New Functions in a Development Tool Chain - a User Centered Conceptual Study

Louise Relkman

Master's Thesis

Department of Design Sciences
Lund University
ISRN:LUTMDN/TMAT-5036-SE

EAT 2010

Abstract

The main purpose with this master thesis was to give suggestions for new functions in a tool chain and how they should be designed. The tool chain is used at the company The Astonishing Tribe (TAT) for developing user interfaces (UI:s).

The goal with the new functions was to achieve further improvements of the exchange of information between the persons in the development teams. An investigation of the work process was performed, where the information exchange got mapped out. Areas where new functions could make the work flow more efficient also got defined. One example of such an area is communication around UI prototypes when there is a geographical distance between the members in the development teams.

Usability goals for the suggested functions were defined based on the findings. Some examples are that the user's should experience it as easier to create narration to the prototypes and that the functions should not disturb the experience of the UI.

In order to meet the usability goals low fidelity prototypes were developed. They got evaluated with the use of a focus group and thereafter requirements for the new suggested functions were specified. There should for example be functions for adding comments at the sides of the actual UI in the prototypes, as well as functions for recording a flow of events. High fidelity prototypes then got developed, evaluated and improved in an iterative process. The final evaluation consisted of usability tests, which result led to further recommended improvements. In parallel an investigation concerning the functions placement in the tool chain was executed, as well as a theory collection. The theory collection resulted in information about different methods for adding information in prototypes, as well as a theoretical description of some of the different methods used in the master thesis.

Sammanfattning

Det huvudsakliga syftet med detta examensarbete var att ge förslag på nya funktioner i en verktygskedja och hur funktionerna bör utformas. Verktygskedjan används på företaget The Astonishing Tribe (TAT) för att utveckla användargränssnitt (UI:s).

Målet med de nya funktionerna var att förbättra informationsutbytet ytterligare mellan personerna i utvecklingsteamen. En undersökning av arbetsprocessen utfördes, där informationsutbytet kartlades. Områden där nya funktioner skulle kunna göra arbetsflödet mer effektivt definierades. Ett exempel på ett sådant område är kommunikation kring UI prototyper när det finns ett geografiskt avstånd mellan medlemmarna i utvecklingsteamen.

Användbarhetsmål för de föreslagna funktionerna definierades baserat på resultaten. Några exempel är att användaren ska uppleva det som lättare att skapa förklarande information till prototyper samt att funktionerna inte ska störa upplevelsen av UI:t.

För att möta användbarhetsmålen utvecklades low-fidelity-prototyper. De utvärderades i en fokusgrupp och därefter specificerades krav för de nya funktionerna. Det bör till exempel finnas funktioner för att lägga till kommentarer på sidorna av själva UI:t i prototyperna, samt funktioner för att spela in ett flöde av händelser. Därefter utvecklades, utvärderades och förbättrades high-fidelity-prototyper i en iterativ process. Den slutgiltiga utvärderingen bestod av användbarhetstester vars resultat ledde till ytterligare rekommenderade förbättringar. Parallellt gjordes en utredning angående funktionernas placering i verktygskedjan, liksom en teoriinsamling. Teoriinsamlingen resulterade i information om olika metoder för att lägga till information i prototyper, samt en teoretisk beskrivning av några av de olika metoder som användes i examensarbetet.

Preface

This master's thesis, "New functions in a development tool chain –a user centered conceptual study", was a compulsory part of the Master of Science in Engineering education in Information and Communication Engineering Technologies at Lund's University's Faculty of Engineering. The work was carried out at the company The Astonishing Tribe (TAT) in Malmö during the summer and fall of 2009.

I would like to thank the following persons that have helped me throughout the development of the master's thesis:

The persons at TAT who participated in interviews, focus group, usability tests and meetings. A special thank you to Dan Gärdenfors and Henrik Sandström for valuable tips and feedback throughout the work process.

My supervisor at the Department of Design Sciences at Lund's University, Mattias Wallergård, for the guidance during the writing of this report.

My family and friends, for all of their loving support and encouragement throughout my studies. Glenn, thank you for always standing by me.

Table of contents

1. Introduction

1.1 Background

1.1.1 The company

This master's thesis was carried out at The Astonishing Tribe (TAT), which is a graphical software company. They provide products and services for the creation of user interfaces (UI:s). The customers are mainly mobile phones companies.

A step in the development of user interfaces is to create prototypes, which at TAT is done in tool chain where some of the tools are developed in-house.

1.1.2 The tools

TAT uses a UI framework named TAT Cascades. It is a toolkit which is platform-independent and where the markup language TML (TAT Markup Language) is used to build the UI design. A TML file is a XML (eXtensible Markup Language) file with special tags. According to TAT [16] the name TML is used when the XML describes a UI with TAT Cascades.

The focus when working with TAT Cascades is according to TAT [16] the appearance, expression and usability of the interface. Since mock-up data can be used, the development can move forward without the underlying data services. These can instead get developed simultaneously with the interface.

The development environment for TAT Cascades is called TAT Motion Lab, see figure 1. It supports TML coding in a code view and visual editing in a design view. When simulating the result a viewer is used. There is a standard viewer in TAT Motion Lab as well as different viewers for different devices (mobile devices).

The prototypes developed in TAT Motion Lab play a central part in the collaboration between the persons with different work roles in the development teams.

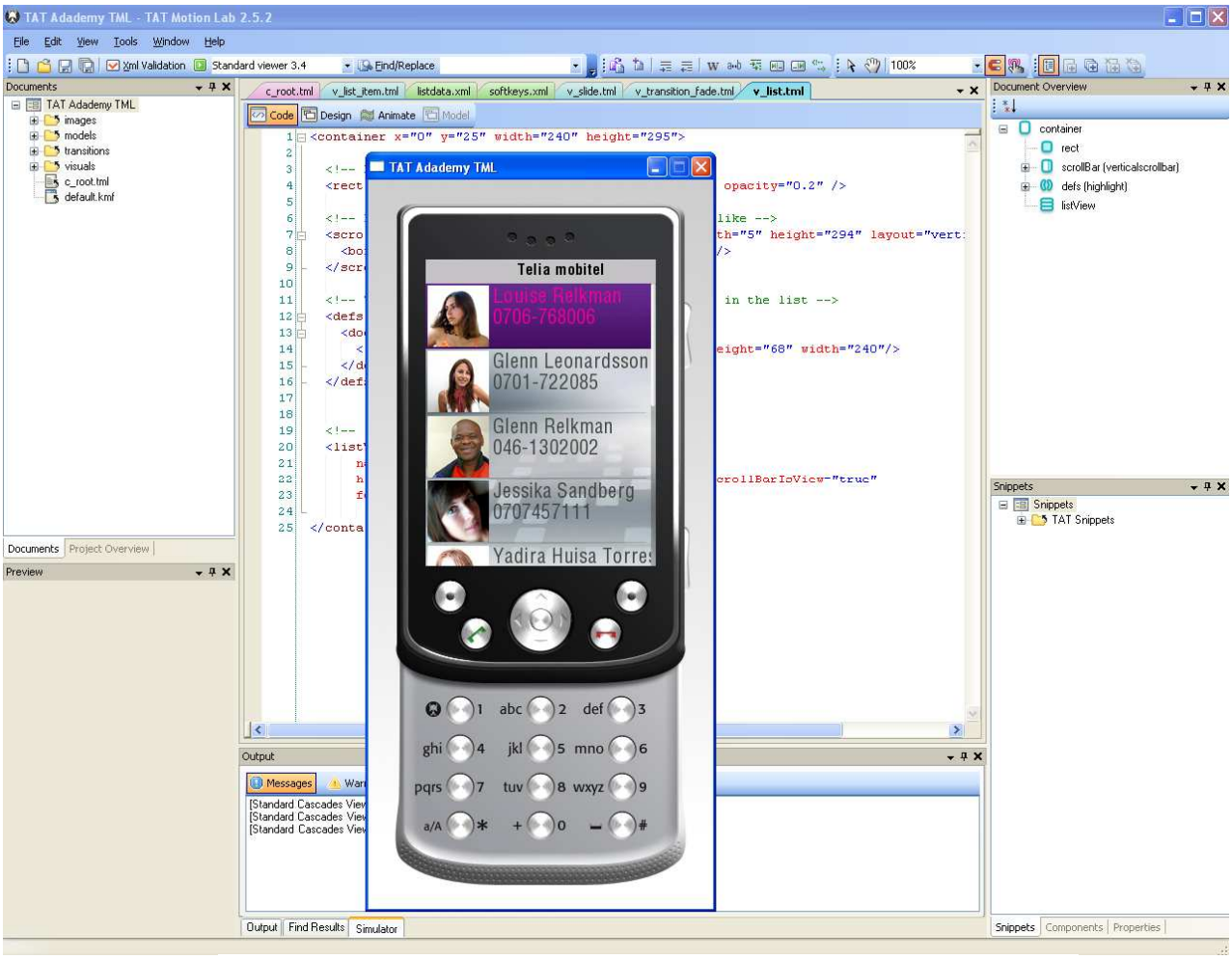

Figure 1: TAT Motion Lab in “Code view” with the simulation running.

1.2 Purpose

The purpose with the master’s thesis was to give suggestions for new functions and their design in the development tool chain. The goal with the functions was to further improve the work flow when the members of the developments teams exchange information about the prototypes developed in TAT Motion Lab.

1.3 Problem definition

The following problem definitions were in focus in the thesis work:

- What areas of the work flow when developing UI prototypes have potential to be improved?
- Which aspects are important to take in consideration when designing the functions?
- Which functions should be added to improve the work flow?
- How should the functions be designed?
- Where in the tool chain should the suggested functions be added?

1.4 Disposition

The report contains the following sections:

Theory
The section includes the result from the theory collection used in the different steps in the thesis work. It includes theory about annotation, narration and usability principles. It also includes theories behind some of the methods used in the thesis work.

Annotation and narration in other programs
Functionalities for annotation and narration in other programs were mapped out. This section contains information about some of them.

Method
The methods used in the different steps of the thesis work are described in this section.

Result
The section contains results from the different steps in the thesis work.

Conclusion
The problem definitions are discussed and answered in this section.

Discussion
Includes an evaluation of the results and if the problem definition have been answered. Also contains a discussion about future development possibilities.

References
Contains references to the sources used in the thesis work.

Appendix
The section contains interview questions, information and questions asked in the final usability tests as well as the data collected during the tests.

2. Theoretical framework

2.1 Narration and Annotation

In the initial task description, suggestions about functionality for narration and annotation were mentioned. In order to get more knowledge and inspiration about things to consider when adding functions for this, a theory search about the subject was conducted.

2.1.2 Narration

Narration is in a dictionary [2] defined as describing text or speech; something that is used in order to inform. In the article *Narrativ multimedia* (2000), Funseth [3] describes narration as something that takes place around an action or around an event.

Linear and interactive narration
Narration can, according to Funseth [3], be divided into two categories: linear and interactive. In the interactive narration the user is a participant in the story, while he/she is a spectator in the linear category. The user participates in the story in the interactive narration and has control over the course of events to some extent. A difficulty with the interactive narration is to find the right balance between giving the user enough control and at the same time keep the narrative context with an engaging story.

The structure of narrative programs
The story in linear and interactive narration is something with a beginning, middle and end. The interactive structure has an added second layer, which can have different degree of complexity. The different degrees of complexity decide how much consequence different choices cause. The following different structures of narrative programs are described by Funseth [3]:

Cul-de-sac
Cul-de-sac is described as a linear structure that allows choices between alternative scenes, with a starting point in the story's main track. This is often used in narrative training and education programs. There are variants where the user can make a pause in the story and explore an optional location.

Arena
Is described as a variant of cul-de-sac, where a certain sequence in the story can be shown in many different ways. This makes it possible for the user to choose between alternative events.

Hierarchic structure
In this structure the user's choices at different places in the story affect the story's direction. The number of choices increases exponential, which makes it inappropriate to use in more than a few parts of a story. The structure is often used to provide stories with alternative endings.

Parallel structure
This structure includes different versions of the story, which take place in parallel to each other. It allows the user to shift between the versions depending on different choices.

2.1.3 Annotation

A dictionary [2] describes annotation as a critical or explaining notation; a comment.

In the book *Sketching User Experiences* (2007), Buxton [1] writes that he believes annotation tools are important in design processes and that it is of big value to invest in the development of appropriate solutions. He further writes that it is important to apply the same skills to the design of the tools, as for the design of the products created with the tools.

Buxton [1] gives two examples of situations where annotation can be used in a design process. The first situation is when colleagues to a designer want to express their ideas based on one of the designer developed concept and this communications only can be efficient by drawing on the actual work. The second situation described is when a designer wants to enhance his/hers drawing with notes and graphical material without changing the original. Hence, annotation can be used for creating narration and giving feedback on a prototype.

Static and dynamic annotation
Annotation can, according to Buxton [1], be used for describing static media, for example a still photo, or dynamic media like a video. He further describes that the annotation itself can be static or dynamic.

The static annotation is defined as something that does not change over time. It can however in different ways indicate a movement.

Buxton [1] gives some examples of dynamic annotation. One of them are DVD videos with a so called "director's commentary". The comments change over time, they do not affect the original and they can be shut off. Another example Buxton mentions is when a sports commentator draws on video sequences and at the same time gives a description of the event with speech.

Important annotation characteristics
Buxton [1] was part of a team that developed a virtual video camera called *Boom Chameleon*, which makes it possible to view and annotate a 3D digital model. The user can walk around the object to see it from all sides on a screen.

When describing the annotation possibilities in the *Boom Chameleon*, Buxton [1] describes a couple of important general annotation characteristics:

- It should be easy to distinguish the original background image from the foreground annotation layer.
- The users must know that the annotations exist, where to find them and how to view them.

2.2 Usability

Usability is an important criterion when developing systems, products or adding new functionality. It is not something that can be added after a system has been developed; it has to permeate throughout the development process. When defining the functions and developing a concept in this master thesis, usability played a central part.

There is no unequivocal definition or rules that describe exactly how usability can be achieved. In this master thesis a selection of the many existing definitions and was used, which are presented in this section.

2.2.1 ISO standard

Usability is described in a standard defined by the International Organization for Standardization (ISO); the ISO standard ISO 9241-11 [6]. It is an international standard that has been developed in collaboration between several different parties and is part of a bigger system standard called “ISO 9241 - ergonomic requirements for office work with visual display terminals (VDTs)" [6]. The standard was completed in 1998 and defines usability as follows: “The 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.”

2.2.2 REAL

In his book, *Human-Computer Interaction: What every system developer should know* (1993), Löwgren [9] writes that usability is the result of relevance, efficiency, attitude and learnability. The four parameters are shortened to REAL and they are described to have the following significances:

Relevance
Corresponds to how well a system/product supports the users’ needs.

Efficiency
Decides how effective the users can perform their tasks.

Attitude
The parameter corresponds to the users' subjective feelings for the system/product.

Learnability
A system's/product's learnability imply how easy it is to learn from the beginning and to what extent the users remember it over time.

2.2.3 Eight golden rules of interface design

Shneiderman [15] has in his book, *Designing the User Interface* (1992), defined a set of rules to achieve and improve the usability and interaction design of an application. The eight rules are:

1. Strive for consistency
Similar situations should require consistent actions, the terminology should be identical and consistent commands should be employed.

2. Enable frequent users to use shortcuts
When using a system frequently, the user gets a desire to reduce the number of interactions and increase the pace of the interaction. Function keys and hidden commands are because of this very helpful to an expert user.

3. Offer informative feedback
Every user action should generate feedback in the interface. For frequent and minor actions the feedback can be modest, while infrequent and major actions should generate more substantial response.

4. Design dialog to yield closure
Sequences of actions should be organized into groups and include a beginning, a middle and an end. Feedback in the end of the actions offers satisfaction and a signal to prepare for the next group of actions.

5. Offer simple error handling
The user should not be able to make a serious error. The system should detect errors and offer simple mechanisms for handling them.

6. Permit easy reversal of actions
When the user knows that errors can be undone, the anxiety is relieved and exploration of unfamiliar options is encouraged. The reversibility can be a single action or a complete set of actions.

7. Support internal locus of control
The user should get the sense of being in control of the system and that it responds to the actions.

8. Reduce short-term memory load
The limitations in the human short-term memory requires that displays are kept simple, multiple page displays are merged together, window-motion frequency reduced and sufficient training times are assigned for sequences of actions.

