Application Development from Prototype to Beta

A Case Study of the Application SeafarerCV

By

RONNY REINHARDTSEN
NIKOLAI ROBSTAD

Thesis Submitted In Partial Fulfillment of Requirements for the

DEGREE OF MASTER OF MULTIMEDIA AND EDUCATIONAL TECHNOLOGY

Under the Supervision of

RUNE ANDERSEN, ASSISTANT PROFESSOR

At the

UNIVERSITY OF AGDER
FACULTY OF ENGINEERING AND SCIENCE
SPRING 2018

UNIVERSITY OF AGDER
FACULTY OF ENGINEERING AND SCIENCE

ABSTRACT

When the prototype of a software product is completed and approved, development of a beta-ready resource is often the natural step toward a future final release. This process contains many steps that may prove vital to the eventual success or downfall of a future product. Technologies must be chosen, design methodologies must be followed, and care must be given to provide the final product with a solid foundation to build upon. This thesis follows such a development process for the development of resources for beta-testing. This process, as well as the human centered design approaches, and the resulting resources, are evaluated compared to relevant theory of the field. In addition, an external team was outsourced development of a mobile application. This thesis gives insight into of some of the considerations, as well as some of the pitfalls, that can occur during a process to take a prototype to a beta-ready stage. It concludes that although the research questions can be answered and the goals of stakeholders were largely met, there is still a significant amount of structured work required to elevate the result to a desired level of quality.

Keywords: Application development, human-centered design, single authentication service, development process, application success factors.

PREFACE

This thesis is our final work for the degree of Master in Multimedia And Educational Technology at the University of Agder. The five years leading up to this thesis has been a journey which has provided us with the knowledge, skills and experience. This project has had a steep learning curve for both the students, and has often been both challenging and rewarding as many new theoretical fields and practical concepts were explored. We would like to give a special thanks to Rune Andersen at the University of Agder, for his guidance and support throughout this project. Thanks also go to Jan Morten Eskilt and Ruben Hansen for giving us the opportunity and the trust to work on SeafarerCV. This has given us a tremendous amount of experience from a "real world" development project, as well as the opportunity to work in a project where both students could work in their field of interest. Lastly we would like to thank each other for the cooperation throughout the project.

Ronny Reinhardtsen & Nikolai Robstad
2018

GLOSSARY

.NET Core	Free and open-source managed software framework for Windows, macOS and Linux. 6
Application Programming Interface	Set of subroutine definitions, protocols, and tools for building application software. 4
Authentication	The process of determining whether someone or something is, in fact, who or what it is declared to be. 4–6, 8, 9, 27, 29
Authorization	The function of specifying access rights/privileges to resources. 4–6, 27
European Union	Political and economic union of 28 member states that are located primarily in Europe. 13
Framework	Abstraction in which software providing generic functionality can be selectively changed by additional user-written code, thus providing application-specific software. 6, 7
General Data Protection Regulation	Regulation in EU law on data protection and privacy for all individuals within the European Union. 5
High-fidelity prototype	Computer-based prototype, that usually allow realistic user interactions. High-fidelity prototypes take you as close as possible to a true representation of the user interface . 7, 9
IdentityServer4	OpenID Connect and OAuth 2.0 framework for ASP.NET Core 2. 6

International Transport Workers' Federation	Global union federation of transport workers' trade unions. 1
Linux	Family of free and open-source software operating systems. 29
Model–view–controller	Architectural pattern commonly used for developing user interfaces that divides an application into three interconnected parts. 5
OAuth 2.0	Industry-standard protocol for authorization. 6, 8, 27
Object-relational mapping	Programming technique for converting data between incompatible type systems using object-oriented programming languages. 6
Open-source software	Type of computer software with its source code made available with a license in which the copyright holder provides the rights to study, change, and distribute the software to anyone and for any purpose. 6
OpenID	Open standard and decentralized authentication protocol. 6
OpenID Connect	OpenID authentication layer on top of OAuth 2.0. 6, 8, 27
Outsourcing	Agreement in which one company contracts its own internal activity to different company. 7, 8
Quip	Collaborative productivity software suite for mobile and the Web. 27
React	JavaScript library for building user interfaces. 7, 9
React Native	Framework for building native apps using React. 15

Representational state transfer	Architectural style that defines a set of constraints and properties based on HTTP. 6
Seagull Maritime	Global provider of e-learning and competence management. 1, 28
SemanticUI	Web-development framework for user interface layout and design. 7
Single sign-on	Property of access control of multiple related, yet independent, software systems. 8
Single-page application	Web application or web site that interacts with the user by dynamically rewriting the current page rather than loading entire new pages from a server. 5
University of Agder	Public university with campuses in Kristiansand and Grimstad, Norway. 1
Virtual private server	Virtual machine sold as a service by an Internet hosting service. 29
WordPress	Open-source content management system. 8

ACRONYMS

API Application Programming Interface. 4–9, 26, 27, 29

EU European Union. 13, 14

GDPR General Data Protection Regulation. 5, 13, 14

HCD Human-centered design. 12, 17

ITF International Transport Workers' Federation. 1

MVC Model–view–controller. 5

ORM Object-relational mapping. 6
OSS Open-source software. 6, 7

PbD Privacy by Design. 13

REST Representational state transfer. 6, 8

SFCV SeafarerCV. 1, 2, 4, 6–8, 10, 18, 21, 22, 26–29
SPA Single-page application. 5, 7, 9
SSO Single sign-on. 8

UCD User-centered design. 17
UI User interface. 4, 5, 7, 11
UiA University of Agder. 1, 27, 29
UX User experience. 3, 11, 12, 15, 27, 29

VPS Virtual private server. 29

LIST OF FIGURES

LIST OF TABLES

TABLE OF CONTENTS

1. Introduction

1.1 Background

SeafarerCV (SFCV) is a company that was founded in 2017 and is currently based in Arendal, Norway. The goal of the company as stated by the stakeholders is to provide a centralized service for maritime document verification and storage, as well as a digital platform for maritime document exchange and access in order to aid in the recruitment process of potential seafarers (SeafarerCV, 2018). Their motivation is to improve the time and effort currently required by both the seafarer and the employer during the process of maritime paperwork processing that is largely done manually. For the seafarer, this process involves delivering original documents to potential employers, resulting in temporary loss of original documents that could delay or hinder other employment opportunities. For the employer this means costly, time consuming and potentially error-prone manual verification of large amounts of documents. In addition, the leap towards such a digital solution may be smaller than perceived, due to the fact that much of the maritime documentation that is currently issued is already available in a digital format. An example of this is one of their shareholders, Seagull Maritime[1], who already provide a digital solution for verification of their issued documents. Finally, the SFCV stakeholders hope that the partnership with International Transport Workers' Federation[2] (ITF) will both ensure that SFCV becomes the digital standard for maritime document verification, and also serve as a seal of approval from one of the largest trade-union federations, representing nearly twenty million workers (International Transport Workers' Federation, 2018).

