User Experience and Its Importance in Banking Software

A Study of User Experience During Development

By

Majlinda Haziri

Department of Electrical and Information Technology
Faculty of Engineering, LTH, Lund University
SE-221 00 Lund, Sweden

Abstract

This study was conducted to provide an explanation as to why user experience is important when designing software, specifically for a banking company undergoing a complete redesign of their banking client. Neglecting the user experience design in the development of any application or webpage can lead to user frustration and abandonment. Frustration can lead to terrible customer service and reliability, such as an employee too focused on an issue with the banking client rather than the issue the customer is hoping to resolve. Companies have realized the importance of user experience when designing their software and the industry has incurred a large demand for front-end developers and user experience designers. From gaming to finance to marketing and retail, the benefits in designing an exceptional user experience is worth the time and effort.

Besides user experience being the core concept discussed in this thesis, gathering information about user interfaces was necessary in order to understand the design of a good user experience. Information gathering was done on front-end frameworks and their importance on the way people access media and applications. Development of front-end frameworks, their use, and their impact on the technological landscape can be explained further, but the focus of the thesis is on the benefits of user experience in banking software. The result was also used to give an extensive recommendation to Resurs Bank, comparing it to their current client, with the goal of creating better usability and less frustration.

The data gathered through the survey and interviews shows that Resurs Bank can improve the UX of their software. Designing a new web client with the feedback gathered would help boost productivity by solving issues that would hinder an employee’s ability to help a customer. Besides an increase in functionality and usability, maintenance can be substantially easier depending on the front-end framework chosen. Many of the new modern frameworks have enough documentation and updates to make maintenance much easier compared to older frameworks.

Using Bootstrap as the front-end framework would make it easier to create a modern, cohesive UI and make it easier to design the UX. Once the UI is visually updated, the developers can design a UX tailored to the data gathered through the survey and interviews, such as keeping popular shortcuts and removing clutter. Resurs Bank can also use the analysis done on their current client and consider it when designing the layout of their new web client.

Keywords: User Experience (UX), Front-End Framework, Model-View-Controller (MVC), Web Client, User Interface (UI)

Sammanfattning

Denna studie genomfördes för att hjälpa till att förklara varför användarupplevelse är viktig när man utformar programvara, speciellt för ett bankföretag som vill genomgå en fullständig omformning av sin bankklient. Det kan leda till frustration och övergivande hos användarna om man försummar användarupplevelsedesignen när man utvecklar en applikation eller webbsida. Frustration kan leda till dålig kundservice och tillförlitlighet, till exempel en anställd som är för inriktad på att lösa problem i bankklienten snarare än att hjälpa lösa kundens problem. Företagen har insett vikten av användarupplevelse i sin programvaruutveckling och det finns nu en stor efterfrågan på front-end utvecklare och användarupplevelsedesigners. Från spel till finans till marknadsföring och detaljhandel är fördelarna med att designa en exceptionell användarupplevelse värt den tid och ansträngning som behövs.

Användarupplevelse är kärnkonceptet som diskuteras i denna uppsats, men det var även nödvändigt att samla information om användargränssnitt för att förstå utformningen av bra användarupplevelser. Olika front-end ramverk och deras betydelse för hur människor får tillgång till media och applikationer i modern tid, undersöktes också. Utveckling av front-end ramverk, deras användning och deras inverkan på det tekniska landskapet kan definitivt förklaras ytterligare, men fokusen i uppsatsen är att ta upp fördelarna med bra användarupplevelse i bankprogramvara. Resultatet användes också för att ge en omfattande rekommendation till Resurs Bank, en jämförelse med den nuvarande klienten, med målet att skapa bättre användbarhet och mindre frustration.

Data som samlats in genom enkäten och intervjuer visar att Resurs Bank kan förbättra användarupplevelsen för deras programvara.

Utformning av en ny webbklient med hjälp av den samlade feedbacken skulle bidra till högre produktivitet genom att lösa problem som annars skulle hindra en anställd att hjälpa en kund. Förutom ökning av funktionalitet och användbarhet kan underhållet bli väsentligt lättare beroende på vilken ram man väljer. Många av de nya moderna ramverken har tillräckligt med dokumentation och uppdateringar för att göra underhåll mycket enklare jämfört med äldre ramverk.

Att använda Bootstrap som front-end-ramverk skulle det bli enklare att skapa ett modernt, sammanhängande användargränssnitt och att utforma en bättre användarupplevelse. När användargränssnittet är visuellt uppdaterat kan utvecklarna designa en användarupplevelse skräddarsydd för data som samlas in genom enkäten och intervjuerna, till exempel att behålla populära genvägar och ta bort det som är rörigt. Resurs Bank kan också använda analysen som gjorts på den nuvarande klienten och överväga den när man utformar layouten på sin nya webbklient.

Nyckelord: Användarupplevelse, Front-End Ramverk, Model-View-Controller (MVC), Webbklient, Användargränssnitt

Acknowledgments

I would like to thank Resurs Bank, especially my supervisor Tobias Friberg, for allowing me the opportunity to shape my thesis around the development of their new web client, and providing avenues for data gathering and helpful feedback. Without access to a real-world example of development, this thesis would have been much more difficult to complete.

I also want to extend my gratitude to the employees working over at Resurs Bank for taking time out of their days to provide valuable data and providing me the opportunity to observe them during their daily routines. The data gathered from their experiences and feedback was invaluable to the success of this thesis.

Lastly, I would like to thank Christin Lindholm and Christian Nyberg from LTH, for looking over and guiding me through the entire process.

Majlinda Haziri

List of Contents

1. Introduction

User experience is often neglected when designing a new application or system. Most developers deal with functionality and not with how the functionality is presented. Without an emphasis on user experience, companies run the risk of losing out on potential business to competitors. A good user experience will keep users focused on the purpose of the application rather than asking questions or feeling confused. This section provides some insight into the motivation and purpose of the thesis. It also defines some of the problems and limitations. The emphasis of the thesis is the importance of user experience in regard to development of a new web client for Resurs Bank.

1.1. Background

Resurs Bank is a leader in the consumer credit market in the Nordic region, offering payment-solutions and consumer loans to over 5.5 million customers. Resurs Bank is a large company with connections in finance and commerce. To support their large user base, Resurs Bank uses a banking system, Bank2010, they developed and have maintained through the years. Resurs Bank seeks to improve all facets of the company, including technology.

For Resurs Bank to better serve their customers, they are looking into developing a modernized web client. To ensure that their needs are met and within budget, Resurs Bank wants to create a proof of concept, PoC. The PoC will be divided into two different sections, the back-end programming and the user experience. This thesis will be focusing on the user experience of that web client.

Although there have been previous theses focusing on user experience with websites, this thesis will be unique in that sense that the focus will be on the user experience for banking software.

1.1.1. User Experience

User experience is defined as the overall experience of a person using a product such as a website or computer application, especially in terms of how easy or pleasing it is to use. The main use of user experience when designing a user interface is to keep the user pleased and informed. Users that experience frustration and confusion are more likely to abandon an application even if all the necessary features are built into it. So, while an application needs to provide features or functions that make tasks easier, it also needs to be easy to use and predictable. Applications that manage to combine both functionality and usability tend to have the highest retention and satisfaction rates [1][2].

1.2. Purpose

When developing an entirely new web client, it is important to increase efficiency. One of the ways to ensure that the new web client is efficient is by creating a user experience that is easy to learn and filled with the necessary tools. If the new web client is too difficult to learn or convoluted, then issues may arise that could make the new web client inferior to Bank2010, the current client. If Resurs Bank wants a better return for their investment, it would be beneficial for them to create a web client that also increases productivity. Ease of use, clarity, and simplicity are all concepts that help create a positive user experience, which can help increase efficiency and productivity. Whether it be teaching new employees to use the new web client, or providing customers with quicker service, creating a good user experience is paramount.

To ensure that the user experience is improved, this thesis will focus on gathering data to determine necessary tools that should be intuitively presented, as well as an avenue for feedback for future improvements. If the focus on user experience is determined to be of value, other companies may start to focus on this area as well. It acts as an incentive for companies to create easy to understand and useful systems for employees. This could give Resurs Bank an edge in the market as frustration and difficulty can cause consumer remorse, which is incredibly difficult to resolve.

1.3. Objective

The main goal for this thesis is to gather data from a variety of sources, in order to analyze and determine what the user experience needs. Interviews and surveys will provide the majority of the data. Information gathereing on common user experience methodology and implementation will also be used to better analyze the data. The user experience should not hinder the current users and would ideally be easier to understand and navigate than Bank2010.

Using the gathered data, the thesis will provide recommendations to Resurs Bank based on analysis and research. Additionally, client examples were taken from their current client, analyzed, and criticized. A draft of this thesis will then be presented to the developers for initial feedback, and then finalized to give the developers an overview of what they should focus on when developing the user experience. This finalized data will be used to decide on a front-end framework in order to future proof the new web client and avoid problems similar to the ones they are experiencing with Bank2010.

1.4. Problems

The thesis will answer the following questions:

1. What can be done to give Bank2010 a more modern look and would it be beneficial?
2. What do the users like and dislike about Bank2010?
3. Which tools or functions are used most and least often?
4. Which front-end framework would work best with the new web client, and is it future proof?

1.5. The Motivation of the Thesis

The reason this thesis will focus on user experience is because it is often disregarded when developing applications, since most developers focus on the functionality rather than the user's feelings when interacting with the application. It allows for the development of skills regarding dataset analysis. Resurs Bank also has a large user base which makes the task of creating a good user experience all that more challenging and rewarding.

Resurs Bank finds it beneficial to understand how to create a user experience that will allow their users to be more efficient. An increase in efficiency would allow them to better serve their customers and improve their productivity, which could lead to larger profits in the future. If proven to be successful, it could be used to urge other companies, especially those without a technological background, to put some effort into their user experience.

1.6. Limitations

Since this thesis emphasizes on user experience, it will not focus on the back-end programming of the PoC.

The main limitations will be the size of the dataset received through the methods detailed in chapter three. Depending on the availability of the current users and the developers, the dataset may vary largely in size. Getting in contact with developers, users, and stakeholders and inquiring about the user experience may prove to be a challenge, and therefore the thesis will focus mostly on the customer support department. Additionally, some feedback may not be thorough enough to draw any conclusions.

2. Technical Background

This chapter details the most common front-end frameworks and libraries used in the industry in order to provide some background before finalizing the recommendation for Resurs Bank.

The backbone behind the UI and UX, front-end frameworks and functional libraries are extremely important. Developers need the tools to build modern and attractive UI, while also having the tools to build the functionality that drives the entire application.

Most front-end frameworks have a common goal: give developers the ability to develop simple and modern UI and UX while also having the performance to be responsive and scalable. Most are based on either the Model-View-Controller, MVC, architecture or components. Choosing a front-end framework or library is an important decision that affects the future maintenance and longevity of the application. From different programming languages to different package sizes and feature sets, each framework has a purpose and it is important to identify what the new web client will require before making a final decision.

2.1. React

React is a widely used and powerful JavaScript library that enables developers to create elegant and responsive UI. It was first developed by Facebook before eventually becoming available to the public in 2013 [3]. It aimed to move away from the MVC framework model that was prevalent in the industry with prior UI frameworks by building around components [3]. React was built to be much quicker and easier for developers to learn. It is entirely JavaScript based, meaning there is no requirement to learn any outside API calls or functions. Additionally, React enables developers to use the JSX extension, an XML-like syntax which can make code easier to visually read and understand.

Most other frameworks rely on enhancing HTML through JavaScript. React generates HTML from JavaScript. This allows any experienced JavaScript developer to develop a fast and efficient, as well as modern, UI. React also made it much faster to update the DOM with reconciliation, where any changes that are made between the first render and the updated render are implemented [3]. This is much faster than completely rendering the representation with data that may not have changed.

Ever since its release, React has grown in popularity, see Figure 1 [4]. Many companies have taken notice of its ability to create well-designed UI and in effect, allow designers to focus more on the UX. Companies like Netflix, Facebook, Instagram, Dropbox, Khan Academy, and Code Academy have chosen React to develop their webpages or UI elements [5].

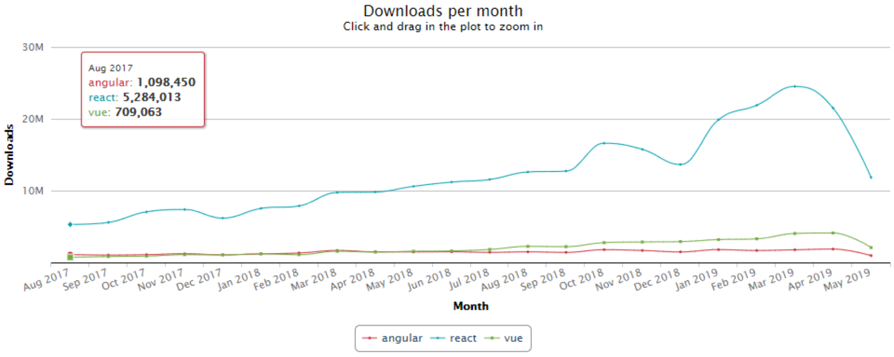


Figure 1: Measurements of front-end framework popularity based on npmjs.com downloads, a popular package manager. Chart generated by npm-stat.com [4]

As mobile apps became more common, React Native was developed to allow developers the ability to create responsive and scalable UI that was already possible on React, but on mobile platforms [6]. It was released in 2015 and has become a staple for application development ever since, one of the reasons being the ability to develop cross-platform applications. Discord, AirBNB, UberEats, Skype, Instagram, Townske, Myntra, Gyroscope, and many others have all built their mobile applications on React Native [7].