2.3 Identifying needs and establishing requirements

In the book *Interaction design: beyond human-computer interaction* (2002), Preece et al. [12] describe the design process as an iterative process which includes cycles of various activities. One of the first activities in the process should involve an information gathering about the users' needs. In order to identify the needs and specify requirements, whether the project goal is to update an established system or develop a new product, an understanding for the users and their tasks is crucial.

2.3.1 Data gathering techniques

The purpose with data gathering is to collect relevant and appropriate data in order to produce a set of requirements. Each of the different techniques for data gathering will yield different kinds of data, which are useful in different situations [12]. The techniques can besides being used in requirement activities, in most cases also be used for evaluation.

Data is often divided into two categories: quantitative and qualitative. Hodgson [4] writes in the text *Quantitative and Qualitative data – getting it Straight* (2003) that quantitative data involves precise measurements and includes numbers. He [4] further describes qualitative data as something that concerns verbal descriptions of persons' experiences, opinions, feelings and knowledge. The current phase in the development process often determines what kind of data that is suitable, together with what kind of analysis that needs to be done [12]. The data gathering techniques used in this master's thesis are described below.

Interviews
Preece et al. [12] describe interviews as conversations with a purpose. Interviews are especially a good data gathering technique for exploration. They mostly generate qualitative data.

There are four main types of interviews: unstructured, structured, semi-structured and group interviews. The first three types are named after the degree of control the interviewer imposes on the conversation by following predetermined questions. The forth type involves a group of persons guided by an interviewer. The appropriate type of interview to use depends on the evaluation goals and the questions that need to be addressed. [12]

Unstructured interviews are like conversations, focused on a special topic. The included questions are open, which means the format and content of the answers are not predetermined. The participants can answer as fully or briefly as they

wish. A benefit with unstructured interviews is that they generate rich data; the data can however be time-consuming and difficult to analyze. [12]

Structured interviews include predetermined questions, similar to a questionnaire. They are useful when the goals of the studies are clear and when specific questions can be identified. The questions often require precise answers and the same questions are used for all participants. [12]

Semi-structured interviews combine features from unstructured and structured interviews. The interviewer has a set of preplanned questions and encourages the participants to express ideas and provide relevant information. [12]

Focus groups
A focus group is a type of group interview. They are a good alternative to traditional interviews in order to collect data from several viewpoints. They encourage contact between different stakeholders and generate qualitative data. [12]

Focus groups normally involve three to ten persons and the participants should be selected to create a representative sample of typical users. The facilitator has an agenda to guide the discussion, but unexpected issues are followed when raised. The facilitator guides the discussion and the session is usually recorded for later analysis. Focus groups provide quick results, but the facilitator must be skillful so no time is wasted on irrelevant issues. [12]

2.3.2 Usability goals and requirements

In *Människa-datorinteraktion* (1997), Kindborg [7] writes that usability goals can be abstract or concrete. An example of an abstract goal is: “The system shall be easy to learn”, while the following is an example of a concrete goal: “The user shall be able to write and print a letter within an hour”.

Requirements can be used as a tool to fulfil defined usability goals. Requirements are, according to Preece et al [12], statements about an intended product that specifies what it should do or how it should perform. Requirements should be as specific and clear as possible. The level of abstraction of the requirements varies depending on the situation.

There are different kinds of requirements. Functional requirements capture what the product shall do, while non-functional requirements define the product’s performance. One kind of non-functional requirements is environmental requirements, which describes the circumstances where the product is expected to operate. [12]

2.4 Prototyping

When discussing ideas with stakeholders, prototypes are useful. They function as a communication tool among team members and as an effective way to test ideas. Different kinds of prototypes serve different purposes.

2.4.1 Low fidelity prototypes

Low fidelity prototypes do not look like the final product. Materials very different from the final product are often used, paper can for example be used to illustrate a software. Low fidelity prototypes are useful because of their simplicity; they can quickly be produced and modified. This is important in early stages of the development, when they should encourage exploration. They can be used for identifying requirements and for evaluating multiple design concepts. They have however limited usefulness for usability tests and navigation. [12]

2.4.2 High fidelity prototypes

In high fidelity prototypes materials that would be expected in the final product are used. High fidelity prototypes are often built with a software tool. They are useful for tests and for demonstrating ideas. The prototypes are however more time-consuming to create and not effective for requirements gathering. [12]

2.5 Usability testing

In his book, *Handbook of usability testing: how to plan, design and conduct effective tests* (1994), Rubin [14] describes four different kinds of usability tests; exploratory test, assessment test, validation test and comparison test. Exploratory tests are conducted early in the development cycle in order to examine and explore preliminary design concepts. In order to expand the findings of the exploratory tests, assessment tests can be conducted. This is usually done when the high-level design of the product has been established. Validations tests are carried out close to the release of the product, with the intention to certify the product's usability. To compare different design alternatives or make comparisons to competitor's products comparison tests can be conducted, often in conjunction with any of the other three tests. [14]

The most suitable test to use in this thesis work was exploratory testing since the purpose was to develop suggestions for requirements and design, which is an early part of the development process. A more thorough theoretical framework of exploratory testing is given in this section.

2.5.1 Exploratory test

The exploratory test is according to Rubin [14] conducted early in the development cycle, in a product's preliminary stages of definition and design. The user profile and usage profile should have been defined.

The objective with the test is to evaluate the effectiveness of preliminary high-level design concepts. Rubin [14] emphasizes the importance of the early analysis and research, because of the critical design decisions made at this stage. The product will, according to Rubin [14], have usability problems later if the project begins with wrong assumptions.

Test methodology
Rubin [14] describes that preliminary versions of the product's interface can be developed for evaluation by representative users. It could for example involve a prototype simulation or a mockup of the product that represent its basic layout, organization of functions and high-level operations. Prior to a working prototype, static representations or paper prototypes can be used.

Prototypes
Rubin [14] writes that it is not necessary to represent the entire functionality of the product. It is only necessary to show enough functionality to address the test objectives. When the user can move left and right in the prototype, but is limited in moving deeper, the prototype is referred to as a horizontal prototype. More functions can be necessary if the test objectives include vertical usability, in other words moving down several layers. Rubin [14] describes that both kinds of objectives can be achieved by creating a representation of all large functions and a vertical representation of a few of the functions.

Exploratory test are often conducted as comparison tests, where different prototypes are compared to each other. Rubin [14] writes that a reason for this is to prevent too early commitment to one single design and later on discover problems with it.

Test procedure
When carrying out an exploratory test the process often is informal and includes much interaction between the test monitor and participant. An exploration of the users' thought process is according to Rubin [14] an important part of the testing. The test monitor and participant can explore the product together. The test monitors role is to conduct an ongoing interrogation and/or encouraging the participant to "think aloud". The goal with the test is to discover confusing areas and to understand why the users perform in a certain way. This is done by collecting qualitative data. [14]

2.5.2 The test plan

Rubin [14] writes that the test plan is the foundation for the entire test. It serves as a blue print for the test, describes the required resources and gives the test its focal point. According to Rubin [14] the test plan should include the following sections:

- Purpose: A short description of the reason for performing the test, often from the organization's viewpoint.
- Problem statements: Describes the issues and questions that need to be resolved. The problem statements should be as precise, accurate, clear and measurable/observable as possible.
- User profile: Describes the end users of the product that shall be tested. Should include a description of the users' different characteristics.
- Method: A detailed description of how the test should be executed. Should provide an overview and description of the testing methods that shall be used.
- Task list: Describes the tasks that the test participants shall perform during the test. The tasks should be developed making it possible to answer the defined problem statement.
- Test environment: A description of the simulated environment and the required equipment.
- Test monitor role: Clarifies what the test monitor shall do during the tests. Not a necessary part of the test plan, it is primarily important when other persons shall be observing the tests.
- Evaluation measures: Provides an overview of the measure types that shall be collected during the test, both preference and performance.

3. Annotation and narration in other programs

An investigation of annotation and narration in other programs and areas were performed. Some of the findings, which also work as examples of different kinds of narration and annotation, are presented in this section.

3.1 Annotation in Microsoft Word

A commonly used annotation tool is the one in Microsoft Word. By clicking on a button that says “New comment”, alternatively choosing the option in a drop-down menu, the document’s right marginal expands and a comment box is generated in the expanded area. A dotted line connects the box to the currently selected area alternatively to the word closest to the current location of the cursor (see figure 2).

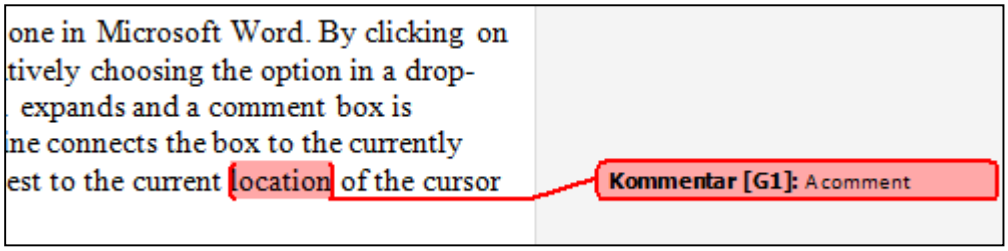

Figure 2: Annotation in Microsoft Word

3.2 Annotation in Microsoft Expression Blend 3

Microsoft Expression Blend 3 is a visual tool for designing and prototyping applications for Windows using the .NET framework or for the web using Silverlight. It has a user experience prototyping module called Sketchflow. In Sketchflow, according to Microsoft [10], it is possible to create prototypes by drawing flows, screens and states of an application UI. The Sketchflow player makes it possible to review the prototype through standard browsers.

In the Sketchflow player it is possible to add feedback and annotations on created prototypes. In order to add a comment, the user writes the comment in a feedback pane and it is then possible to add visual cues, for example draw on the prototype or highlight an area (see figure 3). The feedback can then be exported and saved as a small feedback file, which can be emailed to the receiver and integrated in the Sketchflow project.

The player contains a map of the different pages in the prototype. The pages that have annotation added to them are marked with a symbol.

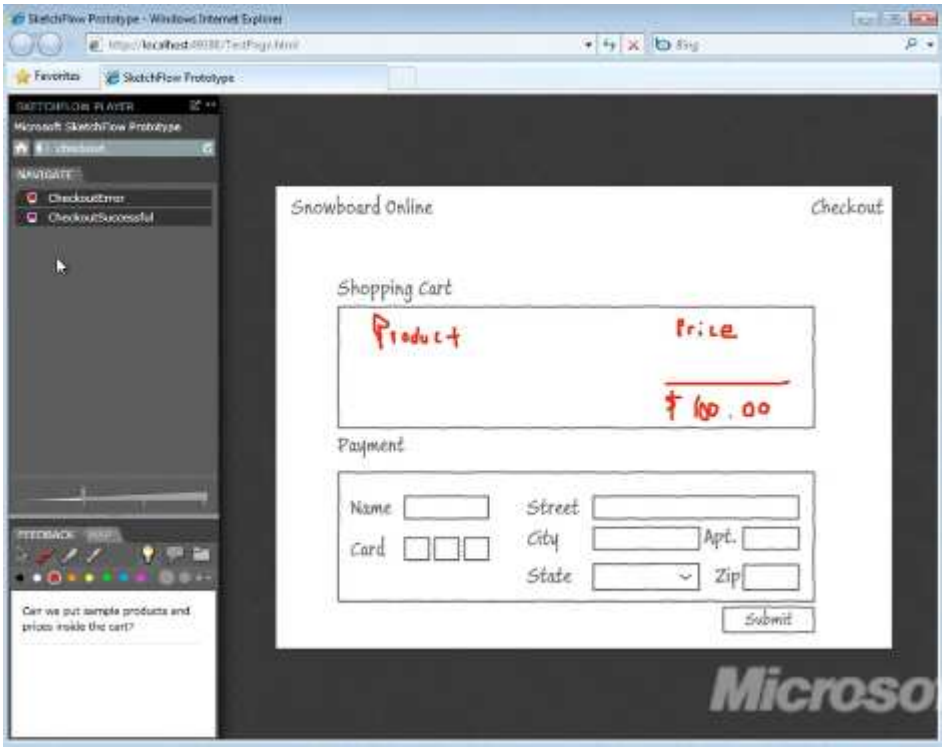

Figure 3: Added annotation in Microsoft's Sketchflow player [11]

3.3 Protonotes

Protonotes [13] is a tool for annotating on prototypes when developing HTML pages. By adding four lines of JavaScript code in the prototype, Protonotes is installed. A menu bar is shown when reviewing the prototype in the web browser and it is possible to add so called *Notes* (see figure 4).

A demo on Protonotes' [13] webpage shows that the menu bar contains one button for creating a new node, one button for hiding and one for showing the added notes and an index button. When clicking on the index button in the menu bar a list of all pages that have notes are displayed. The menu bar can be minimized by clicking on a button.

It is possible to place the added notes anywhere on the prototype by clicking on upper area and drag-and drop. Each note contains two checkboxes, *reviewed* and *completed*, as well as a button for removing the note. By dragging the left corner it is possible to resize the notes.

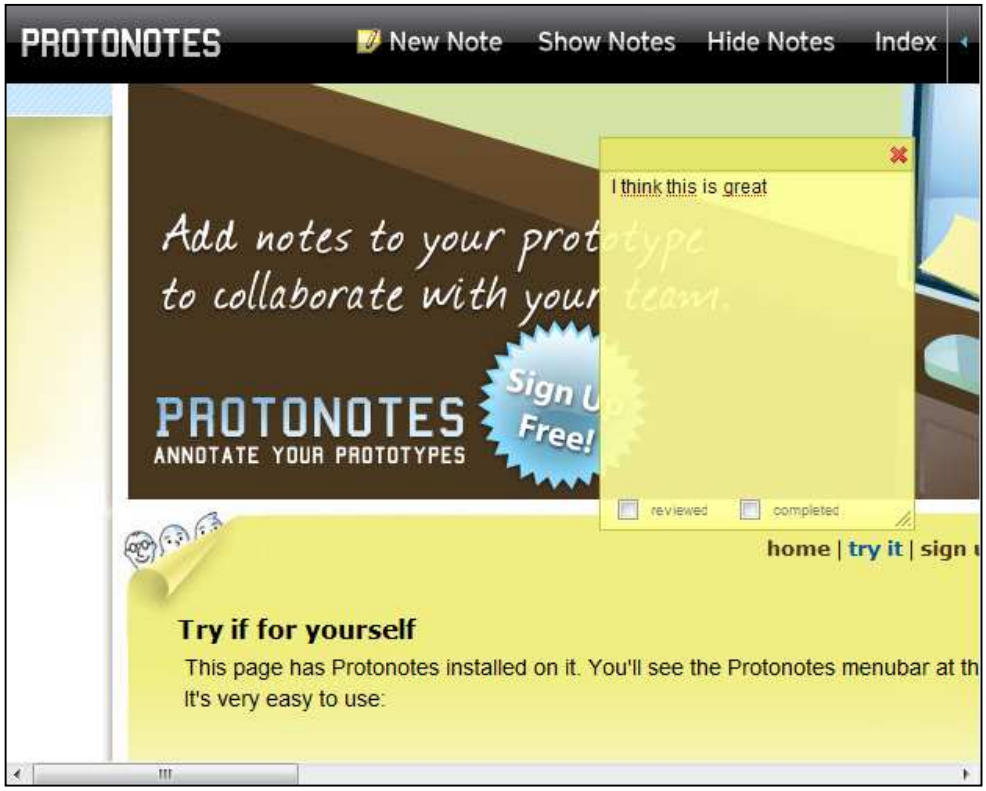

Figure 4: Protonotes' [13] annotation on a HTML page

3.4 YouTube's video clip comments

YouTube is a popular website for video sharing; twenty hours of video get uploaded on the website every minute, according the website [17]. YouTube has a function for annotating videos called video clip comments.

The annotation function makes it possible to add text or allowing the viewer to choose which the next scene should be. It is also possible to add links to other video clips, channels or search results. The user can decide where and when the comments should be shown in the video clip. [17]

To add a comment, the user clicks on a button to show the menu for adding buttons and then on a button to add a comment. The user then chooses among three different kinds of comments; speech bubble, note or focus. The speech bubble and the note are different shaped, while focus means that the comment only appears on a mouse over. It is possible to place the selected comment anywhere in the currently shown image and write text in it. A menu next to the comment makes it possible to change the kind of the comment, add a link, change the color of the comment or remove it. The user chooses where the comments should be displayed in the video clip, by pulling two levers next to the time line.

