Usability and user generated content: Web 2.0 sites with complex data structures

Wouter Kool
w.l.c.kool@student.utwente.nl

ABSTRACT
The use of the Web is changing and users demand more collaborative and interactive web environments. A philosophy in which several ideas and techniques are combined to meet these expectations is called Web 2.0. The success of Web 2.0 applications depends on the amount of people that use it. Therefore website usability becomes even more important than it already was. This research focuses on finding usability issues of Web 2.0 websites with complex data structures. Theorymaps, an existing Web application in which users can create, visualize and compare research theories has been tested using ‘Thinking aloud’ testing. The retrieved problems are analyzed and classified. Based on the results found in this case study future research is proposed.

Keywords
Web 2.0, Rich Internet Application, usability, human computer interaction

1. INTRODUCTION
During the last few years the use of the Web has been changing rapidly. As websites exclusively based on HTML are no longer considered useful enough, more websites are changed to interactive services. When the Web was invented and the first web client was developed by Tim Berners-Lee [27], it was not only used to read the Web, but also to write it. However, when the popular web browser Mosaic made the Web accessible for a large community, the possibility to easily create content within the Web browser disappeared [27]. Now, a new trend in web application design, which focuses on user participation and collective intelligence, instead of one way communication between website owner and user, arises. This trend conveys a broad area of techniques and ideas which often is referred to as Web 2.0 [25].

With Web 2.0 great possibilities and opportunities in communication and knowledge sharing arise. Consequently more and more companies adapt their websites to the new Web 2.0 ideas to give the users the experience they want, by creating interactive websites and web applications which provide participation, collaboration and rich information. This change will result in new usability issues, because people are used to and understand the page-based model of the Web [26].

However, due to the fast changing environment and the great technical possibilities, designers focus on efficiency and effectiveness, but often overlook principles of good design and usability of their applications [26].

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

12th Twente Student Conference on IT, January 22, 2010.
Copyright 2010, University of Twente, Faculty of Electrical Engineering, Mathematics and Computer Science.

In Web 2.0 the user is the main value creator and therefore user experience is an important factor, however little research has been done about usability concerning Web 2.0 or user generated content.

Many articles can be found that discuss the phenomenon Web 2.0, its characteristics and the techniques behind it [5,9,12,17,25,26]. Some of these articles also say that Web 2.0 will result in new usability issues; however the kind of issues is not discussed. Only one article, written by Nielsen [24], about the kind of problems that occur with these websites, was found. Nielsen says that many websites fail to get the basics of their site right because they rush too much to Web 2.0 functionality. This functionality does not work without good basics. Although these problems occur with Web 2.0 websites, they are not directly related to Web 2.0 technologies. They exist because old principles of design are neglected.

Research about usability regarding Web 2.0 environments with complex data was not found.

This research aims to explore and analyze the usability issues of a Web 2.0 application with complex data structures. A web application called ‘Theorymaps’ [33], which people can use to create, visualize and compare complex data structures will be used [18,19].

In this study complex data structures are causal relations proposed in scientific articles. Further clarification on complex data structures will be given in Chapter 2.

The main research question is:
What usability issues occur using a Web 2.0 application, working with complex data structures?

2. RESEARCH BACKGROUND

2.1 The Web

2.1.1 Web 2.0

The term Web 2.0 was first used by Tim O’Reilly in 2004 [26]. He concluded that the dot-com crash was a turning point for the Web. The characteristics of the websites that survived the crash marked and started a new philosophy. Many researchers have tried to specifically describe what the term includes, but a complete consensus is not yet reached. This is partly due to the fact that Web 2.0 consists of a broad area of new ideas to use the Web rather than a new technology [31].

Hoeg et al. [35] describe Web 2.0 as “the philosophy of mutually maximizing collective intelligence and added value for each participant by formalized and dynamic information sharing and creation”. Collective intelligence can be described as all available knowledge in a network of people and computers, which combined contains more intelligence than every individual person, group or computer can have by itself [13]. Hoeg’s description means that the Web must not longer be seen as a one way communication network from companies and web designers to consumers, but a platform where everybody can easily access, share and design their own content. A Web 2.0 platform consists of providers which facilitate the way of
communication, but the main factor that creates value is the user, who provides to the platform vast amount of content which is available for every other user in the world [12].

Implementing the ideas of Web 2.0 in their websites and Web applications will help organizations to adapt to changes in their environment and keep their competitive positions. Globalization and a shift towards a knowledge-based economy require a need for change in information management and value creation which Web 2.0 can provide [35]. In today's Web environment users demand free, democratic, interactive and rich services and content [17]. The fulfillments of these requirements determine the user experience or the level of user satisfaction. A good experience will keep users coming back and will let the amount of valuable information that is created by them grow. In the knowledge-based economy this information has become a main source of value creation [35].