During the first half of 2017, a prototype was developed by students at the University of Agder[3] (UiA) under the direction of SFCV. This prototype was meant to demonstrate the potential of the ideas presented by SFCV to the maritime industry, and serve as a base for further development. In November of 2017, UiA was again approached by SFCV with the intent of involving a group of master's students in the process of furthering the work done in the prototype to something that would be considered ready for beta testing. During this period, we were presented with the idea and offered the opportunity to accept the project as the basis of our master's thesis. Shortly following the initial meeting, the offer was accepted and the

[1]Seagull Maritime: Global provider of e-learning and competence management
[2]International Transport Workers' Federation: Global union federation of transport workers' trade unions
[3]University of Agder: Public university with campuses in Kristiansand and Grimstad, Norway

planning phase of the project was started in early December of 2017.

The first task was to establish what practical resources SFCV expected from our involvement with the project, as well as what academic approach that would be taken with regards to the practical projects. It was clear that interaction design could be a major part of the thesis, as well as to look at the development process itself and explore whether or not successes and failures in the interaction design of the resource could be attributed to the development process or the management of this process. Although these ideas were discussed, a final decision was not settled upon as it was decided that the project would start with a practical period of two months consisting of January and February of 2018. In this period, development of the resource under the direction of SFCV would be the main focus. After this period, development would cease and user testing and academic work based on the resource and development process would start.

It was decided that during and following the conclusion of our practical project period, the resulting resource would be used as a base for security testing by bachelor students Bentsen and Holte for their bachelors project. This would provide useful insight into the threat model of the resource at the end of development, as well as recommendations on future work to further ensure security of the products. Following this, academical work would be based upon the resources created.

Because of the large emphasis of the practical resources of this thesis, half the available time was spent developing these resources during the first two months of the project. Because of this, this process is described in detail in chapter 2.

1.2 Research Questions

Based on the discussions around the project, the following research questions were chosen. These questions are divided into three areas, namely interaction design, technical implementation of authentication and authorization, and the development process.

- To what degree can a HCD-based methodology be followed in the external development of a mobile application?

Due to the mobile applications being developed by external developers, it was the wish of SFCV to see how the quality of such a process could be maintained, and in particular the Human-centered design (HCD) methodology. This research question tries to answer to what extent this can be achieved.

- How can SeafarerID be implemented as a single authentication service for all internal and external SeafarerCV services?

One of the resources described in chapter 2 revolves around the implementation of an authentication[4] and authorization[5] service. It was the wish of SFCV to implement this as a single service used by all internal and external services, as well as as an OpenID Connect[6] provider. This research question revolves around this implementation.

- To what degree can the eventual success of SeafarerCV be predicted based on the development process and known success factors?

As most of this thesis revolves around a development process, it was desirable to retroactively look at this process and see to what extent the development process fulfills criteria theory suggests might affect the success of a development process and its end-product. This research question tries to answer this with regard to the process and products of SFCV.

[4]authentication: The process of determining whether someone or something is, in fact, who or what it is declared to be

[5]authorization: The function of specifying access rights/privileges to resources

[6]OpenID Connect: OpenID authentication layer on top of OAuth 2.0

1.3 Existing Research in the Field

Essential to our work on HCD in the thesis, the research of Benyon functioned as a guide to HCD. This was complemented by the work of Rogers, Sharp, & Preece. This research has also been helpful for definitions and methods. The work of Lazar, Feng, & Hochheiser has been a helpful to explore which research methods to be used in a HCD process. Due to the imposing implementation of the General Data Protection Regulation[7] (GDPR), the official specification provided by the European Union serves as the authoritative document for consideration. For practical implementations of these principles, the research of Cavoukian and the Privacy by Design (PbD) principle provides valuable solutions for how to lead a privacy-oriented development process.

1.4 Limitations and Restrictions

There are several elements that affects a project. The main limitation of this thesis, and in particular the practical development period, is time and personell. Due to the process of taking a prototype and developing a beta-ready resource in less than two months with a single software developer, there were severe limitations on desired functionality, and the state of completion of such functionality, that could be incorporated. In addition, as discussed in chapter 6.1, the sudden change of focus from a beta-ready resource to a release-ready product limited our ability to focus on preperations and functionality for beta-testing. Another limiting factor was communication difficulties between us and the team working on the outsourced mobile applications. Due to the outsourced applications relying on and using resources developed by us, communication was vital and unfortunately at times lacking, causing misunderstandings and delays due to culture differences and language barriers.

[7]General Data Protection Regulation: Regulation in EU law on data protection and privacy for all individuals within the European Union

1.5 Thesis Structure

This thesis is structured according to guidelines for Master thesis, provided by UiA. First, chapter 1 introduces the company of SFCV and the project and its motivations, as well as the previous work that has been done leading up to the thesis period. Following this, chapter 2 talks about the planning and execution of the practical resource project. The theory of chapter 3 explores different theoretical foundations relevant to this thesis, such as application development processes and User experience (UX). Chapter 4 describes the different methods and approaches for gathering of data as well as the reasoning behind the chosen methods. This data is presented in chapter 5. In chapter 6 the findings and results are elaborated upon and discussed with relation to the theory in chapter 3. Finally, in chapter 7 a conclusion is drawn from the results and discussions in chapter 6, as well as suggestions for future work. Following this is both the list of references and the appendices.

2. SeafarerCV Resources

In January and February 2018, several resources were developed for SFCV based on the existing prototype and with the goal of reaching beta-ready functionality. The intent of this process was both to create a foundation upon which to build a solid master's thesis, but also as valuable initial resources for SFCV to build upon in the future. A key component of this thesis, are the actual resources that were developed. This chapter describes the resources created, the process taken to develop the resources, as well as the background necessary to understand the background for the research questions.

2.1 Planning

The first step in the planning phase of the resource was to examine the work previously done on the prototype, and contact was established with the developers of the original prototype. The developers gave a thorough briefing of the project and the state of all prototype functionality, and their thoughts on future work that they felt needed to be done. Functionality that was demonstrated was a web-application where users had the ability to log in, upload documents, and verify them manually. There was also a native mobile application that was purely used to demonstrate push-notification functionality. The developers were clear that some work that had already been completed was merely for demonstrative purposes, and thus could largely be discarded. Following this, time was spent analyzing how to further develop the prototype. During this period, a conclusion was reached that the functionality should be split up into several self-contained microservices. In this case the microservices would be firstly an authentication and authorization service for authenticating users, as well as an authorization authority for other services. This would be connected to the Application Programming Interface[1] (API) service to be used by the web-application, the mobile application and possible future clients. There would also be an account service for user registration and administration. Finally, a web-application would be developed as both an alternative to the mobile application, as well as an User interface (UI) providing more advanced functionality than the limited size of a mobile device screen allows. Due to this decision, it was decided to start entirely new projects, and use the prototype as a reference rather than a starting point. This also made it possible to think differently with

[1] Application Programming Interface: Set of subroutine definitions, protocols, and tools for building application software

regard to how to further develop the prototype. For instance, the prototype was a traditional Model–view–controller[2] (MVC) application, meaning each navigation and action required a page-reload. In contrast to this, it was decided to develop the beta-ready web-application as a Single-page application[3] (SPA) that was in direct contact with the API. Although the main reasons for this decision was interaction design motivated, it was also felt that this was an ideal choice, as this would make it possible to actively test the API as it was in development and discover any deficiencies real-world use would expose.