2.2. Angular

The original AngularJS was developed by Google as a JavaScript front-end framework that used the traditional MVC architecture with declarative programming being the backbone for designing and creating UI [8]. It was one of the first front-end frameworks released, available since 2010. Angular has been described as an HTML enhancer, a framework that enhances HTML with features such as loops and conditionals [9]. Support for the framework was eventually scrapped as Google switched directions with its development.

In 2016, Angular 2.0, or Angular, was released as an entirely new front-end framework built with TypeScript. It was a complete rewrite of AngularJS and caused a ton of outrage from previous developers, mainly due to the lack of backwards compatibility and migration paths [10]. Even as developers dropped Angular and adopted other frameworks, development has remained consistent throughout the years. IndieGogo, Overleaf, E-Trade, Forbes, Autodesk, VMware, and McDonald’s are just some of the large companies that rely on Angular to run their web pages [11].

With Angular being written in TypeScript, it makes learning it cumbersome compared to JavaScript, which has become the standard for front-end development in the past decade [12]. Its popularity has declined as other frameworks are much easier to learn and provide similar or faster performance, see Figure 2.

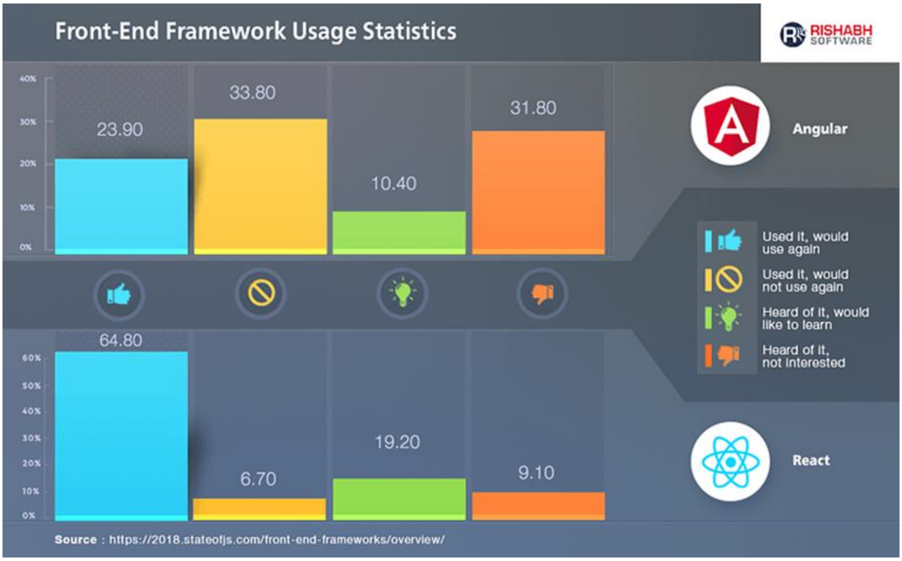


Figure 2: Graphs showing the differences in opinions between Angular and React by users and developers [13].

2.3. VueJS

One of the front-end frameworks to rival React and Bootstrap in front-end development, VueJS was released in 2014. The original creator, Evan You, aimed to make a framework that was similar to Angular but lightweight [14]. It is an MVC framework but unlike Angular, VueJS is written in JavaScript. Similar to React, VueJS also aims to find efficient re-renders with it tracking reactive dependencies.

Its large adoption rate is primarily attributed to its simple learning curve and combination of the primary benefits from Angular and React [15]. As more and more developers flocked to VueJS, the community flourished and lowered the barrier to entry even further.

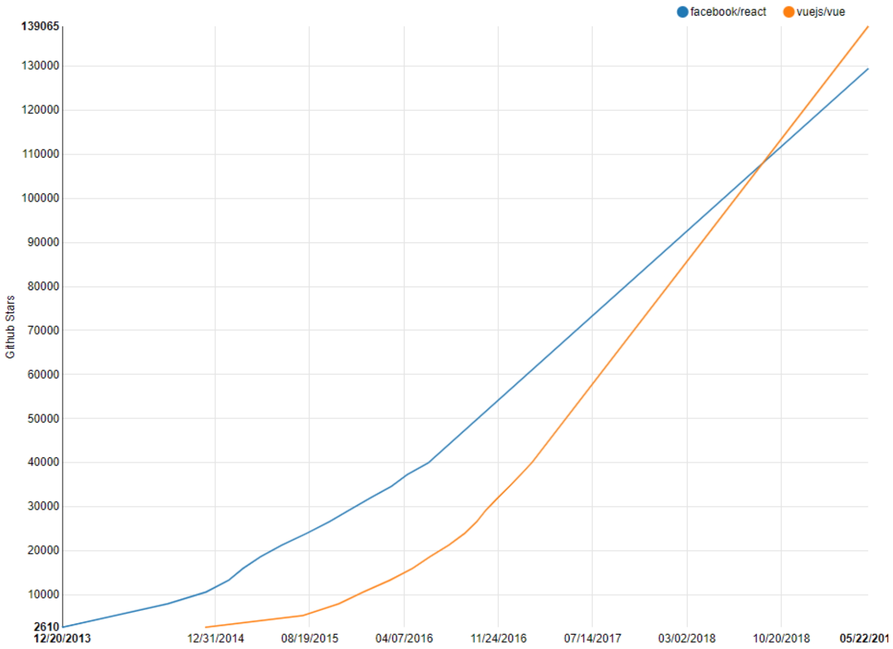


Figure 3: Graph showing the rise in popularity between React and VueJS based on GitHub stars. Created with a tool called star history. [16]

Some of the companies that have already adopted VueJS are Alibaba, Grammarly, Xiaomi, Reuters, Adobe, and Laracasts [17]. With its high adoption rate, ease of use, and efficiency, VueJS is a major player in the front-end framework space, see Figure 3. The only caveat currently is that the job market is primarily based in China, which can be problematic for developers aiming to work in Europe or the United States [18].

2.4. Bootstrap

Bootstrap was originally developed by Twitter with the goal to encourage consistency across internal tools [19]. Massive libraries are not uncommon for interface development, which is a huge burden when it comes to maintenance and consistency. It was released in 2011 and has had three major updates, with Bootstrap 4 being the latest version.

Bootstrap's main draw is the uniformity and simplicity of its HTML elements. The result is an appearance that is consistent across web browsers. Depending on the project, this can cut down on development time and provide an interface that is easier to maintain.

Besides basic styles, Bootstrap is equipped with several JavaScript components in the form of JQuery plugins. These plugins can provide additional functionality, such as an auto-complete function for input fields. Additionally, layout components called "containers" store all the elements inside them. These containers have a width that is fixed or fluid, with fixed having a few select sizes and fluid filling up the entire page size.

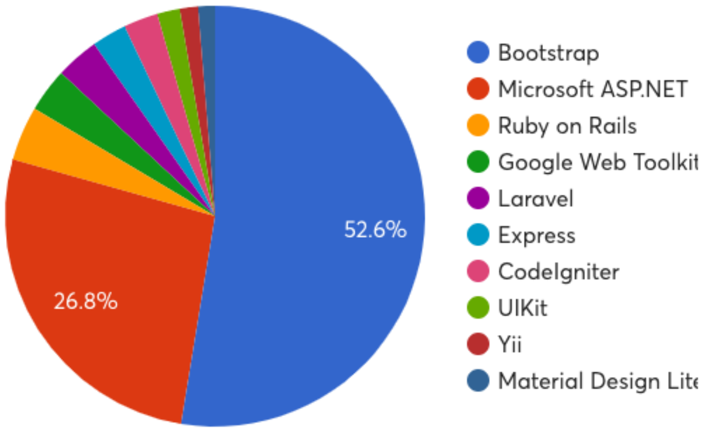


Figure 4: Bootstrap popularity in 2018 as a web framework. Analysis by Wappalyzer [20].

One of Bootstrap's drawbacks is its size or "heaviness" [21]. Bootstrap includes CSS and JavaScript, which while great for functionality, can increase load times. If speed is the primary goal, a leaner framework would be best. If functionality and ease of use are top priorities, Bootstrap is simple to use and is filled with a variety of functions. Additionally, due to its popularity as a front-end framework, see Figure 4, and basic styles, many websites share common themes or appearances [22]. If a unique look with unique functionality is the aim of the project, it could pose a challenge trying to create it with Bootstrap's structured environment. Finally, Bootstrap does not work without JavaScript. While it is rare for a user to not have JavaScript running on a browser, it does mean a backup made entirely in CSS would be required for the site to function properly.

2.5. Foundation

Originally released in 2011 by ZURB, Foundation is a front-end framework that uses HTML and CSS components to create UI elements. It has had major updates since its release, with the most recent update being in 2015. It can also run JavaScript extensions for additional functionality. It has functions unique to mobile devices, such as touch input, built into it natively [23]. One of its greatest strengths is its ability to rapidly prototype or build front-end experiences [24].

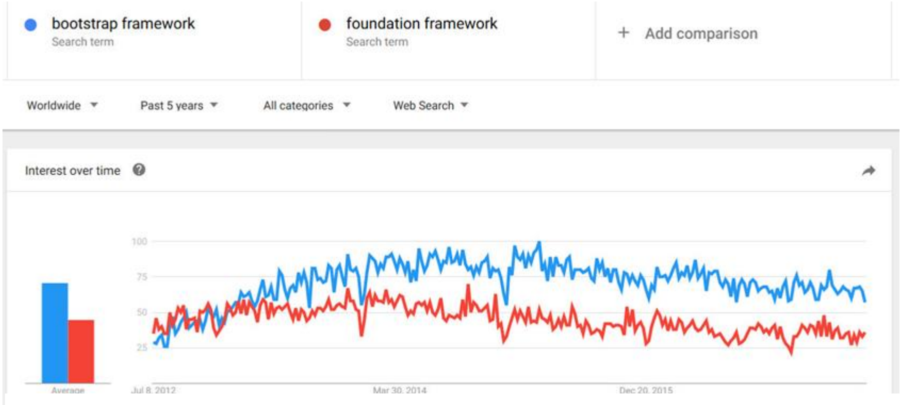


Figure 5: Google search results showing interest overtime between Foundation and market leader Bootstrap [25].

Foundation can also be used with SASS, a CSS extension that allows the use of tools not traditionally found in CSS, such as variables and nested rules. It also is compatible with every version of CSS and helps maintain a clean and organized style sheets.

Although not as popular as other frameworks, see Figure 5, companies like Runescape, Crowdrise, New Balance, Disney, BMW, HTC, and Fandango all use Foundation for their webpage designs

[26]. Overall, a solid choice without much downside besides its low popularity leading to less support and troubleshooting solutions available to the public.

3. Methods

In this chapter, the methods used in the thesis are explained in detail. Figure 6 shows the five different phases the thesis went through before being published: thesis proposal, pre-analysis, data collection, first draft, and finalization. Each phase was allocated a time slot on a time plan created to keep the thesis on track. The thesis proposal took about four weeks, the pre-analysis three weeks, data collection five weeks, the first draft around eight weeks, and the finalization about one week. Overall, the time plan allowed each phase to feel complete and avoid feeling rushed due to time constraints.

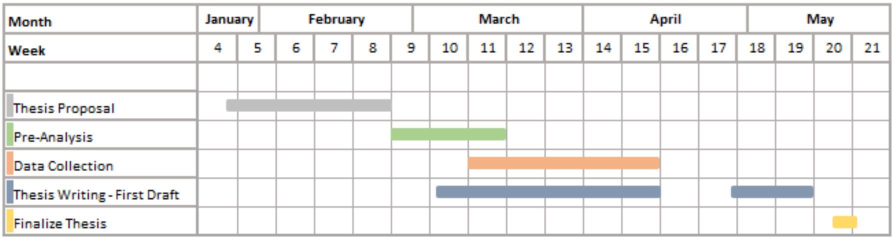


Figure 6: Time Plan

Interviews were conducted with the managers and developers of Resurs Bank, in order to narrow down the topic for the thesis. After the thesis topic was finalized, the focus shifted to gathering data unique to Resurs Bank. As UX is inherently subjective, a personal data set would be required to provide a thorough analysis and recommendation. A survey was created based on observations in the Resurs Bank office and information about UX design. After the data was collected, it was analyzed and served as a basis when eventually analyzing the Bank2010 client. Once the problems and issues were established, the shift was made to gather information about front-end frameworks and their role in UI and UX design. The recommendation was then written, based on the information gathered on Resurs Bank, UI and UX design, and front-end frameworks.

Four different methods were used when it came to collecting data for the thesis. It is important for any research-based thesis to gather data and information to provide proof or reason without it being completely opinionated. The primary goal was to gather information regarding user experience and user interfaces. This was necessary as UX design does not have a single solution that solves every problem.