2.1.2 Characteristics

The features of websites that fit the Web 2.0 philosophy are fundamentally different than those of websites which do not. Web 2.0 providers know that it is important to reach as many people as possible, because the more people will use their product, the more people are attracted and the more rich, extended and dynamic the information shared on their website will be [12]. To make their website accessible for a wide variety of users which want to participate and collaborate with each other, they make their service straightforward, because “users will not spend too much time or effort into a website they cannot rapidly decipher” [12].

Besides intuitivism the website has to appeal, which means it has to be dynamic and interactive, it has to contain rich information and it has to have a social network structure. In this context a network structure means the users are not only connected to the service provider, but also to other users and their created content. Websites or web applications that fulfill these requirements are behaving more and more like desktop applications [36]. However, the user has the benefits of being connected and does not have to download and install software. Furthermore, services can be updated and tested in real time with real users and without explicit user participation. Christopher Alexander [25] described this as ‘the perpetual beta’.

Another important characteristic of Web 2.0 is ‘users as co-developers’: for example, many Web 2.0 sites have a user-generated ‘folksonomy’ instead of the traditional hierarchical taxonomy [32].

The last important difference between Web 2.0 and Web 1.0 is the use of data. Web 2.0 has similarities with the open source movement, while traditional software developers keep their know-how desperately within the company. Open Source software can be used and changed by any user. This results in dynamic software that can be easily adapted to user’s different needs. Web 2.0 providers often also implement the idea of free distribution and use to gain a large amount of user data. For example, Youtube uses closed source software, but everybody can embed the videos on their own site. Moreover Web 2.0 providers sometimes provide users with open Web Application Programming Interface (API), a set of definitions which can be used for communication between an application and another software program. An API can also be used to develop applications that use the data of the Web 2.0 site. A good example is Google Maps [10], which has an API that can be used to embed the application on a website and to manipulate and add content to the maps. Thus in this trend not only the official producer can develop an interface, but everybody can [32]. As a result, more web applications that collect data from several sources and display them in one interface are developed. Such application is called a mashup [5].

2.1.3 Rich Internet Applications

Traditional internet applications are build on the 2-tier client-server model and use exclusively HyperText Markup Language (HTML) to present content. However, HTML alone cannot provide users with dynamic applications. An important technological trend related to Web 2.0 is the Rich Internet Application (RIA). RIAs use asynchronous techniques such as AJAX to enable real time data communication and dynamic rendering [9]. These applications provide the user with a dynamic interface which is quite similar to an interface of a desktop application.

The most well-known type of RIA is created using ‘Asynchronous Javascript and XML’ (AJAX) [9]. AJAX is not a technique on its own; it is a combination of technologies. The most important technologies are JavaScript, XHTML, XML, CSS, XML, XSLT and XMLHttpRequest. These technologies make it possible to update only the necessary part of a webpage, which decreases the required bandwidth [36]. AJAX also makes it possible to fetch data in the background. This results in smooth interaction, like a desktop application, because the user interface remains usable while handling the request.

Another aspect of RIAs is the 3-tier client-server model. Many Web 2.0 applications are based on this model [9], which consists of a client tier, an application tier and a database tier. In a 3-tier structure client requests are not directly sent to the database server, but to the application tier which can consist of Web servers and application servers. The Web servers can ensure load balancing by passing client requests to application servers which provide the application logic. To do this the application server uses data from the database server. This results in smaller response times. For example, if in a 2-tier structure a large group of users send requests both simultaneously and directly to a database server, the server could be overloaded. If instead requests are sent to a ‘middle tier’ which can do the calculations and logic and balance the requests over more different servers, the server load can be decreased.

Improved performance of RIA’s is also accomplished because more of the application’s functionality has been placed in the client browser. This means that the client provides extended functionality which reduces the amount of server communication [36].

2.2 Theorymaps

To explore usability problems in a Web 2.0 platform with complex data structures, a web application called Theorymaps [33] has been used to find these kinds of problems. In this section Theorymaps will be described and its relevance to Web 2.0 and complex data structures will be explained.

2.2.1 Description

Theorymaps is a tool to transform traditional scientific articles into causal maps [18, 19]. A user can create his/her own causal map by using a wizard which consists of three different steps:

1. Describe theory
2. Add variables
3. Add causal links

Completing these steps results in an automatically generated causal map. Now the theory can be compared to other theories
which use the same variables. It is also possible to map the correlation between variables when observing them or when doing experimental interventions. Henceforth different theories can be compared and resemblances and contradictions between them are detected. Besides correlation users can adjust the state of a variable (increasing, decreasing, constant) when observing it or with making interventions to see the effect on other variables within that theory. The tool provides users of new methods of visualizing, analyzing, comparing and searching for theories.

