UNIVERSITY OF GOTHENBURG

Intuitive Design Principles:
guidelines for how to build an instinctual web application

Bachelor of Science Thesis in Software Engineering and Management

ALINA BUTKO
MARTIN MOLIN

Intuitive Design Principles:
guidelines for how to build an instinctual web application

ALINA BUTKO,
MARTIN MOLIN.

Intuitive design principles:
guidelines for how to build an instinctual web application

Alina Butko
Department of Computer Science and Engineering
Forskningsgången 6
Gothenburg, Sweden
butkoalina@gmail.com

Martin Molin
Department of Computer Science and Engineering
Forskningsgången 6
Gothenburg, Sweden
gusmartimo@student.gu.se

Abstract - This paper investigates interaction design overall and intuitive design, in particular. In this research, we aim to gain an understanding of intuitive design principles and based on this, we compose guidelines. These guidelines will help the designers to comprehend the main principles of intuitive design and give them a possibility to be able to develop intuitive web applications without the need of going through hundreds of books and articles. We believe that this research will be useful for a broad audience considering both designers and researchers will be able to find information of how actual people perceive design and what makes design intuitive. Considering the outcome of this study it is incautious to say that the guidelines will support anyone who is working with design and if anyone would want to continue our research, one possible direction could be to try other types of applications or usage areas while applying our guidelines. Even though this study is conducted within one company it will be possible to use the results for other cases where intuitive design is applicable.

Key-words: intuitive design, human-computer interaction, guideline, web application, intuitive design principles

I. INTRODUCTION

There is more software and hardware in the world today than ever before. They are designed by people, who are involved into the IT world. In small companies developers often become designers; here is what Hanselman (2012) says about this: "The difference between a Designer and Developer, when it comes to design skills, is the difference between shooting a bullet and throwing it."

However, often designing web interfaces and web application interfaces in particular can become a big problem, because of the importance of informational content, lack of space, or impermissibility of style. As a result, we may get a very complicated interface for very simple actions.

This topic is interesting because intuitive design can really help when designing user interfaces for applications. It can be used to make the applications simple, yet effective to use because it is designed with human reasoning in mind. We chose this topic because we want to learn more about design and how things should be designed. There are many bad examples of how not to design something, both hardware and software related. We want to make sure that we can avoid such mistakes when designing our software in the future.

According to *Software Engineering Body of Knowledge* (SWEBOK, 2012), our research will be conducted within the Software Design knowledge area. There we defined two research areas: (1) Software Design Quality Analysis and Evaluation and (2) Software Design Strategies and Methods. Researching this topic will help us find and describe design principles in the intuitive design world.

The main purpose of the research is to figure out the importance of the intuitive design by understanding its principles and creating guidelines for building web applications with intuitive design in mind. **Research question to be investigated in this paper:** What principles of interaction design are conducive to the development of intuitive web applications?

In the field of the human-computer interaction (hereinafter referred to as "HCI") and intuitive design, in particular, there has been done a lot of work. A lot of different principles and guidelines supposed to help designers to Research papers touch upon different aspects of intuitive design, but all of them look into the interaction overall without looking specifically into web application design.

A. STRUCTURE OF THE PAPER

The introduction section is followed by method where we discuss the methods used to conduct the research. After that literature review where interaction and intuitive design will be investigated to help us to build the base of the research and highlight the principles of intuitive design. Results section will be next in the paper to present the results of the research and the research question answer will be provided including whether our principles seem to hold. After that, the results of the research will be examined in the discussion section, and conclusions will be drawn. At the end of the paper, you can find Appendices which contain preliminary and final interview questions, the

questions of the survey with pictures, and the actual result of the research: guidelines with five principles and short description.

II. METHOD

We came into contact with a company called Systemite from Göteborg, Sweden, which provides the first high-performance platform (SystemWeaver) for systems development - enabling integration of design processes for complex computer-based systems. They asked us to develop a web application for issue management, which will be working towards a SystemWeaver server and use their own API for issue management. When they presented the original PC client, we noticed some small intuitive design faults, which gave us an idea for this research. The aim of this research is to identify key principles of the intuitive design and create guidelines for designer who wish to create intuitive systems without the need of additional literature.

To conduct our research, we have decided to go with a mixed method approach, which means that we will both collect and analyse qualitative and quantitative data (Creswell, 2009). We will start by doing a literature review to collect a robust base of qualitative data regarding intuitive design which we will interpret and compose into principles that can help designers to avoid common mistakes. By utilizing these principles, we can construct interview and questions for an online survey to confirm whether the principles help or not. The interviews will collect qualitative data for our research, and the survey will collect quantitative data about our principles, and if they contribute to an intuitive design.

To do the literature review, we defined a systematic way of searching for articles, which should be included. Two sources which you can find below are used to identify relevant articles. Source 1 is keyword searches, and Source 2 is references from the articles found from Source 1. These Source 2 articles provided more keywords to use with Source 1, thus establishing a method loop, making it a continuous process as the two sources are complementing each other.

Source 1: Database keyword search

IEEE Xplore (2012) is a database of articles and it was used to identify relevant articles. This database was accessed through school network The initial keywords were motivated by our research question.

The following keyword phrases are identified:

- Intuitive design
- Intuitive design principles
- Human-computer interaction
- Design guidelines
- Web application design

Source 2: References

When a relevant article is found using Source 1, the references of that article is checked as well to see if they are of value to this literature review. If they are, then they will be included and there is a possibility to identify new keywords to improve Source 1.

