A research on usability problems of frequent and infrequent users

A case study of the RaboShop

Author
Sanne Derckx

Student number
3014738

Date
31-03-2015

Study
Master Information Science
Radboud University Nijmegen

Supervisor
prof. dr. Erik Barendsen

Second examiner
dr. Stijn Hoppenbrouwers

External supervisors
Jolanda de Mooij (Rabobank)
Job van der Sijs (Rabobank)

Rabobank

Radboud University

Abstract

This research tried to find the differences in usability issues of ERP systems between frequent and infrequent users. The RaboShop from the Rabobank was used as a case study. Multiple research methods were used to find an answer to this question: heuristic evaluation, interviews and user tests. First the heuristic evaluation was conducted which resulted in a list of usability issues of the RaboShop. After that interviews and user tests were conducted with 7 test persons, 4 infrequent and 3 frequent users. The results of these methods were coded and this resulted in a list of usability criteria that described the usability issues of the system. The following usability criteria were found: navigation, task support, errors, search function, learnability and UI presentation. There was found a slight difference in usability issues between the frequent and infrequent users. They mostly experience the same usability issues, however they occurred more often with the infrequent users and they did experience more trouble because of those issues.

Acknowledgements

The following research was conducted as part of the Master Information Science at the Radboud University in Nijmegen. This thesis was written as a graduation assignment of the Rabobank. I would like to thank Jolanda de Mooij and Job van der Sijs for giving me the opportunity and support to graduate at the Rabobank and also all employees who helped and supported me. A special thank you for Erik Barendsen, who guided me when I was lost and was a great help in the writing of this thesis. I also would like to thank my parents, who always support me no matter what, and my sister for the coffee breaks. And last but not least I would like to thank Jasper for giving me inspiration and motivation and all my friends for the support.

Index

1. Introduction

Nowadays a lot of companies have implemented an Enterprise Resource Planning system (ERP). An ERP system can be thought of as a companywide information system that integrates all aspects of a business. It promises one database, one application, and a unified interface across the entire enterprise. An entire company under one application roof means everything from human resources, accounting, sales, manufacturing, distribution, and supply-chain management are tightly integrated. This integration benefits companies in many ways: quick reaction to competitive pressures and market opportunities, more flexible product configurations, reduced inventory, and tightened supply chain links. (Bingi, Sharma & Godla, 1999). ERP systems are very expensive, but they can increase the productivity of employees since all information can be found in one system. Commonly used ERP systems are SAP, Oracle and Microsoft Dynamics. Because of the widespread demand and the magnitude expenses, academic researchers and practitioners have paid a great attention to ERP systems (Kwak, Park, Chung & Ghosh, 2012). The main topic of the studies is the implementation of ERP systems. Because of the size of such systems, implementing it is complex. This has led to a number of big failures (Calisir & Calisir, 2000). Causes of this failures are: poorly defined business goals, inadequate training, the lack of strong and adequate sponsorship, changes in the project scope, and incompatibility among the various computer hardware and software systems (Calisir & Calisir, 2000).

But what does make an ERP system a success? Studies suggested that user satisfaction is one of the key factors leading to information system success and that the usability of interfaces can be seen as one of the factors that influences end-user satisfaction (Calisir & Calisir, 2000). Usability problems may not lead directly to a large-scale failure, but they can interfere with an individual's or workgroup's productivity, making it harder for users to achieve their goals as effectively or efficiently as is desirable. They may also make system acceptance a more difficult and lengthy process (Topi, Lucas & Babaian, 2005).

Because of the importance and costs that are involved with ERP systems, the system should be easy to work with and increase the productivity of employees. Therefore research about the usability of large-scale information systems, like ERP systems, is necessary. Researchers already found some usability issues using multiple research methods (Topi, Lucas and Babaian, 2005) and also improved some research methods to find these issues (Singh and Wesson, 2009). However they all mentioned that more research is needed with more different systems to discover all usability issues.

1.1 Problem statement

If systems are not user-friendly one of the possible outcomes is that the return on investment will decline when very few users actually adopt the system or only use it infrequently. According to SAP their systems are also used much by infrequent users and it is important to also involve these users in the implementation of a system to maximize the benefits and value of the system ("Drive SAP HCM adoption", n.d.; Johnson, 2012). Topi, Lucas and Babaian identified some usability issues of ERP systems. In order to get these issues they interviewed a number of users who used a particular ERP system on a daily basis. Involving infrequent users in such research could lead to more insight in the usability issues of a system, at least if there is a difference between their issues. In already existing literature on the topic of usability of ERP systems the differences between frequent and infrequent users were never researched. This raised the following research question:

Do infrequent users experience the same usability issues as frequent users?

In the already existing literature about this topic only interviews and heuristic evaluations were used. This raised the question if a different method, like user tests, would lead to different results. In this research besides interviews and heuristic evaluation also user tests were used. This research also tried to find an answer to the question:

Do different research methods lead to different results?

1.2 Context of case study

For this research the RaboShop was used as a case study. Multiple research methods were used to get a complete view on the usability issues of the RaboShop. The RaboShop is an internal shop in which employees of the Rabobank can order products they need to do their job. This shop contains an extensive range of products from pencils and staplers to laptops and ATMs. Each month the Rabobank sends out a questionnaire to a number of employees to test the user satisfaction of their internal systems, which also includes the RaboShop. The RaboShop suffers from low grades according to this questionnaire. After several attempts to improve the user satisfaction the grades of the shop are still low. An in-depth research could help to find out what should be done to increase the user satisfaction of the RaboShop.

1.2.1 Rabobank

The founding father of cooperative agricultural credit is Friedrich Wilhelm Raiffeisen, a German rural mayor in the latter half of the nineteenth century. In the 1860s, he founded an agricultural credit union that extends local farmers credit from savings collected from local communities. The ideal of standing stronger together catches on in the Netherlands and across the Dutch countryside farmers and horticulturists begin establishing their own local agricultural cooperative banks. They become owners, members and managers of the bank, sharing responsibility accordingly. Rather than paying out the profits to the members, they are added to the reserves year by year, gradually building up a solid foundation for hard times and lean years. By the end of the nineteenth century the first Dutch local banks established two umbrella organisations: the Coöperatieve Centrale Raiffeisen-Bank in Utrecht and the Coöperatieve Centrale Boerenleenbank in Eindhoven. These two organisations become the central bank for the local banks and play a facilitating role in a number of areas. The two merge in 1972 to become Rabobank Nederland, a cooperative in which all local Rabobanks are members and shareholders. (History of Rabobank, n.d.)

Nowadays the Rabobank is one of the biggest financial services providers of the Netherlands and is a global leader in Food and Agri financing and sustainability-oriented banking. The group consists of 123 independent local Dutch Rabobanks, a central organizations (Rabobank Nederland), and a large number of specialized international offices and subsidiaries. Food and Agribusiness are still the prime international focus of the Rabobank Group. (http://en.wikipedia.org/wiki/Rabobank)

1.2.2 Raboshop

The RaboShop is an internal shop of the Rabobank. The RaboShop facilitates the operational procurement of acquisition (order to delivery) to pay of goods and services of third parties. With the system users are able to order products in the several catalogues and it is possible to see previous orders. The RaboShop consists of several catalogues with products:

- the Algemene catalogus: this catalogue contains the ICT products and services. Also products for which Rabobank specific data is needed can be found here.
- Kantoorartikelen: this catalogue contains most of the products. All products that do not need Rabobank specific data can be found here.
- ‘Software catalogus’: workspace related software can be ordered here.

Furthermore there is also the possibility to place an order with the help of the free order form. With this form users can order products or services that are not default. For example a painter for painting the walls in a building.

Some products need approval before they can be ordered. In the RaboShop there is a functionality to approve such orders. The manager of the user that ordered the product can approve or disapprove an order.

Based on the functionalities of the RaboShop four different user groups can be distinguished:

- Algemene catalogus bestellers: this user group orders the standard catalogue products which do cost money. These products can be found in the the Algemene catalogus and the Kantoorartikelen.
- ICT diensten catalogus bestellers: this user group orders ICT products and services. These products have a symbolic amount of €0,01. Also the users that order products from the 'Software Catalogus' belong to this category of users.
- Free order bestellers: these are users that order products that are not standard.
- Goedkeurders: this user group mostly uses the RaboShop to approve or disapprove orders. The majority of this group are managers.

	Bestellers					**Goedkeurders**		
		algemene catalogus *	ICT diensten catalogus*	Free order*	Totaal		product specialisten	managers
Identiteit	Aantal gebruikers < =5 bestellingen/jaar	2.333	1.514	172	2.976		X	X
	Aantal gebruikers > 5 bestellingen/jaar	2.282	1.15	110	3.076		X	X
	RN Aantal bestellers	1.023	1.442	282	2.033			
	LB Aantal bestellers	3.592	1.222	0	4.019			
	Bestellingen waar geen goedkeuring voor nodig is (procuratie bij besteller)	x	x	x	x		X	X
Gebruik totaal	Totaal aantal bestellingen/jaar	163.994	65.724	3.278	232.996		x	x
	Gemiddelde bestelwaarde	€ 303	€ 0,01	€ 166.39			x	x
	Gebruik primair laptop of mobiel	X	X	X	X			

Table 1

As can be seen in Table 1 the number of users that use the RaboShop and order less than five products a year is almost as big as the number of users that order more than five products a year. So we could state that the infrequent users (<= 5 orders) are at least as important as the frequent users (>5 orders). The questionnaire about user satisfaction at the Rabobank is filled in by all employees, so both frequent as infrequent users. Previous researches done within the Rabobank about the user satisfaction of the RaboShop is focused on the frequent users. It could be possible that because of

this focus they miss out about issues that only infrequent users experience. Which could be a reason that the score of the RaboShop does not increase.

Because of the distribution of frequent and infrequent users his system is definitely suited to be used as a case study to find differences between frequent and infrequent users. The answers of this research will also be of great value for the Rabobank to improve their system and increase their score.

The next section will discuss already existing theory about usability, ERP systems and the usability of ERP systems. In Chapter 4 the research methods used for this research are explained. After that the results of the research are presented. In this Chapter also the differences between the frequent and infrequent users are discussed. The last Chapter contains the conclusion and discussion of this research and also suggestions for future research.

2. Usability

Usability is a broad term and various characterizations have been proposed.