2.2.2 Web 2.0
Theorymaps can be labeled a Web 2.0 web application. It contains all the main characteristics mentioned in section 2.1, which are:
- Intuitive/straightforward
- Dynamic
- Interactive
- Collaborative

These characteristics should lead to a large pool of users and that leads to a website which has rich and extended information. Theorymaps asks users to create their own theories and compare them with theories of others. This is an interactive and dynamic process in which the provider of the website is only the facilitator and users together create a collective intelligence which becomes greater with every new theory that is added to the application. Being labeled Web 2.0 means that the website and the interactive functions have to be intuitive and therefore straightforward. That is the main factor the usability depends on.

2.2.3 Complex data structures
Theorymaps is a web application in which causal maps can be created. A causal map is an example of a complex data structure. A data structure can be defined as: a way in which sets of data are organized in a particular system. Data structures that are simple do not have an extended and complicated organization. An example is a ‘tag’. Tags are keywords used to describe a part of content, for example a picture on Flickr or a video on YouTube. They can be used to easily find other content that contains the information the tag refers to. The tag and the content it refers to have a one link structure, a simple data structure. A complex data structure consists of a data organization which looks like a graph. In Theorymaps variables of a theory can be linked to every other variable created. This can result in a complex structure in which not only variables are connected but different theories can be linked to each other as well. These structures are complex data structures.

2.3 Usability

2.3.1 Definition
A definition of usability is given by Dumas and Redish [7]: “Usability means that the people who use a product can do so quickly and easily to accomplish their own tasks”. The definition presumes four points:
- Usability means focusing on users
- People use products to be productive
- Users are busy people trying to accomplish tasks
- Users decide when a product is easy to use

Another definition more specialized to websites is given by Krug [15]. He states that a website is usable if it is self evident. A user should be able to understand what the website is about, how it should be used, without expending any effort thinking about it. He presumes three facts:

1. People do not read pages. They scan them
2. People do not make optimal choices. They satsifice.
3. People do not figure out how things work. They muddle through.

Satisficing is a word coined by Herbert Simon [15]. It is a combination of the words satisfying and sufficing, and it refers to a strategy in which a user chooses the first reasonable option, instead of searching for the best option. The three facts happen because for most people it does not matter if they understand how something works as long as they can use it. Furthermore people do not search for better ways once they found one. If a better options comes across they will use it, but they will seldom look for one [15].

2.3.2 Usability testing
Usability greatly influences user satisfaction and the users’ willingness to revisit the website [1]. Research [4] showed that poor usability resulted in failure of 40% of shop transactions and 50% of potential sales. Therefore usability is one of the most important quality aspects, especially for popular websites [2]. Because Web 2.0 websites (including Theorymaps) focuses on attracting as many users as possible, the importance of usability testing is even greater.

Many website designers know that their website has to be straightforward and intuitive, but they have the feeling that the results of usability testing will not even match the time and costs it absorbs [15]. However, still many website are not sufficiently usable. Designers fail to think from a user perspective in the right way. Knowledge about how the website works and moreover contextual information about the subjects and technical design makes it almost impossible to understand the way a user without this knowledge will react to the designer’s website [15].

2.3.3 Techniques
Usability testing is one of many techniques to make sure a website is well designed and people will be able to use the website without real problems. Other techniques are the heuristic evaluation or the cognitive walkthrough. These techniques however do not involve user testing. Nielsen [30] stated that the most effective source of information concerning the usability of a website is provided by laboratory testing of users. There are several methods which can be applied to user testing. Two of the most well known and widely used, are ‘Undertaking tasks’ and ‘Thinking aloud’ [1, 34].

Undertaking tasks is a method in which a test user browses a web page using one or more tasks created by the test designer. Thinking aloud means the test user is asked to say everything he/she thinks out loud during the test. To be able to detect and analyze problems, thinking aloud has proved to be a valuable observation method [34]. Both Undertaking tasks and Thinking aloud were used in this research.