The interview questions were composed from the literature review and were designed to give us in-depth information about how particular users perceive and understand web application design. The aim of the interviews was to help us understand if our intuitive design principles are beneficial to use. The questions were targeted at the employees of Systemite because for this research they have a very large spectrum of people: managers, developers and consultants working on a customer side. This strategy helped us to target a few different parts of the IT world in a short amount of time. Before we did the interviews, we created a preliminary list of questions to be asked after that we made two try out interviews and revised the questions according to the feedback which we got from the interviewees. The two first interviews were done face-to-face, and the rest of the interviews were done via email after we had revised the questions. This way, we saved time and avoided transcribing mistakes and personal interpretation. We managed to interview eight people, and they gave us very good material to work with.

The survey was also based on the principles from the literature review, and the questions were targeted at students of Gothenburg University. The questions of the survey were formed to test size, grouping, positions, color and presentation of the information of the user interface controls such as buttons, menus, labels and overall impression of the design in order to get an understanding of what different people find intuitive and natural. This survey was done online at Free Online Survey (2012). We sent out an email with the request for respondents to answer the survey together with a link to the actual survey. This strategy allowed us to get feedback from a large amount of users; we were aiming for 100 answers, and at the end we managed to get 126. This means that we accurately could collect a percentage of how many users liked a

particular design over another and test our principles.

III. LITERATURE REVIEW

Studies show that well designed system should be easy to learn, easy to remember, useful, easy and pleasant to use (Gould & Lewis, 1985). These are well known factors of a "good" design. It is possible to see that design is intuitive but not so many people are able to define intuitive other than intuitive (McKay, 2010). It was hard to find straight definition of intuitive design, so we looked at the definition of the word intuition on Wikipedia (2012) and based on it, we defined intuitive design as "design where a user has clear understanding of UI behavior and effect without any assistance, training or experimenting." Spool (2006) gives following definition of intuitive design: "A design is intuitive when people just know what to do and they don't have to go through any training to get there… When a design is not intuitive, our attention moves away from what we're trying to accomplish to how we can get the interface to accomplish what we want." If we look at this definition closely it becomes obvious that for such intuitive design, it is necessary that users should have so-called prior knowledge. This knowledge can be obtained either from real-life experience or from previous experience with similar software. Norman (2006) in his book "Design of Everyday Things" gives amazing examples of how things which we are using every day are designed so that they lose all the "intuitiveness" and become a disaster to use. One example that he mentioned was when his friend got stuck between two sets of glass doors only because it was impossible to understand push/pull/slide/right/left/up. So when we design a system, we should keep in mind who is going to use it, how they are going to use it, and where they are going to use it (Sharp, Rogers & Preece, 2006). Looking through related literature, it comes to mind that intuitive design is the design which you spend hours, days, weeks or even months to develop, but in the end it seems like you have done nothing. Intuitive design is invisible, it allows the users to concentrate on the task that needs to be done and not on the design itself (Shneiderman, 1992).

Designers who reached a certain point of success know that different users learn, think, and solve problems in numerous different ways. Some people may have much easier time with tables rather than graphs; with numbers instead of words; with slower display rates not faster; or with a rigid structure rather than an open-ended form (Shneiderman, 1992).

Design becomes intuitive when the users who will use this design find it intuitive. Spool (2005) suggests that we look at an interface's knowledge space as a continuum which goes from knowing nothing about the interface to knowing everything someone could possibly know (Figure 1).
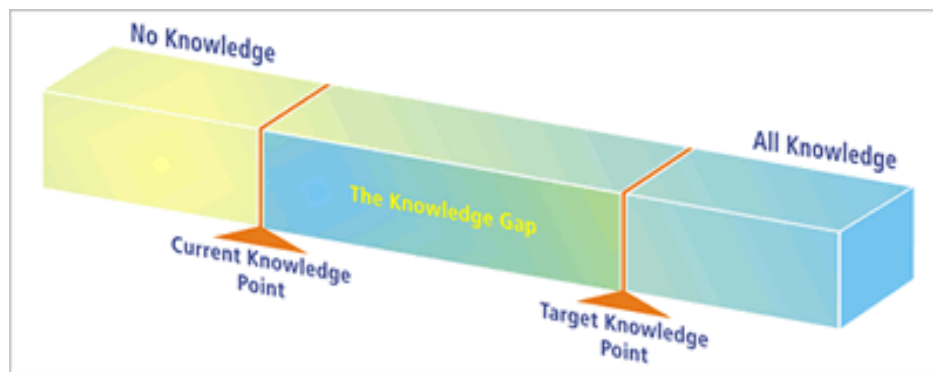

Figure 1. Interface's knowledge space continuum (Spool, 2005)

If we line up all the users who are going to use future design. Then organize them so on the left side we have those who know nothing about how to use the interface and on the right side we have users who know everything about an interface. Every user who is in this continuum has some certain knowledge of design, this is called the current knowledge point. There is one more point which represents what user needs to know that is called target knowledge point. The distance between these two points is the knowledge gap which is what we are concerned with when designing interfaces. When the gap between the current and the target points no longer exists, the design becomes "intuitive" in the eyes of users. The design of the interface should become a bridge between those two knowledge points and help the user to come to the target knowledge without noticing that he was trained. That is when the design of an interface becomes intuitive for users (Spool, 2005).