4. Method

The thesis work consisted of several steps in order to identify areas with potential improvement possibilities and develop a concept with solutions to achieve the improvements. In the first step, information about the development process was gathered and usability goals were defined. Thereafter an iterative process began, where different kinds of prototypes were produced. In parallel to this, an evaluation of the possibilities concerning where in the tool chain the functionality could be placed was performed.

The iterative process made it possible to evaluate different concepts, while each step also resulted in more advanced prototypes. The first prototypes were low fidelity, while the last step resulted in a high fidelity prototype.

4.1 Investigation of the work flow

Since TAT requested a new perspective and encouraged new ideas, the work flow during the development of prototypes was investigated and evaluated. This made it possible to get more input about the process and which functionalities that could improve it.

In order to evaluate the work flow, an information gathering was necessary. A first meeting was held with a concept designer and a project manager, where the project manager described the general development process at TAT.

To obtain more detailed information, semi-structured interviews were performed. The purpose with the interviews was to identify the tasks and interactions between the persons in the development teams. One aspect that was investigated was what information is exchanged and over what communication mediums.

The interview participants consisted of persons with different roles in the development process. The distribution of the participants was as follows (further details about the different roles can be found under “Result”):

- One interaction designer
- One interaction designer/graphic designer
- Three TML engineers
- One developer

Some of the interview questions concerned the work tasks in general, whilst others concerned what type of information the different persons’ work tasks are based on and what type of information get passed on to other team members (see Appendix A).

In order to obtain information about the communication with customers, additional meetings were held with two project leaders.

After the interviews and meetings, the answers got compiled and the development process and information flow got mapped out. Based on the result from the interviews and the theory collection, usability goals for the concept prototype developed in the master thesis were defined.

4.2 Low fidelity prototypes

4.2.1 Development

Based on the defined usability goals, low fidelity prototypes were developed. The prototypes consisted of several simple paper prototypes. They were based on screenshots of a prototype made in TAT Motion Lab with added simple graphics to demonstrate different possible functions.

The prototypes where intentionally created in a simple way and printed in black-and-white, in order to avoid distraction and to keep the focus on the functionality when presenting the ideas.

The goal with the prototypes was partly to discover problems and possibilities with the suggested solutions, but mostly to create a possibility to get input from persons from the different target groups. With the prototypes it was possible to discuss the defined functions, which made it easier to get concrete feedback.

4.2.2 Evaluation

When the prototypes had been constructed and printed, a focus group was put together. The goal was to get as much input as possible from all of the possible targets groups. An invitation was sent by email to several different persons. A group of four persons was put together, consisting of two TML engineers, one graphic/interaction designer and one project leader.

The focus group started with some questions about the participants' way of work. This got the discussion started and gave a confirmation to the results from the interviews. The next step was a brainstorming session, in order to gather ideas from the potential users, without affecting them with ideas that already had been defined. The goal was to keep their thoughts "as clean as possible".

When the participants' ideas came to an end, the discussion continued around the developed low fidelity prototypes. These were presented to the participants and it started a discussion around the different illustrated functions.

The session got recorded with a video camera, which made it possible to review and summarize the discussion concerning the different topics afterwards.

4.3 Specification of functions and requirements

In order to simplify the further development of prototypes, the different possible functions got prioritized based on the result from the focus group, the interviews and different gathered theories. Functional and non-functional requirements for the prototype developed in this master thesis got specified.

4.4 High fidelity prototypes

4.4.1 Development

Step 1: Video clips
The high fidelity prototypes got developed in two steps. In the first step several videos were made in Adobe Flash. They illustrated the use of the different functions; each video illustrated one of the functions. During the development issues concerning the functions' design got discovered, which led to further development and new videos.

Step 2: Interactive prototype
In the second step an interactive prototype was developed. The prototype was horizontal, which means that all functions were represented, but with limited depth. All functions were not available in all parts of the prototype.

Since the new functionality should be used by different persons in different ways, the prototype was altered to illustrate the various alternatives.

4.4.2 Evaluation

Step 1: Video clips
The result from the first step mostly served as a help to discover strengths and weaknesses with the suggested functions, they got evaluated by the author of this report. Some evaluation was also made during informal discussions with different persons at TAT.

Step 2: Interactive prototype
The interactive prototype was, compared to the first step, more fully evaluated with usability tests. The tests were of the kind by Rubin [14] described as explorative test. The tests were conducted according to the test plan below. A pilot test was conducted, which according to Rubin [14] is important in order to test the chosen methods.

Purpose
The purpose with the usability tests was to evaluate the functions shown in the prototype and the functions' design. The tests worked as verification for the

suggestions and the evaluation was based on the identified usability goals; how well the prototype's functionality and its design met the goals.

Problem statements

- Do the suggested functions meet the specified usability goals?
- Do any misunderstandings concerning the functionalities occur when the test participants explore the prototype?
- What are the participants' other opinions about the suggested functions and their design?

User profile

The test participants were persons that represented the different roles involved in the communication concerning the UI and therefore could benefit from the suggestions for new functionality.

Rubin [14] writes that research shows that four to five participants will expose 80 percent of the usability deficiencies when performing a less formal usability test, for example an exploratory test. In these tests four persons participated; two TML engineers, one interaction designer and one project leader from the sales department. Including the TML engineers and the interaction designer as participants made it possible to evaluate the impact on the internal communication, while the project leader made tests with external communication in focus possible.

Method

The background information for the prototype was explained to the test participants; the purpose with the prototype and the usability tests. The prototype was thereafter explored by the participant in close interaction with the test leader. Subsequent questions and an open discussion finished the tests; the questions were formulated in such a way that they gave an indication if the solution fulfilled the usability goals concerning the user's experiences. The questions were modified depending on what work role the test participant has. The initial information and the subsequent questions are listed in Appendix B.

Task list

Since the test was an exploratory test, it did not include a task list.

Test environment

In order to get best resolution with the correct settings, the prototype was displayed on the computer where it had been developed. The tests were performed in an office similar to the participant's daily working spaces. One difference however was that the office where the tests took place was empty, while the test participants often work with several other persons in the same room. Background noises had to be avoided to make it possible to capture the test session. The test participants reactions, comments and answers during the tests were captured with a video camera.

Evaluation measures

The following qualitative data was collected during the tests:

- The test participant’s spontaneous reactions and comments during the exploration of the prototype.
- The test participants’ comments after the exploration of the prototype.
- The test participants’ answers to the subsequent questions.

4.5 The suggested functions’ placement in the tool chain

In parallel with the development of prototypes, an investigation was made in order to give suggestions for where the functionality could be placed in the tool chain. The evaluation consisted of meetings with persons with expert knowledge in different areas. Discussions about the technical possibilities were also continuously held with different persons throughout the thesis work.

5. Results

5.1 Investigation of the work flow

The following section includes the result from the investigation of a general work flow at TAT when developing prototypes.

5.1.1 General work flow

Project leader Henrik Sandström [19] described a general work flow at TAT. Typically and simplified, the following steps are included:

Prototype phase

- Application planning
- Concept lead
- Interaction design
- Prototype workshop (to mock-up or heavy prototype)
- Test (early)
- Management (first evaluation)

Real application

- Graphic design (final design)
- Integration
- Application logics
- Data services
- Test (QA)

It is in the fourth step of the prototype phase, the prototype workshop, where the prototypes get developed and which is addressed in this master thesis.

5.1.2 Development teams - “four-leaf clovers”

The goal at TAT is according to Sandström [19] to work in tight teams during the Prototype workshop step. This means that the employees work in smaller groups consisting of persons with different professions and roles in the projects. The persons often sit close to each other in tight teams, in most cases in the same room.

The tight teams are referred to as "four-leaf clovers" by Sandström [19]. Figure 5 illustrates the different parts of a typical four-leaf clover.

The inner circle demonstrates the persons with different roles in a development team, the central part of the development process. Each role is normally represented by one or two persons. The different roles are described in detail in

the next section. The outer circle shows the persons from the customer with the same roles, who also are involved in the development process. [19]

The development teams often include one internal and one external project leader, in other words one project leader from TAT and one project leader from the customer. They often have different responsibility areas: the external project leader is often in charge of the development process, while the internal project leader mostly has responsibility for more internal tasks like the distribution of the workload. [19]

Another part of the development process is the so called product support. The product support consists of persons that handle the internal tools used in the development, for example Cascades. [19]

The described four-leaf clover is an ideal case which according to TAT works well, it is best practice. In some cases the persons in the inner circle of the four-leaf clover are located on a geographic distance from the other persons. In some cases the team members are not only represented by persons from TAT, but of a mix of persons from TAT and the customer. The TML engineer and developer can for example be employees from TAT, while the designers are employed at the customer. [19]

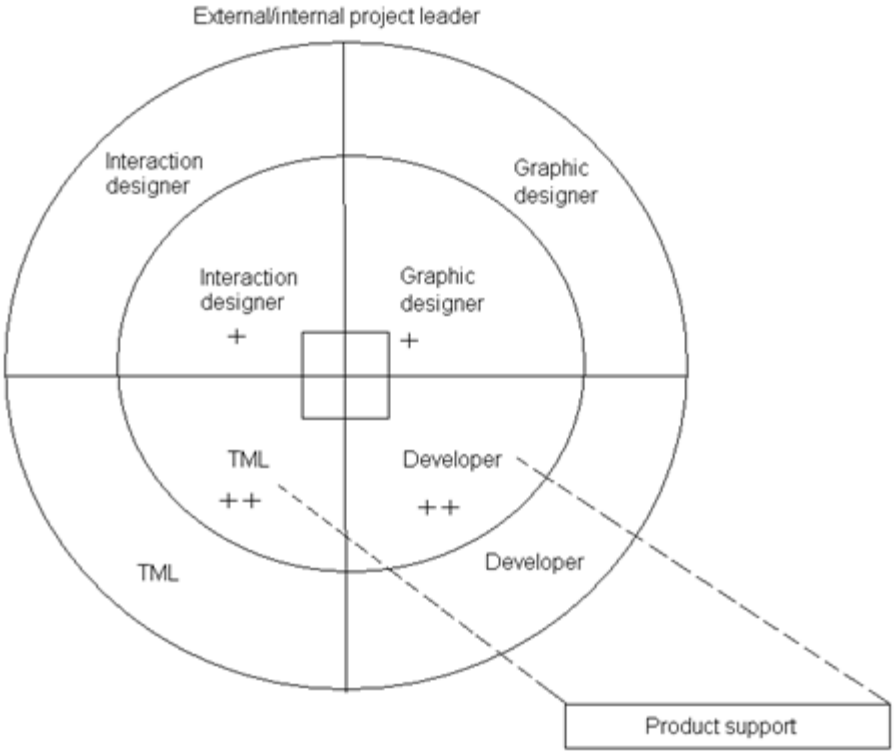

Figure 5: The four-leaf clover

5.1.3 Roles in the development teams

This section includes an explanation of the different roles in the development teams (four-leaf clovers).

Graphic designer
The graphic designers draw the graphics of the UI. They work in the programs Adobe Photoshop and Adobe Illustrator. When working with 3D-graphics they sometimes use Autodesk Maya, which is a program for 3D modeling.

Interaction designer
The interaction designers design and draw wireframes[1]. Wireframes show the flow, for example how different functions should work step-by-step. They work with paper sketches as well as the programs Adobe Illustrator and Adobe Photoshop.

Some persons have the competence for both graphic design and interaction design. This means that one person can represent both roles in some projects.

TML engineer
The persons that work as TML engineers write the UI with TML in TAT Motion Lab.

Developer
The developers create the application with C/C++ based on Cascades. They use Visual Studios when writing the code and the program Jira for bug- and issue tracking.

There is a fifth profession that plays a part in the development process, integrators. They integrate Cascades against different platforms. They work with integration against OpenGL and fine tuning for best performance. The integrators are however not a direct part of the prototype workshop.

5.1.4 Information flow in the prototype workshop

Figure 6 shows a graph over how the information in general is sent between the different roles in the four-leaf clover. It also shows what information is sent and in what format. The arrows in the graph show the direction of the information. The order of the different information exchanges is illustrated from the top and down. The lines with numbers to the left in the graph are guidelines to simplify discussions regarding the graph.

The graph is based on the information collected in the conducted interviews. The interaction designer and the graphic designer are illustrated together, since the roles in some cases are represented by one person. A more detailed description of the information sent between the roles is given in the section below the graph.

[1]Wireframes are illustrations of screens with indications of functionality and screen flow [5]

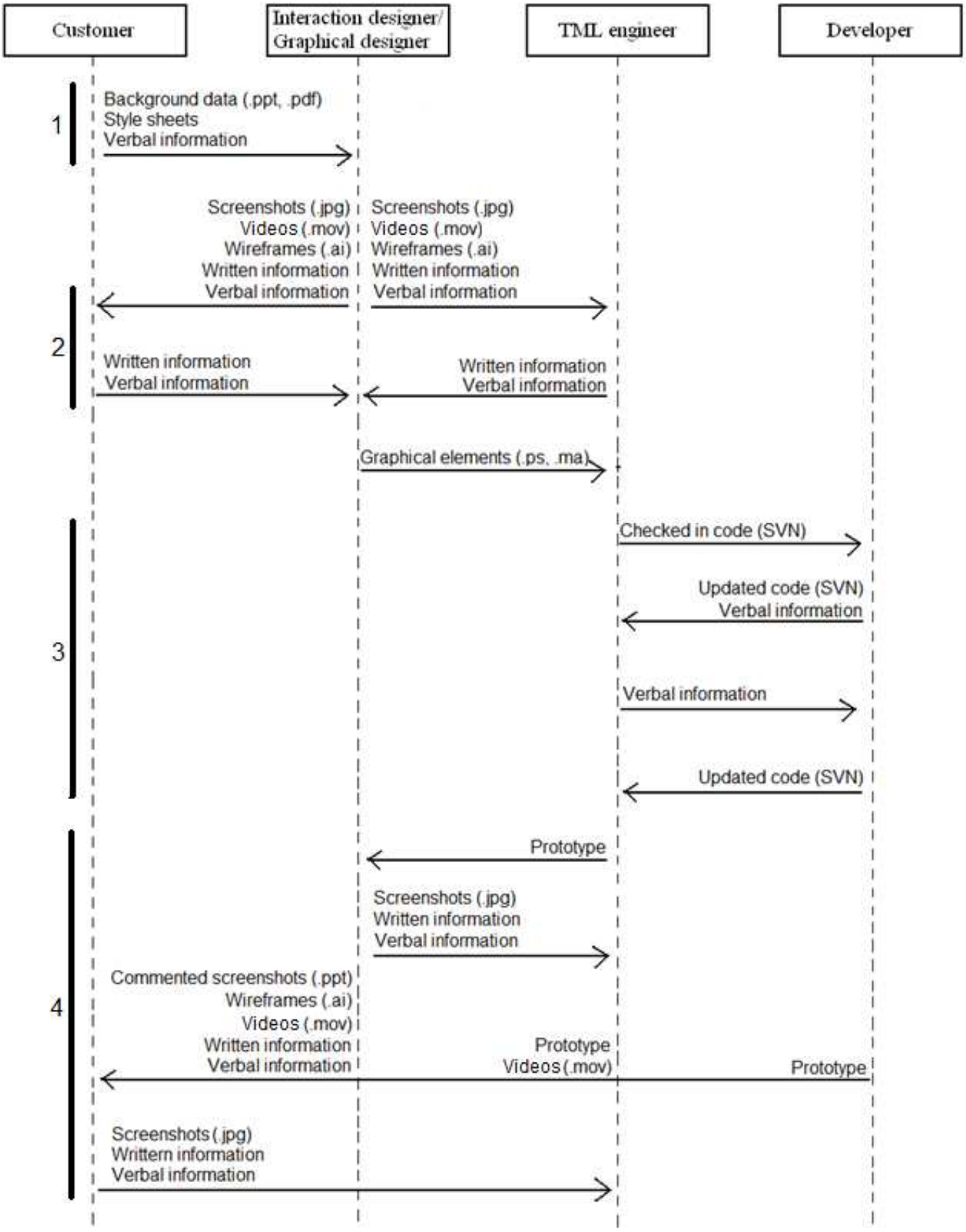

Figure 6: Information flow

Initial information from the customers to the designers (Line number 1 in figure 6)
The designers get information about the project during meetings (verbal information). Style sheets and background data (Microsoft PowerPoint, pdf) are sometimes sent to provide more detailed information. In some cases the work is based on previous projects.