3. METHODOLOGY

3.1 Qualitative research
There are different ways to do research depending on the research situation and desired results. In this paper a qualitative research instead of a quantitative research has been done. The aim of qualitative research is a complete and detailed description (of a situation). It is a subjective research which means the individual interpretation is import and the resulting data is more rich, but less useful for generalization. Quantitative
research consists of explaining situations in which events can be generalized, counted and statistically analyzed [28]. In usability testing a qualitative research is often used. Only a few test users are questioned in depth with the purpose of finding as much different individual problems instead of patterns or other generalizations. These problems and their causes are unpredictable and therefore every single result is useful on its own. For example, if a website designer wants to know if his navigation structure is clear to users, he does not want to know what percentage of a large group of test users says ‘yes’ or ‘no’, he wants to know what problems occur and why these problems occur.

3.2 Test users

3.2.1 Number of test users

Because a qualitative research aims to produce rich information only a few test users will be necessary to find the results I want. Nielsen [20, 21, 23] discovered that a good usability study does not have to involve a large number of test users. He studied the relationship between the number of test users and the percentage of usability problems found. He concluded that with a number of fifteen test users almost all problems would be found, but that the optimum is around four test users. The contribution of every extra user would fall exponentially (Figure 1). Nielsen calculated the optimum based on three assumptions:

1. Usability testing involves costs. Therefore using test users which do not discover new problems will make test less efficient and discovering all problems would be expensive.
2. The researchers attend to do iterative testing which means that after the first test round has taken place, the web application will be adjusted and another test round will be done.
3. The most severe problems are among the first problems found.

In case of the optimum amount of users the percentage of problems found would be approximately 75%. This is also supported by Krug [15].

In my research I did not face user compensation or other costs, however because of a short time span and the intention to do more test rounds I chose to use five test users.

The amount of five users also seemed sufficient according to the problems found during testing. The fourth test user discovered 26 problems. Only 6 of them were new problems. The last user found 4 new problems of a total of 30.

3.2.2 Profile

The test users were chosen without special requirements except that they had to be students or University staff. Using test users with a different background would not improve the quality of the results, because Theorymaps mainly aims at this target group.

Four of five participating test users are students and one is part of the University Staff. I intended to use test subjects with different backgrounds, interests and internet experience. Based on the information I had in advance about their profiles the subjects were chosen. All of them were asked about their study and online behavior to identify their internet experience and background. This information indeed turned out to be very varied from no technological background and few hours spend online per week to an (Information) Technology background and much internet experience (Table 1). This makes the pool of users a good reflection of the variety that will be encountered in the real target group.

<table>
<thead>
<tr>
<th>Table 1. User profile</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hour browsing</td>
</tr>
<tr>
<td>(week)</td>
</tr>
<tr>
<td>User 1</td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td>User 2</td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td>User 3</td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td>User 4</td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td>User 5</td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
</tbody>
</table>

3.3 Methods

The methods used in the testing are ‘thinking aloud’, face expression and screen capturing. The test users were asked to tell everything they thought while exploring the website. This sound was recorded and next to that a film camera recorded the facial expressions. Special software recorded every action on the screen. Afterwards these recordings could be synchronized which made it possible to analyze the data together at the same time.

During the tests the users had to complete two types of tasks: problematic tasks and structured tasks. Structured tasks are tasks in which the user is directed from page to page and has extensive guidelines. Problematic tasks have a goal, but no guidelines. For example, a user who has to order a specific book on an online book store faces a problematic task and a structured task is asking a user to find the search feature on the current web page to search for a specific book. After this task more structured tasks follow and eventually the same goal will be reached [1].

Research [1] showed that structured tasks reveal more problems on average, but problematic tasks reveal more severe problems.

In my tests the main tasks were problematic and between these tasks structured tasks were carried out. During both tasks direct questions about website appearance, choices and problems were posed. After completing all tasks I asked the users to mark the usability of Theorymaps. The tests took approximately fifty minutes. A script of the test can be found in Appendix A.
4. FINDINGS
The results of five usability tests are presented in the following. The first section lists the actual problems that were found. Section 4.2 and 4.3 illustrate a severity assessment and a heuristic classification, which is followed by an analysis of the results. Finally, in section 4.5 the test users’ satisfaction is presented.

4.1 Problems
Five tests resulted in a total of 27 usability problems. A general classification is used to structure the problems. The classes are ‘problems in website structure or appearance’, ‘problems in understanding the website’ and ‘problems in programming’.

4.1.1 Problems in website structure or appearance
P1. Clicking on the website logo does not navigate to the start page.
   The logo navigates to ‘Theories’ which is not the start page. Because the start page itself is not clear, this leads to extra confusion.

P2. White buttons on top of the page do not stand out enough.
   The white buttons are not seen by the test users unless they want to make a theory or adjust it. The buttons ‘Correlation’ and ‘What-if’ are not recognized and therefore the test users do not know what the possibilities of the website are. It looks like they do not expect the website functions having the same kind of button to make or adjust a theory.