The definition of usability in the ISO 9241 standard: "*The extent to which a product can be used by specified users to achieve specified goals with effectiveness, efficiency, and satisfaction in a specified context of use."* (ISO/IEC, 1998). This is a very abstract definition. Other definitions come from Nielsen and Quesenbery *"Usability is a quality attribute that assesses how easy user interfaces are to use. The word "usability" also refers to methods for improving ease-of-use during the design process."* (Nielsen, 1994) and *"The quality or characteristic of a product that meets the needs of the people who use it, allowing them to work – or play- with it for their own purposes and in a way that is appropriate for them."* (Quesenbery, 2001)

The definition from ISO is very abstract, while the definitions of Quesenbery and Nielsen are more specific. They both define usability as the quality of a product, however Quesenbery is more extensive in her definition. For this research I will stick to the definitions from Quesenbery and Nielsen.

This section will contain the attributes of usability and the ways of measuring usability.

2.1 Attributes of usability

Quesenbery and Nielsen both described five attributes of usability (see table 1). The attributes of Quesenbery are based on the attributes of Nielsen, which also can be noticed looking at the descriptions of the attributes. Four out of five attributes are almost the same, so they both have one attribute that is different.

Nielsen	**5E's of Quesenbery**
Learnability	Easy to learn
Efficiency	Efficient
Errors	Error tolerant
Satisfaction	Engaging
Memorability	Effective

Table 2

2.1.1 Learnability & Easy to learn

According to Nielsen learnability indicates how novice users work with a system. It can be measured by letting test persons perform a couple of tasks and measure the time they need to successfully fulfill the assignments. A very important criteria for this attribute is that the test persons cannot have any experience with the system.
Quesenbery says that a system is easy to learn if users can work with it with their own knowledge without deliberate effort. To create an easy to learn system it is important to have a consistent interface. By analyzing and observing users it is possible to predict the behavior of the users, which makes it easier to create an easy to learn interface.

Nielsen focuses more on how learnability can be measured and Quesenbery on how an easy to use system could be achieved.

2.1.2 Efficiency & Efficient

Efficiency tells us how experienced users work with a system. It can be measured in the same way as learnability can be measured, but instead of a novice user you need an experienced user. Note to this attribute is that it is important to define when a user is experienced.

The efficiency is the speed in which users can complete the task for which they use the system. Ways to measure are the number of clicks or keystrokes that are required or the total 'time on task'. There are a few things that can improve the efficiency of a system: shortcuts, menus, links and other buttons that can make navigation faster; know the preferences of your users.

2.1.3 Errors & Error tolerant

An error tolerant program is designed to prevent errors caused by the user's interaction, and to help the user recovering from any errors that do occur. An option to design an error tolerant program is by making it difficult to take incorrect, invalid and irreversible actions, also plan that the unexpected can happen.(Quesenbery)

To measure the number of errors you can let someone perform some tasks with the system and count the errors that are made. Look at how severe the errors are and how the user recovers from the errors made. (Nielsen)

2.1.4 Satisfaction & Engaging

Nielsen and Quesenbery use different names for this attribute but they have the same meaning. They both describe this point as the indication if an interface is pleasant and satisfying to use. Quesenbery says that visual design is the most obvious element of this characteristic. An interface is pleasant and satisfying to use if it meets the expectations and needs of the people who have to use it. This attribute can be measured with a questionnaire.

2.1.5 Memorability & Effective

These attributes are the only ones that Quesenbery and Nielsen do not have in common. Nielsen has as a fifth attribute memorability. This attribute looks at how much a user memorizes about the system. It can be measured in two ways. One way is to look at how a user works with a system after not using it for a while. Another way is to let a user work with the system and ask some questions about the system afterwards.

Quesenbery has as fifth attribute effectiveness. Effectiveness means how complete and accurate users can achieve goals. It can be measured by looking at whether the users goals were met successfully and whether all work is correct. The effectiveness of an interface often relies on the clear presentation of choices. Also an informative interface is important for effectiveness. To increase effectiveness it is also possible to offer redundant navigation, which is especially useful for systems that support infrequent users.

Quesenbery was inspired by Nielsen in the development of the 5 E's, which is clearly visible. The attributes of both researchers are pretty similar, only one of them is completely different. Where Nielsen focuses mostly on the way of measuring the attributes, Quesenbery is more broad in her way of describing them.

2.2 Measuring usability

There are four basic ways of evaluating user interfaces (Nielsen, 1994):

- Automatically: usability measures computed by running a user interface specification through a program
- Empirically: usability assessed by testing the interface with real users;
- Formally: using exact models and formulas to calculate usability measures;
- Informally: based on rules of thumb and the general skill and experience of the evaluators.

Empirical methods are the most used methods, and especially the user tests. However sometimes it is hard to recruit real users in sufficient numbers, which is why informal methods, like inspection, are sometimes preferable. Studies have shown that usability inspection methods are able to find many usability problems that are overlooked by user testing, but user testing also finds some problems that are not found by inspection methods. So the best results often can be found by using several methods. (Nielsen, 1994)

For this research the system has been evaluated with empirical and informal methods. The empirical method used are user tests and for completeness also an informal method, a heuristic evaluation, is used.

2.2.1 User testing

In user testing, sample size turns out to be less relevant than the number of test cases. In fact, Nielsen and Landauer (1993) show that the best results are obtained from testing no more than five users and running as many small tests as possible. The number of usability problems found in a usability test with *n* users can be calculated with:

$$N\,(1-(1-L)^n)$$

Where N is the total number of usability problems and L is the proportion of usability problems discovered while testing a single user. After a large number of projects studied Nielsen and Landauer concluded that the typical value of L is 31%.

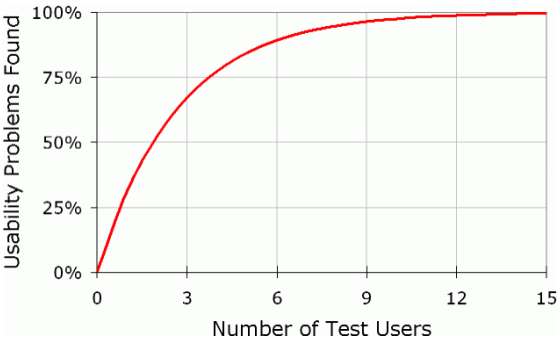


Figure 1

If data is collected from a single user, almost one third of the usability of the interface is learned. With every new user the amount of usability problems found will decrease. After fifth users time will be wasted by observing the same findings repeatedly but not learning much new. Nielsen and Landauer recommend to stop testing after five users, then improve the design and test again with five users. In this way the budget can be distributed over many small tests instead of one big test. When testing multiple user groups the formula only holds for comparable users. But even when groups are very different, there will still be great similarities between the observations. That is why it is not necessary to include as many members of each group as would be done in a single test of a single group of users. The overlap between observations will ensure a better outcome from testing a smaller number of people in each group.

2.2.2 Heuristic evaluation

By doing a heuristic evaluation a usability expert evaluates a user interface with a set of guidelines. Heuristic evaluation was originally developed as a usability engineering method for evaluators who had some knowledge of usability principles but were not necessarily usability experts. However it turned out that usability experts were better at finding usability issues than the evaluators without usability expertise (Nielsen, 1992). In the same research Nielsen found out that major usability

problems have a higher probability than minor problems of being found in a heuristic evaluation. One of the most used heuristic sets are Nielsen's ten heuristics (Nielsen, 1995).

Nielsen's ten heuristics

1. *Visibility of system status*
   The user should be informed about what is going on. The system should give appropriate feedback within reasonable time.
2. *Match between system and the real* world
   The system should speak the same language as the user, words, phrases and concepts of the system should be familiar.
3. *User control and freedom*
   If a user makes a mistake there should be a clear 'emergency exit', like an undo button, which makes it easy to leave the unwanted state without having to go through an extended dialogue.
4. *Consistency and standards*
   Words, situations or actions should in the whole system have the same meaning. The users should not have to wonder if it would mean the same thing at other places in the system.
5. *Error prevention*
   It is better to have a careful design which prevents a problem from occurring than a good error message.
6. *Recognition rather than recall*
   The user's memory load should be minimized by making objects, actions and options visible. The instructions for use of the system should be visible or easily retrievable.
7. *Flexibility and efficiency of use*
   Flexibility and efficiency of use should help expert users to speed up the interaction by enable the use of accelerators, which are unseen by the novice users. In this way the system can cater to both inexperienced and experienced users.
8. *Aesthetic and minimalist design*
   Dialogues should not contain information which is irrelevant or rarely needed.
9. *Help users recognize, diagnose and recover from errors*
   Error messages should be expressed in plain language, precisely indicate the problem and constructively suggest a solution.
10. *Help and documentation*
    It is better if a system can be used without needing documentation, but it maybe is necessary to have some. This information should be easy to search, focused on the user's task and not be too large.

Evaluators evaluate a system with such a set of heuristics. If a system violates one of the heuristics this will be identified as a usability issue. A usability issues severity can be determined with a severity scale, which is mostly a scale from 0-4 (Singh & Wesson, 2009):

0. Not a usability problem;
1. Cosmetic problem: need not to be fixed unless extra time is available;
2. Minor usability problem: fixing this should be given a low priority;
3. Major usability problem: important to fix, should be given a high priority;
4. Usability catastrophe: imperative to fix this problem before product can be released.

Most of the heuristics of Nielsen explain major usability problems. The heuristics 'speak the users language' and 'aesthetic integrity' are more suitable for the discovery of minor usability problems (Nielsen, 1994).

3. Usability and ERP systems

3.1 Enterprise Resource Planning (ERP)

Enterprise Resource Planning are programs that aim to provide integrated software to handle multiple corporate functions as finance, human resources, manufacturing, materials management, and sales and distribution (Amoako-Gyampah & Salam, 2004). Its goal is to maximize productivity, manage costs and serve customer needs in an optimal way. This is realized by bringing the automatic handling of logistical, administrative and financial processes together in one system (What is ERP, n.d.). ERP systems require significant organizational resources and the implementation of such system is very risky because of the large investments. They represent a completely different class of IT application compared with traditional and simple systems (Amoako-Gyampah & Salam, 2004). Because of the size of an ERP system, the implementation of it is not easy and this has led to a number of big failures (Calisir & Calisir, 2000). Causes of this failures are poorly defined business goals, inadequate training, the lack of strong and adequate sponsorship, changes in the project scope, and incompatibility among the various computer hardware and software systems (Calisir & Calisir, 2000). But what does make an ERP system a success? A number of researchers suggested that user satisfaction is one of the key factors leading to information system success and that the usability of interfaces can be seen as one of the factors that influences end-user satisfaction (Calisir & Calisir, 2000).