According to Spool (2005), the first step to make design seem intuitive is to understand what the user's current knowledge point is and what the target knowledge is. "A lot of people in our industry haven't had very diverse experiences. So they don't have enough dots to connect, and they end up with very linear solutions without a broad perspective on the problem. The broader one's understanding of the human experience, the better design we will have." Steve Jobs

First principle: The designer needs to understand what the current and the target knowledge point of the future user is.

However, this is far from the only thing which should be taken into consideration. As it was mentioned before it is extremely important that the application is as easy to navigate and use as possible, but unfortunately in the development cycle, there is rarely any space for UI designers (Windows Dev Center, 2012). At first glance, intuitive design is invisible, which is why so many companies tend to invest as little money and time as possible into the development of the UI side of the application. In this situation designers start to use too much graphical content, which distracts user from his primary goal. Blackler et al. (2007) describes an experiment of redesigning a video iPod. The experiment group did the initial testing and interviews, and they recognized that changing the circular scrolling pad which is the central method of interaction would adversely affect the brand as it has become well-loved by so many people. The things they decided to change were those which have been revealed as the most problematic: the hold button, the on/off button and few others. The initial design of the iPod was simple and effective, but a few things were not "intuitive" enough. So designers decided to keep familiar things like the circular scrolling pad but redesign the problematic components to be simpler. There are guidelines that mention designer should avoid use of unfamiliar symbols in the software. Blackler et al. (2006) stated in their paper that intuitive interaction may work through the similar features from the same or differing domains. They conducted an experiment which showed that "familiarity with a feature will allow a person to use it more quickly and intuitively." For example, if we compare Windows Word to Emacs it becomes obvious that they do not use same key combinations for the same functions. The most common problem is that when a person needs to save the document every five minutes, it becomes inconvenient for him to do that through the menu each time he needs to save it. That is why users are comfortable to use key shortcuts. In a lot of other programs to be able to save the document you need to press Ctrl+S but in Emacs this combination was responsible for

incremental search, if you want to save your document, you need to press Ctrl+X Ctrl+S. For people who are used to the Word this can become the worst experience in their life with Emacs because all the combination users used to in other programs in Emacs were responsible for absolutely different things. This fact can become a reason for slowing down work and causing frustration but on the bright side for those who were using Emacs all the time it wasn't a problem because they got used to it.

Shneiderman (1992) mentioned that pictures and visual representation of the information are not necessarily an improvement over text. Improper use of the visual part may cause information to spread out too much, create off-page connectors or even become confusing. Another problem with visual presentation may appear when designers overthink all known and simple things and create some custom icons, which require that the users learn the meaning of the icon before they can start using it with full force. One more problem with visual representation, mentioned by Shneiderman (1992), is that the visual representation may become misleading for the users. They can use their previous experiences with another software and draw incorrect conclusions about the actions. "Simple metaphors, analogies, or models with minimal set of concepts seem most appropriate to start." Shneiderman (1992).

Second principle: Design should be simple and familiar.

Computer screens become bigger and with higher resolution this gives a huge opportunity for design improvements. Having all this free space designer tends to put it into inappropriate use: they might use large buttons or garish non-standard colors to draw user attention to the important function (Windows Dev Center, 2012). Depending on the overall theme of your page you may change the size of the text to draw user attention to it and show that this is the most important thing on this page (Krug, 2006). It is important to be careful with sizes, Krug (2006) states: "Use as much space as necessary" but it is still important to think about the purpose of the used space. It should lead, help, explain user where he is, what he should do now and so on, and at the same time "... don't use any more space than necessary," said Krug (2006). He is assuring us that it is important to keep messages short - just long enough to get the point across, and no longer.

Another important thing to think about is positioning: where and why - Where do you want to put it? Why do you want to put it there? If we are talking about window applications, it is possible to

notice that most windows have identifying titles at top-center, top-left or bottom-center position. Scroll bars are usually at right and bottom position, which allows a user to scroll through the content and give access to the information which did not fit onto the screen (Shneiderman, 1992). Usually information that is significant to the user is placed in the center of the screen that is why it is a good practice to place information which the user desires to get there.

When we read information for the first time we tend to scan the page first to understand the content and what benefits we gain from it, considering that, it is a good practice for designers to group similar things under one heading, display them in similar visual style or put them in a clearly defined area (Krug, 2006). Grouping is not new to the design world and most designers intuitively group information on the page. Right now, any modern application has a lot of different controls. Only by proper, intuitive grouping it is possible to make all these controls easier to use (Windows Dev Center, 2012). However, Shneiderman (1992) is warning that "there is always danger that some users may not grasp the designer's organizational framework."

Card (1982) conducted experiment where a single 18-item vertical menu of text-editing commands was presented to a subject. This subject was given a command to find the certain item on the menu and click it. The menu items were organized in three different ways: alphabetically, in function groups and random. Each of the people who took part in the experiment made 86 trials with each type of the menu. The results were following: alphabetic - 0.81 seconds; functional - 1.28 seconds; random - 3.23. After a while, it became noticeable that users' memory for the menu organized in function groups was outstripping their memory for the alphabetic or random organized menus.

Everything that has been written above makes a page look attractive to the user, easier to use and more intuitive. However black and white pages can be boring to use, that is why it is well known practice to apply different colors when designing a page. There is no doubt that color makes information and pages more attractive to users, sometimes users even begin to associate color with information, certain action or brand, for example, Facebook - blue, Twitter - light blue, YouTube - red and so on. Shneiderman (1992) mentions few rules for color use, which are worth mentioning here: use color conservatively, limit the number of colors, recognize the power of color as a coding technique, ensure that color coding supports the task, have color coding appear with minimal user effort, place color coding under user control, use color to help in formatting, be consistent in color

coding, be alert to common expectations about color codes, be alert to problems with color pairings, use color changes to indicate status changes and use color in graphic displays for greater information density.