P3. It is not clear enough that the variables are on the left and that you can drag them.
   Because the page has a lot of ‘attractive’ elements, the variables are not seen directly, while in the best situation to understand the function, the variables and corresponding explanation should be focused on first.

P4. The ‘what if’ button is interpreted as a help function.
   The question mark button is related by users to a help function. The label ‘what if’ enhances that believe.

P5. It should be possible to move variables directly within the table.
   There were situations where it was possible to move the variables directly, but not in all situations. This does not work efficiently.

P6. The possibility to drag a map during the wizard gives the idea it is directly adjustable.

P7. The titles in the colored area on top of the page are not read.
   The titles in the colored areas are not read because it seems to be connected with the headers. Apparently users do not connect the text in the header with the information on that web page. Therefore important information is unknown. For example, the concept ‘variable’ is not completely clear for every user and not knowing that a variable is able to increase or decrease, which is noted in the title, can lead to mistakes.

P8. It is not possible to connect variables to creators.
   If a user adds a variable that already exists in the system, a definition is added to this existing variable. The definitions can be different, but only the first creator of the variable (and first definition) is named.

If a second creator wants to use the variable in his theory, he may want to use the second definition, but he cannot indicate this.

P9. Error messages are not very prominent.

4.1.2 Problems in understanding the website
P10. It is not clear enough what the correlation function does.
   The term correlation itself is not sufficient to understand the function. The results were not understood or misinterpreted and even the fact that different theories could be compared was not always clear.

P11. It is not clear enough how to remove a variable.
   Test users did find the box after a short while, but it was clear that in an interactive environment it annoys people if they cannot find a simple function quickly.

P12. It is not clear enough what is meant by intervening (what if function).
   Test users did not really understand the difference between ‘observations’ and ‘experimental interventions’

P13. The navigation in the wizard is not clear enough.
   After completing the first step it is not possible to go back to this step. It is possible to go further, but there are two different ways to do this. One of the two options however does not have a button to finish the wizard. Using this option confuses the user about finishing the wizard. Almost all the test users expected to see navigation buttons at the bottom of the web page.

P14. The concept tag is not understood.
   The concept ‘tag’ is not known or understood by every user. Hence, people do not add tags and confuse them with variables and theories. Especially tags at the variables are confusing, because both often consist of only one word.

P15. It is not exactly clear what a scope is.

P16. The bookmark function is not clear and not interesting enough.
   There are a few problems concerning the ‘add bookmarklet’ page. The browser names can give the user the idea that the information is about browser compatibility which many users do not find interesting. The second problem is the use of the bookmarklet; it is not clear how to use the button. If the user found an article in Scopus it is not clear that the button has to be clicked. The third reason why users are not willing to use the feature is because they do not know what the function is and why it is advantageous to them. Some users will see it as a normal favorite link. The fourth problem is that some users do not have an English browser which can make it hard to find the bookmarks toolbar and because this is the first step, users can be discouraged. The last problem is that not every user knows differences in browsers and does not know which one to take or even the fact that they only have to do the steps of their own browser.

P17. The website does not contain a general explanation about its purpose and functionality.
   The main problem that occurred was the insufficient amount of information. The homepage is not
recognized as a homepage because there is not any information about how the website works. People expect more explanation and some help function which they can use when something is unclear. It is not possible to find out how the website works by just clicking around. Because users do not understand the idea and the working of the website other problems occur:

Problem: What is Scopus and what can I do exactly on this webpage?
Problem: Who are the ‘others’, mentioned in the first wizard form?
Problem: What is the connection between the theories on the ‘theory’ page and the website itself?
Problem: What are the relationships between the theories?
Problem: Can I enter my own theory or is it just a database/wiki?

These problems will result in a quickly abandonment of the website.

P18. It is not clear what changes if you register and log in.
It is not clear why a user should sign up. This is because the idea of the website is not clear at the beginning, but also because the users are not told and do not see the difference.

P19. It is hard to understand the meaning of ‘because’ in the ‘cause-and-effect’ links.

P20. The symbols + and - are seen as positive and negative, the concept variable is hard to understand.
As mentioned in the problem about the titles, the short explanation about a variable does not stand out. Also the words ‘cause’ and ‘effect’ are overlooked.

4.1.3 Problems in programming

P21. When enlarging a map from the relations graph it can fall off the screen.
The idea is good to use scroll bars, but the maps only occur when pointing the mouse to the relations graph. This makes it impossible to use the scrollbars.

P22. You lose the Scopus page after submitting the article.
It can be useful to check, adjust or add information from Scopus after using the ‘post to Theorymaps’ button.