3.2 Usability of ERP systems

ERP systems do have significant benefits for organizations, like improving operational efficiency. However this systems have been criticized as being too complex. This complexity is mostly due to these systems having to integrate and process large amounts of data, which resulted in ERP systems possessing user interfaces which suffer from poor usability. However a lot of research has been done on the topic of ERP implementation failures, the usability of ERP systems and other large-scale systems has not been widely studied.

By interviewing users who worked with ERP systems on a daily basis, Topi, Lucas and Babaian (2005) identified usability issues the users experienced. The following usability issues did come out of this study:

- Identification of and access to the correct functionality.
- Transaction execution support
- System output limitations
- Support in error situations
- Terminology problems
- Overall system complexity

These usability issues were used in a heuristic evaluation of an ERP system by Singh and Wesson (2009). They evaluated the system based on the general usability heuristics of Nielsen and a set of own developed usability heuristics specifically for ERP systems. To develop this specific set of heuristics they used the usability issues of Topi, Lucas and Babaian, mixed them with other findings (Matthews, 2008) and mapped them in five criteria: navigation, learnability, task support, presentation and customization. They transformed these criteria into measurable heuristics:

- Navigation and access to information;
- Presentation of screen and output;
- Appropriateness of task support;
- Degree of ease to learn how to use the system effectively;

- Ease of customizing the system to ensure alignment between the system, the user and business processes.

The set of ERP usability heuristics identified different usability issues than Nielsen's ten heuristics for general usability. So combining the two sets of heuristics gives the most complete set of usability issues.

4. Method

In this research the differences in usability issues between frequent and infrequent users of ERP systems have been measured. This is done at the hand of three different methods. The results are compared at the end of the research. Two formal methods have been used: interviews and user tests. But the research started with one informal method: heuristic evaluation.

4.1 Heuristic evaluation

For the heuristic evaluation the system was evaluated with the help of a set of guidelines. The guidelines used are based on the research of Singh and Wesson (2009). They used for their evaluation the ten heuristics of Nielson and they developed five own heuristics specifically for ERP systems. In their research they concluded that for a good heuristic evaluation of an ERP system both lists of heuristics should be used, they complement each other. For that reason they both are used in this research.

Ten heuristics of Nielsen	**ERP heuristics of Singh and Wesson**
1. Visibility of system status 2. Match between system and the real world 3. User control and freedom 4. Consistency and standards 5. Error prevention 6. Recognition rather than recall 7. Flexibility and efficiency of use 8. Aesthetic and minimalist design 9. Help users recognize, diagnose and recover from errors 10. Help and documentation	1. Navigation and access to information; 2. Presentation of screen and output; 3. Appropriateness of task support; 4. Degree of ease to learn how to use the system effectively; 5. Ease of customizing the system to ensure alignment between the system, the user and business processes.

Table 3

To evaluate the system a few tasks were performed. For this evaluation similar tasks were performed as for the user tests.

- Order some products in the Algemene catalogus. (laptop, phone)
- Order some products in the Kantoorartikelen. (Parker pen, stapler)
- Order a piece of software in the Software catalogus.
- Set a substitute in Takenlijst.

With these tasks all parts of the RaboShop were tested. During the execution of the tasks, notes about possible usability issues were made. These issues later were classified within the heuristics. Within these heuristics the usability issues were graded at the hand of the following grading scale:

1. Cosmetic problem: need not to be fixed unless extra time is available;
2. Minor usability problem: fixing this should be given a low priority;
3. Major usability problem: important to fix, should be given a high priority;
4. Usability catastrophe: imperative to fix this problem before product can be released.

After the scoring the mean of the score per usability criteria was noted, which gives an indication of the criteria that has the most severe problems. At the end these results were compared to the results of the interviews and user tests.

4.2 User tests

One of the formal usability evaluating methods used are user tests. These tests give insight in how users work with a system. Because the goal of this research is to find out if there is a difference in usability issues between frequent and infrequent users, the user tests are conducted with some frequent and infrequent users. Seven users participated in this research, three frequent and four infrequent users. Nielsen and Landauer (1993) recommend that with two user groups three to four users of each category should be tested.

For this test the participants were asked to perform five tasks in the RaboShop. These tasks were prepared on the basis of information gathered from the heuristic evaluation, information from an expert of the RaboShop and exploring conversations with RaboShop users. The heuristic evaluation gave a good impression of the system and the tasks that could be performed. An expert of the RaboShop told about the most used functionalities of the RaboShop and also the tasks that frequently lead to problems. Also there were some exploratory talks with users of the RaboShop about their experiences with the system. With this information the following tasks were formulated:

1. Buy a present for one of your colleagues who is going to retire.
2. Buy items for a new colleague.
3. Order a piece of software.
4. Buy tickets for the parking area near the Rabobank office.
5. Change the settings in such way that a colleague can do your tasks (approve orders, check shopping carts) in the RaboShop in your absence.

The tasks were formulated in such way that it does not give away how the participant could reach the goal. In this way the way of working came as close as possible to the normal way of working. The names of products also were not noted because this would give away the search word. If specific products were requested, pictures were used instead.

The tests were conducted in a quiet environment and were recorded with a digital camera. The participants were asked to think aloud so actions could be better interpret afterwards. At the hand of the video the tests were coded later on. During the test I was present but only as observer.

4.3 Interviews

After the user tests the interviews were conducted. The interviews were semi-structured. The main questions were defined beforehand, and more detailed questions about the user test and given answers were defined on the spot. The goal was to gain information about working with the RaboShop and experienced problems, and also why the user performed certain actions during the test. The following main questions were asked during the interview:

1. For which purpose do you use the RaboShop? How often do you use it? Which parts do you use?
2. What do you think about working with the RaboShop?
3. What do you think about working with the the Algemene catalogus/Kantoorartikelen/'Software catalogus'/'Free order'? (dependent on what parts the user used)
4. What do you think about the presentation of the the Algemene catalogus/ Kantoorartikelen/ 'Software catalogus'/'Free order'?
5. Does it happen that you cannot find something in the the Algemene catalogus/ Kantoorartikelen/ 'Software catalogus'/'Free order'?
6. Do you think the RaboShop works efficient?

7. Do you sometimes get error messages? If yes, what do you think about these messages? Are they clear? Do you know what to do to solve the problem?
8. Do you check the help documents when you are having problems? If yes, what do you think about them? Do they help you solving the problem?
9. Is the system easy to use? Can you work with it easily if you have not worked with it for a while?

The questions are based on the usability criteria used in the heuristic evaluation. In the beginning the questions are more global, to get a first impression of what the user thinks of the system. In this way the participant is not guided in a certain direction, and tells what he really thinks of the system. Later on the questions are getting more specific. Besides the standard questions a part of the interview was based on stimulated recall. Questions about interesting parts of the user tests were asked to get more insight about the motives of the actions.

4.4 Analysis

After conducting the interviews and user tests the interviews were transcribed. The transcripts and the videos were coded in Atlas.ti. The first round of coding existed of open coding. If a problem was detected or if something went wrong a code was given to that part. This code was a description of the problem, like 'Wrong catalogue chosen'. After this first round a second round of coding was conducted. In this round analytic coding was applied to the already existing codes. This meant that codes that belonged together were taken together into a new more general code. For example 'Cannot find Vervanging' and 'Cannot find Takenlijst' were taken together in the code 'Cannot find functionality'. After de codes were generalized to code groups, they could be linked to usability criteria. For this the usability criteria of the heuristic evaluation were used: navigation, task support, UI presentation, learnability and errors. Also the criteria 'search function' was used because a lot of codes were related to the search function.

To get better insight in the problems that occurred in the tests and interviews with the frequent users or in those of the infrequent users, we looked at the co-occurrence of codes. First all interviews and user tests were assigned the code 'Infrequent user' or 'Frequent user'. After that the program can visualize the co-occurrence of a selected code and the 'Frequent user' or 'Infrequent user' code.

5. Results

5.1 Heuristic evaluation

During the heuristic evaluation tasks were performed to check the whole system and meanwhile the system was checked at the hand of the heuristics of Singh & Wesson and Nielsen. The results of this evaluation are presented in this chapter.

5.1.1 Singh & Wesson

Usability criteria	Potential usability issues
Navigation	• Functionalities cannot be found quickly and easily. • Lot of scrolling needed • Search function does not work well • System does not support guidance-type of information
Presentation	• UI is not intuitive
Task support	• Terminology of the system is not consistent with the terminology of the user. • The system does not support efficient completion of tasks • The system does not improve user productivity • System is not easy to use
Learnability	• Different interfaces, so more to learn • System is not easy to understand • The system is intimidating and complex to learn and use.
Customization	• Only a little customization is possible

Table 4

Navigation

Functionalities cannot be found quickly and easily. The first step of the tasks is to go to the catalogues, which can be entered with the 'Inkopen' button. This buttons is the entrance to the main functionality of the RaboShop, so it should have a prominent place on the homepage. However the 'Inkopen' button is just a regular button, and has to be searched. Another important option in the RaboShop is the option to set a substitute for approving orders. This option is hidden behind another button and is hard to find.

A lot of scrolling is needed. In the Algemene catalogus a lot of scrolling is needed because the search results cannot be seen without scrolling. And if you want to search again or see if something is in the shopping cart you need to be on top of the page again.

The search function does not work well. The search name for example has to be in the product description, otherwise the system will not find the item you want. When searching on 'telefoon', you will not find an iPhone. Also in the the Algemene catalogus the search options 'Hiërarchie' and 'Productgroep' do have categories that do not match with the expectations of the products that will be in it. For example when searching a laptop and you select 'Aanvraag laptop…' the right item will be filtered out.

The system does not support guidance-type of information. The system does not contain much help with guidance while working with it. Some names of buttons are not clear, for example 'Positie overzetten' which means to place an order. People do not see positions as products. For these kind of buttons it would be helpful if there was some information available within the system.