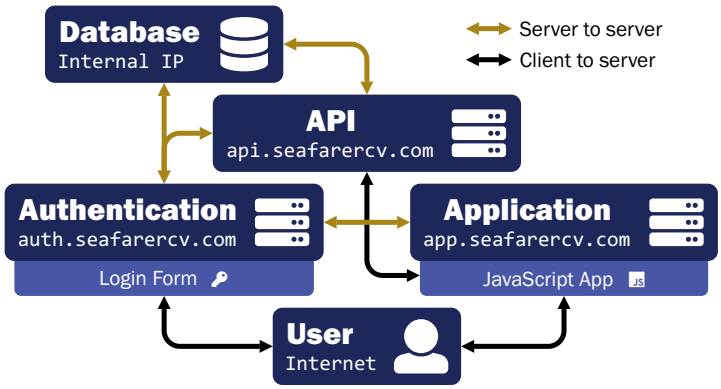

Figure 2.1: Planned service architecture

As shown in figure 2.1, the planned architecture consisted of several interconnected services, where the goal was to logically limit the responsibilities of each service. The most fundamental of all the services is the database. This is a service that would only be used directly by both the authentication service and the API service without external access. The authentication service is responsible for both presenting the user with login forms when required and serving as an authentication and authorization authority for other services. The API service is where clients may access data from the database. Each request to the API service requires an access token that has been issued by the authentication service. The application service may be any client application, but in the case of the figure, the example of a web-application is used.

With the new GDPR directive taking effect in May of 2018, it was vital that also this was given consideration both in the technical and architectural planning phase as well as in the UI design phase. Although GDPR was not one of the main considerations for the beta resource development, the back-end architecture needed to be developed in such a way as to not hinder future implementations of GDPR functionality.

[2]Model–view–controller: Architectural pattern commonly used for developing user interfaces that divides an application into three interconnected parts

[3]Single-page application: Web application or web site that interacts with the user by dynamically rewriting the current page rather than loading entire new pages from a server

Although it was initially planned we would develop mobile applications, it was decided by SFCV to use external developers for this task. To aid in this, external designers produced additional sketches and mock-ups to various developers. To supplement this we would also provide sketches and High-fidelity prototypes[4] to function as a resource for the developers as well as a resource for us with regard to usability testing and further work on the interaction design.

Originally, SFCV had used external developers to construct a website which was to be used as an early sign-up page. From the sign-up page, SFCV intended to collect names and email addresses from potential users and provide them with a short informational text about the project. The website was given to a team of external developers and the project officially started the 24th of November 2017. The website was to be based on the WordPress[5] content management system and would, in addition to collect names and email addresses, function as a communication channel, providing information about progress of development. In addition, it would later be used to invite users for beta testing of the mobile and web-applications. Throughout the development of this website, SFCV decided there no longer was a need for a sign-up page. Instead, the website would only function as an informational website for SFCV. As the project neared completion, the SFCV stakeholders were displeased with both the design and the structure of the website. In addition, the development of this website served as a bottleneck for the rest of the project, as this website was required to be in place before SFCV could initiate the marketing of their project to seafarers. This also caused delays for us, as it reduced the ability of getting responses from user demographic surveys, which were important for the work on the project. As a result of slow progress, it was decided in March of 2018 to offer to finish development of the landing page for SFCV. The goal for the website was to give SFCV a resource where they were in control of the content and could easily and quickly update this. As SFCV already had decided on WordPress, this was a quite easy goal to achieve.

[4]High-fidelity prototypes: Computer-based prototype, that usually allow realistic user interactions. High-fidelity prototypes take you as close as possible to a true representation of the user interface

[5]WordPress: Open-source content management system

2.2 Execution

One of the principles that were adopted for the development of the resource was to use as many Open-source software[6] (OSS) frameworks[7] and solutions as possible. The first choice to be made was what frameworks to use. It was decided to use the .NET Core[8] framework as the main development framework. The reason for this was both our familiarity with the framework and the rich package ecosystem that would greatly benefit the short development period. The OAuth 2.0[9] and OpenID Connect protocols were chosen as as the basis of the authentication and authorization services. The reason for this was the wish of SFCV stakeholders to introduce an industry-standard SeafarerID, a goal that would be aided by making SFCV an OpenID[10] provider. In addition to internal services, this would also allow for other websites to let users log in with their SFCV credentials, in a similar fashion as to what currently many social networking sites offer. This presented a complicated learning-curve early in the development process, as we were not familiar with neither the OAuth 2.0 or OpenID Connect standards. Using the IdentityServer4[11] framework for .NET Core allowed us to bypass most of the headaches one would associate with trying to manually implement such a comprehensive standard, and was solely responsible for quick implementation of this early in development. The log-in form of the authentication and authorization service can be seen in figure 2.2. As the service with most inbound requests, the API needed to be fast, stable and secure. The API would be based on the Representational state transfer[12] (REST) principles. The .NET Core framework already contained most of the functionality necessary to achieve this, with a few exceptions. For communication with the database, an Object-relational mapping[13] (ORM) framework was used, allowing for easy and fast database operations. For flexibility between the data models and the representational models, a framework was used to map data from one object to another. For authorization, .NET Core already contained functionality to interface with the OAuth 2.0 and OpenID Connect server to check validity of access tokens included in requests to the API.

[6]Open-source software: Type of computer software with its source code made available with a license in which the copyright holder provides the rights to study, change, and distribute the software to anyone and for any purpose

[7]frameworks: Abstraction in which software providing generic functionality can be selectively changed by additional user-written code, thus providing application-specific software

[8].NET Core: Free and open-source managed software framework for Windows, macOS and Linux

[9]OAuth 2.0: Industry-standard protocol for authorization

[10]OpenID: Open standard and decentralized authentication protocol

[11]IdentityServer4: OpenID Connect and OAuth 2.0 framework for ASP.NET Core 2

[12]Representational state transfer: Architectural style that defines a set of constraints and properties based on HTTP

[13]Object-relational mapping: Programming technique for converting data between incompatible type systems using object-oriented programming languages