P23. Password has to be entered again after clicking on an external link.
I do not know if this is possible by adjusting the code, but it is annoying if you enter a password and it is gone after visiting a provided external link.

P24. The update button in the wizard does not work all the time.
There are possible situations in the wizard when the update button does not work properly. It can be fixed after going back to the start, but it is not only a one time problem or delay.

P25. Website disappears during step 2 and 3 of the wizard.

P26. After clicking on ‘add citation’ it is not possible to use the browsers return button and to return to the same page you were before.
Many users are so used to the browsers ‘previous page’ button that they rather use that than a button on the webpage.

P27. Within the Safari explanation an extra space is needed: “Post to Theorymaps to your Bookmarks” Toolbar.

4.2 Severity assessment

A severity assessment classifies problems looking at their impact on the users’ ability to use the website and to accomplish the tasks they want to undertake. The classification I used has been used by Nielsen [1]. The five classes are:
0: A problem but not a usability problem
1: Superficial problem: should be fixed if there is enough time
2: Minor problem: low priority to be fixed
3: Major problem: important to be fixed
4: A usability disaster: imperative to be fixed

The assignment of a problem to a class is based on three factors: problem impact, persistence and frequency.

Table 2: problem severity

<table>
<thead>
<tr>
<th>Problem</th>
<th>Class</th>
</tr>
</thead>
<tbody>
<tr>
<td>P1</td>
<td></td>
</tr>
<tr>
<td>P2</td>
<td></td>
</tr>
<tr>
<td>P3</td>
<td></td>
</tr>
<tr>
<td>P4</td>
<td></td>
</tr>
<tr>
<td>P5</td>
<td></td>
</tr>
<tr>
<td>P6</td>
<td></td>
</tr>
<tr>
<td>P7</td>
<td></td>
</tr>
<tr>
<td>P8</td>
<td></td>
</tr>
<tr>
<td>P9</td>
<td></td>
</tr>
<tr>
<td>P10</td>
<td></td>
</tr>
<tr>
<td>P11</td>
<td></td>
</tr>
<tr>
<td>P12</td>
<td></td>
</tr>
<tr>
<td>P13</td>
<td></td>
</tr>
<tr>
<td>P14</td>
<td></td>
</tr>
<tr>
<td>P15</td>
<td></td>
</tr>
<tr>
<td>P16</td>
<td></td>
</tr>
<tr>
<td>P17</td>
<td></td>
</tr>
<tr>
<td>P18</td>
<td></td>
</tr>
<tr>
<td>P19</td>
<td></td>
</tr>
<tr>
<td>P20</td>
<td></td>
</tr>
<tr>
<td>P21</td>
<td></td>
</tr>
<tr>
<td>P22</td>
<td></td>
</tr>
<tr>
<td>P23</td>
<td></td>
</tr>
<tr>
<td>P24</td>
<td></td>
</tr>
<tr>
<td>P25</td>
<td></td>
</tr>
<tr>
<td>P26</td>
<td></td>
</tr>
<tr>
<td>P27</td>
<td></td>
</tr>
</tbody>
</table>
4.3 Heuristic classification

Heuristic evaluation is a usability evaluation technique that evaluates human-computer interaction. This means that it mainly focuses on the interface of an application. The technique, developed by Nielsen [11, 22], consists of ten heuristics which all represent a set of website characteristics. During an evaluation a panel of expert users uses the heuristics as a checklist to test the interface of an application. The heuristics can be found in Table 3.

In this research I did not test with experts, but with users that represent the actual audience, so I did not use this technique itself. However I did use the ten heuristics for classification purposes. The heuristics cover the important characteristics of a website. Therefore, every (interface) problem that was discovered during my tests would be related to one of the ten heuristics. This classification would give a good view on which kind of characteristics the focus should be regarding usability improvement. For this reason I classified the problems found using Nielsen’s heuristics (Table 3).

<table>
<thead>
<tr>
<th>Heuristic Classification</th>
<th>Number of problems</th>
<th>Problems</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Visibility of system status</td>
<td>1</td>
<td>P7</td>
</tr>
<tr>
<td>2. Match between system and the real world</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>3. User control and freedom</td>
<td>3</td>
<td>P8, P13, P22</td>
</tr>
<tr>
<td>4. Consistency and standards</td>
<td>3</td>
<td>P1, P3, P5</td>
</tr>
<tr>
<td>5. Error prevention</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>6. Recognition rather than recall</td>
<td>3</td>
<td>P2, P4, P25</td>
</tr>
<tr>
<td>7. Flexibility and efficiency of use</td>
<td>2</td>
<td>P6, P11</td>
</tr>
<tr>
<td>8. Aesthetic and minimalist design</td>
<td>1</td>
<td>P16</td>
</tr>
<tr>
<td>9. Help user recognize, diagnose and recover from errors</td>
<td>1</td>
<td>P9</td>
</tr>
<tr>
<td>10. Help and documentation</td>
<td>8</td>
<td>P10, P12, P14, P15, P17, P18, P19, P20</td>
</tr>
</tbody>
</table>

Note: P21, P23, P24, P26 and P27 are programming problems which cannot be classified using these heuristics.