Presentation

The RaboShop consists of a few parts, which all have other interfaces. When first ordering some products in the Algemene catalogus and you continue to the Kantoorartikelen the interface is very surprising. It is completely different from the Algemene catalogus. When heading next to the Software catalogus you will face again a completely different interface. The Algemene catalogus has an old-fashioned, more tool-based kind of interface. It cannot be compared with a webshop. In this catalogue it is possible to search with the help of a search field and two different lists with categories you can select to refine the search results. In the Kantoorartikelen it is also possible to search with an search field, but here you can also search with a menu at the left side of the window. The Software catalogus is more form-based. There is a list of fields that need to be filled and at the end the software can be ordered.

The diversity of interfaces makes the system not intuitive. Every catalogue requires a new learning process, because they all work differently.

Task support

The terminology of the system is not consistent with the terminology of the user. In the Algemene catalogus the term used for a product is ‘positie’. It took a while before this was noticed, because it is not a term you would immediately relate to a product. Also the product descriptions are in a different language then a user would use, more jargon. This also makes it harder to find product because the search name does have to match the product description to find it.

The system does not support efficient completion of tasks. To order a product a lot of steps are necessary. First the right catalogue should be chosen. In the catalogue the product should be found and put in the shopping cart. For some articles it is mandatory to fill in a form, which requires an extra manual handling. In the shopping cart the order has to be transferred to the main part of the RaboShop. Back in the main part the overall shopping cart is viewed with the products that has been transferred from the catalogues. Here you can continue to the next page and here the products can finally be ordered.

The system does not improve the users productivity. Because of the inefficient way of working with a lot of steps and also the not well working search function the system does not improve the productivity of the user.

Learnability

Different interfaces, so more to learn. The system consists of multiple interfaces which all work differently, as said earlier. This requires more learning, which makes it harder to immediately work with the system.

The system is not easy to understand. The terms that are used are not always self-explaining, and there is no guidance-type information available. Also not all functionalities can be found in logic places. Like the function to set somebody as a substitute cannot be found easily. The system consists of multiple parts which are tied together in the main part. It will cost some time to understand how the system works. The different parts also cause problems with what to find in which catalogue. This is not clear and also not described.

The system is intimidating and complex to learn and use. Based on what is said earlier the system is complex to learn and use. The system also is intimidating because it has a lot of options and the windows contain a lot of information.

Customization

During the performed tasks it was not really necessary to customize the system. There are also not much options for customization in the RaboShop. The view of products can be changed and also the extra search functions can be collapsed.

So the most problems in the RaboShop are caused by navigation, task support and learnability.

Afterwards the scores of the usability issues are added.

Usability criteria	Potential usability issues	Score (1-4)
Navigation	• Functionalities cannot be found quickly and easily. • Lot of scrolling needed • Search function does not work well • System does not support guidance-type of information	3 1 3 2
Presentation	• UI is not intuitive	2
Task support	• Terminology of the system is not consistent with the terminology of the user. • The system does not support efficient completion of tasks • The system does not improve user productivity • System is not easy to use	2 2 2 2
Learnability	• Different interfaces, so more to learn • System is not easy to understand • The system is intimidating and complex to learn and use.	1 2 2
Customization	• Only a little customization is possible	1

Table 5

Navigation

The issues 'Functionalities cannot be found quickly and easily' and 'Search function does not work well' both received the score 3, which means a major usability issue. Because of these issues it is possible that users cannot find products or functions and therefore are not able to order the products or change certain settings. Trying to find these functionalities or products can cost a lot of time and maybe the user has to call to the helpdesk. All of this does cost time and time is money.
'System does not support guidance-type of information' is a minor usability issue. Because functionalities cannot be found quickly and easily it would be handy to have guidance-type of information which could help you while working with the system. This could also lead to less problems with finding functionalities.
'Lot of scrolling needed' is graded as cosmetic issue. It is not really nice while working and a bit inefficient, but it does not cause real problems.

Presentation

'UI is not intuitive' is a minor usability issue. This slows down the ordering process, however with some practice this probably will not be a big problem anymore.

Task support

'The system does not support efficient completion of tasks' and 'The system does not improve user productivity' are minor usability issues. It will cost some more clicks and scrolling to order a product, but you will get there.
'System is not easy to use' is also a minor usability issue. Certainly for a first time it is really not easy to use. After working a while with the system it will be more logical and easier to work with.
'Terminology of the system is not consistent with the terminology of the user' is graded as a minor usability issue. Some terms used or names of buttons are not clear. Because of that some functionalities are hard to find or maybe not to find.
'The system does not improve user productivity' is also a minor usability issue.

Learnability

'Different interfaces, so more to learn' is graded as a cosmetic usability issue. The RaboShop consists of different catalogues which all have another interface. Because of these different interfaces the user has to learn everything all over again, which costs extra time. However it is a matter of time before you know all interfaces.

'System is not easy to understand' is graded as a minor usability issue. Because of the different parts of the RaboShop, the functionalities that are not easy to find and the search function that does not work well the system is not easy to understand. Those issues are also the reason that 'The system is intimidating and complex to learn and use' is graded as a minor usability issue.

Customization

'Only a little customization is possible' is a cosmetic usability issue. There are not much customization options available within the RaboShop, but this also is not really necessary. So it is not a big issue.

Usability criteria	Mean
Navigation	2,25
Presentation	2
Task support	2
Learnability	1,667
Customization	1

Table 6

In the scores we can see that the most severe problems have to deal with navigation. This criteria also contains issues with the score 3, standing for major usability issue. Presentation and Task support are also criteria with serious problems. Customizations has the least priority. Also we can state that more usability issues doesn't necessarily mean that this criteria also has the most severe problems.

5.1.2 Nielsen

Usability criteria	Potential usability issues
Visibility of system status	• In the Algemene catalogus not immediately visible when something is added in the shopping cart. • In the Algemene catalogus the search results are not visible without scrolling. • System status in Kantoorartikelen not visible.
Match between system and real world	• Terminology is too technical for users
User control	• No emergency exit when your session is expired.
Consistency and standards	• No consistent UI
Error recovery	• Error messages only vaguely explain the cause of the error • Error messages do not indicate how to solve the problem
Error prevention	• The system does not prevent you from errors. • No information about format of values for fields.
Memory load	• There is no logical grouping of items. • Options not visible enough
System flexibility	• System does not support multiple level of users

Minimalist design	• Search functions are collapsed by default.
Help and documentation	• No help within system. • There is no search functionality in the help documents

Table 6

Visibility of system status

If you want to search something in the Algemene catalogus it looks like nothing happens. The first and visible part of the window consists of search functionalities and only if when scrolling down the search results are visible. Also if a product is selected in the search results and it is added to the shopping cart it is not visible if something really is added to the shopping cart. To see if something happened it is required to scroll all the way up to the beginning of the page.

The Kantoorartikelen is very slow in responses and the status of the system cannot be seen, so it is not clear if it is doing something or not.

Match between system and real world

The terminology used in the RaboShop does not match with the real world, there is too much jargon for users. The naming of buttons sometimes is not clear and also product descriptions are not like users would describe it.

User control

The control of the user is mostly fine, but if the session expires (after 15 minutes of inactivity) an error message will pop up as soon as you click somewhere. The only option is to refresh the page and start all over again, because the system does not save the progress. The system does not give a warning that the session expires, and it is not even clear that it is possible.

Consistency and standards

Every catalogue looks different and works different, there is no consistency between them.

Error recovery and error prevention

The error messages in the RaboShop are also not clear. In the shopping cart it is possible to get error messages but it is not clear what exactly is wrong or how to fix it. For example when ordering some items the error message 'Please add a template' was shown. However it is not clear about what template they are talking or where it can be found. Later it turned out that it was required to fill in a form for some products. The order cannot be ordered until it is fixed, so it is important that it is possible to do that. The system also does not prevent for getting these errors. For example it should force users to fill in the form that belongs to a product.
In some forms it is required to fill in a data, but this should be done in a specific format that is illogical. The correct way of filling this field is not shown.

Memory load

The memory load is high for this system. The find something it is really needed to know where it can be found to find it in a reasonable amount of time. Also some functionalities are hard to find. Therefore users are looking for this in the help documents, but there are a lot of documents in there so it can take a lot of time to find out where something can be found.

System flexibility

The system does not support multiple user levels. It is not possible for an experienced user to work faster in some way then a novice user. However experienced users probably can work faster because they know where to find products, but there are no shortcuts available.

Minimalist design

The design of the RaboShop is not minimalistic. Functions that can be collapsed are all shown by default. Some interfaces are also a bit chaotic with a lot of information.

Help and documentation

In the RaboShop there is help documentation available, but it can only be found in the first page of the RaboShop, before entering the real shop. Inside the RaboShop (catalogues etc.) there is no information available that could help. Not about which catalogue to choose and not about the meaning of buttons. This information would be nice to have because the names of some buttons or functions are not clear.
If users cannot find something in the RaboShop they will look in the help documentation to find out where they can find it. The help functionality contains of a lot of documents. It would be helpful if there was a search function, so it is not needed to search for a long time there.

Usability criteria	**Potential usability issues**	**Score (1-4)**
Visibility of system status	• In the Algemene catalogus it is not immediately visible when something is added in the shopping cart. • In the Algemene catalogus the results are not visible without scrolling. • System status in Kantoorartikelen is not visible.	1 1 1
Match between system and real world	• Terminology is too technical for users	2
User control	• No emergency exit when your session is expired.	2
Consistency and standards	• No consistent UI	2
Error recovery	• Error messages only vaguely explain the cause of the error • Error messages do not indicate how to solve the problem	3 3
Error prevention	• The system does not prevent you from errors. • No information about format of values for fields.	2 1
Memory load	• There is no logical grouping of items. • Options not visible enough	2 2
System flexibility	• System does not support multiple level of users • There is no search functionality in the help documents	1 1
Minimalist design	• Search functions are collapsed by default.	1

Help and documentation	• No help within system.	1

Table 7

Visibility of system status

'In the Algemene catalogus it is not immediately visible when something is added in the shopping cart' and 'In the Algemene catalogus the results are not visible without scrolling' are both cosmetic problems. It is not convenient that it is not visible, but it does not really have an impact on the way of working.
'System status in Kantoorartikelen is not visible' is also a cosmetic problem. This is also not convenient but does not have an impact on the way of working. Just be patient.