Information between the designers and the TML engineers/customers (Line number 2 in figure 6)
The designers send wireframes (Adobe Illustrator) that describe flows and screenshots (jpg) which describe design of the different pages. In some cases videos are sent in order to describe transitions and animations. The information is discussed face-to-face, via email/Skype and during meetings.

The TML engineers give suggestions about how the graphical elements should be cut out by the designers. The elements, made in Adobe Photoshop (.ps) or in a few cases Autodesk Maya (.ma), are then sent to the TML engineers. In some cases the TML engineers cut out the graphical elements.

Information between the TML engineers and the developers (Line number 3 in figure 6)
When the TML engineers discover that there is a need for C programming, they contact a C programmer via face-to-face or email/Skype. The code is checked in via SVN, which makes it accessible for other persons in the development team. The developers create solutions and then check in the updated code.

The developers often show the TML engineer the solution on his/hers computer. If there is a need for changes, the updated solution is created and checked in via SVN.

The TML engineers also have contact with developers for Cascades and Kastor. If there is a problem in TAT Motion Lab, an issue is created in Jira. Jira is also used for bug reporting in bigger projects.

The prototype (Line number 4 in figure 6)
When the development of the prototype in TAT Motion Lab is finished, it is sent from the TML engineers to the designers. The designers compare the prototype in detail with the material made in Adobe Photoshop. They also compare it to possible videos of transitions and effects. Feedback is then given to the TML engineer. The feedback is in most cases provided face-to-face and in some cases with commented screenshots.

The prototype is also sent to the customer and discussed during meetings (face-to-face or via telephone). In order to provide more information, the prototype sometimes is complemented with commented screenshots. In some cases videos are created in the program Adobe After Effects or by recording the prototype with a video camera. The customers give feedback on the prototype during the meetings. Sometimes written comments also are given in combination with screenshots.

5.1.5 Information flow in the sales department

Minna Gedin [18], project leader in a team in the sales department, answered questions about their typical information flow. The team consists of engineers and designers, who design and develop pitches and prototypes which are delivered to customers. This can be done after special requests from customers or proactively. Proactive development means that the demos/prototypes are developed without any influence from the customers. [18]

The prototypes are typically shown on meetings, where a sales person from TAT demonstrates it. The sales persons have been provided with information about the prototype, in general via telephone or email. The feedback from the customers is then passed along to the project leader of the team via the sales persons and sometimes also via a translator. [18]

Sometimes information is delivered in pdf format to the customer, where the metaphor, design and interaction are explained. Sometimes concept videos or prototypes are sent. [18]

5.1.6 Areas with potential improvement possibilities

The initial task description included a suggestion for functionality for annotation, in order to make it possible for designers and customers to give feedback on the prototype in a more efficient way. According to the description this could shorten the development time and improve the quality of the given feedback even more. Narration was defined as a way to explain complex UI:s to the customer in a more efficient way.

In order to create an even more efficient work flow, the task description stated that the new functions should fulfill the following goals:

Functionality for annotation:

- Faster than the current solutions
- Easier to use than the current solutions
- Improve the quality of the given feedback
- Integrated in the existing tool chain

Functionality for narration:

- Faster than the current solutions
- Easier to use than the current solutions
- Decrease the risk for missed information in the UI
- Improve the quality of information sent to the customer
- Improve the quality of the given feedback
- Integrated in the existing tool chain

During the interviews possible areas where new functions could improve the work flow were identified. The areas were divided into internal and external

communication. Internal communication concerns the communication between the members of the four-leaf clover that is employees at TAT (interaction designer, graphic designer, TML engineer and developer). The external communication is the communication with persons from the customer.

Internal communication
The communication in the four-leaf clover works very well when the persons sit close to each other, according to the interview participants. It is experienced as easy to ask questions as well as give and receive feedback. Functionality that offers an easy way to make notes when receiving feedback face-to-face could possibly make the communication even more time-efficient.

The interviews showed that when there is a geographical distance between the persons, the communication could possibly be improved further by creating even more efficient and qualitative possibilities to give feedback and add information to the prototypes. The test participants experience it as complicated to explain features and changes via email or Skype, complementing meetings are often required. Other external programs that could make the information clearer, like recording tools, are often skipped since they are experienced as time consuming and laborious to use.

External communication
As for the internal communication, the interview results indicated that there is a possibility that the communication could be improved by creating even more efficient ways to give feedback and add information to the prototypes. Since geographical distances occur more often in the external communication, there is more room for improvement. A more efficient possibility to give feedback could also work as a way to make it easier for customers to prepare before meetings.

The interviews showed that adding information to the prototypes is an important area, since this decreases the risk of the receiver missing important parts of the UI, especially if it is not yet fully developed and information in it is missing. The interviews showed that it could be positive if the information could be in direct connection to the prototypes, in order to decrease the risk of it being separated from the prototype when it reaches the receiver.

Parts of the information flow that should be covered by the new suggested functions
In order to improve the existing work flow, it was important to be aware of which parts of the described information flow the new suggested functions should make more efficient. Based on this, the usability goals were defined and knowledge about what information that needs to be transferred was gathered. The parts are described below and the lines in bold style, marked with red color, in figure 7 show these parts of the information flow.

Annotation
The new suggested functionality for annotation should replace, alternatively complement, the following parts of the information exchange in the internal and external communication:

- Verbal feedback (meetings face-to-face and via telephone).
- Written feedback (via email and Skype)
- Screenshots with comments

The areas of use for the new functionality are:

- When designers are providing feedback on the prototypes to the TML engineers (internal communication).
- When customers are providing feedback on the prototypes to the designers/TML engineers (external communication).
- To encourage the receivers of the prototype to give feedback.
- To draw the receivers attention to wanted areas of the UI
- When TML engineers need to take notes, for example during verbal communication.

Narration
The suggested new functionality for narration should replace, alternatively complement, the following parts of the information exchange in the external communication:

- Prototype
- Videos that show scenarios (made in Adobe After Effects)
- Videos that show scenarios (recorded prototypes)
- Verbal feedback (meetings face-to-face and via telephone).
- Written feedback (via email and Skype)
- Screenshots with comments

The areas of use for the new functionality are:

- When a complex prototype needs to be explained to a customer
- When a prototype is not completely finished and there is a need for complementing information.

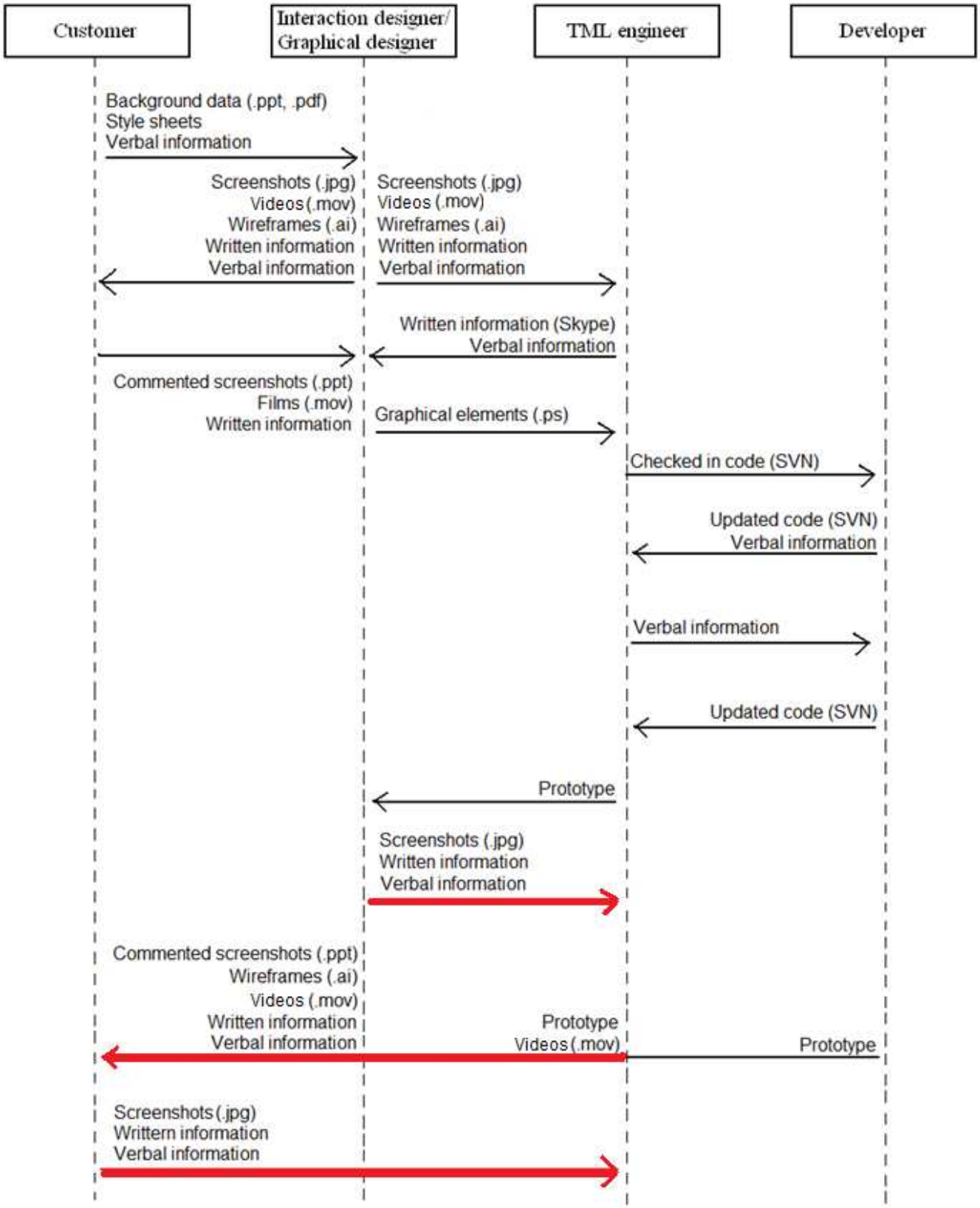

Figure 7: Information flow – the lines in bold and marked with red color show which parts the new suggested functionality potentially could make even more efficient.

5.1.7 Abstract usability goals and requirements

In order to suggest a solution with good usability, the solution should fulfill the four usability parameters defined in Löwgrens [9] REAL model. To accomplish this, abstract usability goals were defined based on the task description and the areas that were identified during the initial meetings and interviews.

Since the suggested new functionality is supposed to replace, alternatively complement, the currently used methods, a majority of the usability goals define some sort of improvement in comparison to them. The usability goals were defined as follows, categorized after the REAL parameters:

Abstract usability goals

Relevance
U1. The new functionality should replace or complement the currently used methods
U2. The functionality for feedback and narration should not disturb the designers' or customer's experience of the prototypes.

Efficiency
U3. The designers and customers should experience it as faster to give feedback on the prototypes.
U4. The designers should experience it as faster to create narration to the prototypes.

Attitude
U5. The designers and TML engineer should experience the given feedback as more qualitative.

Learnability
U6. The designers and customer should experience it as easier to give feedback on the prototypes.
U7. The designers and TML engineers should experience it as easier to create narration on the prototypes.
U8. The TML engineers should experience it as easier to understand the given feedback.

Abstract requirements
AR1. The new functionality should be integrated in the existing tool chain
AR2. The new functionality should improve the quality of the narration sent to the customer.
AR3. It should be possible to add information to the prototype that can increase the customer's understanding for it.
AR4. It should be possible to add information to the prototype that can decrease the risk for customers' to miss important parts of it.
AR5. It should be possible to add information to the prototype

5.2 Low fidelity prototypes

The low fidelity prototypes were used as a first step to meet the defined usability goals. They were however mainly created in order to get material which could be used as a base for further discussions with the users and for specifying more concrete goals and requirements.

5.2.1 Development

The new functionality's placement in relation to the prototype
When deciding where the suggested functionality should be placed in relation to the prototype, the choice to place it in direct connection with the actual prototype was made. The choice was crucial in order to meet the defined usability goals and requirements. The usability goals U3-U7 is for example partly met; the users should experience it as faster and easier to give feedback and create narration on the prototype as well as experiencing the given feedback as more qualitative.

If the feedback and narration could be created in direct connection with the prototype, the number of required steps would be decreased. By creating feedback in direct connection with the prototype, the chance for the user's to experience it as more qualitative would increase. This since there would not be a need for as vivid explanation of the location of the area which the feedback concerns. The close placement to the prototype would also follow requirement AR1, that the new functionality should be integrated in the existing tool chain.

When developing the low fidelity prototypes, the functionality, for above described reasons, was placed in the prototype. This made it possible to investigate the users' opinions of the placement when evaluating the prototypes.

Functions for annotation
The functionality for annotation would have several fields of use. It should function as a tool for providing feedback internally, mainly between the designers and the TML engineers. It should also work as a narration tool to provide more information when sending prototypes to the customer companies as well as a tool for the customer companies when giving feedback on the prototype.

A prerequisite for being able to provide feedback is a communication medium, either audio, images or text. When people in the development team provide feedback to each other face-to-face, it is often combined with finger pointing. This can be interpreted as being perceived as the easiest way to quickly explain what part of the prototype the comment concerns. The finger pointing shows a precise location of the prototype and makes it easier to explain. Face-to-face communication also provides an opportunity to quickly get feedback on the comment, hence ensuring that the recipient of the information understands the meaning of the comment.

A disadvantage with face-to-face communication is that the recipients do not receive the comment in writing, which increases the risk of the persons forgetting it. Another disadvantage is that the persons who want to provide feedback is

forced to interrupt their work and physically move to the recipient or, alternatively, await an appointment.

Alternatives to face-to-face communication, such as sending written information via Skype and email, have some disadvantages. One example is that it requires a text description of where and when the unwanted feature appears in the prototype.

Depending on the design, the functionality could also meet usability goal U5; that the given feedback should be experienced as more qualitative. Creating good conditions to give feedback could encourage the users to provide feedback.

It is important that the users perceive adding comments to be quick (usability goals U3 and U4) and easy (usability goals U6, U7 and U8). The current solutions include making screenshots with added comments, suggesting that there is a need to link the comments to a certain point in the prototype. This would also make a description of the location of the feature unnecessary. Since the new functionality should improve the usability compared to the currently used methods, it is important that the comments can be linked to a certain point in the prototype. It will also be perceived as more qualitative by the recipient if it clearly appears what part and section of the prototype the comment refers to.

The intention with the low fidelity prototypes with annotation functionality was to investigate the users' thoughts about annotation in general and about different variants and placements of the annotation. This made it possible to examine the limits for what the users experienced as disturbing (in order to follow usability goal U2) respective helpful.

Comments with arrows (see figure 8)
The first prototype described functionality with comments in boxes on the sides of the UI. The functionality would be reached via three buttons; one for creating a box, one for writing and one for adding arrows to the comments. The arrows would make it possible to connect a comment to a special point of the prototype.

Figure 8: Low-fidelity prototype -comments with arrows

Comments with status (see figure 9)
Another prototype showed functionality for comments with the possibility to set status. It also illustrated a function for showing the UI with or without the comments, which the user could choose by pressing a button. This functionality was included to avoid the risk of the comments disturbing the impression of the UI (usability goal U2).

Figure 9: Low-fidelity prototype -comments with status

Advanced comments (see figure 10)
One of the prototypes showed more advanced functionality; it showed functionality for removing and editing comments, possibility to create graphical annotation (drawing), signatures on each added comment and different filters when viewing the comments.

Figure 10: Low-fidelity prototype –advanced comments

Functions for recording flows
The text annotation could have other areas of use than to provide feedback on the prototype. Depending on the annotation format it could also be used to improve customers' understanding of the prototype, requirement AR3. There is however a need for more functionality other than annotation. An explanatory comment that is associated with a particular point in the prototype is not very helpful if the user never reaches a page where the point is located. If a UI for example is very complex, there is the risk that the user/customer cannot find all pages in the UI. Another example is if feedback is needed earlier in the development process, when the UI is not finished. This could cause missed pages or that the customer gets a false impression of the prototype. Making it possible to create a flow, a recording of clicks, could help demonstrating complex UI:s avoiding the receiver to miss important parts. Therefore, low fidelity prototypes that showed functions for recording flows were developed.