Interviews were carried out to better understand the issues that customer service representatives faced when using Bank2010. If employees were able to connect to a human element when asked questions, they could possibly provide more honesty and urgency. It would also provide a better incentive for the employees to answer the survey that might otherwise be considered a waste of time.

Literature and online research was conducted to give reasoning and validity to the data gathered through the interviews and survey. Most of the information was gathered from online sources, but two publications [1][27] contain a large amount of useful explanations and concepts. A large amount of the information found online in the form of blogs, articles, and data repositories were either confirmed or repeated in the literature. The research was necessary to understand UX before writing relevant interview and survey questions. Without the knowledge acquired from the research, it would be difficult to propose any sort of question and expect useful feedback. It would also make analysis of the data much more difficult.

Observations clarified some issues that were not reported. Customer service representatives gave consent to being observed during their daily routines. Each new day presented new experiences for customer service representatives and their customers. These daily interactions that were observed helped underline the necessary functions and features that were consistently used, especially for someone that had no experience with Bank2010. Observing the employees also helped shape some of the questions in the survey.

The survey was the final piece of data that was used to provide a recommendation to Resurs Bank. It provided the unique data set that is often required to improve upon the current UX. There needed to be a unique data set that could be analyzed in a more discrete matter because UX design is subjective. Thankfully, the survey was answered by the Resurs Bank employees in a prompt manner.

3.1. Source Criticism

When looking for sources, effort was made to find multiple sources with similar outcomes or thoughts. This was important to ensure that the data contained in the sources was common or widespread. If there was an opposing view or criticism, research was done to determine whether the criticism was valid. After much research, the selected sources were determined to be legitimate with the consensus and written in the field by developers experienced with the subject. The sources provided legitimates examples or data that is openly available to the public.

[1], [2], [27], [28] and [29] are published literature that are reputable about interviews, surveys, and user experience. The information contained in these sources served as a background when formulating the questions used during the observations, interviews, and survey. [5], [8], [9], [10], [11], [12], [13], [14], [17], [18], [20], [21] and [22] are researched tech blogs, articles, and websites that share common information with other sources of similar topics. They were crucial in providing data about the popularity and use of the selected frameworks. [6], [7], [15], [19], [23], [24] and [26] are repositories or

websites owned by the creators of their respective framework. They provided concrete documentation and information about the technology inside their frameworks. It also helped validate the content from the previous sources. [30], [31], [32], [33], [34], [35], [36], [37], [38] and [39] are tech websites with concrete documentation regarding definitions. Definitions are required to verify that the information is valid and unchanged. [16] and [25] are tools that provide results that can be replicated.

3.2. Interviews

The first interview was conducted with the manager and a developer. It was centered on developing a PoC for a new web client. Notes were taken during the discussion and reflected on in private. Additionally, questions regarding the company's employees, business, culture, statistics, and goals were asked. These questions were asked to create a more personal thesis topic that could benefit Resurs Bank. As the meetings progressed, the idea of a UX focused thesis was proposed. After a few more interviews, it was decided that the thesis would focus on the importance of the UX in developing their new web client.

Learning about Bank2010 and its daily use was important. It allowed more thorough research to be done and helped provide some clues into what exactly the developers needed to work on to modernize the software. Interviews were conducted among the employees in order to gain some personal insight into the thoughts going into the redesign, and to better understand the goal they hoped to achieve.

Semi-structured interviews [28] were conducted due to limited knowledge of Bank2010, allowing the information to be controlled primarily by the interviewee. These face-to-face interviews created a dynamic where questions and concerns were built upon previously answered questions or comments, which was helpful in getting useful information. The information was then used to develop a survey that would be distributed at a later date.

3.3. Literature Reviews and Online Research

When deciding on the necessary research for this thesis, it was important to search for a variety of articles and literature that explained the concept of front-end frameworks and user experience due to the nature of personal taste and opinion. User experience is a field that delves into the methods and appearances that best retain and satisfy the user, which can be quite difficult to explain objectively without broad research. Research was done on the two different interview types in order to decide which method to use for the initial interview: structured interviews or semi-structured interviews. Additionally, research on the differences between different front-end frameworks was necessary, as user interface and user experience rely heavily on these technologies. The main front-end frameworks that were researched were Bootstrap, Angular, VueJS, and Foundation, due to their popularity in the industry. Research was also done on React, a JavaScript library, in order to understand the usability advantages it brings into the user experience field.

3.4. Observations

6 customer service employees were observed as they helped customers. Issues were documented without interfering with the workers. The observations were held during the day and lasted for three to four hours before switching to another customer service representative. An average of 35 customers were helped by each

customer service representative during each observation. These observations were conducted for two weeks.

Observing the customer support representatives as they used the client provided some important information. It helped understand some of the issues that they may not have realized were apparent due to habit. Understanding the perspective of the expert users is important when redesigning an application. Functions, shortcuts, and visual cues need to be accounted for if the goal is to improve the quality of current expert users, while minimizing the learning curve for new users. Additionally, observing the day-to-day operations gave some insight into the frustrations and shortcomings of the user experience that are rarely disclosed after the issue is resolved or forgotten. This information helped create specific questions in the survey that would remind users of some of the problems or frustrations with Bank2010, and how it can be solved with a greater focus on a simpler and more modern user experience.

3.5. Survey

The survey, see appendix I, was created with the help of the observations and research [29] explained previously. It was designed to provide some variance in responses in order to better understand the individual concerns from each user. This individuality can be recorded and tallied in order to distinguish any patterns. The survey also provided some questions that were based on a scale to better understand the different levels of frustration the users experienced. For the last section of the survey, participants were to provide their own answers in an attempt to urge some unique insight and gather information that may have been overlooked. Miscellaneous data was also gathered, such as work experience, age, and gender, in an attempt to find patterns or correlations between groups of people. The survey was anonymous in order to provide safety and encourage honesty.

The survey was written over the course of a week before being finalized with Google Forms. The survey was then given to the manager for review. It was then distributed by the manager to the customer service employees through an e-mail list. They were encouraged to complete it within a week. The survey was closed by the second week.

The data gathered from this survey was significant and helped propose a solution for the future Resurs Bank user experience.

4. Analysis

Analysis was crucial in determining the final recommendations for the Resurs Bank web client in development. A variety of data was analyzed, including personal observations, survey data, user experience research, and front-end libraries and frameworks.

Analyzing the personal observations gave insight into the struggles and frustrations that users or developers may not report. It also gave information about subconscious decisions made in their daily routine, such as function placement and layout navigation. These decisions are habitual and important to keep in the future web client. Removal of habitual UI placement, functions, or features can lead to frustration from previously experienced users. While it is important to keep certain functions and features, it does not mean that all functions and features need to remain. Functions can be moved around, retooled to different shortcuts, and presented in a different way. The challenge is determining what tools are the most common and how are they presented. Through prototyping, it is possible to develop many layouts in the new web client in order to test which have the greatest success in mimicking the UX that users enjoy from the current client.

Survey data was able to provide some information that could help develop prototypes and eventually, after some testing and feedback, the final web client. Anonymity is a tool that can best be used to garner honest thoughts and feelings on questions and topics that may otherwise never be mentioned. The survey included questions that had fields where the participants could write their own answers. This was included in addition to multiple choice questions to try to gain some data that may fall through the cracks if a question fails to mention a particular issue. Some of the data gathered through this method provided clear proof that some of the assumptions originally made in the observations were correct, such as requiring multiple devices or windows to successfully handle certain situations. The write-in answers provided a window into the thoughts that some employees have in regards to their experience with Bank2010, which is crucial when suggesting prototypes or solutions.

4.1. Observations

Shortcuts were quite important for many of the employees. Many employees showed signs of memorization and tactile proficiency. Knowing shortcuts helped provide better customer service and lowered the time between customers. With that in mind, the new web client that is being developed should maintain a focus on shortcuts, preferably keeping those that already exist. This would make transitioning easier for current employees and is evidence that some of the UX elements from the current client are salvageable.

Bank2010 is not a complete solution. Many other applications or devices were used while working with customers. Some employees used physical calculators while others used alternative software calculators. Resurs Bank even developed an application that uses BankID to identify customers, separate from Bank2010. It would be beneficial for Bank2010 to be redesigned with consolidation in mind. Take the purpose of all the external applications and devices that the employees use and provide them naturally in the new web client. This would help relieve some confusion or frustration that comes from feeling lost between all the different windows or from feeling like a tool is unavailable. Combining separate applications, especially those with a simple function such as customer verification, would improve the UX drastically.

While some features are missing, such as loan calculators or easy access to signed contracts for each account, some areas are filled with too many features or functions. Feature bloat can be a major issue but is a minor one compared to the others. Some functions rarely get used but are presented alongside functions that are much more common, which might confuse users or cause accidental clicks. If the features are mandatory, a possible solution would be to solidify user roles and determine functions for each department. This would maintain a cleaner look but also maintain the most common functions for each user, depending on their department.

4.2. Survey

The data from the survey, see Appendix II, was helpful in determining the changes necessary for the new web client to better serve its users. A personalized data set is essential to creating a user experience that meets all the expectations of the users, and the survey provides that data set for the thesis. Without a personalized data set for Resurs Bank, designing an improved UX would be more difficult and could end up being worse than the current UX.

The survey was given out to the manager and then distributed through e-mail to the customer support department before being filled out anonymously. They were given a week to answer nineteen questions in the survey. Some of the questions were multiple choice, and others were graded on a spectrum. The questions that were graded on a spectrum were graded with the two options above and below the center, a score of three, as for and against respectively. The last section were fill-in questions that encouraged discussion. The survey was designed to be completed quickly as to not bother the daily routines of the employees. It also had some questions aimed at providing information to build demographics, such as gender and age.

4.3. User Experience

A good UX is difficult to define as it differs on a case-by-case basis. In simplest terms, a good UX aims to fulfill the user's needs and create a pleasant experience. The best way to design a good UX is by listening to the users and reiterating based on feedback [27]. It is important to note that developers must understand that they are not the target audience and should be willing to adapt to feedback.

Considering that the web client is primarily designed for customer service representatives, communication between the development team and the customer service team is crucial. Without communication, the developers are at risk of developing a UX that leaves the users feeling frustrated or lost, which makes the entire development cost much less valuable to the company.

A personal analysis on Bank2010's current UX would be beneficial for the developers, providing feedback from the perspective of new user without prior experience. This would help them understand the initial learning curve or questions that a new user may have but not speak about as they are new to the job and not willing to risk a bad first impression.

For the analysis, the current UX was criticized and given feedback Questions, that the developers should keep in mind whenever developing the new web client, were also proposed. This analysis, in addition to the data gathered from the survey, interviews, and observations should provide the basis of their new UX design.

5. Results

According to the data gathered from the survey, 63.6% did not have any prior experience with banking software. And even though 95.5% received training, 72.7% reported to have been confused when using the client. 63.7% reported that Bank 2010 was easy to navigate but only 13.6% believed it looked modern. This indicates that the current layout can be taught and is acceptable but not perfect. The lack of a modern visual may alienate some users as they navigate through the client. A more modern client might increase the ease of navigation.

When asked whether Bank 2010 had the necessary features to help customers, there was a split decision with the same 36.4% in agreement and disagreement. This is a major issue that needs to be addressed. Having the tools necessary to help customers is crucial to users as it improves customer satisfaction, a priority for any business. And while there is a split on whether Bank 2010 has all the necessary features to help customers, 81.8% agreed that it needed an update. This gives executives and managers proof that an updated client should be developed.

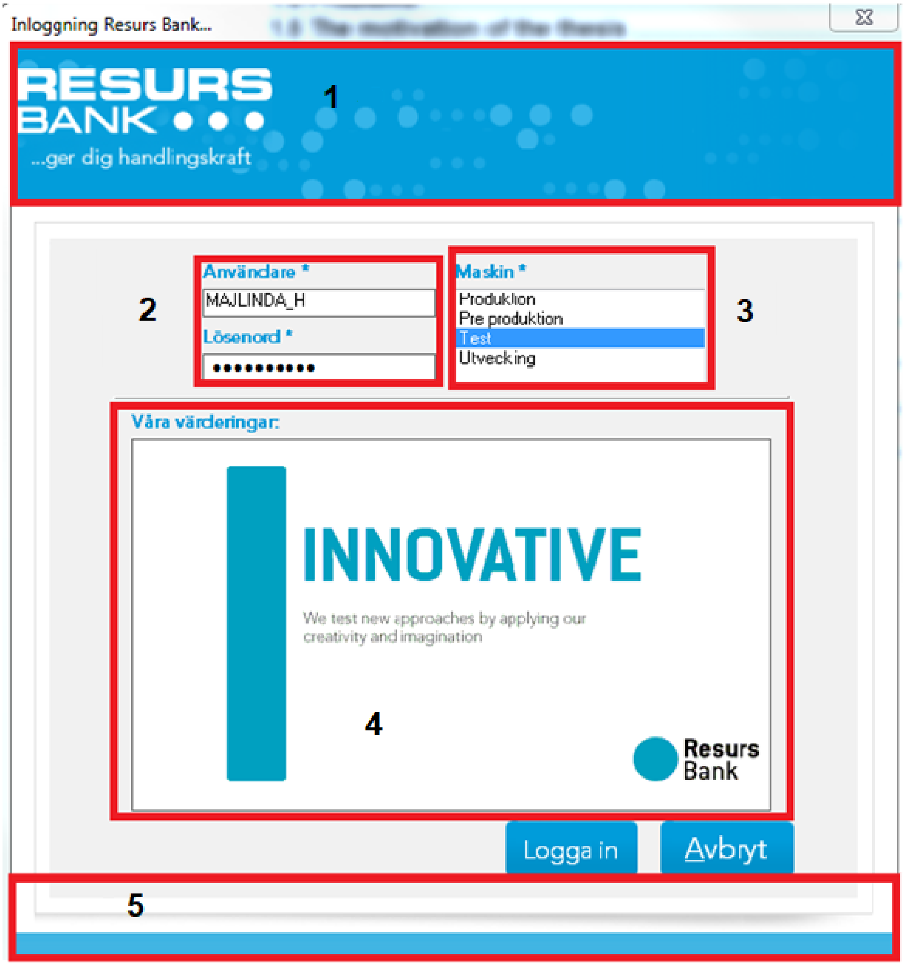
The tools that were used most often, according to the survey, were the “Fortress”, “Historik”, “Transar”, and “Kundinfo” functions. The least common were not named directly, but simply alluded to as functions with niche uses. The reason this is important information is so that the new web client can be built upon the knowledge of what is important to the users. This information can help developers understand what is currently working well with their current client, which they can then build it into the new web client.