Match between system and real world

'Terminology is too technical for users' is a minor usability problem. Some terms used or names of buttons are not clear. Because of that some functionalities are hard to find or maybe not to find.

User control

'No emergency exit when your session is expired' is a minor usability problem. Starting all over again is very inefficient and annoying, but it is still possible to order products.

Consistency and standards

'No consistent UI' is also a minor usability issue. The RaboShop consists of different catalogues which all have another interface. Because of these different interfaces the user has to learn everything all over again, which costs extra time. However it is a matter of time before users know all interfaces.

Error recovery

'Error messages only vaguely explain the cause of the error' and 'Error messages do not indicate how to solve the problem' are both major usability issues. Error messages that pop up are vague and they do not explicitly tell how to solve the problem. This errors make it impossible to order products, so it is very inconvenient to get such an error.

Error prevention

'The system does not prevent you from errors' is a minor usability problem. The system does not require users to fill in a mandatory form. If it did errors could be prevented and products could be ordered smoothly.
'No information about format of values for fields' is a cosmetic problem. Some fields require unusual notation formats. Mistakes could easily be avoided by inform the user about the format.

Memory load

'There is no logical grouping of items' is a minor usability problem. Search categories do not have logical groupings which makes it possible that the searched product is filtered out of the search results. This will slow down the working process and will lead to irritation.
'Options not visible enough' is also a minor usability problem. Some options do not stand out while they are fairly important. This also will slow down the working process.

System flexibility

'System does not support multiple level of users' is a cosmetic problem. It is not really necessary to support multiple levels of users.
'There is no search functionality in the help documents' is also a cosmetic problem. It would be nice to have a search function to find the right help document. This would help to increase irritations of an earlier problem. However it has no priority.

Minimalist design

'Search functions are collapsed by default' is a cosmetic problem. By collapsing by default the window looks chaotic and full.

Help and documentation

'No help within system' is a cosmetic problem. More help within the system would help users to work with the system more smoothly. Also they immediately have the right help information on the right moment.

Usability criteria	**Mean**
Visibility of system status	1
Match between system and real world	2
User control	2
Consistency and standards	2
Error recovery	3
Error prevention	1,5
Memory load	2
System flexibility	1
Minimalist design	1
Help and documentation	1

Table 8

In this table it can be seen that some usability criteria have more severe problems than others. The error recovery is the most severe usability issue. In the RaboShop it is hard to solve an error, because it is not clear what the problem is or how to solve it. Also the match between system and real world, user control, consistency and standards, and the memory load are usability issues in this system. These are the criteria that should get most attention when improving the RaboShop.

5.2 User tests & interviews

In Table 9 the main categories are listed with the occurrence of their code in the interviews and tests and the differences between the frequent and infrequent users. Also a table with an overview of the test users is given (Table 10). The results of the categories will be presented in this Chapter and at the end the differences between the frequent and infrequent users will be discussed.

	Frequent user	Infrequent user	Total
Navigation	18	20	38
Task support	19	39	58
Search function	36	50	86
Errors	5	8	13
Learnability	11	8	19
UI presentation	13	27	40

Table 9

	Gender	Frequent or infrequent	Type of user
User 1	Female	Frequent	Local bank
User 2	Female	Frequent	Catalogue orderer
User 3	Female	Infrequent (never uses the catalogues)	Free order
User 4	Female	Frequent	Catalogue orderer
User 5	Male	Infrequent	-
User 6	Male	Infrequent	-
User 7	Male	Infrequent	-

Table 10

5.2.1 Navigation

All users experienced problems with the navigation of the RaboShop. One of the biggest problem was finding functionalities.

> *"Het is gewoon heel lastig om uit te vogelen waar wat zit, als je dat dan nog nooit gedaan hebt dan ben je eigenlijk aan je lot over gelaten." – User 6*

> *"Ik weet nooit zo goed waar ik nu naartoe moet." – User 1*

The RaboShop has a lot functionalities and in practice only a few are often used. Users indicate that they would appreciate it if unused functionalities would be removed.

> *"De werking.. in mijn geval heb ik dit niet nodig (zoekopties), altijd als ik iets bestel gebruik ik alleen trefwoord en zoeken, meer gebruik ik eigenlijk niet, alle overbodige aanhangselen mogen er wat mij betreft uit." – User 4*

All users think that the Kantoorartikelen works better and more user friendly then the Algemene catalogus. This is also due to a better navigation.

> *"Dit is duidelijker, dit geeft sneller een idee van oh ik moet daar of daar zoeken." – User 3*

> *"Ja die gaat wel goed, dat is verder wel duidelijk en je moet het even overzetten, ja, vind ik op zich wel prima." – User 2*

But most problems with navigation have to do with finding functionalities.

Cannot find functionality

All of the users had trouble with finding functionalities. The biggest problem occurred with setting a substitute. Nobody completed this task successfully without any hints. The first problem is knowing that this functions has to be found in the tab 'Takenlijst'. One of the users said:

> *"Ja, ook de vervanging regelen is niet iets wat je onder een takenlijst zoekt. Een takenlijst vind ik echt iets van dingen die je moet doen, en vervanging regelen zit meer in de autorisatiehoek." – User 7*

In 'Takenlijst' the button 'Vervanging' had to be found. 6 out of 7 users could not find this button, or it took them a long time. It is a rather small and unnoted button for such an important functionality.

> *"Met de vervanging wist ik dat het in takenlijst stond en nog kon ik het niet vinden." – User 2*

If a user found the 'Vervanging' button a new rule had to be created, for which the 'Regel creeëren' button was needed. Nobody found this button without hints. Most users tried to add a new rule by clicking on the empty rules, unfortunately this does not work.

Another problem with finding functionality occurred with confirming the chosen software in the software catalogue. The 'Ok' button is visible when scrolling all the way down, but it took a while before users found that button. 2 out of 7 users had trouble finding this functionality.

5.2.2 Task support

Task support is another issue of the RaboShop. All users do have trouble with the sequence of steps of ordering products.

> *"Ik zeg eerlijk omdat ik het vaker doe gaat het prima, maar ik kan me heel goed voorstellen dat mensen die hier nieuw komen werken of nieuw met de RaboShop gaan werken het lastig vinden. Ook met dat overzetten bijvoorbeeld, ik weet toevallig dat je even op een site van Staples zit, en dat die daarom overgezet moet worden, maar dat is natuurlijk omdat ik op deze afdeling werk. Maar anders is het niet logisch." – User 2*

Also in which catalogue one can find the right article.

> *"Het is niet duidelijk wat je waar moet bestellen. Hoezo haal je jaarbeurskaarten bij de inkopen van Staples die potloden en pennen heeft? Die logica is er gewoon niet." – User 3*

The users find the Kantoorartikelen and Software catalogus easier to work with.

> *"Ik vind dit [..] makkelijker werken. Dit lijkt veel op de nieuwe RaboShop. Dit werkt ook wat fijner. Je hoeft minder handelingen te verrichten om iets in je winkelwagen te krijgen." (Kantoorartikelen) – User 4*

> *"Die is wel gebruikersvriendelijker, want er wordt heel duidelijk aangegeven wat je moet invullen. Het is niet of dat of dat, de stappen die hier worden aangegeven vind ik makkelijker dan de stappen in.. omdat je hier formulier-based werkt, makkelijker stappen voor de gebruiker." (Software catalogus) – User 3*

One suggestion to improve the task support came from user 7:

> *"Eigenlijk heb je een soort van stappenplan nodig dat als je een laptop besteld dat je stap voor stap de zaken door moet. Nu moet je echt kijken van ik bestel een laptop en je vergeet heel snel rechts het formulier in te vullen. Als je sequentieel alle stappen door moet lopen is daar ook geen verwarring over mogelijk."*

Adding more guidance-type information would also be a good option.

> *"Het zou het mooiste zijn als er gewoon een hulpje zou zijn bij hetgene wat je besteld of zo, of uitleg." – User 4*

Other usability issues that decrease the task support are efficiency and language, which will be discussed next.

Efficiency

None of the test users thought that the RaboShop works efficient. It does not save them time. In fact, it even takes more time, especially when unusual products have to be ordered.

> *"Maar als je een nieuw item bestelt dan kan dat heel tijdrovend zijn. Als je niet weet waar je iets moet gaan zoeken, in welk gedeelte of hoe het precies heet, [..] dan ga ik dat wel op een moment doen dat ik daar tijd voor heb." – User 2*

Not only the search function makes the system inefficient. A lot of scrolling is needed in the system.

> *"Je moet veel scrollen op welk scherm je ook zit." – User 1*

Mostly in the Algemene catalogus a lot of scrolling is needed because the search options are collapsed by default.

Also a lot of handlings are needed, a lot of clicks, to order a product.

> *"Veel handelingen nodig om een resultaat te zien of veel klikken om resultaat te zien." – User 1*

For some products filling in a form is required. Filling in this form is not a part of the process flow. This has to be filled in and submitted and then the product has to be added to the shopping cart.

> *"Zodra het ingevuld is dan mag t gelijk in het winkelmandje, dat zou het meest makkelijk zijn, je vult dat formulier niet voor jan lul in, je wilt dat het meteen in het winkelmandje komt. Dat bedoel ik met zoveel klikken, scherm na scherm na scherm. Zou mooi zijn als dat formulier als je het afsluit het meteen in het winkelmandje gaat." – User 4*

But also the process from finding a product to ordering it contains of a lot of steps.

Language

5 out of 7 users did have trouble with the terminology used in the RaboShop. This came up in the conducted interviews and also in the user tests. In the tests some users had trouble with finding functionalities because the names used for the buttons were not clear. For example user 5 did not succeed in transferring his order from the Algemene catalogus to the main shopping cart, because the name of that button did not correspond with his expectation. Also the description of the products are not always clear for most users.

> *"Is wel heel erg inkoopterm gericht, het is geen eindgebruikerstaal. Er staan dingen bij waarvan je denkt van huh, niet duidelijk." – User 3*

And this also causes problems while searching products. In the search words:

> *"Ik had een klein beetje de indruk dat de namen van de producten vanuit een technisch perspectief ingevoerd zijn, niet door een gebruiker. Als ik parkeerkaarten zoek voor de jaarbeurs zoek ik niet naar uitrijkaarten, ik weet niet dat het een uitrijkaart heet." – User 6*

But also in the search options, in which user 5 and 6 several times filtered out the right product by choosing the wrong category.