Third principle: Think about color, size, positioning and grouping.

Nevertheless, all above means nothing if the information that the designer is trying to convey to the user is not split into appropriate amounts. The rule of thumb states that humans can remember seven plus or minus two chunks of information, this requires that displays should be kept simple, multiple page displays be consolidated, window-motion frequency be reduced. The user should not be required to remember information from one screen for use on another (Shneiderman, 1992). If the designer wishes to use some self-designed custom icons or functions, they should be designed in the way that they are related to the ones the user already knows so he is not required to remember new information to be able to use the software. It is easier for user to recognize things then to learn them. Mandel (1997) says that reducing the users' memory load is a golden rule of the design. The users do a lot of tasks at the same time so it is hard for them to keep all the information they need to use in mind, so it is a bad practice to force them to remember everything while they switch between tasks. Program elements such as undo and redo, and clipboard actions like cut, copy, and paste, allow users to manipulate pieces of information needed in multiple places and within a particular task and makes people less afraid of making mistakes. Even better, programs should automatically save and transfer data when needed at different times and in different places during user tasks (Mandel, 1997). When the user works with an application, he usually uses short-term memory, but it is possible to include long-term memory support in the interface by using recognition. For example, it is much easier for the user to browse through a list of items and choose the one he needs rather than to keep all the items in mind and try to recall them from long-term memory.

It is also important for the user to know where he is, what he is doing and what he can do next (Krug, 2006). Visual interface is the best help to the user in this situation - designer should use so-called visual or textual cues. The example of such cues can be Google docs (2012). When you open the document in Google docs you are able to see at the top-left corner the name of the document and in what folder the document is. A little lower you can see if you are using normal text or headings, what font, what size, align and so on. All these small things unload user's memory, which makes it so much easier to

concentrate on the task and do not think about anything else. In the applications, it is a good practice to give the user a possibility to use shortcuts, but the designer needs to remember that using unusual shortcuts may confuse the user, and he could close your application and decide to never touch it again. Use of the simplest shortcuts such as Ctrl+S (save), Ctrl+O (open) or Ctrl+P (print) can make the users work much easier, and they will not need to spend time on searching for those commands in the menu. Even if it is fairly easy to find such commands in the dropdown menus it is still a lot faster to press well known combination of buttons (Mandel, 1997).

To be able to unload the user's memory and avoid the need to learn how the new systems work Mandel (1997) suggests using real-world metaphors. They will allow the user to transfer knowledge about things look and work. This will help a lot of users who are not used to computer systems, or if they are not encountered by it everyday, to avoid remembering new things or recalling an experience of previous use of this application. However, it is important to be careful while using metaphors so you do not confuse a user even more. Do not try to overwhelm a user by showing all the functions at once because he will not be able to concentrate on it and remember things which he needs. Try to show users what they need, when they need it, and where they want it (Shneiderman, 1992).

Fourth principle: Minimise memory load on user.

Not once in this paper it is mentioned the importance of the visual presentation of information to the user. The Web Credibility Project (2012) investigated very interesting questions regarding the users' trust/mistrust to websites and programs. They found out that there are a lot of different reasons for people trusting websites, applications and so on, but "appearance of a company's product" took the first place in their list. Users believe that if a company spends time and effort to create an appropriate design, they can be trusted.

It is known that the same content may be perceived differently because the design is different. Css Zen Garden (2012) has a good demonstration of this principle; they use a single HTML page, but they have different CSS style sheets so the participants of the project could try out different visual designs for this HTML page. Different color schemes, graphic elements, positions give the impression that different pages are visited and different information is given, but the text is the same. Developers make sure that the application they developed has all the functionality needed and that all functions working properly, but they do not care about design, and that is where designers begin their work. They do not need to think about functionality anymore because developers took care of it, what the designers care about is the emotional and visceral reaction of the users on the product (Tidwell, 2006). Tidwell (2006) also talks about color to be one of the factors influencing the user's emotional perception of the application. In our paper, we already discussed the important position of color, but we discussed that with color, you can draw attention of the user, evoke associations and so on. Color scheme is the first thing users see when they open application so it is important to ensure that color will not irritate user and cause him to immediately close your application because of fear and never come close to it.

The informational part of your application should be written with a font that is easy to read, you can make it as beautiful as you want, but if it is impossible to read it, there is no purpose to have it. You can write two of the same words in Photoshop (2012) and try to stylize them differently. Let them be same size but one word can be in thicker strokes, have lighter color or different textures; all of these settings will change the emotional background of the user. It is also possible to use different shapes and angles to cause the needed emotions from the user. If you have a rectangular form with round corners, it will make design to feel softer and vice versa if you draw a rectangle with the sides pushed a bit in the middle to center design becomes sharper (Tidwell, 2006).