4.4 Analysis

Although the website seemed really clear and usable before testing, 6 of the 27 problems that occurred are a usability disaster. Only 3 problems will not obstruct users of smoothly using the web application. Considering the heuristic classification I see that 8 of the 22 problems lie within heuristic 10: help and documentation.

Combining the severity and the class (heuristic) of the discovered problems, I see that from 10 of the problems with a severity code of 3 or 4, 80% is concentrated on only three heuristics: 3, 6 and 10. User control, Recognition and Help are the most troubled areas. Help and documentation can be considered the most important class; of the 6 disaster problems 3 are classified heuristic 10.

The direct usability problems were not the only outcome of the test sessions. Information about user behavior when browsing a website was also retrieved. This information could be important context information, when considering the obtained view based on the discovered usability problems.

The most interesting discovery was the way the test users browsed and used the web application. They explored most of the website by trial and error. When opening a webpage, they immediately focused on the graphs, maps and tables. They clicked what was logical to them and only if they got stuck they tried to read something. But even then they scanned a text rather than read it.

This means that every interaction that is possible on the website has to be intuitive. For example, every user tried to adjust the map by clicking and dragging it directly. One user said that if a website contains only of text he reads it, but when the website contains graphs or other more attractive parts, that willingness disappears. This explains why the greater part of the problems is related to information about the working, context and purpose of the application. It seems an interactive environment makes a user even more impatient than a more informative website. This impatience has to be compensated with an application that is totally clear and intuitive.

Another remarkable finding is that although the users sometimes spent a lot of time figuring something out (three users said that in real situations they would have decided to leave the website because it was too hard to understand), when I told them once how it was done, they immediately understood and the next time they needed the function, they used it without any problems. If this is true in general the outcome of the test should be interpreted different. Now, even though, most of the problems are related to the lack of information and are also the most severe problems, these are not durable problems. It is a great barrier when a user first uses an application, but once it is made clear, the overall understanding will result in flawless use of the web application. For example a simple help function meant for first timers could greatly improve a website.

4.5 User satisfaction

After the sessions, every test user was asked to mark the idea of working with theories in the way you can in Theorymaps and to mark the perceived usability. The marks presented in the table below are within a scale of 1 – 7 with seven as highest score.
<table>
<thead>
<tr>
<th>Name</th>
<th>Idea</th>
<th>Usability</th>
</tr>
</thead>
<tbody>
<tr>
<td>Test user 1</td>
<td>5</td>
<td>4</td>
</tr>
<tr>
<td>Test user 2</td>
<td>7</td>
<td>5</td>
</tr>
<tr>
<td>Test user 3</td>
<td>6.5</td>
<td>5</td>
</tr>
<tr>
<td>Test user 4</td>
<td>4</td>
<td>6</td>
</tr>
<tr>
<td>Test user 5</td>
<td>6</td>
<td>4</td>
</tr>
</tbody>
</table>

The results presented in the Table 4 are remarkable because most of the test users were asked during the test if they would have given up at that particular moment because they got stuck or they encountered too much difficulty, of which three said: ‘yes’. Furthermore, in some tests the user did not understand the purpose of the website and therefore would have stopped if it was not a test. Still the marks that are given are good to very good. This could be another proof that a user has to have enough information to make the website usable and therefore satisfactory. If a user says half way browsing he would quit if it was not a test and twenty minutes later he gives a good mark not only for the idea but also for the usability, the easiness of influencing the user is clear and the importance of knowing how a website can be useful and how it works is of great importance.

5. DISCUSSION

This research has resulted a clear set of usability issues of a Web 2.0 application with complex data structures. The issues give a small insight in what Web application designers should focus on and how users react on this kind of applications. Nevertheless, this research is mainly a case study and therefore more research to support the findings has to be done. The results from Theorymaps showed a high concentration of problems related to information about the purpose of the website and about its functionality. It seems reasonable to believe that this issue will be experienced in more Web 2.0 website with complex data structures, because the element of interactivity that is shared by all these applications seems to decrease the user’s willingness to read and use explaining documentation. This implies that an important aspect of usability in Web 2.0 with complex data is implementing a successful way of communicating important information to users.