> *"Categorisering vond ik voor mij niet logisch." – User 5*

5.2.3 Search function

The search function is probably one of the biggest usability issues of the RaboShop. All users have trouble with finding products, however if they know where to find something it is doable.

> *"Als je weet hoe je kan zoeken is het makkelijk, maar als je niet weet waar je op moet zoeken is het lastig." – User 2*

> *"De producten die we bestellen zijn doorgaans regulier dus dan weet je waaronder je moet zoeken." - User 1*

Also some user make notes of where they can find products, so they can find it faster next time.

> *"[…] of ik heb ergens een notitie gemaakt van als je dit moet zoeken dan is dat de zoekfunctie ,dan vind ik het wel snel." – User 1*

Choosing between catalogues

One big problem are the different catalogues in the RaboShop. Before searching a product the user needs to choose in which catalogue the product could be found. During the tests it became very clear that this is not always easy. In one task parking tickets were bought, and these could be found in the Kantoorartikelen. The users that never ordered this before could not find them because in most cases they not even did try to find it in the Kantoorartikelen

> *"Nee het is niet duidelijk wat je waar moet bestellen. Hoezo haal je Jaarbeurskaarten bij de inkopen van Staples die potloden en pennen heeft? Die logica is er gewoon niet." – User 3*

The users said that it would be better if there was a search function on a higher level, so products can be searched in all catalogues at the same time.

> *"Wat ik wel mis is dat je op het hoofdniveau een trefwoord kan zoeken en dat je dan automatisch naar de verschillende catalogussen wordt doorgeleid." – User 7*

Search words

Another problem that hinders the searching are the search words. Products can only be found when searching on words that are used in the product description.

> *"Op zich werkt het wel, als je de term weet waarop je moet zoeken dan." – User 6*

And as said before, the terminology of the RaboShop does not match with the users terminology. This was also a problem during the tests, where users were looking for 'batch' instead of 'smartcard' or 'parkeerkaart' instead of 'uitrijkaart'.

> *"Soms heb ik wel problemen dat als je een bepaald artikel zoekt onder welke benaming je het moet zoeken. Je moet echt de juiste benaming van het artikel weten, anders vindt hij hem niet. Als ik een voorzetscherm of spiekscherm moet bestellen, en er komt iemand bij mij iets bestellen en die noemt dat heel anders dan hoe ik het ken, maar onder die naam kan je niet altijd alles terugvinden. Het zou leuk zijn als die parkeerkaarten van de Jaarbeurs kunt vinden op Jaarbeurs en op parkeerkaart. Als het google gebeuren: bedoelt u soms.. kies maar uit." – User 4*

The users would like to have a broader range of search words on which they can found the product they want.

> *"Vind je dat hij dan echt verkeerde dingen zoekt of zou je willen dat hij een wat ruimere benaming, dat als je cadeau of geschenk zoekt hij alle opties daarvoor pakt?" "Ja, juist inderdaad." – User 2*

Search options

There are several options in the RaboShop which should make it easier to search. In the Algemene catalogus the user can filter the results on categories with the 'Hiërarchie' and 'Productgroep' option. In the Kantoorartikelen there is a menu with search categories which can help searching. The tests made clear that frequent users do not use these search options, while infrequent users do use them. The frequent users said that they do not need it and they could get removed from the RaboShop.

> *"De werking.. in mijn geval heb ik dit niet nodig (zoekopties), altijd als ik iets bestel gebruik ik alleen trefwoord en zoeken, meer gebruik ik eigenlijk niet, alle overbodige aanhangselen mogen er wat mij betreft uit." – User 4*

> *"Nee het voegt eigenlijk niets toe, want je gaat toch meteen zoeken en je gaat niet op categorie zoeken." – User 1*

The infrequent users did use the search options, however this did not always led to better searching. As said before the names of the categories are not always clear. It happened a few times that the user searched with the right search name, but the right result was filtered out because the wrong category was chosen.

> *"Op zich dat doorklikken en die categorieën vind ik wel weer handig, maar niet altijd even logisch. Dus het is wel handig om binnen die productgroepen te kunnen kiezen, [..], zodat ik niet 5 pagina's hield maar de beste drie zag die erbij hoorde. Alleen had ik soms het idee dat ik niet de goede categorie koos en het daardoor niet kon vinden." – User 5*

Other problems

A smaller issue is the function to search a name of an employee in for example the Software catalogus. Users needed a lot of clicks before they found out how to search a name.

> *"Alleen dit soort dingen is weer heel erg onduidelijk. Hoe dat werkt en hoe je een naam kunt invoeren en hoe je daar komt." – User 5*

In the Algemene catalogus a wildcard will help the user to search easier. However it is not clear which character is used as the wildcard.

> *"Ik vind het ook niet handig dat die weet wat de wildcard is waarvan ik gebruik moet maken, of het een sterretje is of een vraagteken of een procentje of wat dan ook. Dat kostte me ook moeite om dingen te vinden, tot ik doorhad dat het een sterretje was, toen ging het wat makkelijker." – User 5*

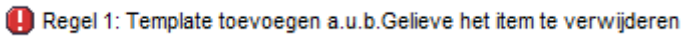
As like with some other issues the users point out that the Kantoorartikelen works better with searching then the the Algemene catalogus.

> *"Ja lijkt wel, lijkt er op dat je hier net iets meer kan vinden dan in de algemene." – User 2*

"Op zich vind ik dit fijner werken, op de een of andere manier heeft deze een wat modernere uitstraling dan de andere, en dit werkte ook beter, wat ik zocht vond ik direct. Op de een of andere manier klopt die indicering hier beter denk ik." – User 5

5.2.4 Errors

During the interviews the only error that came up was the error for time expiration. After fifteen minutes of inactivity the system will give an error after a sign of activity. When getting this error the user has to start over again, because the system does not save the progress. Not much time ago the time of inactivity was increased from 5 minutes to 15, so the users did not see this as an important issue anymore. A more serious issue was detected during the user tests performed by infrequent users. For some products more information is needed so a form needs to be filled in. It is not noted that this is required and the link to the form is not clearly visible. If this form is not filled, the main shopping cart will give the following error:



So a template should be added, or the product should be removed. Firstly, it is not clear that 'a template' is the same as the form. The users were searching for the word 'template' to solve the problem. Secondly it is not clear where this template can be found. As long as the form is not filled the product cannot be ordered.

5.2.5 Learnability

All users agree that the RaboShop is not easy to use for infrequent users, and this is mostly due to the search function:

"Als ik de geijkte dingen ga bestellen dan gaat het prima, want daar ben je inmiddels wel achter hoe dat werkt. Maar als je een nieuw item bestelt dan kan dat heel tijdrovend zijn." – User 2

"Het is veel gezoek en gepuzzel, het is niet makkelijk." – User 4

So even frequent users need a lot of time when they have to find new items. That the RaboShop is not easy to use for novice users was very clear in the user tests with the infrequent users, who clearly had difficulties with the tasks. In the interviews they said:

"Het is gewoon heel lastig om uit te vogelen waar wat zit, als je dat dan nog nooit gedaan hebt dan ben je eigenlijk aan je lot over gelaten." – User 6

"Heb je me bezig gezien? Vijf dingen bestellen en ik ben een uur bezig geweest en ik had het niet eens voor elkaar, [..]." – User 5

To help users there are help documents available at the home page of the RaboShop. But there are a lot documents and the tests showed that it is not easy to quickly find the right help document.

"Maar dan moet je je voorstellen, dan moet je iets doen en dan lukt dat niet en dan ga je het zoeken en dan lukt dat ook niet. Dus dat is dan redelijk frustererend." – User 2

So a search functionality within the help section would help the users to quickly find the right help documentation. Another idea came from user 4:

"Het zou het mooiste zijn als er gewoon een hulpje zou zijn bij hetgene wat je bestelt of zo, of uitleg."

5.2.6 UI presentation

The RaboShop consists of a few catalogues with different interfaces. The results will be divided in three catalogues: Algemene catalogus, Kantoorartikelen and Software catalogus.

Algemene catalogus

The general opinion was that the interface of this catalogue could be improved.

> *"De look and feel zou ik helemaal aanpassen." – User 3*
>
> *"Hoe het eruit ziet? Dat kan wel iets catchier. Een wat vlotter of sneller uiterlijk, beetje oubollig." – User 4*

They also think the interface looks chaotic. There is a lot of information and a lot of scrolling is needed because the results are not visible when the user is on top of the page.

> *"[..]. Vind het redelijk rommelig overkomen, dat het een klein schermpje is, en je hebt eigenlijk de helft van het scherm. Als ik hier dan zeg ik wil bloemen, moet ik echt helemaal naar beneden scrollen, dan denk ik had dit (zoekopties) dan hieronder gezet en dit boven. Zodat het wat overzichtelijker is. [..]." – User 2*
>
> *"Op zich vind ik het niet slecht, wat ik wel vervelend is vind ik dat hieronder, dat je niet kunt zien of er iets bij komt. Het scherm had iets anders opgebouwd moeten worden zodat ik in ieder geval 1 criteria kan zien, maar nu zie ik helemaal niks. Nu lijkt het alsof er niets gebeurd is. En qua opbouw, het kon denk ik wat moderner, maar ik heb er niet echt moeite mee. Ik ben wat gewend." – User 5*
>
> *"Hetzelfde geldt aan de andere kant dat als je hier in je resultaten zit en je doet iets in je winkelwagen dat je niet ziet, dat je je afvraagt wat gebeurt er? Dat je naar boven moet gaan om te zien dat er iets gebeurd is." – User 6*

Kantoorartikelen

This catalogue was better assessed by the users. The interface is better and more familiar.

> *"Deze is sowieso qua lay-out wat meer van deze tijd, dat andere is wat vooroorlogs."*
>
> *"Dit is veel duidelijker, meer wat mensen kennen van normale online winkelwagenomgeving, de bol.com zeg maar."*

But according to the users the difference in interface also leads to confusion.

> *"Ik vind het erg vreemd dat hier een hele andere pagina hebt, dat het lijkt alsof je naar een externe site gaat. Met name de lay-out hier is raar, niet dezelfde look and feel als in de rest. Denk dat deze wel meer overeen komt dan wat je met een normale website, dit lijkt meer op een normale website zeg maar. En die andere is meer een pakket, een tool. Daar heb je veel meer de indruk dat je echt in een tool zit te zoeken en hier heb je echt een website. Dit ziet er bekender uit zeg maar." – User 6*

Also in this catalogue the search menu with different categories, which should make searching easier, is not necessary according to the frequent users.