Another important thing mentioned by Tidwell (2006) is texture. It adds richness to a visual design. Written text creates its own texture, which can be easily controlled by changing font, size, color and so on. Plain color is boring, which is why designers like to fill blank spaces where there is no information with simple textures, different textures can help achieve a different mood on the page. Never use eye hurting colors for textures, most effective textures are subtle (Tidwell, 2006). It is possible to use texture behind text, but it does not work if font size of the text is small or when the font color contrast to the texture is too small. To make it work good, designers fade a texture into a solid color as it approaches a block of text. Often pages have different pictures, either photos or iconic semi-abstract pictures the reason they have it is to support and increase the influence of existing mood created by shapes, forms, sizes, fonts, textures, colors, angles and curves. However, in applications and web applications the information load is on the text and not on visual appearance, content and ease of use is more important than style. However, it does not mean that designers

cannot use any pictures, but they should use purely decorative images sparingly and with great care on functional GUIs, since they tend to be destructive. Designers should look at the functional icons and images in their design, such as toolbar icons and illustrated choices and looking at those the designers will be able to see if all these icons and illustrated choices make the right emotional font which designer wants the whole application to make (Tidwell, 2006). Designers can play with color, sizes, shapes and so on, to make it fit as a whole.

Fifth principle: Make it look good.

IV. RESULTS

In this research paper, our primary research question was *"What principles of interaction design are conducive to the development of intuitive web applications?"* and throughout the research, we were able to conduct a literature review and distinguish principles for intuitive design, after that we proved that our principles worked in the real world, by composing interview and survey questions, which challenged our principles.

As a result of our data collection we got two types of data: qualitative and quantitative. To analyse the quantitative data, we used the method descriptive statistic that is described by Runeson & Höst (2009). Since the mixed method approach is flexible we carried out the analysis of the qualitative data in parallel with the collection of qualitative data (Runeson & Höst, 2009).

The results of the research are very interesting and exciting. On some level, the answers we got are the answers we were expecting to get because we are also users, and we are using different things everyday, both software and hardware. All the interview answers were positive towards our principles, and we were critical in analysing the data which we got from the surveys and interviews. Unfortunately not even one of the answers we got were against the principles, and if there was something contradicting it, it was too minor to influence the results. We have equated the results of the survey, and the interview answers to the principles we distinguished from the literature review. Due to the nature of the mixed method, the results of the research have been summarised and are presented in Table 1. This table is a representation of summarised results of our research where it is possible to see if the principles we distinguish from the literature are truly solid principles.

The results of the survey, which can be found in the Appendix B, presented in the diagram below. Each color of the column corresponds to answer that user have chosen: blue - answer #1 have been chosen, red - answer #2 and some of the questions had more than two answers, there you can see green color, which corresponds to the answer #3. The following diagram 1 shows the percentage ratio of the answers given by the survey respondents. These answers support the description of the principles given in Table 1.

Diagram 1. Percentage ratio of the answers given by the survey respondents
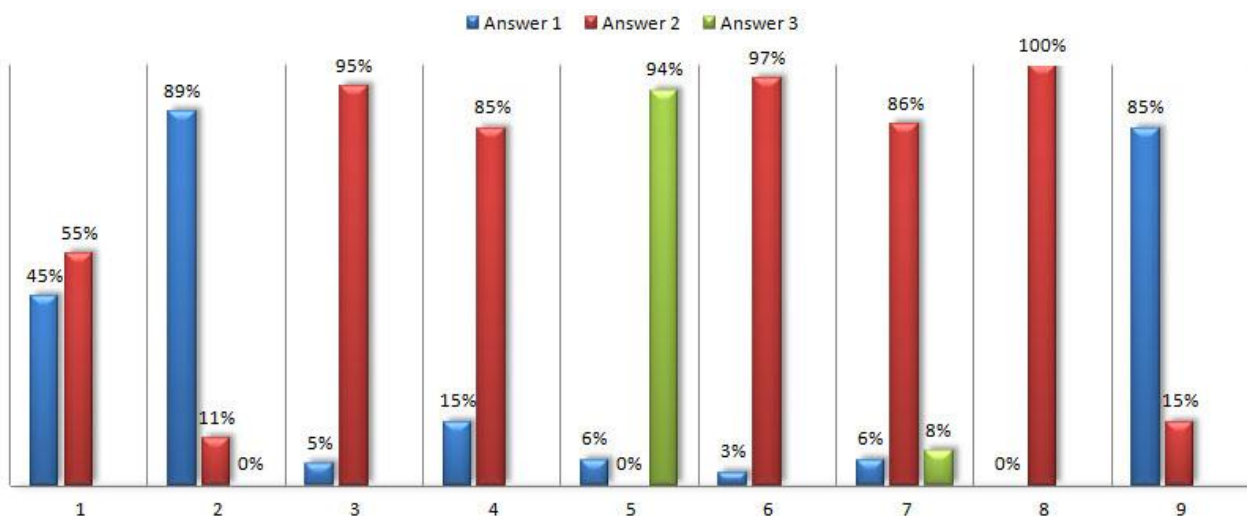
Principles	True(✔)/ False(✘)	Description
1st P: The designer needs to understand what the current and the target knowledge point of the future user is.	✔	Interviews and surveys showed us that some of the respondents demand much more from "intuitive design" than others, which means that they have different current knowledge points. Some of them had more previous experience with computers and software so their knowledge gap will be smaller. One of the interviewees said: "What might be intuitive for someone might not be intuitive for someone else, and it really depends on the end users" When the designer works with a design, he needs to think of how to fill the biggest knowledge gap because it is not irritating for experienced users to use software where every step is explained, but it is frustrating for less experienced user to wonder how to do a certain thing and seek for help. Also from the interview: "Intuitive design for me means that the learning time for using the application is zero," "In intuitive design, you don't get to know GUI, the GUI knows you," these quotes prove the importance of knowing the knowledge gap of each user.
2nd P: Design should be simple and familiar.	✔	The interview respondents were often referring to Windows, Mac, Android, Apple and other well-known brands when they were asked about "intuitive design". "I understand what I see, and understand what I can do," said one of the interviewees, "I do not have to focus on how to do things, just on what I want to achieve." confirming our second principle. Familiar things are always easier to understand "I think that the best icons are the one that are standardized.", said another.
3rd P: Think about color, size, positioning and grouping.	✔	One interviewee said: "...users have an idea of specific meanings tied to colors, icons... " and the survey also proves the importance of this principle: while positioning does not seem to have the highest priority, it has a certain impact of the users' perception of the design, however, the size, color and grouping seem to affect it more.
4th P: Minimise memory load on user.	✔	"There should be hints on what will happen for each action I choose" said our interviewee, he wanted to keep his mind concentrated on the task and not on remembering functions. "There should be a way to get back to where I were one or several actions/steps before, possibly even undo" these are demands from the interface. One more interviewee mentions that it is not so good if different words, situations or actions mean same thing. This makes the user confused.
5th P: Make it look good.	✔	The survey showed us what looks good and seems more solid to use. It is easy to perceive visual information if it is organized in an appropriate way. All our interviewees showed us "their own" icons of simple actions and those icons were simple to use, understand, and they were looking good. "GUI should look nice and give me wish to use the application again."