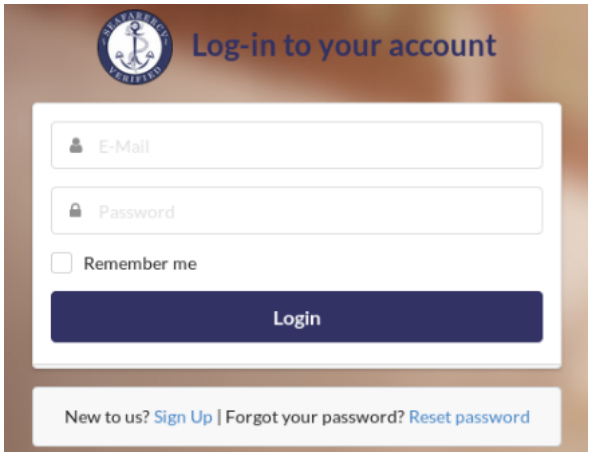

Figure 2.2: The authentication service

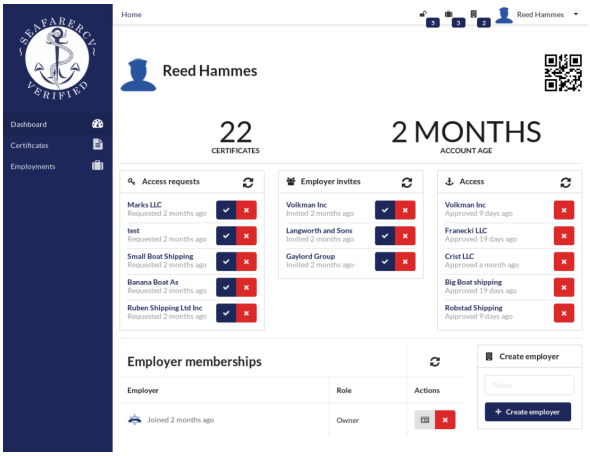

Figure 2.3: The web application

As the web-application was planned as a SPA, there were several frameworks that were considered to achieve this. Eventually it was decided to use the React[14] framework due both to its rich ecosystem and the our familiarity with the framework. Building on the principles of using OSS packages, this idea was extended to the UI as well. SemanticUI[15] was used to build the UI, a web-development library with various components and helpful customizable functionality for quickly building functional and visually modern UI. The dashboard of the web-application can be seen in figure 2.3

[14]React: JavaScript library for building user interfaces
[15]SemanticUI: Web-development framework for user interface layout and design

The High-fidelity prototype sketches were produced in gray-scale to keep the focus on the interaction design as well as layout. Shortly after, color and visual design elements were added to the High-fidelity prototype as shown in figure 2.4.

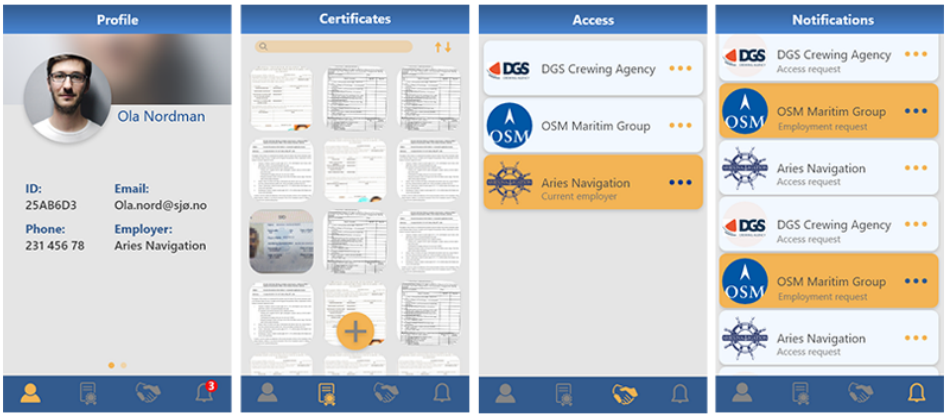

Figure 2.4: Views from the high fidelity prototype

SFCV did not possess an existing visual profile, so the color scheme and design originated from the ideas of the designers, and was considered examples or suggestions on how to proceed. The design appealed to SFCV. The High-fidelity prototype was then adjusted to the wishes SFCV. This included removing employer functionality, resulting in a High-fidelity prototype for an application only consisting of seafarers functionality in the system. With regards to the landing page. Within three days from the start of development, the website was reorganized into two separate areas, one containing information aimed at seafarers with the other presenting information meant for seafarer administrations. A review on the design was done and redesigned for a simpler and responsive overall design, as shown in figure 2.5.

Figure 2.5: SeafarerCV landing page

2.3 End Result

The resource at the end of the development period consists of many parts. Firstly, an authentication service with OAuth 2.0 and OpenID Connect functionality, allowing for Single sign-on[16] (SSO) functionality as well as future option to use SFCV credentials as a login method to other websites and services. The API service is protected by the authentication service and using REST principles data to all current and future clients. The application service is protected by the authentication service and uses the API service through its React SPA. The account service is protected by the authentication service for future management as well as account creation functionality. Finally, a High-fidelity prototype to be used as a base for external developers and a landing page to be used as a front for seafarers and employers alike.

[16]Single sign-on: Property of access control of multiple related, yet independent, software systems

3. Theory

In this chapter the theory used to explain the resources of chapter 2, as well as the grounds for discussion in order to answer the research questions in chapter 1.2, is presented. The theoretical concepts explored range from interaction design concepts such as HCD, privacy and security, project management, and more technical theories regarding application development.

3.1 Human Centered Design

According to Abras, Maloney-Krichmar, and Preece, HCD is based on the physical and psychological needs of the human user (Abras et al., 2004). HCD is not a design style, but a process for design and development that is grounded in information about the people who will be using the product (Abras et al., 2004). HCD is also known under the term User-centered design (UCD) and are often used interchangeably. Both UCD and HCD are somewhat broad terms which describe the design processes in a way where end-users influence how a design takes is developed. There is a spectrum of ways in which users are involved in UCD, but the important concept is that users are involved in one way or another. For example, some types of UCD consult users about their needs, and involve them at specific times during the design process, typically in the requirements gathering and usability testing (Abras et al., 2004).

3.1.1 PACT

When designing or planning the design of an interactive system, the acronym PACT is a useful framework for thinking about the design situation. PACT is an acronym that represents People, Activities, Context and Technology (Benyon, 2014). As a system or product designer it is necessary to understand the people that are going to use the system that is being created. It is also important to understand the activities that people would want to undertake and the context in which those activities would take place, as well as the features of the technologies that are in use (Benyon, 2014). An example of such a flow is pictured in figure 3.1

Figure 3.1: PACT Model

Redrawn from (Benyon, 2014)

People differ in physical characteristics. Variability in the five senses - has a huge effect on how accessible and usable, and how enjoyable using a technology will be for people in different contexts(Benyon, 2014). When designing an application like SFCV, physical differences and variability in the five senses might perhaps not logically be the immediate factors thought to affect the design. However, things such as color blindness is important to take into consideration for design choices. People also have different needs and abilities when it comes to attention and memory. One can assume that most people cannot remember long numbers or complicated instructions. People also have psychological differences, for example people with good spatial ability will find it easier to find their way around and remember a website. Systems should be designed for people with poor ability by providing good signage and clear directions (Benyon, 2014).The psychological differences is also closely connected with the mental models the users have of the system. If people do not have a good mental model of something that can only perform actions by rote. If something goes wrong they will not know why and be able to recover, which is often the case with people using software systems (Benyon, 2014). There are many characteristics of activities that designers need to consider. A designer should focus on the overall purpose of the activity. How regular or infrequent the activities are. Time pressures and troughs of working, when using the system. The response time needed from the system. Are the activities carried out alone or when working with others. How the tasks in the system is defined, as well defined tasks need different designs from more vague tasks and it

is important to think about what happens when people makes mistakes and errors (Benyon, 2014).The activities performed on a system or product always takes place in context. So a designer need to analyze the two together. According to Benyon there are three useful types of context that are distinguishable. The organizational context, which is important as changes in for example technology often alter communication and power structures. The social context, the social environment in which the activity takes place. The physical circumstances, the physical environment in which the activity takes place. A concrete example for our product is the consideration of geographically remote environments, where internet access is slow. Technology is the medium that the system designer work with (Benyon, 2014). A interactive system typically consist of hardware and software components that communicate with one another. Designers of interactive systems need to understand the medium they are working with and how people interact with these mediums (Benyon, 2014).