Flow recording (see figure 11)
The prototype that displayed flow recording showed functions for record, pause, play and stop. It also showed two buttons for capturing a still photo and adding voice annotation to the recording.

Figure 11: Low-fidelity prototype –flow recoding

Functions for annotation and recording flow
The functionality for recording a flow could also, depending on its design, become used to give feedback. Giving feedback with text annotation could be experienced as difficult when it comes to transitions and animations. A recorded flow, where for example a transition in shown, could give valuable information to the receiver.

A recorded flow would not work as feedback on its own, since the receiver would not understand exactly what the recorded flow concerned. It would therefore be necessary with added explanatory information in order to make the flow work as a feedback tool.

By combining functionality for flow recording with written annotation in the prototype, the receiver's understanding for the recorded flow could increase. The functionality would also remove the need for meetings to show and explain recorded flows. Some low fidelity prototypes showed a combination of functionality for recording flows and adding comments .

Flow recording with advanced comments (see figure 12)
The prototype showed functions for recording flows and adding advanced comments. It also displayed functionality for showing the clicks made during the recording.

Figure 12: Low-fidelity prototype –flow recoding with advanced comments

Flow recording with comments on the side (see figure 13)
This prototype showed functions for recording flows. When playing back the flow it would be possible to pause and add a comment to any point in the UI. The comment would be added in a list on the side. When clicking one of the comments in the list, a line to the comment would be displayed.

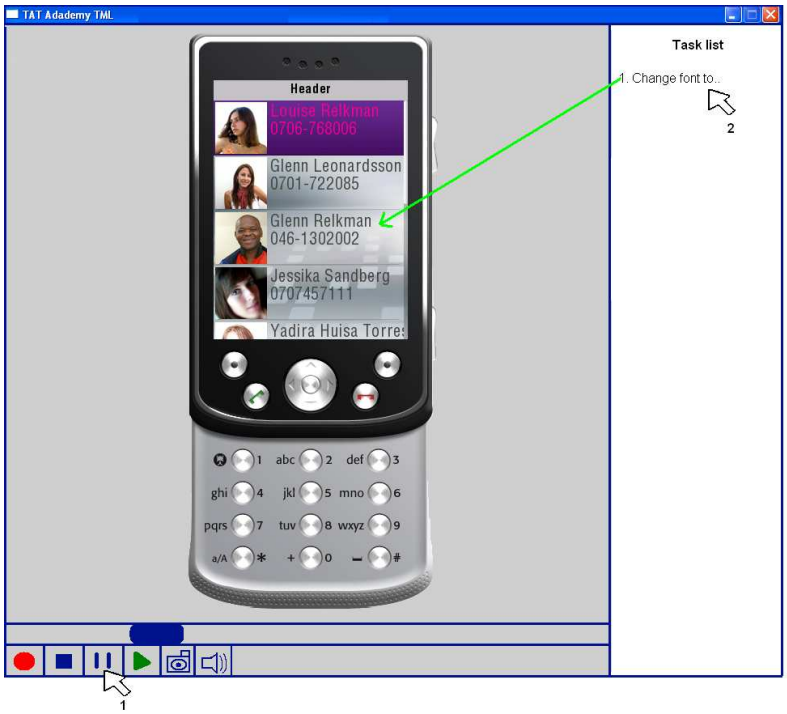

Figure 13: Low-fidelity prototype –flow recoding with comments on the side

Flow recording with comments in document (see figure 14)
This prototype was similar to the one described above (figure 13), with the difference that the comments would be placed in a separate document.

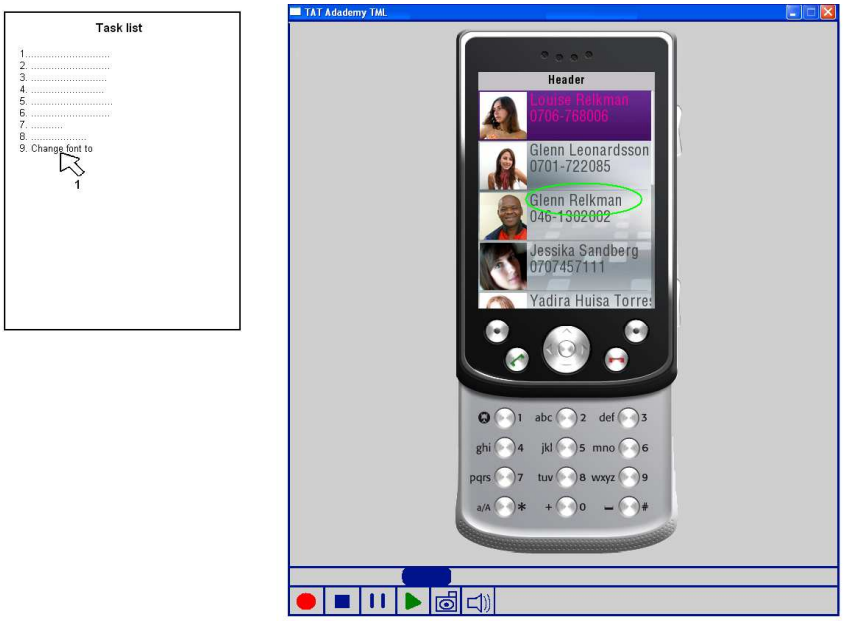

Figure 14: Low-fidelity prototype –flow recoding with comments in document

Flow recording with grid (see figure 15)
This prototype displayed functionality for recording a flow, with comments on the side of the prototype. A grid over the prototype would make it possible to add comments connected to different parts of the prototype. The comments displayed under the heading “1:1” in the task list would for example concern the area in the grid’s square 1:1.

Figure 15: Low-fidelity prototype –flow recoding with grid

5.2.2 Evaluation

The persons in the focus group verified that the work flow identified during the initial interviews was correct. The open discussion and the discussion based on the low fidelity prototypes resulted in examples where new functionality could improve the development process. It also resulted in ideas of new functionality in the tools and some technical recommendations.

Possible improvement areas
Areas in the work flow which could have room for improvement were discussed in the focus group. Communication over a geographical distance was one of the areas. The participants thought new functions that could complement, in some cases even replace, meetings could be beneficial. It could for example cut the time for setting up meetings, provide more direct communication and reduce the need for forwarding information.

Functions to complement written communication could, according to the participants, create time savings if it increased the understandability for the

received information and created an even faster way than the currently used methods to give feedback and information.

The participants also thought it could be beneficial if the functionality provided a complementing solution to communicate problems that appear on the mobile device but not on the PC and vice versa.

The participants discussed the available external programs that can be used for communicating around the prototype. They commented that the programs are expensive, slow and require different fire wall settings.

Brainstorming

When the participants started to discuss ideas, the first thing addressed was to make it possible to create comments directly in the prototype. One of the participants drew an example and the other participants agreed that it would be a good idea. The discussions that followed stayed focused on this, whereof the low fidelity prototypes were shown earlier than planned.

The low fidelity prototypes were met by positive reactions, a comment from one of the participant was: “It would be great”.

The participants thought it would be best to write the comments directly in the prototype, rather than generating them in an external document. They suggested combining it, to have comments as well as a list in the prototype.

The participants discussed the life length of the comments; they did not think it should last so long. They therefore did not find it necessary to have functions for different priorities and signatures.

The participants thought it was a good idea to record a flow. They suggested a function where a click on a certain point in the prototype should result in a recording of ten seconds before and ten seconds after the time of the click. They also thought it was a good idea to make voice annotation possible.

External programs for screen recording were discussed, the participants experience them as being tricky to launch and that they lose frames.

The participants thought the following issues should be addressed when defining a solution:

- How should features that only appears on the device but not on the PC, and vice versa, be communicated? How should it be explained?
- How should transitions, movements and animation be marked?
- How should things, for example a transition, that work too slowly be addressed?
- How should a change be handled, for example if a bug has been corrected? Should the same environment be executed again, with the same executable files, or should the whole environment be rebuilt?

5.2.3 Analysis

Could the suggested new functionality improve the work flow?
Adding the annotation functionality could make it easier to explain features when face-to-face communication is not possible, which also could save time, since there will not be a need for as many meetings. The functionality could also make direct communication between customers and developers possible, since information that in the current situation requires meetings, often via project leaders, could be sent directly in the prototype.

Making it possible to create annotations directly in the prototype could make it easier to describe opinions and giving feedback, in comparison to sending information via email or Skype, since the location in the prototype does not have to be described.

The functionality for recording a flow could provide a complementing possibility to communicate problems with the UI behaving differently on PC and on device, if the functionality also is included in the prototype for the device. This would make it possible for the customers and designers to for example record an experienced behavioral issue and send it to the TML engineers and developers. The functionality could be used to describe problems in general, which could make the problem solving more time efficient.

Refined functionalities and design
Since the low fidelity prototypes were met by positive reactions from the focus group participants, there was not a need for major changes; the possible gain with functionality for adding comments and recording a flow was confirmed. The opinions concerning the different versions of the functionality displayed in the low fidelity prototypes made it possible to refine the parts of the functionalities and their design.

The discussions in the focus group showed that the comments should be placed in the actual prototype and not be generated in an external document. It also showed that it would be positive with some sort of summary of the comments in the prototype.

One important aspect brought up during the focus group was the life length of the prototypes. The prototype will probably not be sent back and forward several times, but mainly be used to send information one time. For example; the prototype is sent by a TML engineer to a designer. The designer reviews the prototype and adds comments in order to give feedback and sends it back to the TML engineer. The TML engineer then make changes to meet the feedback and sends a new prototype to the designer. This makes it unnecessary to have functionalities for prioritizing the different comments, adding signatures and check boxes.

The participants' suggestion concerning the flow recording, that a click on a certain place should create a recording of ten seconds before and ten seconds after the time for the click, could be a good solution when giving feedback. The idea of

only using one single click gives the impression of the solution being time efficient. However, there is a risk that this would break one of Shneidermans [15] eight golden rules: Support internal locus of control. If the recording automatically starts to record ten seconds before a mouse click, the user could find it difficult to control exactly when the recording should start. There is also a risk that there is a need for longer and shorter recordings in different situations. If the recording time is set to a certain amount of time, this could force the receiver to watch unnecessary seconds of recording. Except from being time inefficient, it could also create confusion for the receiver of the prototype since the purpose of the recording can be experienced as unclear if unintentional events have been recorded.

The first issue that the participants thought should be addressed, how features that for example only appear on device should be communicated, could be solved with the flow recording. In order to explain the recording it could be combined with text or voice annotation. This functionality could also be used in order to explain transitions, movements and animations. Text annotation could preferably be added to the page just before a transition or movement.

How changes should be handled, which was the last issue the participants requested should be addressed, is connected with the issue concerning the life length of the prototype. Since the participants did not think that the prototype would be sent back and forward several times, the whole environment could be rebuilt again.

5.3 Requirement specification

The result from the evaluation of the low fidelity prototypes were used to specify more concrete requirements and usability goals for the suggested new functionalities in the concept prototype developed in this master thesis, in order to meet the previously defined abstract usability goals and requirements. The concrete requirements and usability goals are specified in this section.

Functional requirements

R1. It should be possible for the user to create text annotations connected to any point in the UI.

R2. The added text annotations should be possible to reach from an index.

R3. It should be possible to edit added text annotations.

R4. It should be possible to remove added text annotation.

R5. It should be possible for the user to add text annotations that are not connected to a point in the UI.

R6. It should be possible to view the UI without the annotations.

R7. It should be possible to record flows.

R8. It should be possible to add text annotations on recorded flows.

R9. It should be possible to add annotations in direct connection to the prototype

R10. It should be possible to view annotations in direct connection to the prototype

R11. It should be possible to record flow in the same window as the prototype.

R12. It should be possible to view flow in the same window as the prototype

R13. It should be possible to add annotation without starting another application.

R14. It should be possible to view the annotation without starting another application.

R15. It should be possible to record flow without starting another application.

R16. It should be possible to view flow without starting another application.

Non-functional requirements

Performance requirements
R17. The annotation should be easy to distinguish from the UI.

R18. The prototypes' image quality should be maintained in the recorded flow

R19. The annotation should not block the UI more than absolutely necessary

R20. It should be possible to use the annotation to supplement prototypes when information is missing

R21. The number of mouse clicks to create annotation should be as few as possible.

R22. The number of mouse clicks to view the annotation should be as few as possible.

R23. The number of mouse clicks to create flow should be as few as possible.

R24. The number of mouse clicks to view flow should be as few as possible

Environmental requirements
R25. The annotation function should be integrated in the existing tool chain (no external tools).

R26. The function for recording flows should be integrated in the existing tool chain (no external tools).

5.4 High fidelity prototypes

5.4.1 Development

Step 1: Video clips
In order to find solutions to meet the specified requirements, prototypes that illustrated the different functions were created in Adobe Flash. The prototypes were designed as video clips that illustrated imaginary use of the functions.

Since the videos worked as a middle step for the interactive prototype, all design decisions are described in the section about the interactive prototype. Some issues that occurred during the development of the videos are however described in this section.

One issue was to find a way to make added comments in a recorded flow visible. The comments have to be displayed in a way that makes it easy for the user to absorb the information. A solution to this was a function where the flow, when played, automatically pauses at the added comments. This makes it possible for the users to read the comments and pause the playback if necessary. The added comments are marked with a line in the time line (see figure 16).

Figure 16: Comment marked in the time line

Another issue concerned how the comments should follow movement in the UI. When the user for example scrolls in a list, how should added comments act? A solution to this was to make the comment follow the movement. For example when the point it is connected is moved up, the comment also moves up.

Step 2: Interactive prototype
The functions were designed in order to fulfill the previously specified usability goals, which were done by designing the functionalities following the requirements and the usability rules defined in the theoretical framework.

The UI in the high fidelity prototype was based on an actual UI made at TAT. Images from the actual UI were captured and some parts of its functionality were recreated in Adobe Flash. This made it possible to go to different pages in the prototype, as well as experiencing transitions and animations.

The high fidelity prototype includes all new suggested functions. The prototype was developed in order to make it possible to test both vertical and horizontal functionality. Rubin [14] describes this as a way to make it possible for the user to move left and right in the prototype, as well as moving down some layers in some of the functions. This means that all functions are represented in the prototype, but it is not possible to use all of them fully. They are implemented enough to make it possible to fulfill the test objectives. The functions are however in this section described in terms of how they are intended to work in a real implementation. The prototype is described by going through each part of the prototype. Each description contains an explanation of the design decisions made.

The size of the prototype
The area around the UI was expanded in order to create an area to place comments (see figure 17). This in order to follow requirement R19, that the annotation should not block the UI more than necessary. It is important that the users' experiences of the UI are kept intact. Another design decision made in order to follow the requirement was to connect the comment and the point in the UI with a simple line. In order to make the user understand what the meaning of the comment is, it is important that the area is not covered more than necessary. An arrow that was used in the previous prototype versions covered an unnecessary area around the point.

The annotations (comments)
In order to follow requirement R21 and R22, to reduce the number of needed mouse clicks to create an annotation, an annotation is added by right-clicking on the mouse at a point in the UI. When reviewing a UI, the user's attention is directed to the area where a possible problem is perceived. It would therefore be positive to make it possible to make a fast click on a point and add a comment. If the user first had to click on a special button to create a comment, he/she would have to move the attention and risking being disturbed.

When the user right-clicks, a comment box is generated automatically with a line connected to the chosen point in the UI. The idea is that the comment box should be generated at the side of the UI closest to the clicked point and at the first empty

place from the top. The cursor is activated inside the box, which makes instant writing possible (see figure 17).

All added comment boxes have an outline in the same color; a light blue. This gives a uniform design and makes them distinguishable from the UI, which follows requirement R17. The background color in the boxes is white, with a black font color. This gives a strong contrast, which increases the readability.

It is possible to change the content of the comments, by clicking in the boxes. It is also possible to remove comments, by clicking on the button with a cross which is located in the upper right corner of the comments. This follows the requirements R1, R3 and R4; that it should be possible to add, edit and delete added text annotations. It also follows Shneiderman's golden rules number five and six: offer simple error handling and permit easy reversal of actions.