Table 1. Results of the research

V. DISCUSSION

Method Evaluation: The method which we decided to utilise to produce this paper has worked out very well for us, a literature review to act as a foundation for the study which was supported by doing interviews and a survey. The method loop used to systematically collect articles and literature for the literature review also went well and it helped us to keep track of the articles. The interview method which we used also went well, the first face-to-face interviews made us think about how

exactly to structure the questions properly to get the answers that we were expecting. We were also pleased with how the email interviews turned out because the interviewees were given more time to think and formulate their answers. The survey method that was also very successful, we got even more answers than we anticipated.

It is possible to make a research like this by using a different type of method, for example, a literature review would have been enough to find the principles, however that would not support the conclusions more than from the literature, which is exactly what we wanted to avoid by having the interviews and the survey too. Another method of doing this research is to do a literature review, construct principles and then, with those principles you can implement an experiment application and let people use it to see if they think that the design is solid.

Results: One of the interesting sides of the research was that interviewees were asked to use their imagination and create their own icons for the most common used features. This strategy of designing interview questions and survey made a very good contribution into the research process allowing us, while developing guidelines, to focus on what user would like to see. It is possible to continue the research further and apply the guidelines in practice, to conduct an experiment where an application is developed and then collect feedback from the users. There was a lot of theoretical work in this research. Different books and articles describing HCI and design from different aspects. The literature review helped us to summarise books and related articles in five principles, which touch upon the most critical parts of the intuitive design development. The interpretation of the interview answers and survey results support the principles that we extracted from the literature review and showed us that we are on the right track. The research resulted in carefully composed guidelines (See Appendix C) which should help designers not only to develop a good application but develop an appropriate application from the intuitive point of view.

Future research: Limited time frames did not allow us to conduct interviews outside of Systemite, if we would be able to do more interviews, it would let us to target a broader variety of people that would have different knowledge gaps. For future research, it would be interesting to do an experiment applying the guidelines on applications, web pages, web applications and other software, which have user interaction part and compare the results. In our research, we have chosen mixed method approach however it can be possible to conduct another research with different approaches

to prove our principles. A research question that would complement this paper in a future study based on the suggested future research could be: "Would these five principles work in practice?".

References

Blackler, A., Popovic, V. & Mahar, D. (2006) Towards a Design Methodology for Applying Intuitive Interaction. In Friedman, Ken, Love, Terence, Côrte-Real, Eduardo, & Rust, Chris (Eds.) *WonderGround: 2006 Design Research Society International Conference*, November 1-4 2006, Lisbon. [online] Available at: <http://eprints.qut.edu.au/8146/> [Accessed May 9, 2012]

Blackler, A., Popovic, V. & Mahar, D. (2007) Developing and Testing a Methodology for Designing for Intuitive Interaction. In Poggenpohl, Sharon (Ed.) *International Association for Societies of Design Research 2007: Emerging Trends in Design Research*, 12th - 15th November 2007, Hong Kong. [online] Available at: <http://eprints.qut.edu.au/13715/> [Accessed May 8, 2012]

Card, S.K. (1982) User perceptual mechanisms in the search of computer command menus. *Human Factors in Computer Systems.* 190-196.

Creswell, W. J. (2009) *Research Design: Qualitative, Quantitative, and Mixed Methods Approaches* (3rd Edition), Sage Publications: London.

css Zen Garden (2012) *css Zen Garden: The Beauty in CSS Design.* [online] Available at: <http://csszengarden.com> [Accessed May 14, 2012].

Free Online Survey, Web Poll or Quiz (2012) *Create a Free Online Survey, Web Poll or Quiz.* [online] Available at: <http://freeonlinesurveys.com/> [Accessed May 11, 2012]

Google docs (2012) [online] Available at: <http://docs.google.com>

Gould, J.D. & Lewis, C. (1985) Designing for Usability: Key Principles and What Designer Think. *Communications of the ACM*, 28:3, 300-307.