3.2 User Experience

UX is a term that is used for the overall experience a user has with a software application. According to Charland and Leroux, UX can be divided into two primary categories. The context describes the elements that must be understood but cannot be changed or controlled. These include hardware affordance, platform capabilities, UI conventions, and the environment in which the application is used(Charland & Leroux, 2011). The implementation are the elements that can be controlled in the application, such as performance, design and integration with the platform features (Charland & Leroux, 2011). UX derives from the same philosophy as HCD, and as described in the section above, many of the elements and areas for consideration are the same. One can not design a user experience, only design for user experience (Abras et al., 2004). User experience is about how people feel about a product and their pleasure and satisfactions when using it (Abras et al., 2004). How easy a system is to learn to use, is known as learnability and is connected with the experience a user has with a system. People don't like spending a long time learning how to use a system, they wanna start using it straight away, and want to be competent carrying out tasks without to much effort (Rogers et al., 2016).

3.3 Acceptability

Acceptability is about fitting technologies into people's lives and can only be understood in the context of use (Benyon, 2014). The technology acceptance model as shown in figure 3.2 is used to look at technologies and whether or not they will be accepted by communities. The model looks at technology acceptance from the perspective of ease of use and effectiveness (Benyon, 2014).

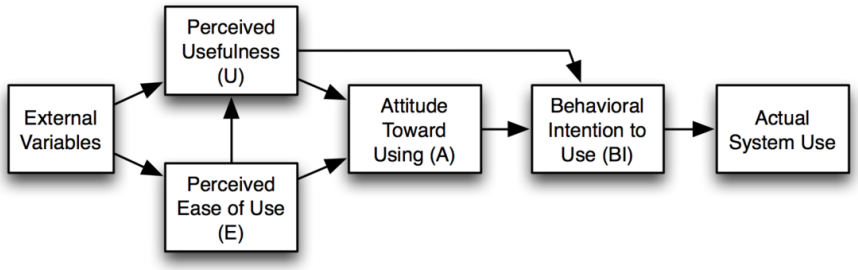

Figure 3.2: Technology acceptance model

The model can tell us to what extent users will accept and use new technology or systems such as SFCV. The model bases this on perceived usefulness which is defined as "the degree to which a person believes that using a particular system would enhance his or her job performance" (Davis, 1989), as well as perceived ease of use which is defined as "the degree to which a person believes that using a particular system would be free of effort" (Davis, 1989). There have been several similar models and theories for motivation and use of technology.

There can be several features that would affect users acceptability towards new systems and technologies. According to Benyon, the key features of acceptability are political, convenience, cultural and social habits, and usefulness and economic (Benyon, 2014). All these features can influence how the technology will be accepted in a community. The unified theory of acceptance and use of technology shown in figure 3.3 was developed based on the eight models and combinations of these different theories and models (Venkatesh, Morris, Davis, & Davis, 2003). These are, the theory of reasoned action, the technology acceptance model, the motivational model, the theory of planned behavior, a model combining the technology acceptance model and the theory of planned behavior, the model of computer utilization, the innovation diffusion theory, and the social cognitive theory (Venkatesh et al., 2003).

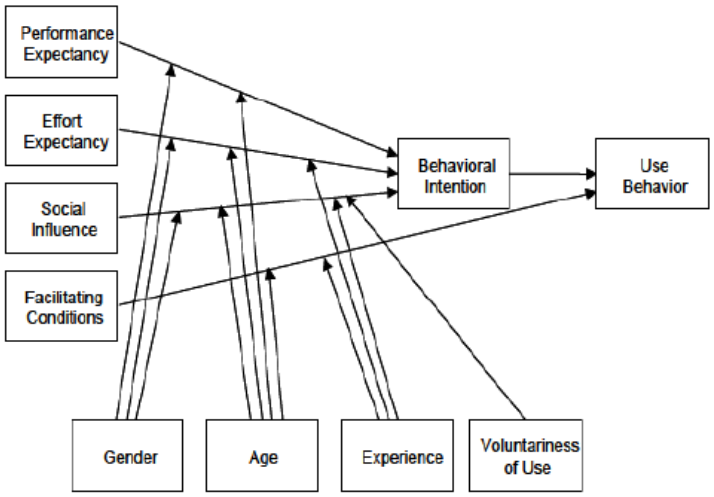

Figure 3.3: The unified theory of acceptance and use of technology (Venkatesh et al., 2003)

3.4 Project Management

For the success of a project, one can assume that some form of management must take place. However, theories and concepts for project management are varied and evolve over time. True and tested methods can at times clash with more modern project management theories, and care must be taken to choose the right approach for each individual project to ensure both a efficient project as well as a successful end-product.

3.4.1 Lean UX

Lean UX is designed to create and deploy innovative products quickly and is linked to Agile UX as Agile software development is one of it underlying philosophies, and champions the importance of providing good user experience (Rogers et al., 2016). Lean UX is based on tight iterations of build-measure-learn, a concept central in lean startup idea, which is inspired by the lean manufacturing process. It's also influenced by design thinking which focus on the understanding of what people want and what technology can deliver, and take a holistic view on the context and product, while evolving a solution through prototyping(Rogers et al., 2016).

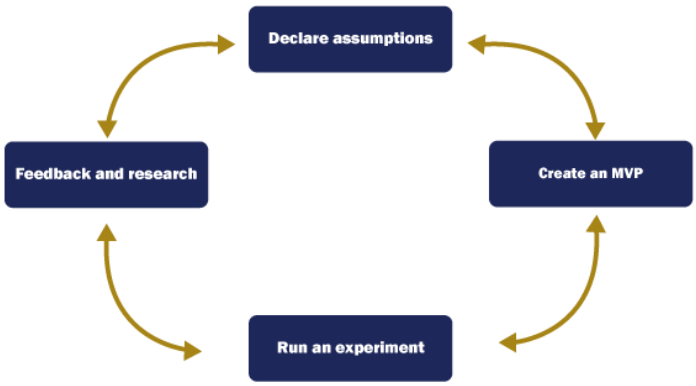

Figure 3.4: The Lean UX process

(Gothelf, 2013)

3.4.2 DevOps and Microservices

Although there are many definitions to the relatively new concept that is DevOps, a word that is clip-compounded by the two words "development" and "operations", Dyck, Penners, and Lichter propose DevOps to be an organizational approach that "stresses empathy and cross-functional collaboration within and between teams — especially development and IT operations — in software development organizations, in order to operate resilient systems and accelerate delivery of changes" (Dyck et al., 2015). The concept of DevOps is strongly linked to the concept of microservices. According to Dmitry and Manfred, the microservice approach to software architecture is to develop applications as small independent services, in contrast to large monolithic services that have been the popular in previous years (Dmitry & Manfred, 2014). According to Balalaie, Heydarnoori, and Jamshidi, microservices enables the DevOps approach to thrive as it minimizes the coordination required between teams responsible for each component, and breaks barriers preventing productive relationships between development and operations teams (Balalaie et al., 2016).