The user can add a comment without connection to a certain point in the UI by clicking on the area outside the UI. This follows requirement R5.

The information in interactive prototype's comments was altered in order to adapt the prototype for different way of use and different users.

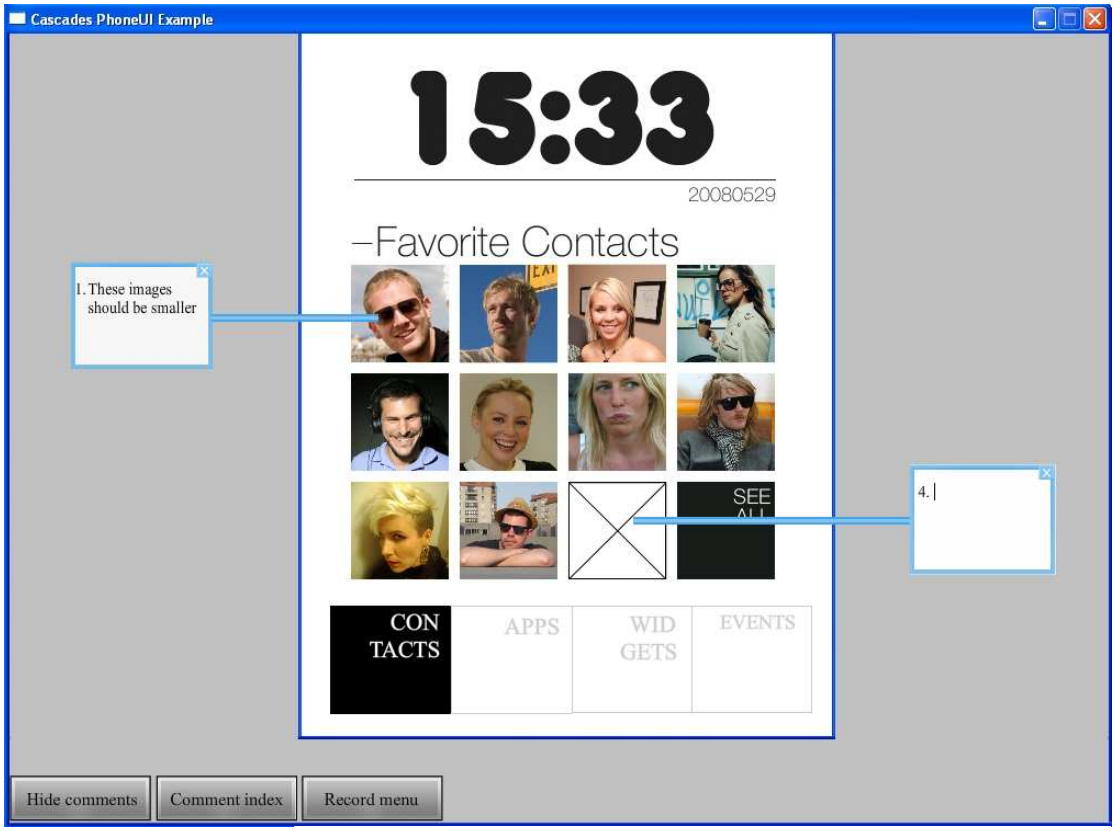

Figure 17: The prototype with comments shown.

The buttons
Three buttons at the bottom of the prototype contains some of the new suggested functions. The reason for using buttons instead of for example placing the functionality in a menu at the top is to reduce the users' short-term memory load (Shneiderman's eighth golden rule). The button is highlighted when the mouse rolls over, in order to offer informative feedback to the user (Shneiderman's third golden rule).

When pressing the button "Show comments", all added comments in the prototype is shown. When clicked, the text on the button changes to "Hide comments". Clicking the button when "Hide comments" is displayed will make the comments invisible again. This functionality follows requirement R6; that it should be possible to view the UI without the annotations.

The button "Comment index" displays a list of all added comments when clicked (see figure 18). This in order to follow requirement R2: The added text annotations should be able to reach from an index. When a comment is added or removed the list is updated.

When pressing the button "Record menu" a new menu appears (see figure 19). The menu contains functionality to record a flow, following requirement R7.

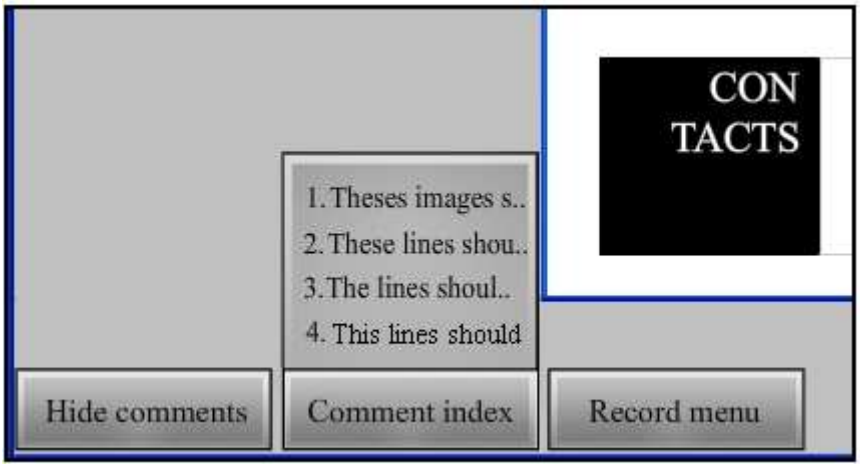

Figure 18: Detail of the prototype: the comment index

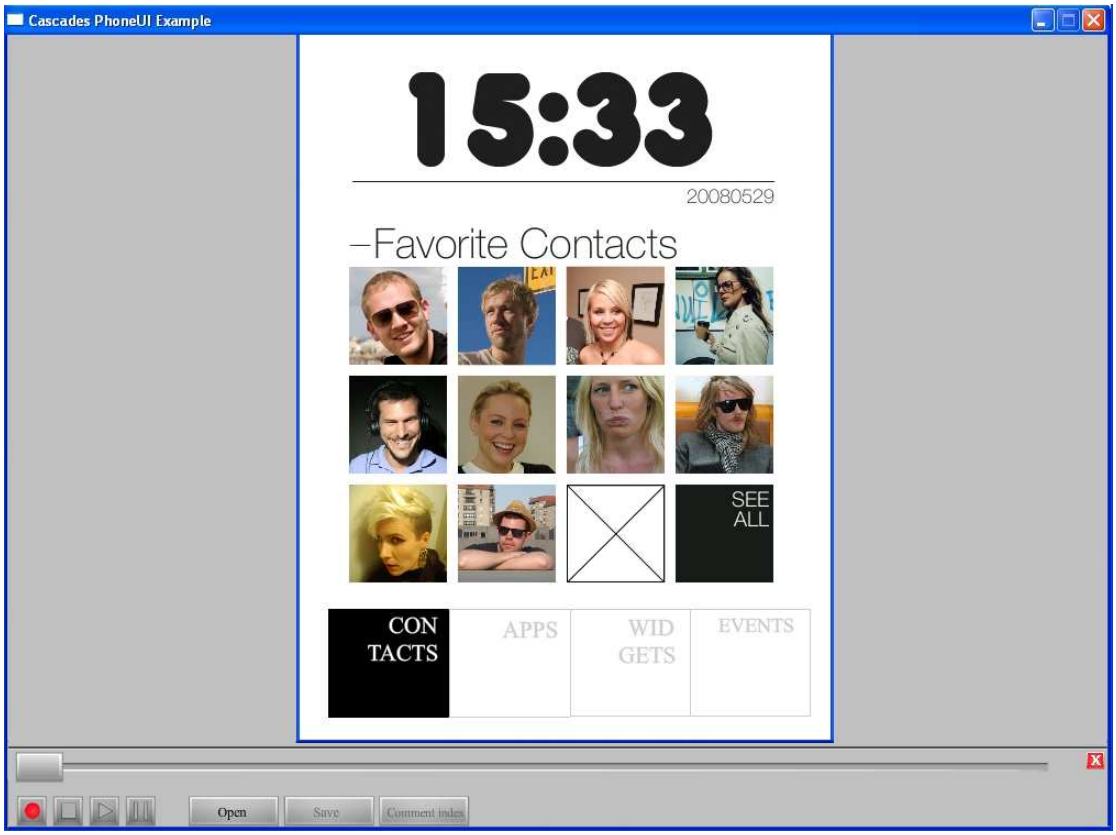

Figure 19: Prototype with the record menu opened

The record menu

The record menu opens when the user presses the button “Record menu” (see figure 20). The menu contains buttons with the following functionalities: record stop, play, pause, open, save, comment index and close. This will henceforth be called “record mode”, while having the menu closed will be called “normal mode”. The options in the menu make it possible to record or view a flow with a minimum of mouse clicks, according to the requirement R21 and R22.

Two of the buttons are active when opening the record menu; the record button and the button “Open”. The record button allows the user to record a flow, while the button “Open” makes it possible to open previously recorded flows. The buttons not active have a light grey color, in order to illustrate that they are not possible to press.

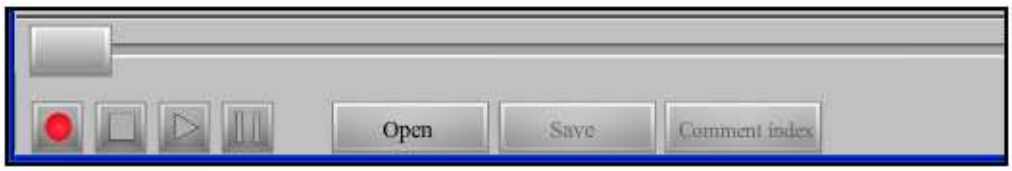

Figure 20: Detail of the prototype: the record menu

When pressing the record button, a lever in the time lime starts to move forward. This and a red text “Rec” in the upper right corner indicate that the recording has started (see figure 21). The buttons for stop and save are displayed as active. This makes it possible to stop or pause the recording, as well as saving the recorded flow.

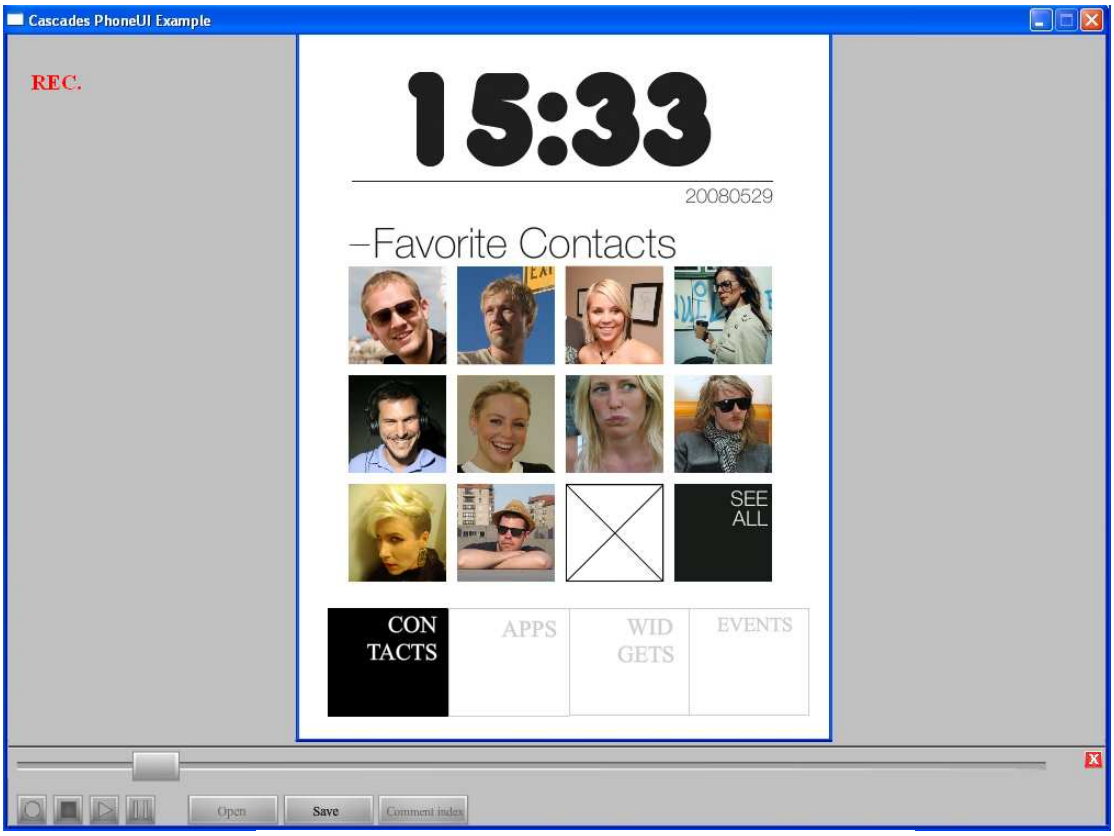

Figure 21: Prototype with the “Rec” button pressed.

When pressing the button “Open”, a list with the available saved recorded flows is shown. When clicking on one of the alternatives, the recorded UI is displayed and a text “Opened: filename” is shown in the upper left corner. Added comments are displayed with lines in the timeline. The mouse clicks made during the recording is shown with graphics (see figure 22).

It is possible to add a comment to any point in the UI by right-clicking. The comments work in the same way as in the normal mode and a line in the timeline mark the comments’ existence.

By clicking on a mark in the timeline, the lever in the time line skips to the location of the mark. This location can also be reached by clicking on the button “Comment index”, where all added comments are listed.

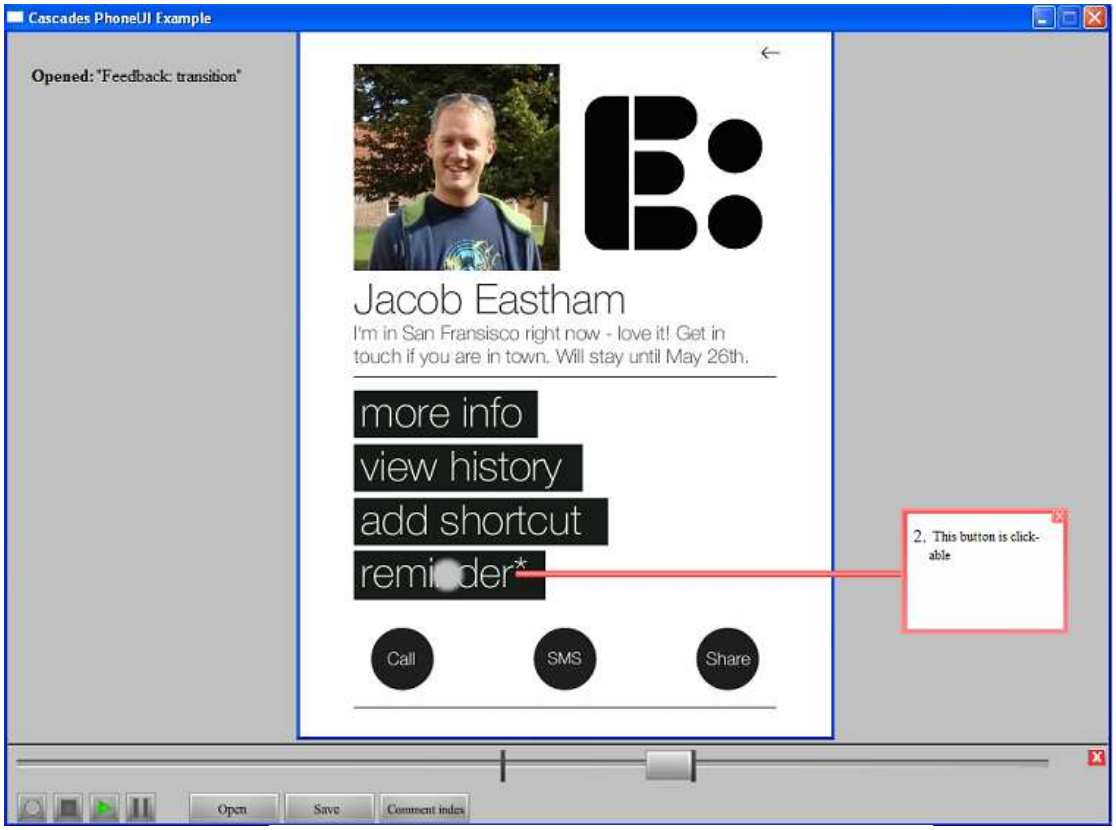

Figure 22: Prototype where a recorded flow is played.