Hanselman, S. (2012) [online] Available at: <http://www.hanselman.com/blog/DeveloperDesigner.aspx> [Accessed May 31, 2012]

Hobart, J. (1995) Principles of Good GUI Design. [online] Available at: <http://www.classicsys.com/css06/cfm/article.cfm?articleid=20> [Accessed May 9, 2012]

IEEE Xplore (2012) *IEEE Xplore - Home.* [online] Available at: <http://ieeexplore.ieee.org> [Accessed 17 May 2012]

Krug, S. (2006) *Don't Make Me Think! A Common Sense Approach to Web Usability* (2nd Edition), New Riders: California.

Mandel, T. (1997) *The Elements of User Interface Design* (1st Edition), Wiley: USA

McKay, E. (2010, June 8th) Intuitive UI: What the heck is it? [Blog post online] Available at: <http://www.uxdesignedge.com/2010/06/intuitive-ui-what-the-heck-is-it/> [Accessed May 4, 2012].

Norman, D.A. (2006) *Design of Everyday Things* (Russian Edition), Wiliams: Moscow.

Photoshop (2012) *Image editor software | Adobe Photoshop CS6* [online] Available at: <http://www.adobe.com/en/products/photoshop.html> [Accessed May 17, 2012].

Runeson, P. & Höst, M. (2009) Guidelines for conducting and reporting case study research in software engineering. *Journal of Empirical Software Engineering*, 14, 131–164.

Sharp, H., Rogers, Y. & Preece, J. (2006) Interaction Design: beyond human-computer interaction (2nd Edition), Graphos: Barcelona.

Shneiderman, B. (1992) *Designing the user interface: strategies for effective human-computer interaction* (2nd Edition), Addison-Wesley Publishing Company: USA.

Software Engineering Body of Knowledge (2012) [online] Available at: <http://www.computer.org/portal/web/swebok> [Accessed May 17, 2012]

Spool, J. M. (2005) What Makes a Design Seem 'Intuitive'? [online] Available at: <http://www.uie.com/articles/design_intuitive> [Accesed May 6, 2012]

The Web Credibility Project - Stanford University (2012) *The Web Credibility Project - Stanford University*. [online] Available at: <http://credibility.stanford.edu/> [Accessed May 11, 2012].

Tidwell, J. (2006) *Designing Interfaces: Patterns for Effective Interaction Design*. O'Reilly: USA.

Wikipedia The Free Encyclopedia, 2012. Intuition (psychology). [online] Available at: <http://en.wikipedia.org/wiki/Intuition_(knowledge)> [Accessed May 4, 2012]

Windows Dev Center (2012, March 7th) User Interface Principles. [online] Available at: <http://msdn.microsoft.com/en-us/library/windows/desktop/ff728831(v=vs.85).aspx> [Accessed May 8, 2012]

Appendix A - Interview

Initial interview questions:

- What do you think when you read the words "intuitive design"?
- What colors, forms and positions do you associate with intuitive design?
- What do you gain from having a design which can be perceived as intuitive?
- Describe or draw icons (your own perception) for following actions: "new document", "open document", "add picture", "delete", "close", "draw", "add diagram", "add new user", "assign to …" (more?)
- Imagine you are the designer: describe how you would design a simple window (i.e. size, color, what buttons used, their position) or draw it.

Revised interview questions:

1. What in your opinion makes a user interface easy to use?
2. What do you think when you read the words "intuitive design"?
3. What colors, forms and positions do you associate with intuitive design?
4. What do you gain from having a design which can be perceived as intuitive?
5. Describe or draw icons (your own perception) for the following actions: "new document", "open document", "add picture", "delete", "close", "add new user" and "assign to …".
6. Imagine that you are a designer: describe how you would design a simple application window (i.e. size, color, what buttons are used, their position) or draw it (it is not a window for a specific application that we are looking for; the important part here is for us to understand what the most standard perception of a window is. I.e. come up with some kind of window with a few controls on it (buttons, textboxes, etc. and place them where you feel they belong).

Appendix B – Survey

1. Which window looks better?

1.

2.

2. Which button is more appropriate to use?

1. Start   2. Start   3. Start
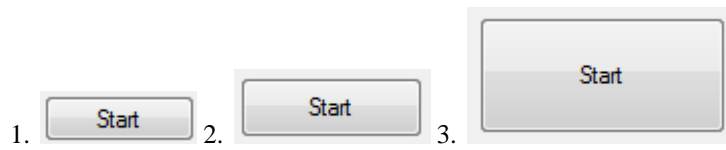
3. Which slider is more appropriate to use?

1. 0 ... 66   2. 66 ... 0
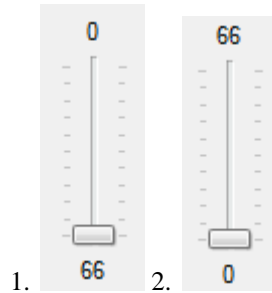
4. Which design is more suitable?

1. Review Raw Data | Create Estimates | Select Report Options | Create Report

2. Create Report | Select Report Options | Create | Review Raw Data
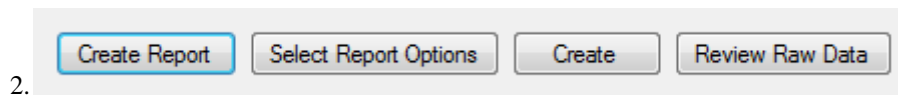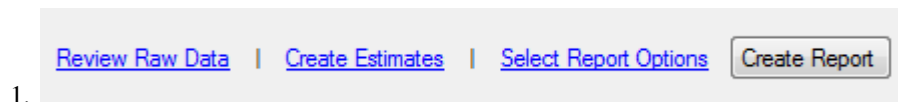
5. Which menu layout would you prefer?