3.4.3 Development Life Cycle

According to Ruparelia, the period of time from the inception of a software development process to its completion and maintenance is called a development life cycle (Ruparelia, 2010). This life cycle can be structured and planned using various models, usually categorized into three different groups, namely linear, iterative and a combination of the two (Ruparelia, 2010). Linear models are a series of stages that are completed synchronously, while an iterative model focuses on actively revisiting previous stages and a combined model is primarily synchronous but revisits previous stages when deemed necessary. One model of the software development life cycle is the waterfall model. This model, while originally a synchronous model, was expanded upon in 1987 by Royce to include iterative qualities as shown in figure 3.5 (Royce, 1987).

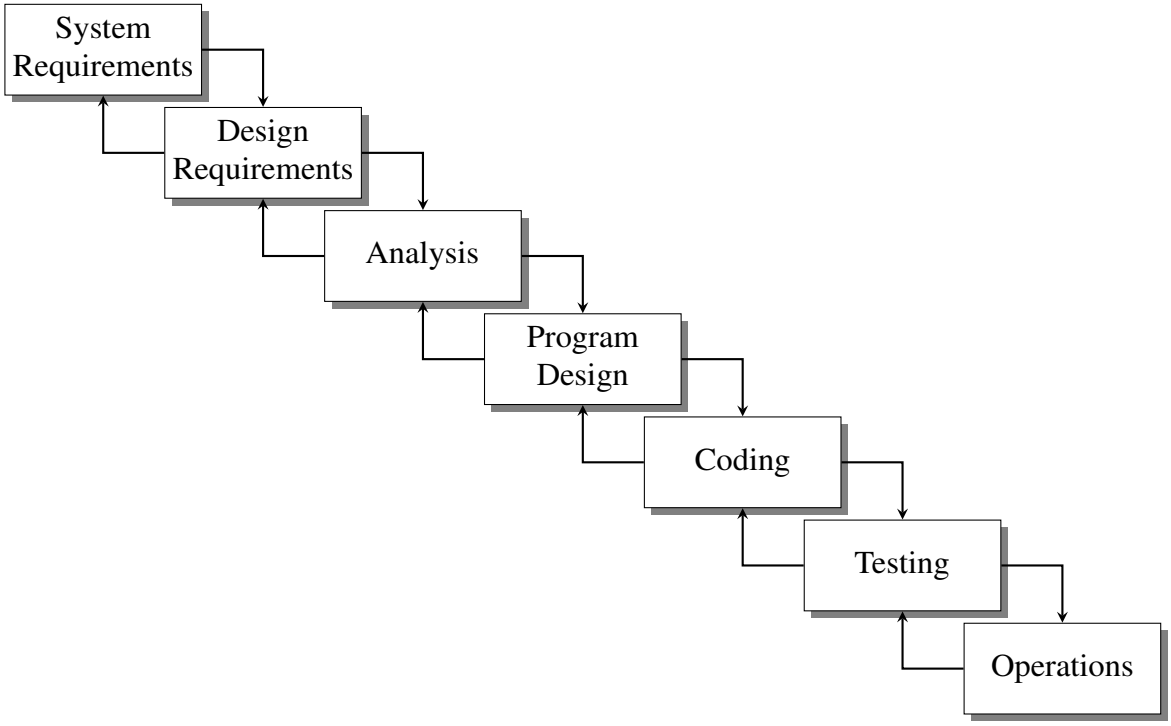

Figure 3.5: An iterative waterfall model

Redrawn from (Royce, 1987)

Another iterative development life cycle model is the Rapid Application Development model. This model is based on prototyping as a mechanism and promotes collaboration between business stakeholders and developers, and has spawned several related methodologies such as Agile (Ruparelia, 2010). The Agile approach to software development is defined by a capability to respond to change, to balance structure and flexibility and to draw innovation and creativity

out of a development team in order to guide an organization through chaotic and uncertain development processes (Highsmith & Cockburn, 2001). In an Agile development life cycle, the same procedure as demonstrated in figure 3.5 can be undertaken, but would be performed repeatedly in many short iterations in contrast to one large process.

3.5 Privacy and security

When developing any form of application directed at users, both privacy and security are crucial both in terms of legal ramifications and to establish trust with users. With the growing public and legislative awareness of both security and privacy, as can be seen with the new GDPR, it is imperative to employ theories and methods to ensure a product is carefully constructed to comply with both these and future regulations, as well as inspiring trust in potential users.

3.5.1 General Data Protection Regulation and Privacy by Design

GDPR is a European Union[1] (EU) directive designed to harmonize data privacy laws across Europe, where the goal is to empower data privacy of all EU citizens and to reshape the way organizations approach data privacy. (European Union, 2016) When designing any type of application that stores and handles personal data it is important to take privacy into account throughout the whole engineering and design process. This process is known as PbD, and is a legal requirement associated with GDPR. According to Cavoukian, Privacy by Design (PbD) is a framework aimed at embedding privacy into information technology, business practices, physical design and networked infrastructures directly (Cavoukian, 2012b). Cavoukian defines the 7 principles of PbD as follows (Cavoukian, 2009).

- Proactive not Reactive; Preventative not Remedial
- Privacy as the Default Setting
- Privacy Embedded into Design
- Full Functionality – Positive-Sum, not Zero-Sum
- End-to-End Security – Full Lifecycle Protection
- Visibility and Transparency – Keep it Open
- Respect for User Privacy – Keep it User-Centric

[1]European Union: Political and economic union of 28 member states that are located primarily in Europe

Although to some these principles may seem like an idealized version of the reality of information technology, according to Breach Level Index, a website that tracks publicly disclosed data breaches, over 9 billion data records have been lost or stolen since 2013, many of which could have been avoided (Breach Level Index, 2018). According to Rubinstein and Good, several data breach incidents at social media websites that they had investigated could have been avoided by following privacy principles and practices such as PbD (Rubinstein & Good, 2013). Not only is this approach to design important from a users perspective, but it is also vital for companies moving forward, as privacy breaches can have a profound financial consequence, as well as permanent damage to brand reputation (Cavoukian, 2012a).