The button for closing is always active in the record menu. When pressing the button, a pop-up window gives the user a possibility to save the recording or the changes made (see figure 23). When closing the record menu the page that was shown when the user pressed the key for opening the recording menu is displayed.

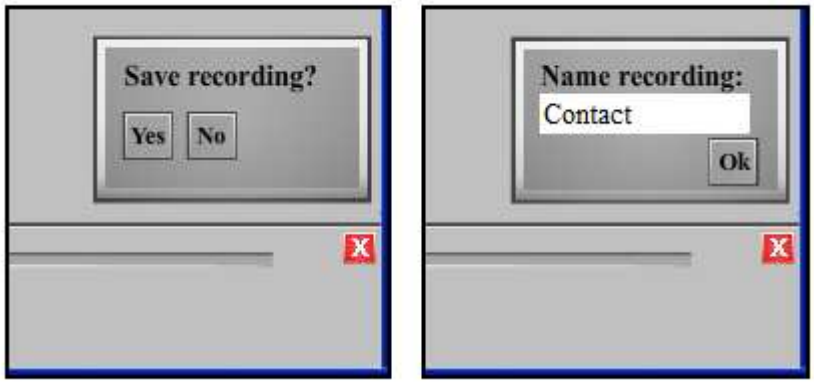

Figure 23: Detail of prototype: menu for saving flow

5.4.2 Evaluation

The test participants' comments and answers to the subsequent questions during the usability tests will be presented in this section, categorized according to the test plan's problem statements. An extended description of the gathered data from the tests can be found in Appendix C. In some cases the subsequent questions got answered spontaneously by the test participants during the exploration part.

Did any misunderstandings concerning the functionalities occur when the test participants explored the prototype?
The results are in this section categorized after the different areas of the prototypes that got commented or where problems were identified during the usability tests.

The two different modes: normal and record
One of the test participants, the interaction designer, did not realize that it was different comments in the normal mode and in the saved flow in the record mode. He thought a new flow should be created when opening the application, which then is saved. He did not understand the meaning of having different comments in the two different modes.

Two other test participants, the two TML engineers, found it unclear that it was different comments in the normal mode and in the recorded flow. One of them thought it should be entirely timeline based and was unsure if the normal mode was needed. He explained that a page often contains many different states and animations; therefore he would like to be able to pause and add a comment at any time. In the normal mode it is not possible to add a comment in the middle of a transition, just before or after. He also thought the current mode should be more visible.

The comment index
One of the test participants, a TML engineer, did at first not understand that it was comments that were listed in the comment index. He first thought it was a menu to something else. However, he thought the functionality was good, especially in the record mode.

Another test participant, the interaction designer, thought the list should be larger to make it possible to read the entire comments.

The comments
One of the participants, the sales project leader, found it difficult to know where in the prototype there were added comments. She suggested a highlighting of the show/hide button when a page has comments.

Another test participant, one of the TML engineers, also made a comment about difficulties to find the comments. His suggested solution to this was an auto-generated UI tree with screenshots of all pages, with the comments shown. A click on one of the pages' display would zoom in on the page.

Do the suggested functions meet the specified usability goals?
In order to verify that the suggestions' functionality meets the previously specified usability goals, subsequent questions were asked after the exploration part of the usability tests. This section presents the result categorized after the usability goals concerning the user's experiences.

U1. The new functionality should replace or complement the currently used methods
All test participants said they would prefer to use the suggested functions to give and receive feedback, compared to information sent via email or Skype.

Concerning the verbal information given in meetings and via telephone, one of the TML engineers who participated in the test thought it is easier to talk about a problem, but that the suggested functionality could partly replace meetings and telephone conferences. He also believed a prototype with the suggested functionalities could be used as a complement; to talk about the comments and explain things concerning them.

The other participating TML engineer thought it would be good as a base in meetings and telephone conferences. The interaction designer who participated in the test thought the suggested comments partially could be a substitute for meetings and telephone conferences.

U2. The functionality for feedback and narration should not disturb the designers' or customer's experience of the prototypes.
The sales project leader was convinced that the customer's experience will not be disturbed, she believed it would increase the communication and make it safer. The interaction designer who participated in the usability test did not give any indications of experiencing the suggested functionalities as disturbing.

U3. The designers and customers should experience it as faster to give feedback on the prototypes.
The interaction designer who participated in the test thought it should be faster to create feedback with the suggested functions.

U4. The designers should experience it as faster to create narration to the prototypes.
Since a majority of the participants did not create narrations, it was difficult to evaluate this usability goal. One participant, a TML engineer, did however explain that the biggest reason for this is that external programs must be used and he thought the features illustrated in the prototype would be very important.

U5. The designers and TML engineer should experience the given feedback as more qualitative.
All of the participants were positive to the new functionality. One of the participants, a TML engineer, thought the solution makes it possible to really see the location intended in the given feedback. He thought it is easy to point out where the problem is and to get a more mutual frame of reference. Another test participant, the interaction designer, said: "Yes, you have what you are talking

about in front of you. You have everything collected in front of you, which is the strength".

U6. The designers and customer should experience it as easier to give feedback on the prototypes.
The sales project leader thought the suggested functions, making it possible to explain more exactly, could offer the customers' a sense of security.

The interaction designer participating in the test believed that it was easier since the comments are connected to the exact intended location, making it unnecessery to explain the location with text.

U7. The designers and TML engineers should experience it as easier to create narration on the prototypes.
One of the TML engineers who participated in the test explained that he does not record prototypes that often. External tools have to be used, which often cause frame rate problems. He thought a recorded flow would be positive. He thought a video that everyone could play and which could be uploaded in different places would be great. The other TML engineer also thought it would be easier to create narration with the suggested functions.

The interaction designer and sales project leader both said videos created in Adobe After Effects mainly are used to show the TML engineers how transitions should look, not to demonstrate how the prototype work.

The sales project leader thought it was very positive to be able to add comments as complementing information, for example to inform the customers that an area of the prototype is not complete. When giving complementing information in the current situation, the describing document often disappears from the prototype.

U8. The TML engineers should experience it as easier to understand the given feedback.
Both participating TML engineers claimed that they would prefer to receive feedback in the way presented in the prototype. One of them explained that it would be much easier since it is easy for the designer to mark details and for the TML engineers to get a general view. He explained that he in the current situation often receives emails containing a list of requested changes and he have to search through the prototype in order to find the concerned area. He thought making it possible to avoid chains of emails would make things even more efficient.

What are the participants' other opinions about the suggested functions and their design?
The participants' spontaneous comments about the suggested functions and their designs, not covered by earlier described misunderstandings or usability goals, are presented in this section.

Positive opinions
One of the test participants, a TML engineer, thought it was positive with the click feedback in the saved recording. He also thought it was good to have buttons, which he was used to from video recording programs like Adobe After Effects. The sales project leader also thought the buttons where positive, since they make it possible to comment fast without losing focus.

The other comments are listed below:

- Simple and good. As simple as possible to be able to record something good. (TML engineer 1)
- I believe this would be great. (TML engineer 2)
- Great initiative (TML engineer 2)
- Great idea. (Interaction designer)
- I really, really believe this would be great. (Sales project leader)

Suggestions for additional function/changes to the functions

- Directly available record menu next to the buttons.
- Connection to a web service with the possibility for several persons viewing and having a conversation about the prototype.
- Hardware buttons shown in the device.
- A function for exporting screens, with automatically drawn arrows in between.
- Functionality for version handling and a possibility to choose which version of comments to view in the prototype.
- An email function, where the video can be presented as an internal link or as an attachment in the email.
- Functionality for creating answers to the comments.

5.5 Placement in the tool chain

The usability goals and requirements show that it is crucial that the suggested functions are placed in direct connection to the UI prototype. It is also very important that the usage of the functions does not require any coding, since a majority of the users will be designers and persons from the customers, who not necessarily have programming experience.

Another important aspect is that the users of the suggested functionalities in many cases do not have TAT Motion Lab installed on their computers, which means that the functionality must be accessible outside TAT Motion Lab.

Some different possible solutions were discussed with Cascades developers Markus Videsson [21] and Neal Shail [20]. The discussion led to the conclusion that in order to make the functions accessible outside TAT Motion Lab, the best solution would be to place the functions in the viewer.

According to Shail [20] the viewer is an executable file. The only connection between the viewer and TAT Motion Lab is a text file that writes to TAT Motion Lab and that the viewer can read from. The text file describes how it should start and what TML code is used. By packaging the TML code, the executable file and the text file together, the UI prototype can be viewed without having to open it in TAT Motion Lab. The viewer automatically searches for the text file. By implementing the functions in the viewer, it would therefore be possible to use them in the executable file, without having access to TAT Motion Lab. [20]

Since different viewers are used in different devices, a framework that includes the functions could with advantage be developed. The framework could be used every time a viewer is built. [21]

6. Discussion

6.1. Findings in the usability test

The results from the usability tests show that the suggested functions and their design could be beneficial for the work process. The fact that all participants were positive to the functions and said that they would prefer to use the suggested functions when giving and receiving feedback instead of using email and Skype, is a strong indication that the new functions could replace those methods and that the users would choose to use them.

Since the majority of the participants stated that they do not create videos for narration and disadvantages with external programs were mentioned, I believe that the suggested functions for recording flows is crucial in order to improve the customers' understanding for the prototype as well as making the communication easier and more time efficient.

The results showed that the participants did not think the new functions could replace the verbal information exchanged during meetings and via telephone. This is not very surprising, since the verbal communication contains many aspects that are not included in the tool chain even with the suggested functions. A major difference is the time aspect for getting answers and discussing features. There are still time delays when sending and receiving the information. Face-to-face communication has many ingredients like facial expressions and body language that the functions cannot replace. Many participants however thought the functionality would be a good basis for meetings. If the meeting participants find it easier to discuss the UI prototype, there is a chance the meetings could be more qualitative and time efficient.

One important aspect when designing the functions was that they should not disturb the experience of the prototypes. Since no indication was given during the tests that this could be a risk, which shows that the functions could enhance the understanding for the UI prototype without interfering with the users' impression of it.

Some of the participants found the normal mode unnecessary. Nevertheless, in order to give feedback fast, I think it is important to be able to add a comment in the prototype without having to record a flow. The downside is, like one of the participants pointed out, that it is not possible to comment animations or transitions. However, since this is covered by the function for recording flows, I think the advantages outweigh the disadvantages. A major upside for the normal mode is it makes it possible to add explaining comments in the UI prototype which could help the customers when examining prototypes. A recorded flow removes all interaction between the users and the UI; this is one reason to why I think it would be important to include the suggested functions in the normal mode.

6.1.1 Changes in the high fidelity prototype

A couple of the test participants thought the button “Comment index” was diffuse. In order to make it more obvious that the menu contains added comments, some changes were done in the high-fidelity prototype. The name on the button has been changed to *Comments*. When clicking on the button the menu is larger, this makes it possible to read the entire comments. The menu also contains a headline stating “Comments:” in order to make the contents clearer.

To make the current mode more visible, a headline stating “Normal mode” and “Record mode” has been added at the top of the prototype.

6.2. Future work

The obvious future work is to transform the concept presented in this thesis work into an implementation in the viewer. I believe that the functions described in this thesis work are absolutely necessary in order to make the work process even more efficient when developing UI prototypes. I also believe that one function more should be added; a UI tree. One of the participants suggested this during the usability test. By adding a function, similar to the one in Microsoft Expression Blend 3 [11], I think the users’ understanding for the prototype would increase since it would give an overview of the whole UI prototype. This would be particularly positive if it is a complex UI.

Figure 24 shows an example of how the UI tree could be designed. The UI tree could be reached by pressing a button called “UI tree” that showed a map of the different pages. Each page should have a unique ID number, in order to facilitate discussions. The pages with comments added to them should preferably be marked in some way; in figure 24 they are marked with the letter C, which stands for comment, after the ID number. This would make it easier for the user to get a picture of where the comments are added in the prototype. A click on one of the images in the UI tree that represent a page should automatically minimize the UI tree and skip to the correct page in the UI prototype. This would make it possible to view the added annotations quick, as well as making it possible to fast go to a page and add a comment.

There are also other functions that could be positive, one participant in the usability tests for example mentioned a web service that allowed several persons to view and annotate the UI prototype at the same time. This could be positive, but I think the first step should be to implement the functions suggested in this thesis. If they are designed in the way described in this report, I am convinced that they will have a positive effect on the time efficiency and the quality of the information exchange.

Later in the development, web services and other functions could be suitable to implement. Another aspect that would be interesting is to create functions that make it possible to edit UI prototypes without requiring any programming knowledge. An example of such a function could be making areas in a UI

prototype non-clickable. It could work like a layer on top of the UI and exactly like the annotation described in this report, the editing would not change the original. This would allow for example designers and project leaders to direct the customers' to different areas of a UI prototype when viewing it.

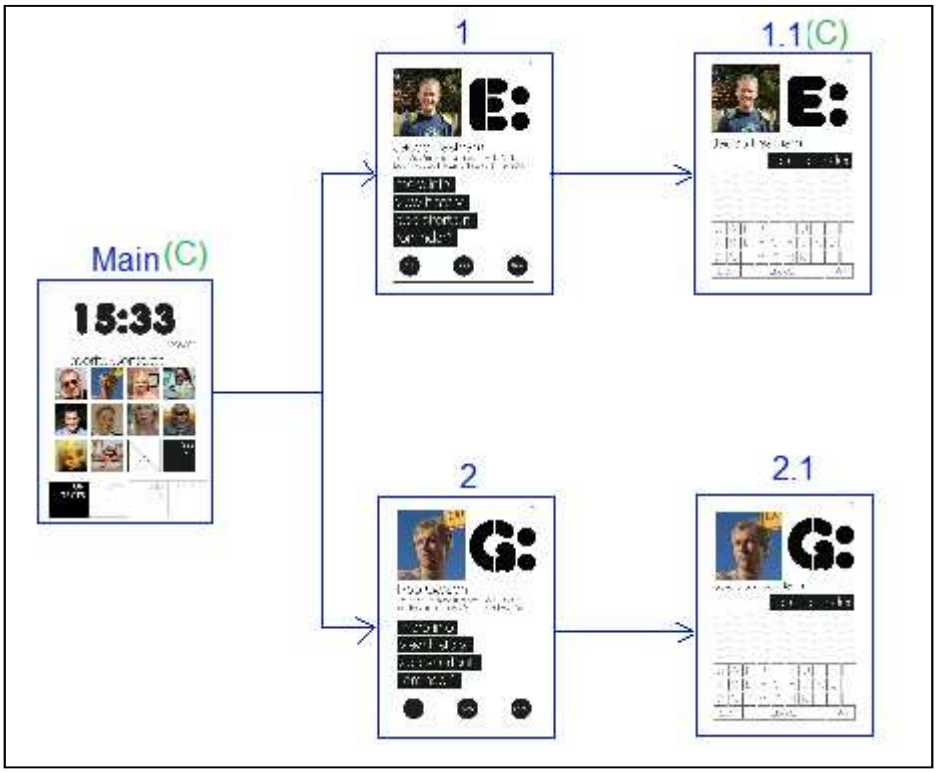

Figure 24: Suggestion for UI tree

6.3. The chosen methods and the result

The investigation of the work flow and information exchange was performed by conducting interviews. This made it possible to get information directly from the persons in the development teams. However, it would have been positive to perform a more extensive evaluation of the work process and make observations of the different persons in the development process when performing their work. This would have made it possible to perform more detailed measurements with quantitative data, for example measuring the time when performing a certain task. The measurements could have led to more discovered possible improvement areas in the process and it could also been used in order to compare the new solution with the currently used solutions. This was however not possible due to secrecy. Another aspect not possible because of this was to include actual customers in the investigations and evaluations.

The developed prototype made it possible to collect valuable qualitative data. It was however not possible to collect any quantitative data like time for completing tasks. This since the prototype was horizontal and all functions were not fully implemented. I prioritized to make it possible to demonstrate all suggested functions. Since it was not possible to make quantitative measurements in the current work flow, it was not possible to make any quantitative comparisons.