> *"Als ik hier iets nodig heb dan, maar dat is mijn persoonlijke mening, ik doe hier (zoekmenu) weer niks mee. Als ik iets nodig heb qua kantoorartikelen dan ga ik meteen hier in het*

zoekveld invullen. En ik snap dat als je denkt waar zou ik het je kunnen vinden dit een soort navigatiedingetje is. Dus ik ja, voor mijn gevoel ik het ook weer heel chaotisch." – User 2

"De werking.. in mijn geval heb ik dit niet nodig (zoekopties), altijd als ik iets bestel gebruik ik alleen trefwoord en zoeken, meer gebruik ik eigenlijk niet, alle overbodige aanhangselen mogen er wat mij betreft uit." – User 4

Software catalogus

This interface is form-based, so totally different from the other catalogues. According to the users this catalogue is very clear.

"En qua overzicht helpt het je wel weer netjes door al die stappen te komen, en ik denk dat ik liever deze vorm heb dan die stappen bovenin, dat ik van de ene stap moet naar de andere en dat ik steeds in de gaten moet houden waar ik nu zit. Hier staat het gewoon netjes onder elkaar, en dat als je hier iets invult krijg je pas het volgende te zien." – User 5

"Die is wel gebruikersvriendelijker, want er wordt heel duidelijk aangegeven wat je moet invullen. Het is niet of dat of dat, de stappen die hier worden aangegeven vind ik makkelijker dan de stappen in.. omdat je hier formulier-based werkt, makkelijker stappen voor de gebruiker." – User 3

5.3 Frequent versus infrequent

During the tests the differences in usability issues between the frequent and infrequent users became visible. The clearest difference was the time needed to finish the user test. Where the frequent users needed about half an hour, the infrequent users needed almost an hour to finish the test. Also in the occurrence of usability issues a difference between the two groups is noticeable. Tables 11 and 12 show the co-occurrences of the usability criteria codes and the 'Frequent user' and 'Infrequent user' codes. The differences in usability issues between the two groups will now be explained.

5.3.1 Navigation

For navigation the most important problem was finding the right functionalities. This issue was present with both user groups, which can be seen in the occurrence of issues and also in the detected usability issues. Unused options could be removed according to frequent users is the only difference between in the usability issues of both groups.

	Frequent user	Infrequent user
Navigation	18	20
Task support	19	39
Errors	5	8
Search function	36	50
Learnability	11	8
UI presentation	13	27

Table 11

5.3.2 Task support

The task support knows more differences, at least in the occurrence of usability issues. The infrequent users have a lot more issues than the frequent users. However the usability issues themselves are not that different for both groups. The only difference in the usability issues is that the categorization of the search functions are not clear according to the infrequent users. In the number of occurrence of the usability categories there is a big difference between the groups. This could tell us that the infrequent users do have more problems with the usability issues in comparison to the frequent users. The infrequent users do have more problems with choosing the right catalogue when searching for a product. They could spend a lot of time with searching in the wrong

catalogue. Also the language used in the system was a bigger problem for the infrequent users. They could not find some buttons because of the name of the button.

5.3.3 Errors

There is not much difference in the occurrence of the usability categories between the frequent and infrequent users. Also in the detected usability issues there is no difference. However the infrequent users have slightly more problems with errors. For example when a form that belongs to a product is not filled in it is not possible to order that product. Frequent users were aware of that problem and know that they did not fill in this form when they see the error. The infrequent users did not know that a form should be filled in so they also did not understand the error and did not know how to solve it.

	Frequent user	Infrequent user
Navigation	18	20
Cannot find functionality	7	8
Task support	19	39
Efficiency	8	3
Language	4	10
Errors	5	8
Search function	36	50
Wrong catalogue	4	15
Wrong search result	17	30
Learnability	11	8
UI presentation	13	27

Table 12

5.3.4 Search function

There is a big difference in the occurrence of usability issues between the frequent and infrequent users in this category. Much more usability issues occurred with the infrequent users than with the frequent users. This is because the frequent users have more experience with the system and they better knew where to find products, and with what search word they could find them. The infrequent users did not know in which catalogue they could find a product. When they chose one they did not know with what search word they could find the product. And it also happened that the right product was filtered out of the results because the wrong category was chosen in the search option. A lot of mistakes were made because of these problems and also a lot more time was needed because of these problems.
The usability issues detected in this category do not differ that much between the two user groups. One difference is that the search options are not necessary according to the frequent users, but the infrequent users thought they were useful. So also for this category the occurrence of the issues show much more difference then the detected usability issues.

5.3.5 Learnability

There is not much difference in occurrence between the frequent and infrequent users in this category. This is possibly due to the fact that this code is mostly used in the interviews, because this code is less applicable to the user tests. The frequent users all mentioned that when the regular items are searched it is easy, but when a new one has to be searched it is hard and time consuming. The infrequent users do not know much about the learnability of the RaboShop yet, however it was clear that they had more trouble using it than the frequent users.
In the detected usability issues we can see a difference between the two user groups. The frequent users said that the RaboShop is easy to use if you know what to search, but searching new products is hard. This is consistent with the opinion of the infrequent users that the RaboShop is very hard to work with without any experience. In addition the frequent users miss a search function in the help documentation.

5.3.6 UI presentation

This category has a big difference in the occurring problems between the two groups, and also in the detected usability issues. Where the frequent users mostly complained about the look and feel of the interface, the infrequent users had more trouble with the visibility of the system status. Also the infrequent users were confused by the change in look and feel when visiting the Kantoorartikelen. They really got the feeling that the RaboShop consisted of a few separate parts that are tied together. The differences in interfaces could explain the difference in the occurrence of usability issues. Each catalogue demands to learn a new interface. The infrequent users could not use what they had learned about a catalogue in the other catalogues, so they made more mistakes.

In Table 13 the differences in usability issues between the frequent and infrequent users are summarized. The usability issues that both groups experience have a grey colour and the issues with different outcomes are black.

Usability issue	**Frequent user**	**Infrequent user**
Navigation	- Some functionalities are hard to find. - Unused functionalities could be removed.	- Some functionalities are hard to find
Task support	- Sequence of steps to order a product is not logical. - It is not clear which products can be found in which catalogue. - Guidance-type information would be helpful. - It does not work efficient. - A lot of steps and actions are needed to order a product. - The language used does not match with the user.	- Sequence of steps to order a product is not logical. - It is not clear which products can be found in which catalogue. - Guidance-type information would be helpful. - It does not work efficient. - A lot of steps and actions are needed to order a product. - The language used does not match with the user. - The categorization in the search options are not clear.
Errors	- Errors are not clear and it is not clear how it can be solved.	- Errors are not clear and it is not clear how it can be solved.
Search function	- It is not clear which products can be found in which catalogue. - The right search word is needed otherwise it is not possible to find products. - Search options are unnecessary. - Search function to search a name of an employee is difficult.	- It is not clear which products can be found in which catalogue. - The right search word is needed otherwise it is not possible to find products. - Search options are useful, but it needs better categorization. - Search function to search a name of an employee is difficult. - Information about wildcard would be helpful.
Learnability	- If the user knows what to search it is easy, but ordering new	- The RaboShop is very hard to work with without experience.

	products is difficult and time consuming. - A search function within the help section would be helpful.	
UI presentation	*Algemene catalogus:* - Oldfashioned and chaotic. - Search options could be removed. - Lot of scrolling needed. *Kantoorartikelen:* - Better than the Algemene catalogus - Search option not necessary. *Software catalogus:* - Clear interface, clear steps.	*Algemene catalogus:* - Lot of scrolling needed. - Results not immediately visible, so not clear if something is changing there. - Shopping cart is not visible when searching through the results. *Kantoorartikelen:* - Better than the Algemene catalogus - Multiple interfaces is confusing, the catalogues do all have other interfaces. *Software catalogus:* - Clear interface, clear steps.

Table 13

6. Conclusion

This research tried to find an answer to the question if infrequent users do experience the same usability issues as frequent users. To find this answer three methods were used: heuristic evaluation, interviews and user tests. The heuristic evaluation was conducted to get a good view of the RaboShop and already listed usability issues. The interviews and user tests were conducted with seven users, three frequent and four infrequent users. The interviews and videos were coded and analyzed. The result was a list of usability issues of the RaboShop.

6.1 Frequent and infrequent users

The main question of this research is: *Do infrequent users experience the same usability issues as frequent users?* In Table 14 the differences between the frequent and infrequent users are shown for the usability criteria. This table is based on Table 11 and 13. If there is no difference between the user groups this is marked with the sign '=', and just a very small difference is marked with '≈'. Bigger differences are marked with '<' for the lowest co-occurrence number and '>' for the highest number. If there was a big difference in usability issues this is marked with the sign '≠'. As can be seen the detected usability issues show a small difference between the frequent and infrequent user. Certain usability issues were only detected at the frequent users or only at infrequent users. However these issues are detailed and specific issues. The usability criteria, so the main categories of usability issues, are all detected in the interviews and user tests of both groups. The frequent users do experience usability issues in these categories and also the infrequent users do. Therefore the difference between the detected usability issues is minimal. The occurrence of the usability issues in the interviews and user tests show more difference between the user groups. Three of the usability criteria do occur a lot more with the infrequent users than with the frequent users.

So the frequent and infrequent users do experience the same usability issues, at least the same usability criteria. However the occurrence of the usability criteria is different in the two groups. Usability issues do occur more at the infrequent users, and they also suffer more from these issues. As could be seen in the user tests the frequent users did run into usability issues, but they could solve it more quickly than the infrequent users. If the infrequent users ran into usability issues they really could get stuck and therefore could not finish the task.