One employee reported in one of the write-in questions that a lack of cohesiveness and consolidation as a problem they faced, stating “instead of having to visit one to three other sites and applications” in regards to handling credit applications. Another employee stated in the same question that an external calculator was required when calculating costs on loans and payments. Besides missing functionality, some reported functions that caused headaches and would be better off removed, such as shortcuts closing the entire web client.

5.1. Client Examples

The current client has an old style that has UI elements that do not enhance the UX. The resulting recommendations and criticisms are based on UX principles. Although UX is naturally subjective, all feedback is still important and should be considered. When given client examples, some of the feedback are posed as questions to encourage the developers to think about their choices.

Figures 7 - 18 are window captures of the Bank2010 client. It is the client that will be replaced by the new web client. The window captures contain feedback and criticisms in the form of statements and questions, found below each figure. The purpose of the statements is to give the developers concrete suggestions that they can implement. The purpose of the questions is to remind developers about their goal when designing their UI.



Inloggning Resurs Bank...
RESURS BANK
...ger dig handlingskraft
1
2
3
Maskin *
Produktion
Pre produktion
Test
Utveckling
Våra värderingar:
INNOVATIVE
We test new approaches by applying our creativity and imagination
4
Resurs Bank
Logga in
Avbryt
5

Användare *: MAJLINDA_H

Lösenord *: ••••••••••

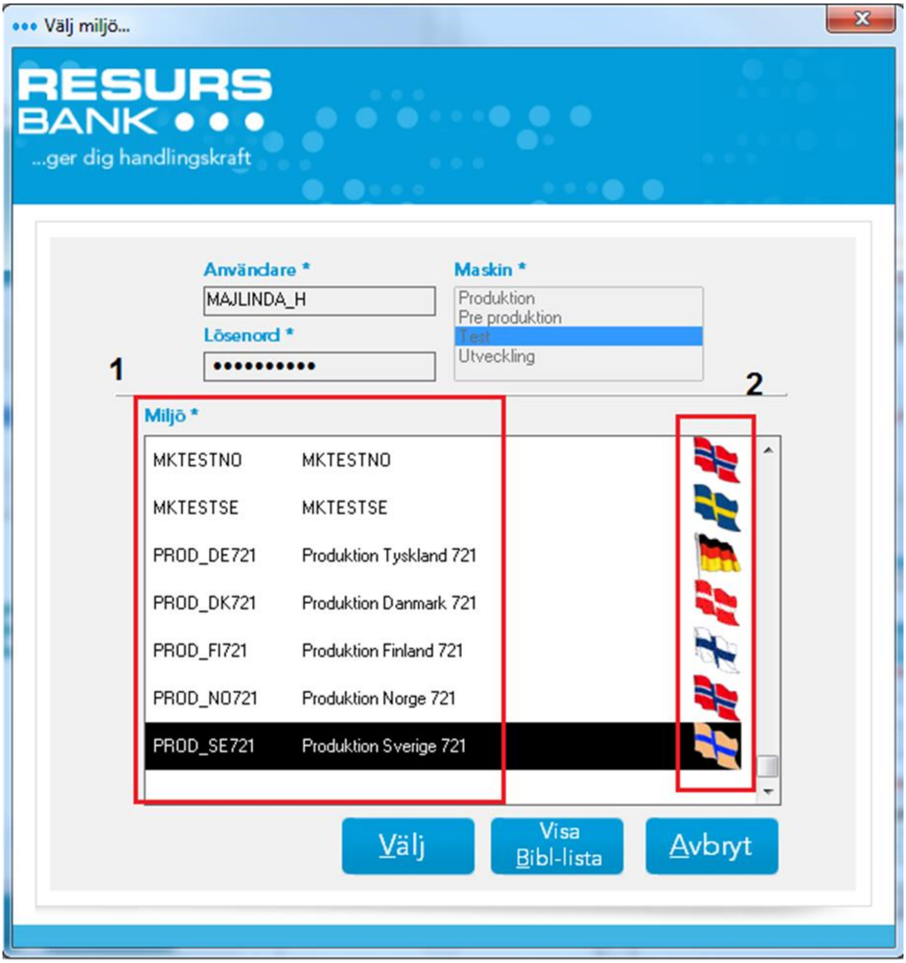
Maskin *: Test

Figure 7: Bank2010 log-in screen

The following criticisms refer to Figure 7.

1. The banner is taking up too much space without any good reason.
2. The fields for logging in could be less wide and be placed side-by-side to save space.
3. A window selection is a bad choice for a list. It limits future expansion and any unused space becomes unusable. Using a dropdown menu would allow future expansion without taking up extra space and not always take the same amount of space as long as one selection is available.

4. Considering this is not a public product or service, having a logo or banner that takes up space in a windowed environment makes little sense. Branding is more important in a product or service that is open to the public.
5. The cascading layered lines are an old style that fails to add anything useful to the user experience.



Användare *
MAJLINDA_H

Lösenord *
•••••••••••

Maskin *
Produktion
Pre produktion
Test
Utveckling

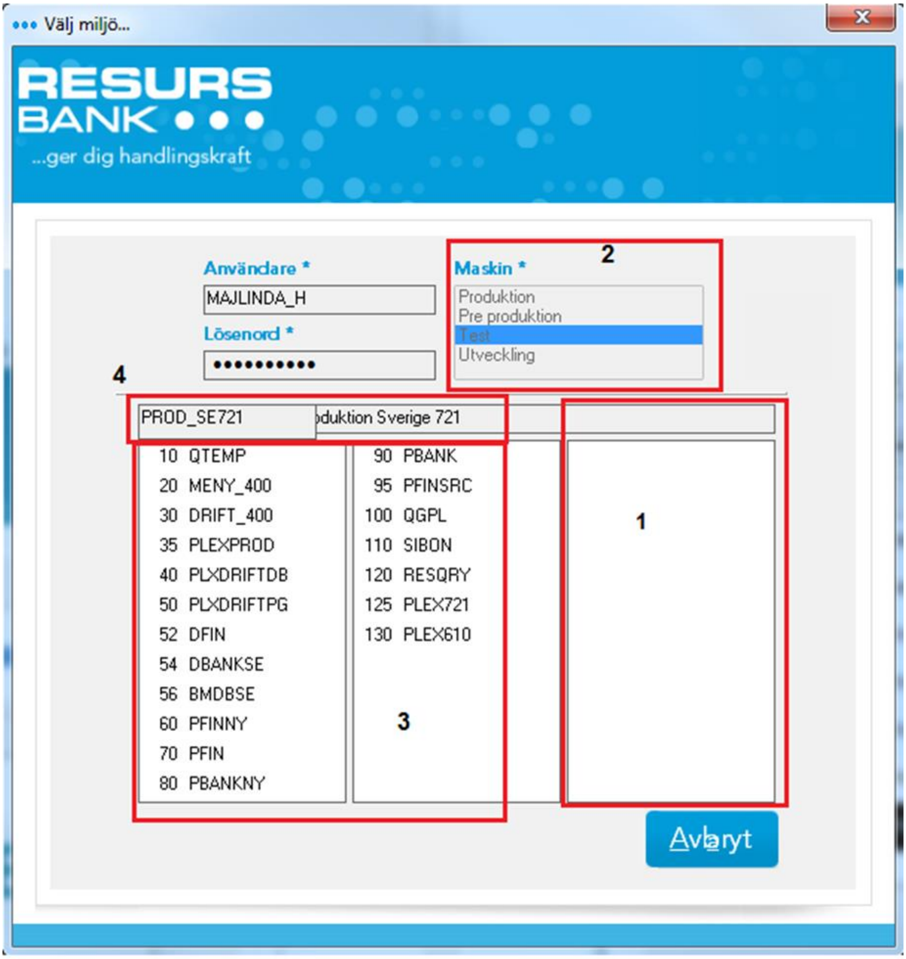
Miljö *

MKTESTNO	MKTESTNO
MKTESTSE	MKTESTSE
PROD_DE721	Produktion Tyskland 721
PROD_DK721	Produktion Danmark 721
PROD_FI721	Produktion Finland 721
PROD_NO721	Produktion Norge 721
PROD_SE721	Produktion Sverige 721

Figure 8: Server selection

The following criticisms refer to Figure 8.

1. This list can be better organized so the user can more easily search the content rather than using a long scroll bar. Arranging each section so that the user fills out one at each step to get to their end goal would be much quicker and easier.
2. The flag icons are outdated. Something simpler or cleaner that removes unnecessary details and emphasizes the important pattern or design would be preferred.



Användare *
MAJLINDA_H

Lösenord *
••••••••••

Maskin *
Produktion
Pre produktion
Test
Utveckling

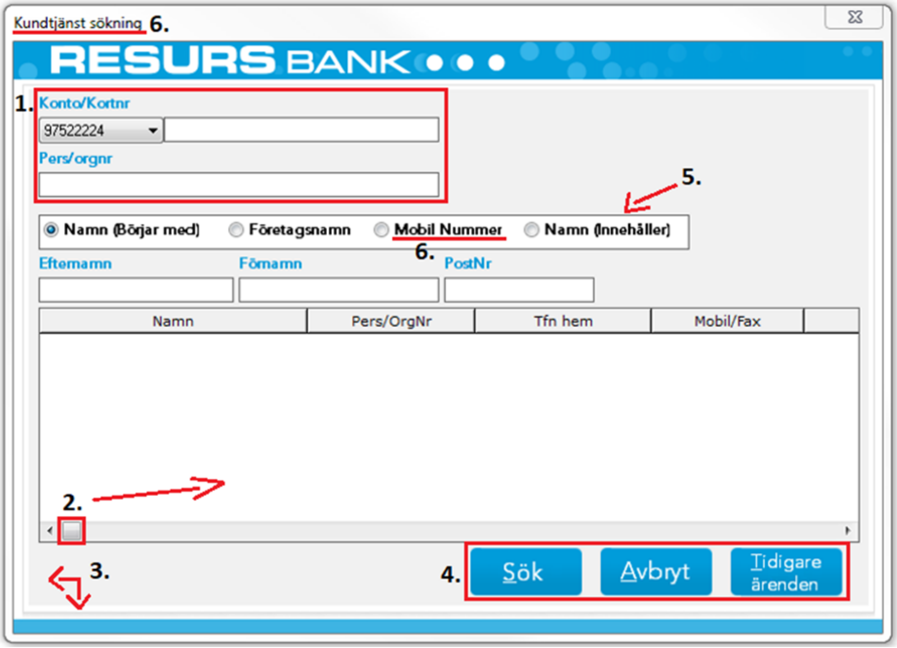
PROD_SE721 | duktion Sverige 721

10 QTEMP
20 MENY_400
30 DRIFT_400
35 PLEXPROD
40 PLXDRIFTDB
50 PLXDRIFTPG
52 DFIN
54 DBANKSE
56 BMDBSE
60 PFINNY
70 PFIN
80 PBANKNY
90 PBANK
95 PFINSRC
100 QGPL
110 SIBON
120 RESQRY
125 PLEX721
130 PLEX610

Figure 9: Library list

The following criticisms refer to Figure 9.

1. This empty unused space has no purpose or function.
2. The color choice, while thematically consistent, does not read well. This can lead to confusion.
3. This list is segmented into 2 different sections, even though they are from the same list. Creating a dropdown menu would help with any expandability issues.
4. This section does not specify any information about the presented data, or why it is presented twice in different formats. Unless the user knew the purpose beforehand, the information is vague.



Kundtjänst sökning

Konto/Kortnr: 97522224

● Namn (Börjar med) ○ Företagsnamn ○ Mobil Nummer ○ Namn (Innehåller)

Namn	Pers/OrgNr	Tfn hem	Mobil/Fax

Figure 10: Search window

The following criticisms refer to Figure 10.