GDPR defines personal data as "any information relating to an identified or identifiable natural person" (European Union, 2016). One of the most comprehensive changes following the GDPR directive is the extended jurisdiction, as it will apply to all companies processing personal data of data subjects residing in the EU regardless of the location of the company. GDPR will also apply to processing of personal data of data subjects in the EU by a controller or processor not established in the EU (European Union, 2016). Conditions for consent have also been strengthened. Companies can no longer use long and complicated terms and conditions. Requests for consent must be given in an intelligible and easily accessible form, with the purpose for data processing attached (European Union, 2016). Consent must also be clear and distinguishable from other unrelated matters. According to GDPR, users have the right to obtain confirmation whether personal data concerning them is being processed, and where and for what purpose (European Union, 2016). In addition, GDPR also introduces data portability, which gives the users the right to transmit the received data to another controller (European Union, 2016). The users have the right to be forgotten, which means that the users is entitled to get his or her data erased. The right to be forgotten also applies to data no longer used or relevant for its original purpose.

3.5.2 Authentication and authorization

In order prevent anauthorized access of digital resources, such resources require the protection of a security layer. This layer usually consists of at least an authentication layer to verify a claimed identity, and an authorization layer to check whether a user or client has access to requested resources or rights (Glass, Hiller, Jacobs, & Perkins, 2000). This process can be executed in many ways, one of which is called Single sign-on (SSO). Hursti state that the purpose of SSO is to enable authentication without the user having to complete multiple different authentication procedures, as well as simpler rights management for administrators (Hursti, 1997). There are also security advantages related to the SSO approach. SSO ensures a shared, centralized and consistent authentication policy throughout an enterprise, making it easier to secure (De Clercq, 2002). To achieve this SSO functionality there are many protocols and frameworks available. One such protocol is the OAuth 2.0 protocol. OAuth 2.0 is an standardized and open authorization protocol to grant third party access to resources (Sun & Beznosov, 2012). Although the OAuth 2.0 protocol offers many advantages, care must be taken as the open and flexible approach of this protocol results in potential insecure implementations without deep understanding of web security (Sun & Beznosov, 2012). The OpenID Connect protocol further extends the OAuth 2.0 protocol with additional identity-related functionality such as verification of the identity users based on authentication from the authorization server (Sakimura, Jones, de Medeiros, & Mortimore, 2013). Due to extending the OAuth 2.0 protocol, the OpenID Connect layer has the additional advantage of easy integration with existing OAuth 2.0 systems (Mladenov, Mainka, Krautwald, Feldmann, & Schwenk, 2015).

3.6 Mobile Application Development

According to Statista there are as of March 2017 5 million mobile applications available in the application stores of the two major mobile operating systems (Statista, 2017). People migrate from their computers to mobile devices, and today many of the most popular mobile properties are mainly accessed via mobile applications instead of mobile browsers (Statista, 2017). In 2016, Localytics performed research on mobile application engagement and retention by measuring the number of times that a person will launch or open an application. According to their research, Only 36% of people are still using an application one month after installation and 23% of applications are only used once before being ignored or deleted (Localytics, 2016). After two months, the data shows that the drop-off of users is 75% and in month three 80%. On average an individual application is launched 13.78 times per month with a average use of 66.14 minute per month which gives an average use of 4.80 minutes per session. There can be a number of reasons why the data for user retention are like this, and can be partially explained by the focus on marketing and application store optimizations done by companies. More importantly, this data can show the importance of building a quality product that engages users and keeps them coming back (Zhang, 2015).

When approaching the design and the development of a mobile application with the goal of building a quality product that users wants to use and keep using, analyzing and researching known criterias for success in the application market can be helpful to see how the product that is developed meets the known criterias to better develop an application that users will keep using and keep the user retention.

3.6.1 Hybrid Mobile Applications

One of the first decisions that must be made when starting work on any form of mobile application is to decide the strategy of the development process. For this there are three choices. Applications can be built natively using the tools, frameworks and programming languages of each mobile platform, it can be built as a hybrid application where most of the application logic is shared between different platforms, or it can be built as a web application that is wrapped in a native runtime environment. According to Wasserman, one of the issues in developing mobile applications is the expense required to develop and maintain native applications for multiple platforms (Wasserman, 2010). A hybrid solution, a layer of abstraction to map shared application logic into native applications for each platform, is one solution to this (Wasserman, 2010). In 2015, merely 3.73% of the top 500 applications on a mobile platform application store were identified as hybrid (Malavolta et al., 2015). This figure may be misleading, as

the results of a survey performed by Ionic Framework suggest. Their findings show that the number of developers working exclusively on native mobile application development has gone down from 20% to 2.9% in two years, with 32.7% expecting to completely abandon native development in favor of hybrid platforms within the next two years (Ionic Framework, 2017). One of the reasons for this trend might be the increased amount of application logic that might be shared between platforms as hybrid development platforms evolve. According to Hansson and Vidhall, they found that when developing a hybrid mobile application using the React Native[2] framework, as much as 75% of their application logic could be shared between platforms without suffering UX drawbacks (Hansson & Vidhall, 2016). It is reasonable to expect this figure to rise even further as technology advances, increasing the advantage of this hybrid approach.

3.6.2 Success Factors

For successful mobile applications, many factors must be considered. MobiDev lists 11 must-have features and characteristics for a successful mobile applications. These are, simplicity for the user, coverage on major mobile platforms, performance, security, off-line functionality, regular updates, feedback, personalization, search, analytics and interoperability (MobiDev, 2016). It is argued that most of the qualities in this list are both applicable to every good application and can be easily facilitated (MobiDev, 2016). These factors and the following are not necessarily based on academic research, but these sources represents what we found, that is representing a form of consensus about important elements and success factors for a successful application. Appdynamics (appdynamics, 2015) has explored application metrics for developers to collect and analyze. Those metrics can say something about what to aim for in development of a mobile application. They have sorted their metrics on a high level, as performance metrics, how the users are experiencing the application. User and usage metrics, data that provide data about the users and their demographics. Engagement metrics, how users are engaging with the application, and business metrics, which is the focus on business, like revenue, to measure the success of a mobile application (appdynamics, 2015). What we can take from this is again the importance of knowing the users and who they are. The importance of having an application with, few application crashes, low load time and low network errors and a focus on end-to-end application and API latency. Measuring the engagement, as sessions length, interval and the retention rate can tell you more than if your application is being downloaded or not, it can tell you how the users are using it or if they actually are using it. Which again emphasising the

[2]React Native: Framework for building native apps using React

importance of designing an application base on users need and requirements.

In addition to features in the actual application, the development process and project management can also be of vital importance on the eventual success of an application. Pinto and Slevin developed a ten factor model to illustrate the critical factors for the success of a project (Pinto & Slevin, 1988).