Further research could verify these findings by testing more applications in the field of Web 2.0 and complex data structures. Hypothesis which could be used are:

**H1:** Users use trial and error instead of reading explaining documentation.

**H2:** Users who do not know the purpose of the website from the home page will encounter more difficulties than user who do know.

**H3:** Users who understand the concept of a function will encounter no problems when using the same function or a function with the same concept again.

**H4:** Users are willing to do an explaining website tour the first time they visit the website (already focusing on a possible solution).

If these hypotheses would be researched, designers would have more supporting evidence and information about the problem of communicating necessary information to user. Knowing more precisely where the difficulties are would be the basis for improved usability.

6. CONCLUSIONS

As discussed in Chapter 1, one of the most important characteristics of Web 2.0 found in literature is the ease of use; a website has to be straightforward. In this research I observed that most severe problems occurred because of the lack of sufficient information about the purpose and working of the web application. The research not only illustrated Web 2.0 usability problems, the research also made clear that in an interactive environment users will use mainly ‘trial and error’ to discover how an application works instead of searching for and reading the text that explains it. This could be an important reason why Web 2.0 applications should be intuitive and straightforward. Consequently, making decisions about what information stands out and the way it stands out becomes more important than it already was. Further research about user behavior in Web 2.0 applications with complex data structures has to be done to verify and define the problems of these applications more precisely.

7. ACKNOWLEDGMENTS

I would like to thank the students and University Staff who participated in the usability tests, for their time and dedication to the test. I also want to thank Roland Müller for his guidance and support throughout the project.

8. REFERENCES


Appendix A: Script test sessions

This script is a test script provided by Steven Krug [15]. The first part is about comforting the test user and getting some information about his surfing experience. After the questions, the user is asked to do five main (problematic) tasks and during each task he will be asked more specific (simple) questions. The text below is only a script and has been continuously adjusted during the sessions in order to retrieve as many information as possible.

Hi, _______. My name is Wouter Kool, and I am going to be walking you through this session.

You probably already know, but let me explain why we have asked you to come here today: We are testing a web site that is already in use, but we want to know how to improve it.

I want to make it clear right away that we are testing the site, not you. You cannot do anything wrong here. In fact, this is probably the one place today where you do not have to worry about making mistakes.

We want to hear exactly what you think, so please do not worry that you are going to hurt our feelings. We want to improve it, so we need to know honestly what you think.

As we go along, I am going to ask you to think out loud, to tell me what is going through your mind. This will help us.

If you have questions, just ask. I may not be able to answer them right away, since we are interested in how people do when they do not have someone sitting next to them, but I will try to answer any questions you still have when we are done.

You may have noticed the Video camera. With your permission, we are going to record what happens on the computer screen and what you have to say. The recording will be used only to help us figure out how to improve the site, and it will not be seen by anyone except me and my supervisor. It also helps me, because I do not have to take many notes.

Do you have any questions before we begin?

Before we look at the site, I would like to ask you just a few quick questions. First, what do you study?

Good. Now, roughly how many hours a week would you say you spend using the Internet?

How do you spend that time? In a typical day, for instance, tell me what you do, at work and at home.

Do you have any favorite Web sites?

OK, great. We are done with the questions, and we can start looking at things.

First, I am just going to ask you to look at this page and tell me what you think it is, what strikes you about it, and what you think you would click on first.

And again, as much as possible, it will help us if you can try to think out loud so we know what you are thinking about.

From this point the text below shows the main questions/subject. Every question will have a conversation and sub questions in reaction of what happens during the test. The sub questions cannot be determined in advance.

- Now you know what the website is about. Can you try to sign up?
- Do you understand what you can do on this page? (Scopus tool)
- Could you do what it says and add a source via Scopus?
- Could you try to add the text I send you in the email as a source?
- Could you try to make your own Theory Map about the text I send you in the email?
- Could you try to find the page where you can predict the behavior of a variable in your theory when adjusting another one?
- Could you try to find the theories about Health?
- Could you find the correlation between ‘Smoking’ and ‘Cancer’ in these theories?

During the testing of the four main questions, more specific questions (simple tasks) will be asked, for example:

- Can you explain what you see?
- Do you like this/that?
- Do you understand what you can do?
- What kind of actions would be the first you would do?
- Why do(n’t) you click that button?
- Do you know what that button does?

I have one last question for you, if you had to rate your satisfaction concerning this website on a scale of 1 which is satisfactory to 7 which is most unsatisfactory, what number would it be?