1. The fields seem much longer than necessary.
2. Scrollbar is not proportionate to window-size. A small scrollbar is hard to click on and even harder to control accurately. It can also be confusing to see inconsistent scrollbars on the bottom and right sides of the window. There is no real pattern that decides which windows or fields get a scroll bar.
3. Unnecessary lines in the design which makes it look cluttered. Figure 11 shows a better example of a clean design.

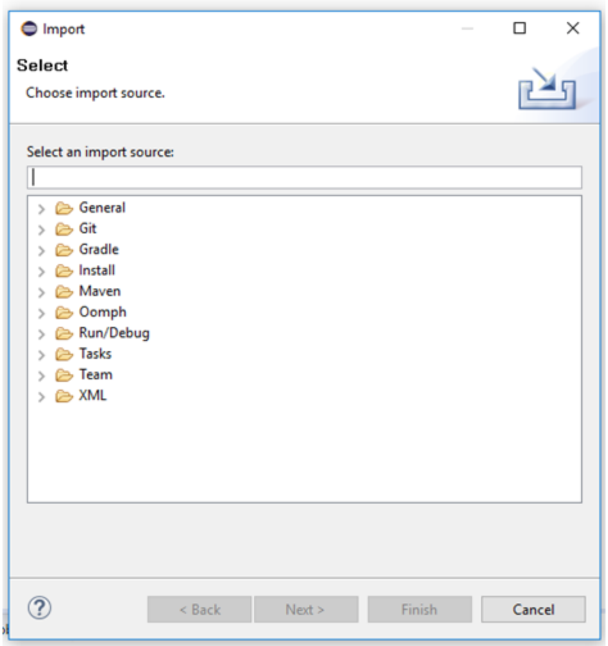
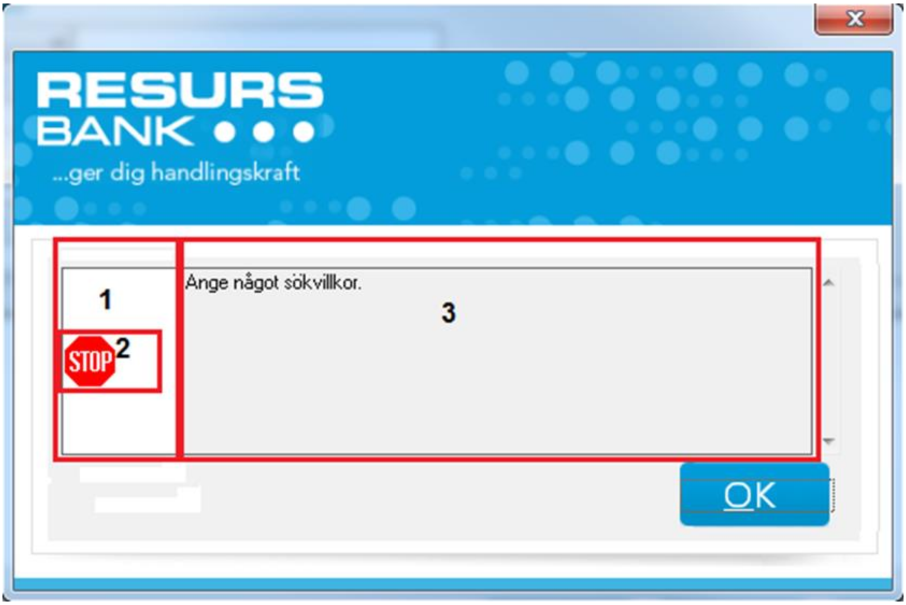


Figure 11: Example from Eclipse

4. Any option to cancel or stop any function should always be furthest to the end when presented in a group, unless the function enables a dangerous function or feature.
5. Is the bubble selection system the best solution here? How would it work if the options were expanded? A dropdown menu would continue to work with any expansion and would remain quickly legible if options remain limited.

6. Some of the word choices are odd, sounding unnatural or spelled incorrectly.



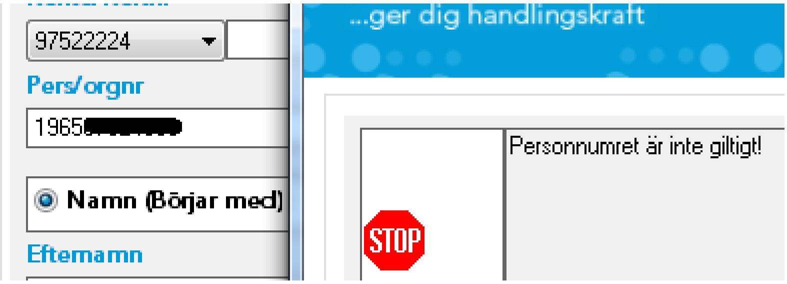
Ange något sökvillkor.

Figure 12: Error pop-up when attempting to search with an empty field

The following criticisms refer to Figure 12.

1. The white block is unnecessary. Even if it is supposed to signify an error, there is no need for it to be separated.
2. The icon is outdated and low quality. Better signs to signify errors exist and are much cleaner and modern.
3. The field is extremely massive for such a small error. Error reports are usually simplified by linking them to numbers that can be looked up. Presenting an in-depth log makes it harder for the user to understand, let alone troubleshoot. Users are not developers.

The whole window itself seems too big for an error prompt and has unnecessary elements such as a scroll bar.



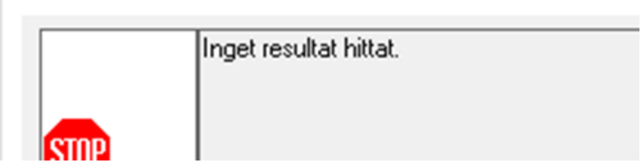
97522224

Pers/orgnr: 1965[redacted]

Personnumret är inte giltigt!

Figure 13: First error when searching for an invalid personal identity number

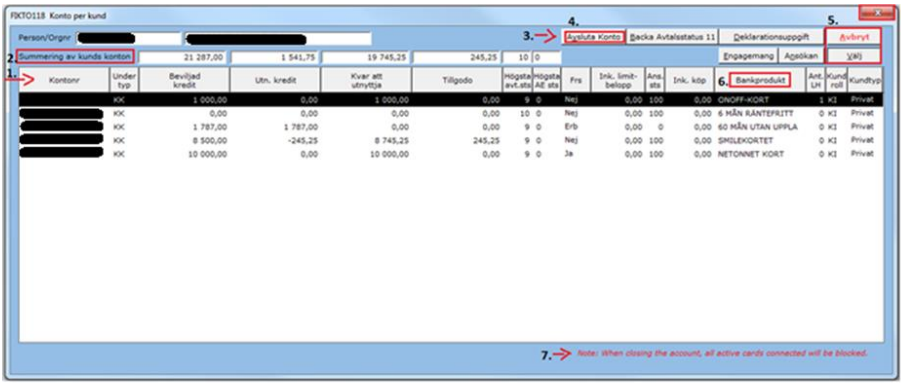
The personal identity number field in Figure 13 does not take non-numerical characters, which can cause confusion as a common method for entering personal identity numbers includes a hyphen. Submitting the complete four-digit year number into the field yields an error. Failed assumptions can lead to frustration.



Inget resultat hittat.

Figure 14: Second error when searching for an invalid personal identity number

Figure 14 is the error prompt that follows Figure 13. There is little use for a second error prompt for an invalid personal identity number when the first error already relays the important information.



Person/Orgnr [redacted] [redacted]

Summering av kunds konton 21 287,00 | 1 541,75 | 19 745,25 | 245,25 | 10 | 0

Kontonr	Under typ	Beviljad kredit	Utn. kredit	Kvar att utnyttja	Tillgodo	Högsta avt.sts	Högsta AE sts	Frs	Ink. limit-belopp	Ans. sts	Ink. köp	Bankprodukt	Ant. LH	Kund roll	Kundtyp
[redacted]	KK	1 000,00	0,00	1 000,00	0,00	9	0	Nej	0,00	100	0,00	ONOFF-KORT	1	KI	Privat
[redacted]	KK	0,00	0,00	0,00	0,00	10	0	Nej	0,00	100	0,00	6 MÅN RÄNTEFRITT	0	KI	Privat
[redacted]	KK	1 787,00	1 787,00	0,00	0,00	9	0	Erb	0,00	0	0,00	60 MÅN UTAN UPPLA	0	KI	Privat
[redacted]	KK	8 500,00	-245,25	8 745,25	245,25	9	0	Nej	0,00	100	0,00	SMILEKORTET	0	KI	Privat
[redacted]	KK	10 000,00	0,00	10 000,00	0,00	9	0	Ja	0,00	100	0,00	NETONNET KORT	0	KI	Privat

Note: When closing the account, all active cards connected will be blocked.

Figure 15: Customer account window

The following criticisms refer to Figure 15.

1. The amount of visible information seems overwhelming at first glance. Can these headers be abbreviated? Are all of them needed?
2. Would it feel more natural for the sum to be displayed below?
3. Not many drop down menus present for the buttons at the top right. Are all these needed? How often are they used? Do they warrant a visual cue?
4. Should the function for closing the account be the first selection? Functions that can cause issues should also stand out more as to not accidentally trigger it.
5. The select and cancel buttons could be easier to distinguish if placed at the bottom right of the window. The space is available, loosens up the clutter at the top, and gives the eye a reason to flow to the bottom of the window.
6. Similar elements should be near each other. From an unexperienced view, “Bank-Product” should be somewhere near account number.
7. The warning is hard to read and has no edge or feature that captures a user’s attention. The warning could be closer to the element it is warning the user about, or be retooled as a prompt after selecting the option.

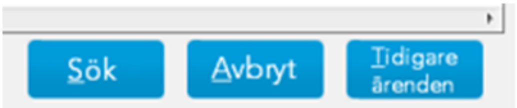


Figure 16: Alignment example 1

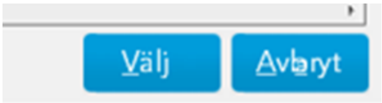
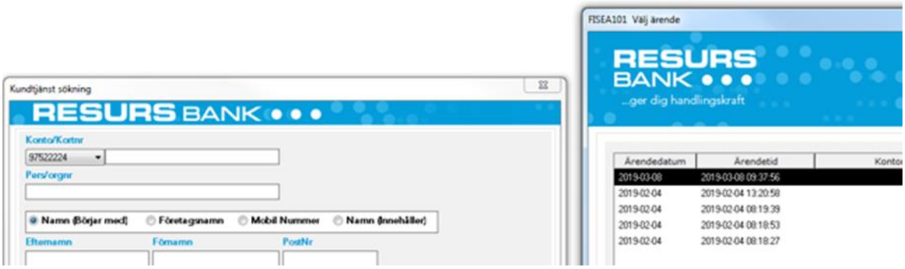


Figure 17: Alignment example 2

The buttons in Figure 16 and Figure 17 are not aligned to a grid or standard.



Konto/Kortnr: 97522224

Pers/orgnr:

● Namn (Börjar med) ○ Företagsnamn ○ Mobil Nummer ○ Namn (innehåller)

Efternamn / Förnamn / PostNr

Ärendedatum	Ärendetid	Konto
2019-03-08	2019-03-08 09:37:56	
2019-02-04	2019-02-04 13:20:58	
2019-02-04	2019-02-04 08:19:39	
2019-02-04	2019-02-04 08:18:53	
2019-02-04	2019-02-04 08:18:27	

Figure 18: Banner examples

The banners in each of the windows in Figure 18 are different sizes. The second window also has a dark highlight that fails to make the text easier to read.

5.2. Recommendations to Resurs Bank

The recommendations to Resurs Bank are:

1. Design the new web client with UX as a primary focus.
2. Use Bootstrap as the front-end framework.
3. Reiterate on the design through feedback and prototyping.

Resurs Bank current client, Bank2010, is relatively outdated. The lack of a cohesive, modern feel can make it feel foreign to new employees. Productivity can be improved if employees had a client built with UX in mind. Additionally, some of the functions necessary to provide customer support are not available. Ease of use should be one of the goals for the new web client. An emphasis on UX in the new web client would help solve the issues regarding confusion and ease of use.

When designing UI with UX in mind, it is important to understand that there is a fine balance between function and form. Too much functionality can cause overstimulation of information and intimidate new users. Too little functionality and employees do not have all the necessary tools to help their customers. In terms of visual appeal, a modern and cohesive look would help new employees bridge the gap as all of them, according to the survey, were familiar with browsing the Web. As most of the front-end frameworks are used in everyday products and services on the Web, new users would feel much more comfortable jumping into a client that seemed familiar.

It is recommended that Resurs Bank create more surveys and conduct more interviews as they develop new layouts for their new web client. Additionally, take the criticisms from their current client and improve upon them. Through iterations, they can develop a much better UX that can increase productivity, longevity, and is easier to use.

5.2.1. Framework of Choice

When taking into account the needs of Resurs Bank, it is important to understand how the data gathered from employees shaped the recommendation for the front-end framework. Considering the employee feedback, the best choice would be Bootstrap with some possible React libraries adopted for any further functionality. The main reason behind recommending this particular front-end framework is due to simplicity and popularity, as well as performance and maintenance ability. If the features are complete and the design is modern and easy to understand, the emphasis can then be placed on the user experience in order to increase workplace efficiency.