1.
File
New   Ctrl+N
Open...   Ctrl+O
Save   Ctrl+S
Save As...
Page Setup...
Print...   Ctrl+P
Exit

2.
File
Exit
New   Ctrl+N
Open...   Ctrl+O
Page Setup...
Print...   Ctrl+P
Save   Ctrl+S
Save As...

3.
File
New   Ctrl+N
Open...   Ctrl+O
Save   Ctrl+S
Save As...
Page Setup...
Print...   Ctrl+P
Exit
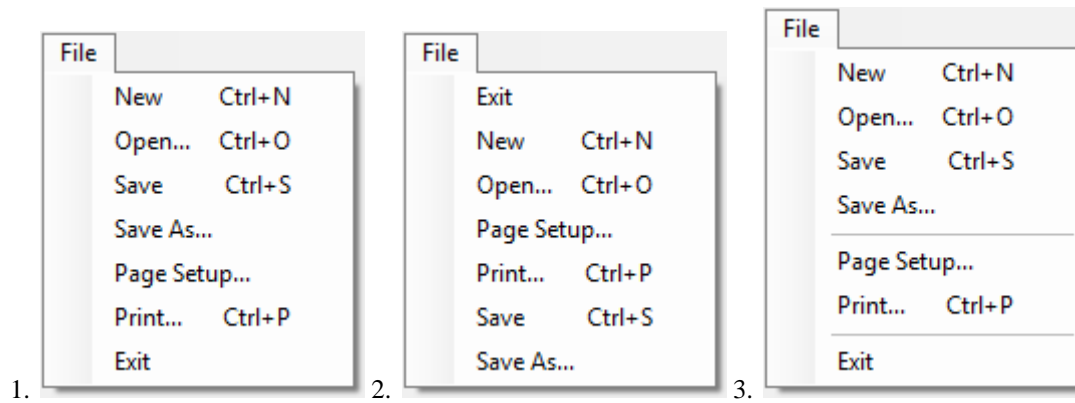
6. Is this warning message appropriate?
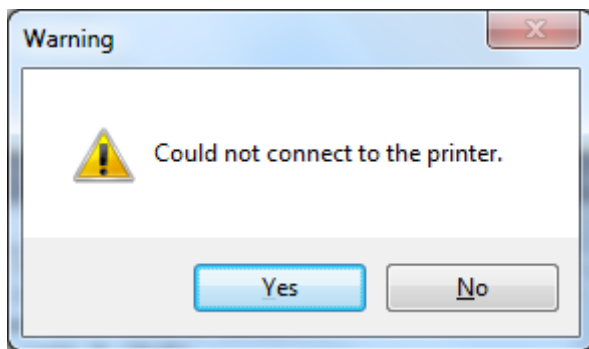
7. Which search looks better?

1.

2.

3.
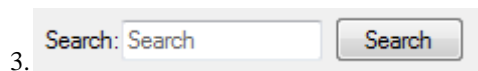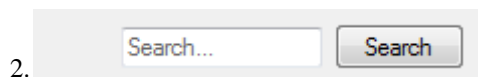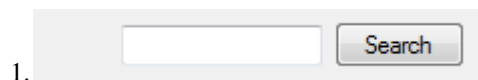
8. This is an appropriate way to present information to the user

9. This is an appropriate way to present information to the user

Appendix C- Guidelines

To develop an appropriate web application you need to follow these principles:

1st P: The designer needs to understand what the current and the target knowledge point of the future user is.

Every user has current and target knowledge. Current is what the user knows right now, and target is what the user needs to know to be able to use a particular software. Distance between these two points is called knowledge gap. The goal of intuitive design is to create a bridge for the user over this knowledge gap so it is easy for him to get from his current knowledge point to the target point. To do so you need to think carefully about the designation of your design and what users will need to know to be able to use your software. For example, to be able to test if any knowledge gap is present in your design you can create a prototype and ask users to give you feedback.

2nd P: Design should be simple and familiar.

When designing something new it is important to keep in mind that not many users like changes. Do not try to dramatically change the design of everyday functions or make them fancier than they should be.

3rd P: Think about color, size, positioning and grouping.

It is always easier for user to perceive information, which is presented in appropriate size and color. Do not try to make it bigger or smaller to attract the user to the information you want the user to see and remember. It is important to use natural colors; however, if your design requires color contrast make sure that it does not irritate a user. The position of controls on the screen is very important; you should remember about cultural differences and that Europeans read information from left to right, from top to bottom. It is a good practice to group similar chunks of information or design to make sure that the user can find whatever he needs.

4th P: Minimal memory load on user.

Keep in mind that user does not want to remember any unrelated information that is might be the reason why he is using your software. Try to avoid situations where a user needs to remember what certain function does, or certain icons means, what he saw on the previous screen or somewhere else. To avoid overloading user memory it is good to give the user hint about what is going to happen if he does certain thing. Try to use more drop down menus with different choices to show him names/things/actions he can use. You also can use association technique, but you should be very careful with it. You do not want the user to associate your serious unique software with application for kids.

5th P: Make it look good.

This is a very important principle because without it is impossible to create a good design. You can add different graphics, which will create a certain mood for your application but also keep in mind that the liberty of creating graphics depends on the purpose of your program.