A challenge during the thesis work was that the persons that participated in the different steps work with development and design of user interfaces on a daily basis. It was positive to get valuable input from experienced person, but it was sometimes difficult to keep their focus on the asked questions. When for example asking questions in order to evaluate the suggested functions, some of the participants automatically started to brainstorm about other possible solutions.

The prototype that together with this report represent the result, works as a tool for demonstrating the developed concept with the suggested functions. Besides making usability tests possible, the prototype is positive to use as a concrete material to show and discuss. Together with the specified requirements it could work as a blue print when starting the implementation.

7. Conclusions

The main purpose with this master thesis was to give suggestions to new functions and their design in the development tool chain used at TAT when developing UI prototypes. The suggestions have been based on an investigation of the work flow during the development. A theory collection has also been conducted, which include information about different ways to add information in a prototype as well as descriptions of some of the different methods used in the master thesis. The result is described in this report and has also been presented with a high fidelity prototype. The answers to the specified problem statements are described below.

What areas of the work flow when developing UI prototypes have potential to be improved?
Areas with possible improvement potential got mapped out by conducting interviews and a focus group. The findings are described in detail in section 5.1.6 and 5.2.2. Some examples are: the communication when there is a geographical distance between the persons in the development teams and the placement of complementing information.

Which aspects are important to take in consideration when designing the functions?
One of the major aspects to take in consideration is what information is exchanged in the current information flow. This in order to know what information that must be possible to give and receive with the new functions. The information flow has been mapped out and is presented in figure 6.

The information flow as well as other important aspects where investigated through interviews and meetings. Usability goals for the functions were produced, which can be found in section 5.1.7. Some examples are that the users should experience it as easier to give feedback and create complementing information and that the functions should not disturb the experience of the prototype.

Which functions should be added to improve the work flow?
The suggestions are presented in section 5.2.1 and in a high fidelity prototype. Some of the major suggested functions are:

- A function for adding comments in the prototypes, connected to certain points in the UI.
- A comment index that shows all added comments in the prototypes.
- A function for recording a flow of events.

How should the functions be designed?
Requirements were developed in order to specify the design of the functions in the prototypes developed in the master thesis. These can be found in section 5.3. An iterative process with prototype development and evaluations was performed, in order to find the best design to meet the specified usability goals and requirements. Some examples of design decisions are:

- The comments should be placed in an area on the sides of the prototype.
- The user should be able to add a comment by right-clicking on a point in the prototype and a comment box should then be generated automatically.
- The comments should be connected to the point in the UI with a line.
- There should be buttons for starting, stopping, playing and pausing a flow of events.

The result is presented in section 5.4 and in a high fidelity prototype. The high fidelity prototype was evaluated with usability tests and the suggested changes based on the test result are discussed in section 6.1.1.

Where in the tool chain should the suggested functions be added?
Meetings with persons with expert knowledge about Cascades and TAT Motion Lab were held. This led to the recommendation to implement the functions in TAT Motion Lab's viewer. The placement is discussed in more detail in section 5.5.

8. References

[1] Buxton, Bill (2007), *Sketching User Experiences*, Morgan Kaufmann.

[2] Dictionary.com (2009), retrieved from <http:// www.dictionary.com> 27 October 2009

[3] Funseth, Barbro (2000), *Narrativ multimedia*, retrieved from <http://www.iml.umu.se> 10 November 2009

[4] Hodgson, Philip (2003), *Quantitative and Qualitative data – getting it straight*, retrieved from <http://www.blueprintusability.com> 9 December 2009

[5] Interaction Design Association (IxDA) (2008), retrieved from <http://www.ixda.org> 9 December 2009

[6] International Organization for Standardization (ISO) (1998), retrieved from <http://www.iso.org> 9 December 2009

[7] Kindborg, Mikael (1997), *Människa-datorinteraktion*, retrieved from <http://www.ida.liu.se/> 6 November 2009

[8] Lunds Tekniska Högskola (2008), *Riktlinjer och råd för examensarbete på civilingenjörsprogrammet*, retrieved from <http://www.student.lth.se> 3 September 2009

[9] Löwgren, Jonas (1993), *Human-Computer Interaction: What every system developer should know*, Studentlitteratur

[10] Microsoft (2009), retrieved from <http://www.microsoft.com> 24 November 2009. "Microsoft product screenshot(s) reprinted with permission from Microsoft Corporation."

[11] Osborn, Jeremy (2009), *Adding Feedback and Annotations in the SketchFlow Player with Expression Blend 3*, retrieved from <http://expression.microsoft.com> 3 December 2009 “Microsoft product screenshot(s) reprinted with permission from Microsoft Corporation.”

[12] Preece, Jennifer, Rogers, Yvonne & Sharp, Helen (2002), *Interaction design: beyond human-computer interaction*, John Wiley & Sons, Inc

[13] Protonotes (2008), retrieved from <http://www.protonotes.com> 16 September 2009

[14] Rubin, Jeffrey (1994), *Handbook of usability testing: how to plan, design and conduct effective tests*, John Wiley & Sons, Inc

[15] Shneiderman, Ben (1992), *Designing the User Interface,* Addison Wesley Publishing Company

[16] TAT (2009), retrieved from <http://www.tat.se> 20 October 2009

[17] YouTube (2009), retrieved from <http://www.youtube.com> 16 November 2009

Verbal references

[18] Gedin, Minna, meeting in July 2009

[19] Sandström, Henrik, meeting in June 2009.

[20] Shail, Neil, meeting in October 2009.

[21] Videsson, Markus, meeting in September 2009

Appendix A: Interview questions

1. Which are your work tasks?

2. Who do you work with?

3. Where are the persons you work with located?

4. How do you communicate with the other persons in the development team?

5. Which tools/programs do you use?

Incoming information

6. What information is your work based on?

7. From whom do you get the information?

8. How do you receive the information?

9. If any files are sent, which file formats?

10. Do you give any feedback on the information you receive? To whom and how?

Outgoing information

11. To whom do you forward the work you have done to? How?

12. Do you receive any feedback on the work you have done? Do you send the work back to the person who gave you the incoming information? To whom and how?

13. If any files are sent, which file formats?

14. How do you explain what in for instance the graphic you mean when discussion something via email or Skype?

19. Any ideas about other ways to give feedback?

Appendix B:
Initial information and subsequent questions in the usability test

Initial information
I have created a prototype that shows functions that are supposed to be used by designers and customer when giving feedback on the prototype made in TAT Motion Lab, as well as creating complementing information when the prototype is displayed for customers.

The idea is that the functions should be used when there is a geographical distance between the designer and TML engineer, instead of sending feedback over email and via Skype. It is also supposed to complement information given during meetings and via telephone.

The focus in this test is to evaluate if the functions could make this possible and if the functions' design is optimal.

Subsequent questions to the TML engineers

1. Do you experience feedback given in this way as more qualitative than information via email or Skype? In what way/Why not?
2. Would you prefer receiving feedback in this way compared to receiving it via email or Skype? Why/Why not?
3. Do you believe information given in this way could replace information given verbally on meetings and via telephone? Why/why not?
4. Do you experience it as easier to create narration than to record a prototype? Why/why not?
5. Do you experience it as faster to create narration than to record a prototype? Why/why not?
6. Do you believe it would be faster to perceive the information in this way?

Subsequent questions to the interaction designer and project leader

1. Do you experience it as easier to give feedback in this way, compared to giving feedback via email or Skype? Why/why not?
2. Do you experience it as faster to give feedback in this way, compared to giving feedback via email or Skype? Why/Why not?
3. Do you believe that information given in this way could replace information given verbally on meetings and via telephone? Why/why not?
4. Do you experience it as easier to create narration than to create videos in Adobe After Effects? Why/why not?
5. Do you experience it as faster to create narration compared to record a prototype? Why/why not?
6. Do you experience the feedback given in this way as more qualitative than information sent via email or Skype? In what way/Why not?
7. Would you prefer receiving feedback in this way compared t

8. Would you prefer receiving feedback in this way compared to receiving it via email or Skype? Why/Why not?

Subsequent questions to the project leader

1. Do you experience it as easier to create narration than to create videos in Adobe After Effects? Why/why not?
2. Do you experience it as faster to create narration than to record a prototype? Why/why not?
3. Do you experience the feedback given in this way as more qualitative than information sent via email or Skype? In what way/Why not?
4. Would you prefer receiving feedback in this way compared to receiving it via email or Skype? Why/Why not?
5. Do you believe this narration could increase the customers' understanding for the UI?
6. Do you believe the functionality could disturb the customers' experience of the UI?

Appendix C: Results from usability tests

Spontaneous comments
The following comments and reactions came up during the test sessions (freely translated):

The record menu

- It makes it possible to see what the user is doing. (TML engineer 1)
- Great with the click feedback (TML engineer 1)
- The comment index is good since there often are many steps back and forward in the video clip as well. (TML engineer 1)
- It is really nice to have the buttons, that you are used to have in video recording program like Adobe After Effects. (TML engineer 1)
- It is important that all steps and animation are included, not just all pages. Sometimes there's an animation that changes a bit on a page and it should be possible to pause when it changes and add a comment. (TML engineer 2)
- When you are testing you often imagine a recording in the head, like: "now I'm pressing rec" etc. If you then have a video, it would be great to be able to record and send out. (TML engineer 2)
- I think that the record menu should be available always, next to the buttons. Then you would save one step and can record at once. And then you get the comments in direct connection to the time line. It would create a better overview, but there is a risk that it will become messy if there are many comments. Writer's comment: misunderstanding concerning the comments. (Interaction designer)
- Do not understand the meaning of having different comments for the recorded flow; instead thinks a new flow should be created when opening the application which then gets saved. The comments should be on that flow. If it is something that is recorded or if it is just comments does not matter. (Interaction designer)

Prototype mode

- I think that it should be completely timeline based. I do not know if the normal mode would be necessery, since it is only possible to add comments on the page 1, page 2 etc. There are often many animations and different states on each page, there could for example be different states if the contact list is full and empty. In those cases you would want to pause at any time and comment. For example press record, press stop and then be able to rewind and comment in that video. Here you cannot comment in the middle of a transition. You would want to pause at any time, so the steps are not to discrete. Here you have to describe "between this and that side". (TML engineer 2)

The comment index

- Did not understand that it was comments that were listed when pressing the button. I first thought it was a menu to something else. (TML engineer 1)
- The list should be larger, so it is possible to read the entire comments. It could actually have an unlimited size, so it would be possible to scroll in it. There is a lot of space. (Interaction designer)

Show/hide button

- Good it is possible to run it without being distracted by the comments. Would like an indication of where there are comments. For example that the button gets highlighted when there are comments on that page. (Sales project leader)

The comments

- It would be positive to be able to write answers on the comments. (Interaction designer)

In general

- I think it is simple and good. As simple as possible to be able to record something good. (TML engineer 1)
- I believe this would be great. (TML engineer 2)
- Great initiative (TML engineer 2)
- It would be positive with a connection to a web service. Making it possible for several persons to have view the prototype at the same time, like a live conversation. It would be positive when for example talking to someone in another country. With different mouse pointers for different persons. Pretty advanced, but would be good. (Interaction designer)
- It would be positive to show the hardware keys that the devices have. A part of the interaction happens through them. The device should be shown around the UI. All telephones do not have touch screens, but all telephones have hardware keys. For example Android phones always have a menu key that is used quite often. (Interaction designer)
- Great idea. (Interaction designer)
- I think this would be great. At this point we often create Microsoft PowerPoint's or pdf:s, where we take screenshots where we create comments and try to explain different elements and the idea with them.
- I really, really believe that this would be a great help. (Sales project leader)
- Positive with buttons, they make it possible to comment fast without losing focus. (Sales project leader)

Suggested changes/improvements/additions

- Would be good with an overview of the prototype. To be able to get an overview of the pages that exists and be able to skip between them. Be able to press on one page and start the recording from there. A little like the project overview in the new version of TAT Motion Lab, but more adapted to this kind of recordings. (TML engineer 1)

- It would be positive, especially for the designers, to be able to export screens. If you have three pages, you press on “export pages” and print screenshots of all of them. And automatically drawn arrows in between them. (TML engineer 2)
- I think there should be two modes: the time-based and the prototype mode with more functions. On one hand make it possible to create comments in an easy way. You would also want a button that auto generated a UI tree with screenshots on all pages. Then you should be able to click on a small screenshot, it zooms in and you could add comments. And the lines in between will be covered by the time line, it is the transitions. For like it is now, you do not know where the comments are in the prototype. (TML engineer 2)
- Version handling: the first time you comment a flow you get version 1, the second time version 2 and so on. Then you could choose which version of comments you want to view, the newest, oldest or all. It would be even better with an addition timeline for this in the prototype, as long as it does not confuse the user. (TML engineer 2)
- More distinct text/information of what the current mode is, for example “you are now in the page view mode”. (TML engineer 2)
- A more advanced recording mode would be needed if you are going to use it in TAT Motion Lab. For example be able to click on rec and send it. And then an email function; “email this to”. And then you get the video as an internal link or as an attachment in the email. (TML engineer 2)
- Would like to weight the buttons a bit, something makes it look a bit messy. (Sales project leader)

Misunderstandings concerning the functionalities during the exploration

- The interaction designer does not realize that it is different comments in the prototype and in the recorded flow.
- Both TML engineers find it difficult to tell which comments that belong to what. One says: “You do not understand that it is different comments in the recorded flow and in the prototype mode”.

Answers to subsequent questions

The following answers were given to the subsequent questions (freely translated):

Is the feedback given in this way experienced as more qualitative than information via email or Skype?

- Absolutely, here you really see visually where it is. Here it is possible to really point out “there is the problem” and “here is what has happened”. You get a more mutual frame of reference. (TML engineer 1)
- Yes, you have what you are talking about in front of you. You have everything collected in one place, which is the strength. (Interaction designer)
- Yes, (Sales project leader)

Would the test participants prefer receiving feedback in this way, compared to receiving it via email or Skype?

- Would absolutely prefer to receive feedback in this way. (TML engineer 1)
- Yes, it is easier than receiving feedback in an email and searching for the concerned area in the prototype. A designer could comment that a marginal is two pixels off, this would make such a thing easy to mark. It is easier to get a general view, compared to receiving an email with listed requested changes which you have to search for in the prototype. It would be much easier with a solution like this. (TML engineer 2)
- Yes, avoiding chains of emails would make it better. absolutely. (TML engineer 2)

Would the test participants prefer creating feedback in this way, compared to sending it via email or Skype?

- Yes. (Interaction designer)

Do the test participants experience it as easier/faster to create feedback in this way compared to giving feedback via email or Skype?

- It is clearer, because you see it directly. Since it is connected to the exact intended location you do not have to explain with text where it is. (Interaction designer)
- It should be faster (Interaction designer).

Do the test participants believe that the narration could increase the customers' understanding for the UI?

- Yes, definitely. (Interaction designer)
- Absolute, great for more complex things. You have made it so easy, so it would not be any effort to add comments. (Sales project leader)

Do the test participants believe that the information given in this way replace information given verbally on meetings and via telephone?

- Yes, I think so. It is always easier to talk about a problem than just receiving a comment like this. But sure, partly. I also believe that it could be used and shown as a complement on meetings. Or perhaps a telephone conference; be able to talk about the comment and explain "here have we done it in this way". (TML engineer 1)
- Would be good as a base, but it is difficult to make notes in it. A base, not more than that. (TML engineer 2)
- Yes, partially. It is smart, like the comments in Microsoft Word and in pdf documents. It is good (Interaction designer)

Do the test participants experience it as easier/faster to create narration in this way compared to recording a prototype?

- A video that everyone could play would be really great. An important feature is to be able to see where clicks have been made. (TML engineer 1)
- I do absolutely believe it would help. First and foremost it will help internally; designers and TML engineers. A great base for meetings and so on. The next step will be that customers start to use it, but that will not happen until later on. (TML engineer 2)

Do the test participants experience it as easier/faster to create narration in this way compared to creating videos in Adobe After Effects?

- Adobe After Effects is mainly used before the prototype is created. Videos are not so commonly created in order to show how the prototype work but in order to show the programmers. (Interaction designer)
- Adobe After Effects is mainly used to show transitions to the TML engineers. (Sales project leader)

Do the test participants believe that the functionality could disturb the customers' experience of the UI?

- It will not disturb the customers' experience; it will instead increase the communication and make it safer. If you need to explain a concept and decisions made, this is absolutely positive. (Sales project leader)