A bank does not have much need for performance over functionality. Considering this, Bootstrap provides a variety of functionality that can be useful in a complex application such as a banking UI. It needs to be feature proof and have the ability to maintain the current functionality. Additionally, it should be modular enough that any future functions that may need to be added should not create too much of a headache.

This web client that is being built by Resurs Bank is closed to the public. Its users will be primarily employees. The look of the web client does not need to compete with other banks. The product is not the web client; the web client is a tool to better provide the service. This is why the simple, premade styles available with Bootstrap can create a cohesive theme without much effort. And since Bootstrap is updated quite regularly, this usually means that the modern trends will most assuredly become a staple in Bootstrap.

6. Conclusions

To improve upon the UX of Bank2010, the new web client should aim to take the feedback found in the survey and iterate accordingly. This thesis only conducted one round of data gathering. Resurs Bank needs to create more surveys and conduct more observations whenever they make an iteration on the UI.

Communication is key in creating a UX that improves productivity and reduces any issues that users may encounter. If users fail to communicate their issues with the current UX, the UX will be difficult to improve in the correct direction. If the developers fail to explain their direction as they iterate, the UX can change in a direction that fails to solve any of the UX issues.

6.1. Goal Outcomes

The recommendation and data gathered during the thesis answered the questions presented in section 1.4.

1. What can be done to give Bank2010 a more modern look and would it be beneficial?

Criticisms were made on the current client’s visual appeal and functionality. Areas were highlighted and commented on, based on UX research. Additionally, the survey asked whether the current client looked modern and the data overwhelmingly agreed it was not. A modern look and feel similar to those seen on the web and in applications makes it seem more familiar to inexperienced users, which increases productivity and makes training new employees easier.

2. What do the users like and dislike about Bank2010?

The survey encouraged employees to reveal some insight about their likes and dislikes of Bank2010. Most of the dissatisfaction comes from lack of functionality or confusion from the current layout. Almost all of the employees have reported being confused during their use of Bank2010, and made comments wondering why certain functions are divided into different windows and buttons. It wasn’t all negative, however, as employees reported that they used the shortcuts often and rarely had issues that required help from others.

3. Which tools or functions are used most and least often?

The tools that were used most often were the “Fortress”, “Historik”, “Transar”, and “Kundinfo” functions. The least common were not named directly, but simply alluded to as functions with niche uses as a general statement. Further surveys or interviews could be conducted to better identify the rarely used functions.

4. Which front-end framework would work best with the new web client, and is it future proof?

The recommendation was to use Bootstrap as the front-end framework for the new web client. This is due to the constant updates, ease of use, and visual cohesiveness. The visual appeal needs uniformity and a touch of modernity rather than expression and uniqueness. The constant updates keep the client up-to-date and the ease of use means maintenance will be easier. It is also full of features that can help with any functions that Resurs Bank may need. Although a bit heavier than other frameworks, the benefits outweigh the negatives in this situation.

6.2. Discussion

The result of the survey seemed to match the assumptions from the observations and interviews previously conducted. There seems to be a frustration present in the customer service department that is finally being acknowledged. There is no doubt that the developers will aim to create a future-proof client that is easier to learn and provides the necessary tools to best serve customers.

Resurs Bank’s current client looks outdated and overwhelming for someone that has never used banking software. Even though the initial impression was not that great, it was good to see that employees were taught on how to use the functions. This shows that the new web client has a higher chance of being an improvement over the current client.

Most modern frameworks will be sufficient to develop their new web client but choosing Bootstrap or React will yield better chances of maintenance as their popularity helps during the hiring process.

6.3. Ethical Reflections

The current political landscape has placed some protections in place in regards to content and data gathering. Considering this is not a public product, the only data gathered is data that is willingly given in exchange for service at the bank. They mainly must worry about keeping the database safe from any outside intruders. The client itself does not store data but instead gives the users a bridge to access the data stored in a database. If Resurs Bank adheres to the GDPR regulations, specifically the one that grants the customer the right to their unique data, there should not be any issues with their client.

6.4. Future Development Opportunities

If the extra effort and time spent on the UX proves to be a beneficial trait during the development phase, it could provide incentive for other areas that may be suffering from a bad UX. Mobile applications are becoming more popular with banks and the experience developing the UX for Resurs Bank new web client could lead to a better mobile application.

Resurs Bank does not have any necessary duty to implement the recommendations given to them, as they are merely suggestions. However, they have stated their interest when the first interviews were conducted to narrow down a thesis topic. As they are investing into a PoC, it would be in their best interest to make the new web client as good as it can be. A better client will increase their productivity, which will increase their employee and customer satisfaction. Whether or not their investment is worth the cost is strictly a decision to be made by Resurs Bank.

7. Terminology

This terminology contains definitions for terms that are used often or are important for understanding this thesis.

Term	Definition
Angular	*Open-source front-end framework built with TypeScript, developed and maintained by a Google-team* [30].
Bank2010	*Resurs Bank's current banking client, used to manage customers, accounts, and financial products.*
Bootstrap	*Open-source front-end framework created by Mark Otto and Jacob Thornton at Twitter* [31].
Foundation	*Front-end framework that uses HTML and CSS components to create UI elements, developed by ZURB* [32].
Front-End Framework	*A collection of HTML, CSS, and JavaScript containing design patterns that can be built upon, with an emphasis on reusability and structure* [32].
JavaScript (JS)	*A high-level programming language that has full integration with HTML and CSS. JS is the most widely used browser interface creation tool* [33].
Model-Viewer-Controller (MVC)	*A famous design pattern used when creating interactive software, its key concept is to part the user interface with the underlying data.* [34].
New Web Client	*Resurs Bank's future web-based banking client.*
Proof of Concept (PoC)	*A proof of concept (POC) is a demonstration aimed at verifying that certain concepts or theories have the potential to be applied in the real world* [35].

React	*JavaScript library originally developed by Jordan Walke at Facebook to create reactive user interfaces* [36].
TypeScript	*A superset programming language developed by Microsoft based on JavaScript. It can trans compile directly into JavaScript without any dependencies* [37].
User Experience (UX)	*User experience is defined as the overall experience of a person using a product such as a website or computer application, especially in terms of how easy or pleasing it is to use* [1][2].
User Interface (UI)	*The space of interaction between a machine, device, application, or tool and user with an emphasis on control and operation* [38].
VueJS	*An incremental front-end framework developed by Evan You written in JavaScript* [39].

8. Bibliography

This section lists all the sources that have been used in this thesis.

[1] Tokkonen, H. and Saariluoma, P. (2013). How User Experience is Understood?. 2013 Science and Information Conference, pp.791-795.

[2] Kiruthika, J., Khaddaj, S., Greenhill, D. and Francik, J. (2016). User Experience Design in Web Applications. 2016 IEEE Intl Conference on Computational Science and Engineering (CSE) and IEEE Intl Conference on Embedded and Ubiquitous Computing (EUC) and 15th Intl Symposium on Distributed Computing and Applications for Business Engineering (DCABES), pp.642-646.

[3] Hunt, P. (2013). Why did we build React? – React Blog. [online] Reactjs.org. Available at: https://reactjs.org/blog/2013/06/05/why-react.html [Accessed 3 Apr. 2019].

[4] Vorbach, P. (n.d.). npm-stat: download statistics for NPM packages. [online] Npm-stat.com. Available at: https://npm-stat.com/charts.html?package=angular&package=react&package=vue&from=2017-08-01&to=2019-05-21 [Accessed 21 May 2019].

[5] Warcholinski, M. (n.d.). 10 Famous Apps Using ReactJS Nowadays. [online] Blog Brainhub.eu. Available at: https://brainhub.eu/blog/10-famous-apps-using-reactjs-nowadays/ [Accessed 1 Apr. 2019].

[6] GitHub. (n.d.). facebook/react-native. [online] Available at: https://github.com/facebook/react-native [Accessed 1 Apr. 2019].

[7] Facebook.github.io. (n.d.). React Native · A framework for building native apps using React. [online] Available at: https://facebook.github.io/react-native/showcase [Accessed 1 Apr. 2019].

[8] M, M. (2018). AngularJS and Angular 2+: a Detailed Comparison — SitePoint. [online] SitePoint. Available at: https://www.sitepoint.com/angularjs-vs-angular/ [Accessed 4 Apr. 2019].

[9] Krill, P. (2013). What's so special about Google's AngularJS. [online] InfoWorld. Available at: https://www.infoworld.com/article/2612801/whats-so-special-about-googles-angularjs.html [Accessed 5 Apr. 2019].

[10] Hamilton, C. (2014). Angular 2.0 announcement backfires - JAXenter. [online] JAXenter. Available at: https://jaxenter.com/angular-2-0-announcement-backfires-112127.html [Accessed 4 Apr. 2019].

[11] Made with Angular. (n.d.). Home. [online] Available at: https://www.madewithangular.com/ [Accessed 6 Apr. 2019].

[12] Static1.squarespace.com. (n.d.). [online] Available at: https://static1.squarespace.com/static/58d20c79725e25b221549193/t/5acdf0d303ce64a75c5fd489/1523445977902/js-github.png?format=750w [Accessed 14 Apr. 2019].

[13] Rishabh Software. (2019). Difference Between AngularJS and ReactJS: A Comparison Guide. [online] Available at: https://www.rishabhsoft.com/blog/reactjs-vs-angularjs [Accessed 21 May 2019].

[14] Cromwell, V. (2016). Evan You. [online] Between the Wires. Available at: https://web.archive.org/web/20170603052649/https:/betweenthewires.org/2016/11/03/evan-you/ [Accessed 14 Apr. 2019].

[15] You, E. (2014). First Week of Launching Vue.js. [online] Evan You. Available at: https://blog.evanyou.me/2014/02/11/first-week-of-launching-an-oss-project/ [Accessed 14 Apr. 2019].

[16] Star-history.t9t.io. (2019). Star history. [online] Available at: https://star-history.t9t.io/#facebook/react&vuejs/vue [Accessed 21 May 2019].

[17] Sajnóg, M. (2018). 13 Top Companies That Have Trusted Vue.js – Examples of Applications. [online] Netguru.com. Available at: https://www.netguru.com/blog/13-top-companies-that-have-trusted-vue.js-examples-of-applications [Accessed 14 Apr. 2019].

[18] Sidorenko, I. (2019). What Are The Pros And Cons Of Using Vue.js. [online] Towards Data Science. Available at: https://towardsdatascience.com/what-are-the-pros-and-cons-of-using-vue-js-3689d00d87b0 [Accessed 14 Apr. 2019].

[19] Otto, M. (2011). Bootstrap from Twitter. [online] Blog.twitter.com. Available at: https://blog.twitter.com/developer/en_us/a/2011/bootstrap-twitter.html [Accessed 12 Apr. 2019].

[20] Burge, S. (2018). How Popular is Bootstrap in 2018?. [online] Ostraining.com. Available at:

https://www.ostraining.com/blog/webdesign/bootstrap-popular/ [Accessed 13 Apr. 2019].

[21] Puzak, D. (2017). Why we stopped using Bootstrap. [online] Borealis.agency. Available at: https://borealis.agency/stopped-using-bootstrap/255 [Accessed 12 Apr. 2019].

[22] Zingdesign.com. (2013). 5 reasons NOT to use Twitter Bootstrap | Zing Design. [online] Available at: http://www.zingdesign.com/5-reasons-not-to-use-twitter-bootstrap/ [Accessed 12 Apr. 2019].

[23] Zurb.com. (n.d.). Start Here, Build Everywhere: Announcing Foundation. [online] Available at: https://zurb.com/blog/start-here-build-everywhere-announcing-fo [Accessed 16 Apr. 2019].

[24] Foundation.zurb.com. (n.d.). Rapid Prototyping | Foundation. [online] Available at: https://foundation.zurb.com/prototyping.html [Accessed 16 Apr. 2019].

[25] Anon, (2019). [online] Available at: https://trends.google.com/trends/explore [Accessed 13 Apr. 2019].

[26] Zurb.com. (n.d.). Foundation | Showcase. [online] Available at: https://zurb.com/responsive [Accessed 16 Apr. 2019].

[27] Krug, S. (2014). Don't Make Me Think, Revisted. 3rd ed. Pearson Education.

[28] Ok Jong, Y. and Kwan Jung, C. (2015). The Development of Interview Techniques in Language Studies: Facilitating the Researchers' Views on Interactive Encounters. English Language Teaching, 8(7), p.31

[29] Trost, J. and Hultåker, O. (2016). Enkätboken. 5th ed. Lund: Studentlitteratur.

[30] Angular.io. (n.d.). Angular. [online] Available at: https://angular.io/about?group=Angular [Accessed 20 Apr. 2019].

[31] Getbootstrap.com. (n.d.). About · Bootstrap. [online] Available at: https://getbootstrap.com/docs/3.4/about/ [Accessed 20 Apr. 2019].

