The timeline shall support different types of events
  - Every type of event should have its own collection of information
  - The user should be able to differentiate different types of events clearly
  - It should be possible to extend the timeline with new kinds of events without changing any code
- The timeline shall fit the graphical profile of iipax for web solutions
- The user should be able to change the time interval displayed, as well as the data density
## APPENDIX 2

### Prototyping tasks

Following are tasks that were used when testing the prototype. These tasks are based on an internal description of what the timeline should be capable of.

An expert in this context is someone who has experience with the case management system. “Time for expert” is a rough approximation of how much time an expert may spend when doing that particular task.

<table>
<thead>
<tr>
<th>Task #1, Which documents has the status ‘incoming’?</th>
<th>Task #2, What happened?</th>
<th>Task #3, Acquire an overview of a case</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Goal:</strong> Find out which documents are marked as ‘incoming’.</td>
<td><strong>Goal:</strong> Determine what happened during a specific week.</td>
<td><strong>Goal:</strong> Gain an impression on how a case is progressing.</td>
</tr>
<tr>
<td><strong>Inputs:</strong> —</td>
<td><strong>Inputs:</strong> The date to look for.</td>
<td><strong>Inputs:</strong> A process flowchart, indicating which steps the case has to go through in order to be considered complete</td>
</tr>
<tr>
<td><strong>Assumptions:</strong> • There exists documents which are marked as ‘incoming’.</td>
<td><strong>Assumptions:</strong> —</td>
<td><strong>Assumptions:</strong> —</td>
</tr>
</tbody>
</table>
| **Steps:** • Toggle automatic events off  
  • Toggle journal events on  
  • Identify all journal events which are incoming | **Steps:** • Toggle on all events  
  • Pan until the the chosen date is visible  
  • Survey all events which happened on that specific date | **Steps:** • Toggle all events so that they show up on the timeline  
  • Pan all the way to the end until there are no more events  
  • Assess, based on the latest events and the process flowchart, how far along the case has gone |
| **Time for expert:** 5 minutes | **Time for expert:** 5 minutes | **Time for expert:** 15 minutes |
| **Instructions for user:** As a manager at the government agency, you are interested in finding out which documents have arrived during the time a case has been active. | **Instructions for user:** As an employee at the government agency, you are tasked with finding out what events that took place during a specific week. | **Instructions for user:** As a manager at the government agency, you are interested in finding out which documents have arrived during the time a case has been active. In order to aid you, you have a process flowchart which indicates all steps that the case needs to go through in order to be considered complete. Estimate how far the case has progressed. |
| **Notes:** — | **Notes:** — | **Notes:** — |
Figure 4: © 2008 IEEE. A system developed by Elmqvist et al. [10] when researching stacked scatterplots is shown. Displayed is the transition between two of the dimensions.
På svenska

Detta dokument hålls tillgängligt på Internet – eller dess framtida ersättare – under en längre tid från publiceringsdatum under förutsättning att inga extraordinära omständigheter uppstår.

Tillgång till dokumentet innebär tillstånd för var och en att läsa, ladda ner, skriva ut enstaka kopior för enskilt bruk och att använda det oförändrat för ickekommersiell forskning och för undervisning. Överföring av upphovsrätten vid en senare tidpunkt kan inte upphäva detta tillstånd. All annan användning av dokumentet kräver upphovsmannens medgivande. För att garantera äktheten, säkerheten och tillgängligheten finns det lösningar av teknisk och administrativ art.

Upphovsmannens ideella rätt innefattar rätt att bli nämnt som upphovsman i den omfattning som god sed kräver vid användning av dokumentet på ovan beskrivna sätt samt skydd mot att dokumentet ändras eller presenteras i sådan form eller i sådant sammanhang som är kränkande för upphovsmannens litterära eller konstnärliga anseende eller egenart.

För ytterligare information om Linköping University Electronic Press se förlagets hemsida http://www.ep.liu.se/

In English

The publishers will keep this document online on the Internet - or its possible replacement - for a considerable time from the date of publication barring exceptional circumstances.

The online availability of the document implies a permanent permission for anyone to read, to download, to print out single copies for your own use and to use it unchanged for any non-commercial research and educational purpose. Subsequent transfers of copyright cannot revoke this permission. All other uses of the document are conditional on the consent of the copyright owner. The publisher has taken technical and administrative measures to assure authenticity, security and accessibility.

According to intellectual property law the author has the right to be mentioned when his/her work is accessed as described above and to be protected against infringement.

For additional information about the Linköping University Electronic Press and its procedures for publication and for assurance of document integrity, please refer to its WWW home page: http://www.ep.liu.se/

© Tobias Jansson