Usability criteria	**Frequent user**		**Infrequent user**	
	Co-occurrence	Issues	Co-occurrence	Issues
Navigation	≈	≈	≈	≈
Task support	<	≈	>	≈
Search function	<	≈	>	≈
Error	≈	=	≈	=
Learnability	≈	≠	≈	≠
UI presentation	<	≠	>	≠

Table 14

6.2 Methods

The second research question was: *Do different research methods lead to different results?*
For this research a heuristic evaluation, interviews and user tests were conducted. When the results of the heuristic evaluation and the interviews and users tests are compared, it can be seen that the results of the heuristic evaluation are more general and those of the interviews and user tests more specific. All usability issues listed in the heuristic evaluation can also be found in the results of the interviews and user tests. The results of the interviews and users tests are also more extensive than

those of the heuristic evaluation. The most important difference between the results of those methods are the usability issues of the search function. The search function is barely mentioned in the results of the heuristic evaluation while this is almost the most important issue that came out of the interviews and user tests.

6.3 Discussion

The expectation of the results was that there would be a difference between the issues of the frequent and infrequent users. The RaboShop is not easy to use so it was expected that it would be harder to work with for infrequent users. According to the results there is a slight difference between the issues of the frequent and infrequent users. They experience about the same amount of issues, but some issues are different. As expected the infrequent users did experience more trouble during the user tests. They needed more time and also did not succeed in some tasks. The usability issues found during this research looked familiar for the department responsible of the RaboShop. The usability issues found confirmed their impression of the problems of the RaboShop. They indicated that some of the usability problems were underestimated by them before and the problems were never listed as clearly as in this research.

The difference in results of the various methods also met the hypothesis. It was expected that the results of the user tests were more detailed than the results of the heuristic evaluation and the interviews. In this research the results of the interviews and user tests are discussed together, so it cannot be stated for sure that the results of both methods would be different. However the expectation is that they would differ. The user tests show better how the user works. This enables to see details that would not have been mentioned in interviews or would not be discovered in a heuristic evaluation.

The results of this research can be compared with those from Topi, Lucas and Babaian. The results differ on some points. The biggest difference is that in their usability issues the search function does not occur while that is a big usability issue here. The search function probably is the biggest usability issue of this system, which makes it surprising that Topi et al. do not even speak of it. Usability issues that do occur in the results of this research and of those of Topi et al. are the navigation (Identification of and access to the correct functionality), task support (Transaction execution support), errors and terminology. For the navigation problem also the problem with finding functionalities was clearly mentioned. With the task support the absence of guidance-type information can clearly be recognized in the results of Topi et al. The unclear error messages and terminology were also important usability issues of the system of their research. There is one usability issue that cannot be found in the results of this research: system output limitations. Probably this is due to the fact that both researches use a different system. The system used for this research does not know any output. The last usability issue of Topi et al. is ‘overall system complexity’. This is an issue that can be recognized in the results of this research. The RaboShop is a complex and intimidating system according to the users.

This research contributed to the better understanding of usability issues with ERP systems, by finding the usability issues of this system. Some of the usability issues found seem not to have been identified in earlier research. However this is just a case study and more research should be conducted before conclusions can be drawn. This research could be a start to bring us closer to a complete list of usability issues in ERP systems. It also gives more insight in which users should be involved when looking at usability problems. To improve the usability of a system not only frequent users but also infrequent users should be involved. By involving infrequent users issues will be found that would not be found by frequent users because they do not experience them as real problems. This research also contributes to the understanding of the use of different methods. For usability

testing, user tests produce more detailed results and give a better understanding about the way users work with a system. Heuristic evaluations give a more general impression of the usability. Interviews are in between those two methods. They give a more general impression than user tests, but they tell more about the way users work with the system than a heuristic evaluation.

This research should be done with multiple ERP systems to get a more general view. The results of this research will probably contain some system specific usability issues. Also this research has been carried out by one person. It should be done by more researches to get a more reliable result. For example the stimulated recall in the interview would have been better with more researchers. After hearing and seeing the interviews and user tests one could conclude that more questions could have been asked about some specific moments in the user tests. The reliability of the heuristic evaluation can be doubted because it is conducted by only one person. For reliable results this should be done by more usability experts.
In future research the listed usability issues could be translated into the interface of the system. The new interfaces could be tested with users of the system to see if the found usability issues really decrease the user satisfaction of systems.

7. Literature

Amoako-Gyampah, K., & Salam, A. F. (2004). An extension of the technology acceptance model in an ERP implementation environment. *Information & Management*, *41*(6), 731-745.

Bingi, P., Sharma, M. K., & Godla, J. K. (1999). Critical issues affecting an ERP implementation. *IS Management*, *16*(3), 7-14.

Calisir, F., & Calisir, F. (2004). The relation of interface usability characteristics, perceived usefulness, and perceived ease of use to end-user satisfaction with enterprise resource planning (ERP) systems. *Computers in Human Behavior*,*20*(4), 505-515.

Drive SAP HCM Adoption: Learn how to make HR self-services easy-to-use, intuitive, and accessible anywhere. (n.d.). Retrieved March 18, 2015, from http://sapinsider.wispubs.com/~/media/Alloy/Whitepapers/Drive_SAP_HCM_Adoption_White_Paper-Sitrion-SAPinsider_041014.pdf.

History of Rabobank. (n.d). Retrieved March 11, 2015, from https://www.rabobank.com/en/about-rabobank/profile/history/index.html.

ISO/IEC, (1998). ISO/IEC 9241-14: 1998 (E) Ergonomic requirements for office work with visual display terminals (VDT)s - Part 14 Menu dialogues.

Johnson, D. (2012). For full ERP benefits use cloud infrastructure and cloud applications. Retrieved March 18, 2015, from http://erpcloudnews.com/2012/07/for-full-erp-benefits-use-cloud-infrastructure-and-cloud-applications/

Kwak, Y. H., Park, J., Chung, B. Y., & Ghosh, S. (2012). Understanding end-users' acceptance of enterprise resource planning (ERP) system in project-based sectors. *Engineering Management, IEEE Transactions on*, *59*(2), 266-277.

Matthews, D. (2008). Usability as an ERP selection criteria. *IFS White Paper, January*.

Nielsen, J. (1992). Finding usability problems through heuristic evaluation. In *Proceedings of the SIGCHI conference on Human factors in computing systems* (pp. 373-380). ACM.

Nielsen, J. (1994). Enhancing the explanatory power of usability heuristics. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (pp. 152-158). ACM.

Nielsen, J. (1994). *Usability engineering*. Elsevier.

Nielsen, J. (1995). Usability inspection methods. In *Conference companion on Human factors in computing systems* (pp. 377-378). ACM.

Nielsen, J. (2005). Ten usability heuristics.

Nielsen, J. & Landauer, K. (1993). A mathematical model of the finding of usability problems, *Proceedings of ACM INTERCHI'93 Conference* (Amsterdam, The Netherlands, 24-29 April 1993), pp. 206-213.

Quesenbery, W. (2001, October). What Does Usability Mean: Looking Beyond Ease of Use'. In *Annual Conference-Society for Technical Communication* (Vol. 48, pp. 432-436). UNKNOWN.

Quesenbery, W. (2003). The five dimensions of usability. *Content and complexity: Information design in technical communication*, 81-102.

Singh, A., & Wesson, J. (2009, October). Evaluation criteria for assessing the usability of ERP systems. In *Proceedings of the 2009 annual research conference of the South African Institute of Computer Scientists and Information Technologists* (pp. 87-95). ACM.

ICT Informatiecentrum, ERP systemen. (n.d.). Retrieved November 17, 2014, from http://www.erpsystemen.nl/wat-is-erp.

8. Appendix

Interviewvragen

1. Waarvoor gebruik je de RaboShop? Hoe vaak? Welke delen/catalogussen gebruik je dan?

2. Wat vind je van het werken met de RaboShop? (algemene indruk van totaal)

3. Wat vind je van het werken met de algemene catalogus/kantoorartikelen/software catalogus? (afhankelijk van wat degene gebruikt) (algemene indruk van aparte delen)

4. Wat vind je van de presentatie van de algemene catalogus/kantoorartikelen/software catalogus? (is het overzichtelijk? Hebben de knoppen goede benamingen? Is het logisch wat wat doet?)

5. Is het wel eens voorgekomen dat je iets niet kon vinden in de algemene catalogus/kantoorartikelen/software catalogus? Wat vind je van de manier van navigeren? (kunnen functies makkelijk gevonden worden? Hoe is de zoekfunctie?)

6. Vind je dat de RaboShop efficiënt werkt? (is dit de snelste manier?)

7. Gaat er vaak iets mis onder het werken? Krijg je wel eens foutmeldingen? Wat vind je van deze foutmeldingen?

8. Raadpleeg je wel eens de hulp documenten? Wat vind je hier van? Wat doe je als je iets niet kunt vinden?

9. Is het systeem makkelijk in gebruik? Weet je na enige tijd er niet mee gewerkt te hebben nog hoe het werkt?

Instructies gebruikerstest RaboShop

1. Een van je collega's gaat met pensioen. Jij hebt de taak toegewezen gekregen om een afscheidscadeau voor haar te kopen in de RaboShop.

2. Volgende week mag je een nieuwe collega verwelkomen. Je nieuwe collega moet voorzien worden van een aantal zaken voor zij echt kan beginnen. Regel dit in de RaboShop. Bestel hiervoor de volgende items:



3. Je nieuwe collega heeft voor haar werkzaamheden ook toegang nodig tot een programma waarmee zij afbeeldingen kan ontwerpen. Bestel dit in de RaboShop.

4. Volgende week komen er een aantal gasten op bezoek. Helaas zijn er geen parkeerplaatsen meer beschikbaar in de parkeergarage bij de locatie Croeselaan en zullen ze bij de Jaarbeurs moeten parkeren. Zorg ervoor dat je over kaartjes beschikt waarmee deze gasten bij de Jaarbeurs kunnen parkeren.

5. Je hebt iets besteld in de RaboShop, maar morgen ga je op vakantie en je bent pas 5 januari weer terug . Je wilt graag dat je collega, Sanne Derckx, je bestelling in de gaten kan houden. Het is daarnaast ook jouw taak om bestellingen van anderen goed te keuren of te weigeren, dus dit kan Sanne dan ook overnemen. Zo kan iedereen rustig doorwerken gedurende jou afwezigheid. Stel in de RaboShop in dat Sanne jou kan vervangen tijdens je vakantie.

Gegevens

Naam gebruiker: Sanne Derckx

Geboortedatum 01-01-1980

Personeelsnummer: 12345

Kostenplaats: 12345

Werkpleknummer: 123456

Emailadres: s.derckx@rn.rabobank.nl

Telefoonnummer: 0612345678

Standplaats: Utrecht Croeselaan