[32] Foundation.zurb.com. (n.d.). About Foundation | Foundation. [online] Available at: https://foundation.zurb.com/showcase/about.html [Accessed 25 Apr. 2019].

[33] Javascript.info. (n.d.). An Introduction to JavaScript. [online] Available at: https://javascript.info/intro [Accessed 12 May 2019].

[34] Leff, A. and Rayfield, J. (2001). Web-application development using the Model/View/Controller design pattern. Proceedings Fifth IEEE International Enterprise Distributed Object Computing Conference, pp.118-127.

[35] Techopedia.com. (n.d.). What is a Proof of Concept (POC)? - Definition from Techopedia. [online] Available at: https://www.techopedia.com/definition/4066/proof-of-concept-poc [Accessed 4 Apr. 2019].

[36] GitHub. (n.d.). facebook/react. [online] Available at: https://github.com/facebook/react/ [Accessed 25 Apr. 2019].

[37] GitHub. (n.d.). microsoft/TypeScript. [online] Available at: https://github.com/Microsoft/TypeScript [Accessed 12 May 2019].

[38] Techterms.com. (2009). User Interface Definition. [online] Available at: https://techterms.com/definition/user_interface# [Accessed 12 May 2019].

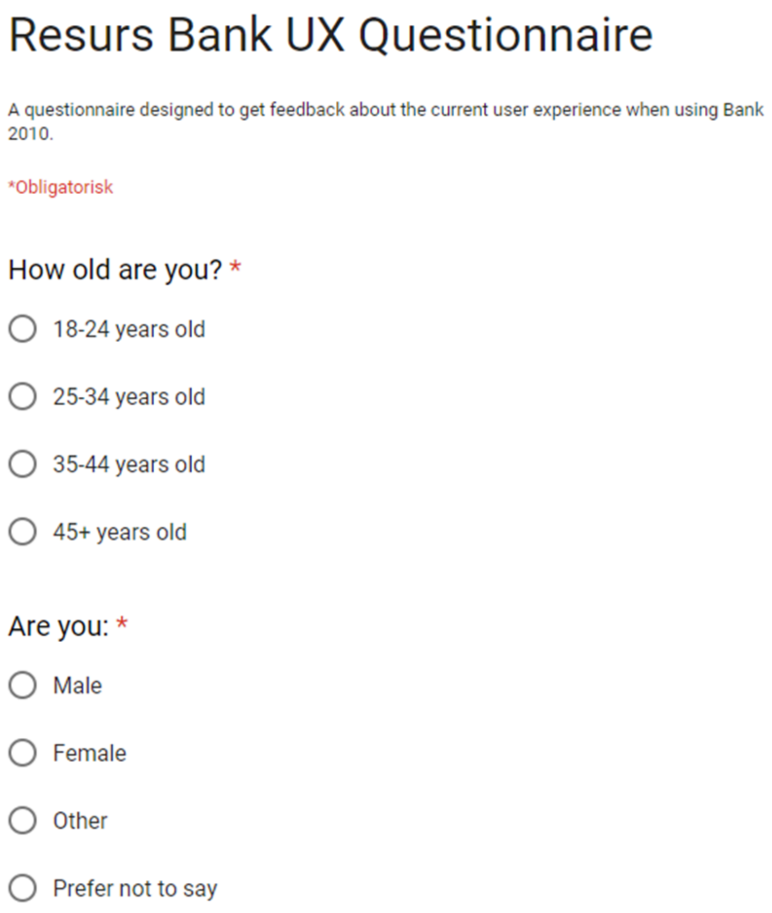
[39] Vuejs.org. (n.d.). Introduction — Vue.js. [online] Available at: https://vuejs.org/v2/guide/ [Accessed 20 Apr. 2019].

Appendices

This section contains the appendices mentioned in this thesis.

Appendix I

This appendix contains the survey that was used to gather data anonymously for the final recommendation.



Resurs Bank UX Questionnaire

A questionnaire designed to get feedback about the current user experience when using Bank 2010.

*Obligatorisk

How old are you? *

○ 18-24 years old

○ 25-34 years old

○ 35-44 years old

○ 45+ years old

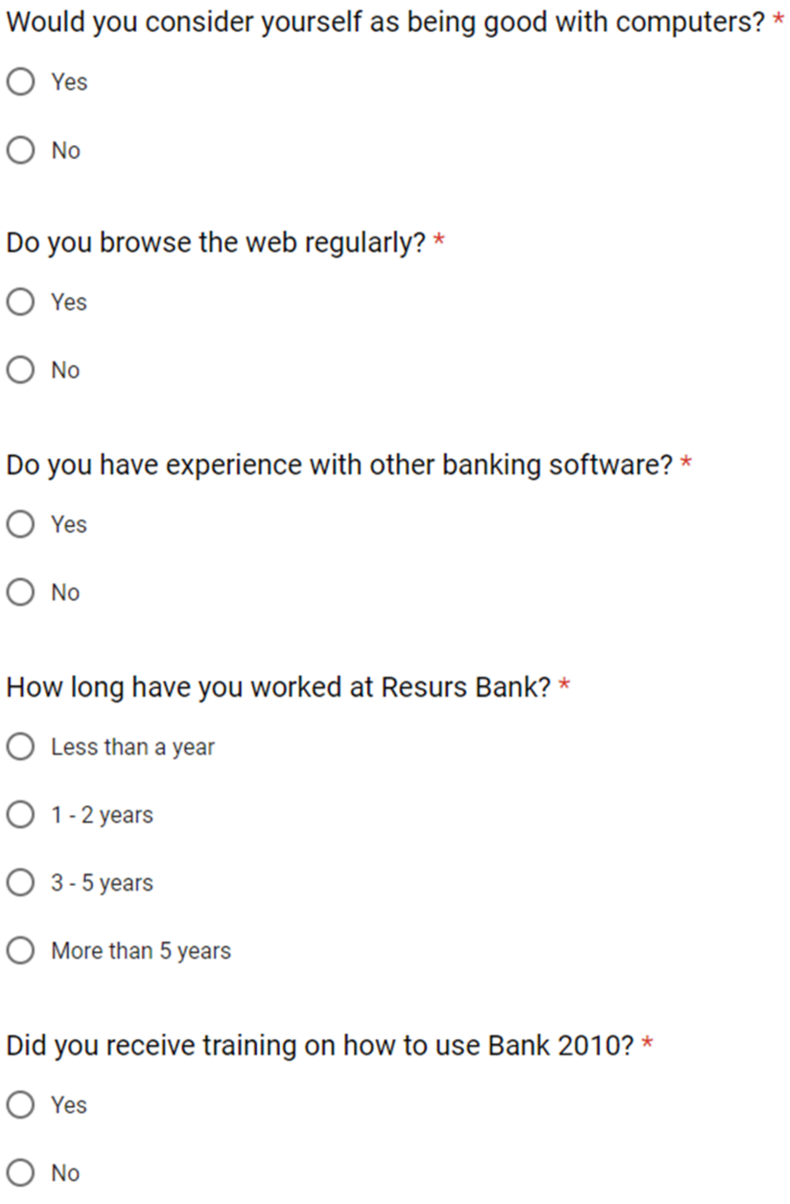
Are you: *

○ Male

○ Female

○ Other

○ Prefer not to say



Would you consider yourself as being good with computers? *

○ Yes

○ No

Do you browse the web regularly? *

○ Yes

○ No

Do you have experience with other banking software? *

○ Yes

○ No

How long have you worked at Resurs Bank? *

○ Less than a year

○ 1 - 2 years

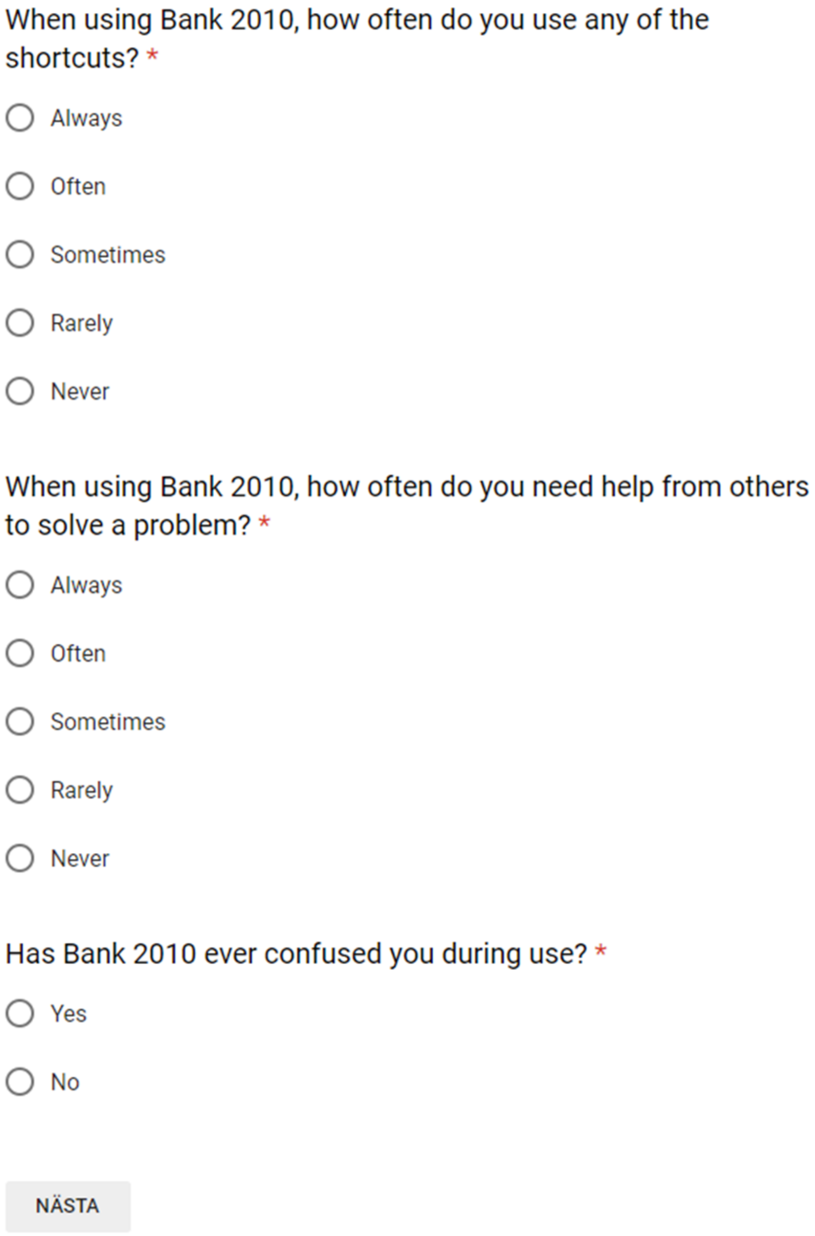
○ 3 - 5 years

○ More than 5 years

Did you receive training on how to use Bank 2010? *

○ Yes

○ No



When using Bank 2010, how often do you use any of the shortcuts? *

○ Always

○ Often

○ Sometimes

○ Rarely

○ Never

When using Bank 2010, how often do you need help from others to solve a problem? *

○ Always

○ Often

○ Sometimes

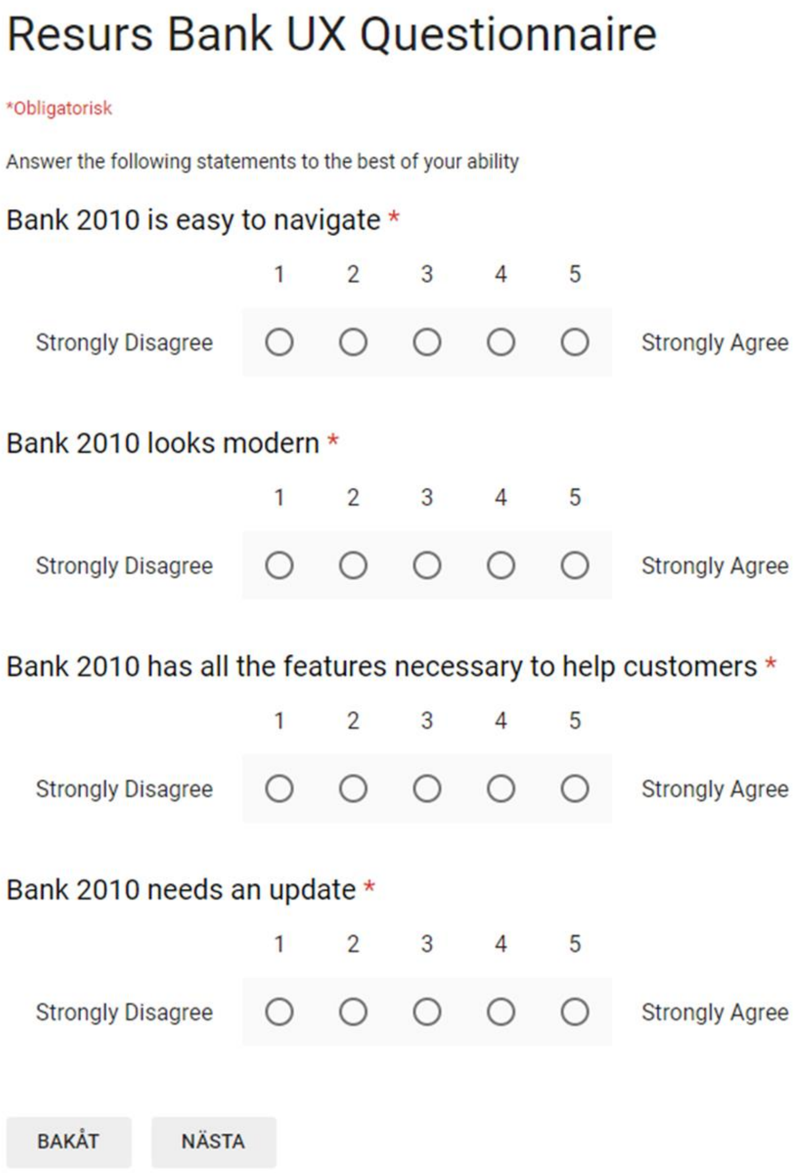
○ Rarely

○ Never

Has Bank 2010 ever confused you during use? *

○ Yes

○ No



Resurs Bank UX Questionnaire

*Obligatorisk

Answer the following statements to the best of your ability

Bank 2010 is easy to navigate *

	1	2	3	4	5	
Strongly Disagree	○	○	○	○	○	Strongly Agree

Bank 2010 looks modern *

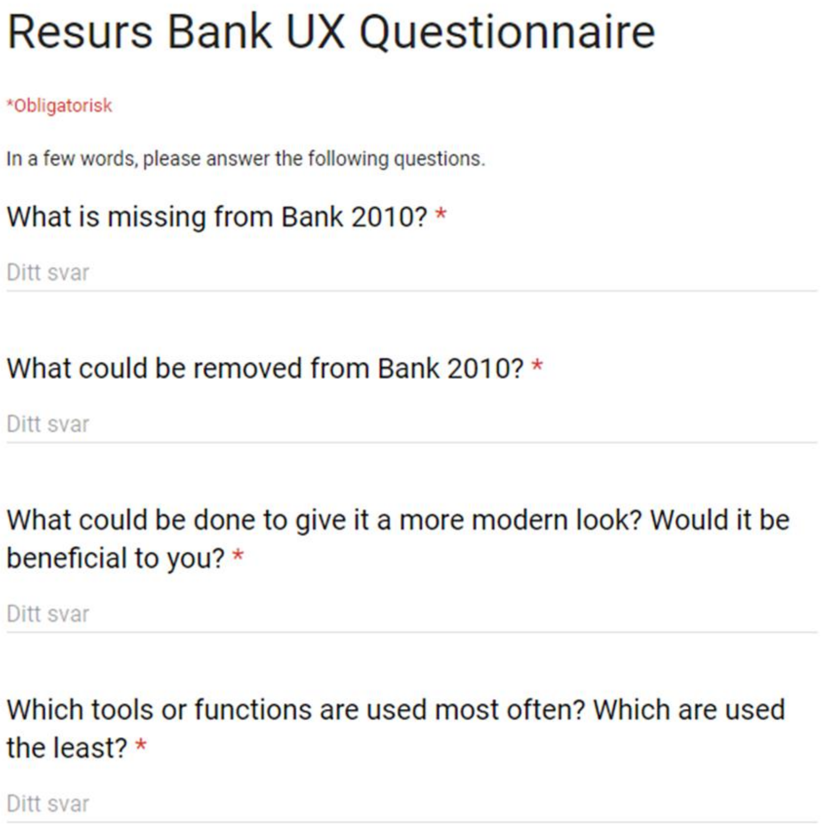
	1	2	3	4	5	
Strongly Disagree	○	○	○	○	○	Strongly Agree

Bank 2010 has all the features necessary to help customers *

	1	2	3	4	5	
Strongly Disagree	○	○	○	○	○	Strongly Agree

Bank 2010 needs an update *

	1	2	3	4	5	
Strongly Disagree	○	○	○	○	○	Strongly Agree



Resurs Bank UX Questionnaire

*Obligatorisk

In a few words, please answer the following questions.

What is missing from Bank 2010? *

Ditt svar

What could be removed from Bank 2010? *

Ditt svar

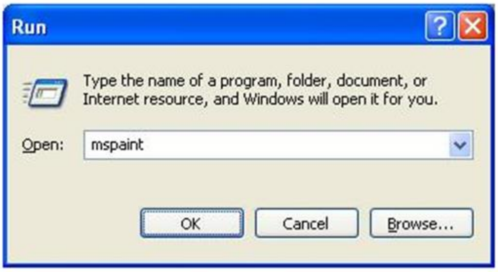
What could be done to give it a more modern look? Would it be beneficial to you? *

Ditt svar

Which tools or functions are used most often? Which are used the least? *

Ditt svar

Choice 1

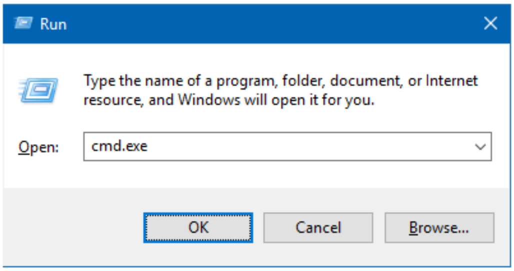


Run

Type the name of a program, folder, document, or Internet resource, and Windows will open it for you.

Open: mspaint

Choice 2



Type the name of a program, folder, document, or Internet resource, and Windows will open it for you.

Open: cmd.exe

Given the above choices, which one do you prefer? Why? *

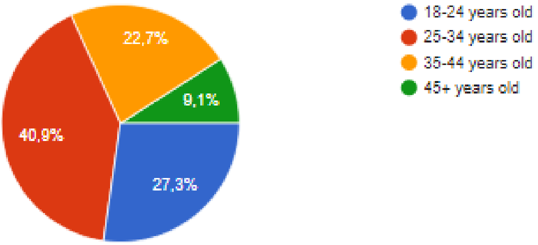
Ditt svar

BAKÅT SKICKA

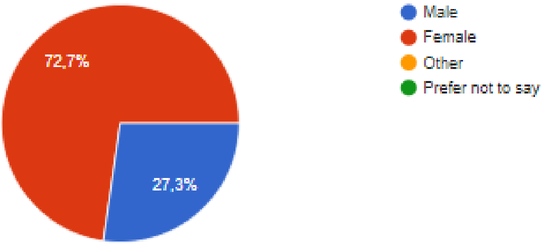
Appendix II

This appendix contains charts and graphs representing the answers from the questions asked in the survey. Questions that required unique written answers were not published as to protect anonymity.

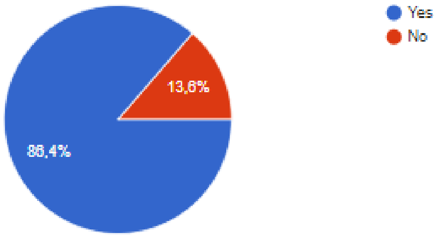
How old are you?



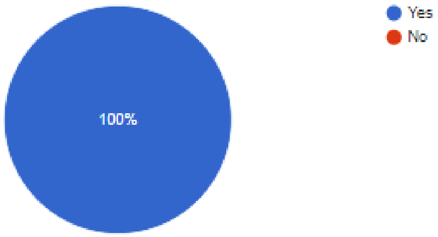
Are you:



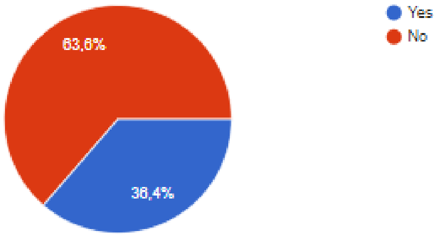
Would you consider yourself as being good with computers?



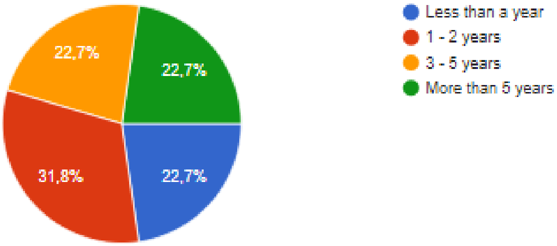
Do you browse the web regularly?



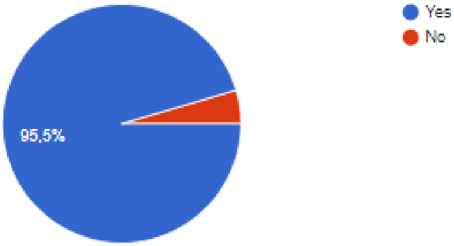
Do you have experience with other banking software?



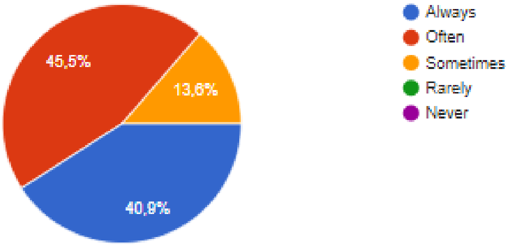
How long have you worked at Resurs Bank?



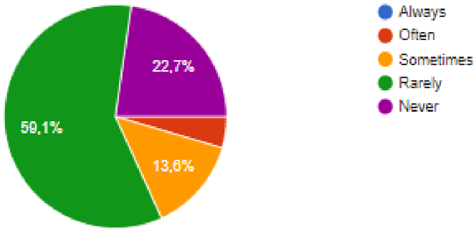
Did you receive training on how to use Bank 2010?



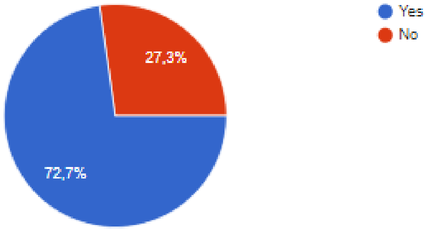
When using Bank 2010, how often do you use any of the shortcuts?



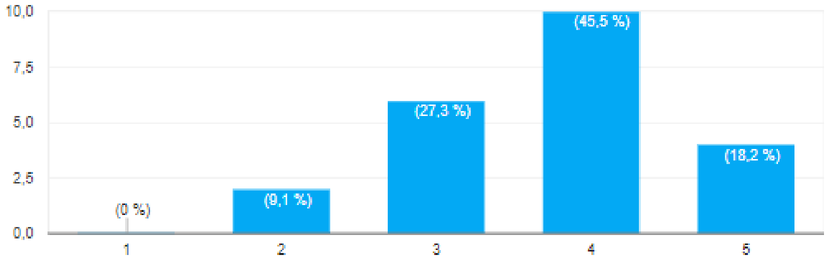
When using Bank 2010, how often do you need help from others to solve a problem?



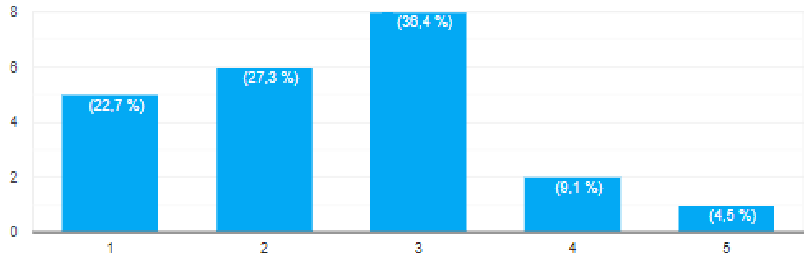
Has Bank 2010 ever confused you during use?



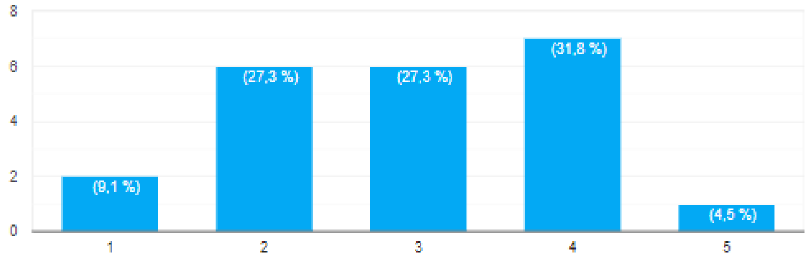
Bank 2010 is easy to navigate



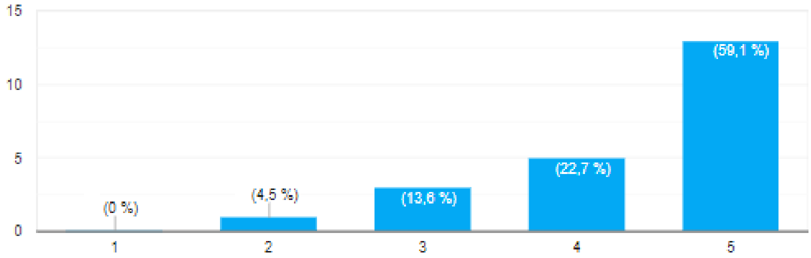
Bank 2010 looks modern



Bank 2010 has all the features necessary to help customers



Bank 2010 needs an update



15
10
5
0
(0 %)
(4,5 %)
(13,6 %)
(22,7 %)
(59,1 %)
1
2
3
4
5