1. Project mission - Initial clarity of goals and general directions
2. Top management support - Willingness of top management to provide the necessary resources and authority/power for project success
3. Project schedule/plans - A detailed specification of the individual action steps required for project implementation
4. Client consultation - Communication, consultation, and active listening to all impacted parties
5. Personnel - Recruitment, selection, and training of the necessary personnel for the project team
6. Technical tasks - Availability of the required technology and expertise to accomplish the specific technical action steps
7. Client acceptance - The act of "selling" the final project to its ultimate intended users
8. Monitoring and feedback - Timely provision of comprehensive control information at each phase in the implementation process
9. Communication - The provision of an appropriate network and necessary data to all key factors in the project implementation
10. Trouble-shooting - Ability to handle unexpected crises and deviations from plan

In addition, Project Management Institute state that in order for a project to be successful, the responsible team must select appropriate processes for meet objectives, use a defined approach to meet requirements, comply with requirements to meet stakeholders expectations, and to balance the demands of resources, risk, cost, time, scope and quality to produce the final product (Project Management Institute, 2008).

4. Method

This chapter describes the methods and the use of these methods in the research of the design of the mobile application.

4.1 Human-centered design

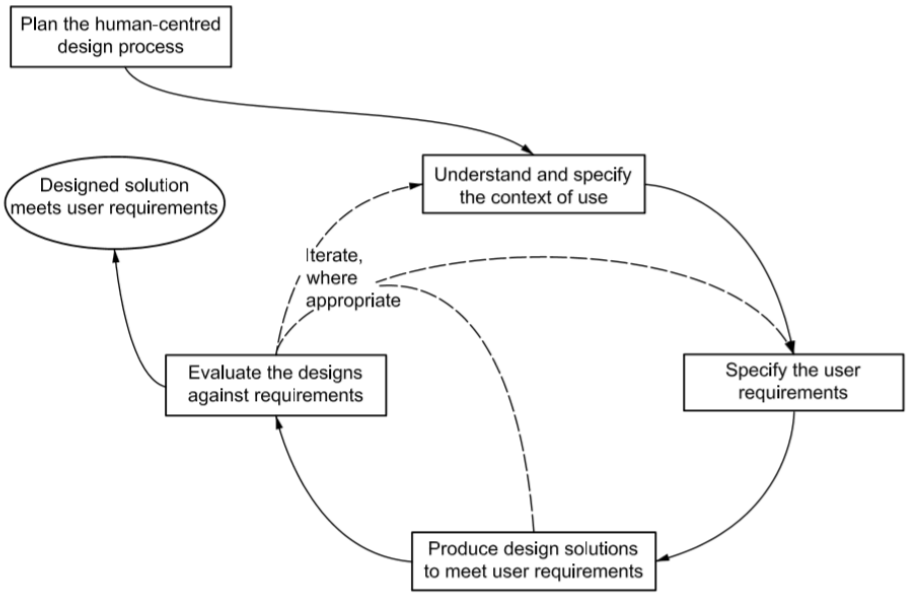

Figure 4.1: HCD Model

(ISO, 2010)

Involving users in design and developments provides a valuable source of knowledge about the context of use, the tasks and how users are likely to work with the future product, system or service. User involvement should be active, whether by participating in design, acting as a source of relevant data or evaluating solutions (ISO, 2010). Human-centred design shall be planned and integrated into all phases of the product life cycle, i.e. conception, analysis, design, implementation, testing and maintenance(ISO, 2010), as shown in figure 4.1.

4.2 Prototype

A prototype is a representation of either all of or part of an interactive system, that, although limited in some respects, can be used for analysis, design and evaluation (ISO, 2010) and is hence for this thesis considered also a method and not only a part or tool of other methods. The prototype developed in this project was a high-fidelity prototype. A high-fidelity prototype is prototype that looks like the final product, allowing realistic user interaction (Rogers et al., 2016).

4.3 Quantitative Methods

Quantitative data are "the numbers" collected through surveys or other measurement techniques (Yauch & Steudel, 2003). In our thesis a quantitative research method was used in the form of an on-line survey. This survey gathered data on potential user demographic, on-line and application habits and familiarity, as well as technical considerations such as internet reliability. Quantitative research is useful for studying a large number of people and is relatively quick research method. In most cases it is also relatively less time consuming with regard to the analysis of the gathered data as it often provides precise, quantitative numerical data. The data collected is also relatively independent of the researches which may lead to it having a higher level of credibility. In a quantitative research, the researcher may miss out on phenomenons occurring because of the focus being on theory or hypothesis testing rather than on theory or hypothesis generation (Johnson & Onwuegbuzie, 2004).

4.3.1 Survey

A survey is a well-defined and well-written set of questions to which an individual is asked to respond. Surveys are typically self-administered by an individual, with no researcher present (Lazar et al., 2017). Surveys, also known as questionnaires, is a way of streamlining the research process if a large number of people are to be queried and there are not enough resources to interview them individually (Benyon, 2014). There are also a few drawbacks when using surveys as a research method. While surveys are very good at retrieving limited "shallow" data from a large number of people, it is not well suited at getting "deep" detailed data. In addition, it is usually not possible to ask follow-up questions, or go back and change the original survey to ask more detailed questions (Lazar et al., 2017). Most people will complete questionnaires that are administered as part of a face-to-face evaluation session. When people take them away

to finish them in their own time, or to do the questions on the web, they very often do not complete them (Benyon, 2014). The survey in this study was digital. The digital questionnaire was distributed through a closed group on social media, and available on-line through the website of SFCV, directed at seafarers. The questionnaire in this development process and study was done as an attempt to collect user demographics and to identify user requirements and needs.

4.4 Qualitative Method

Qualitative data are "the words" collected through interviews, focus groups, participant observation, or related methods (Yauch & Steudel, 2003). In this theisis qualitative research is our usability test, post test interview, and observations made of people interacting with the mobile application in the usability test. With qualitative research the data are based on the participants own categories of meaning, and is useful for studying a limited number of cases in depth. It provides understanding and description of people's personal experiences of a phenomena and the data can usually be collected in a naturalistic setting(Johnson & Onwuegbuzie, 2004). The knowledge produced on quantitative research may also be too abstract and general for direct application to a specific situation, context or individual (Johnson & Onwuegbuzie, 2004). With use of a qualitative research method the knowledge produced may not generalize to other people or settings, the data may be unique to the few people in the research, leading to it having a possible lower credibility. It generally take more time to collect data, relative to a quantitative study and data analysis is also often more time consuming. Most important the result are more easily influenced by the researcher personal biases in qualitative research compared to a qualitative research method (Johnson & Onwuegbuzie, 2004).

4.4.1 Usability Testing

In general, usability testing involves representative users attempting representative task in representative environments on early prototypes of computer interfaces. Usability testing has one basic goal, namely to improve the quality of an interface by finding flaws within it. At the same time, it is also desirable to discover what is working well with an interface design. (Lazar et al., 2017). In our research, usability testing took place late in the development, and can be considered as a combination of a summative test, a test where high level choices have been made, and a validation test with a goal of benchmarking the usability of an interface. In April of 2018, usability testing was conducted on the latest version of the application provided

by the outsourced developers. Six participants, both male and female and from most age groups, participated in our survey. The participants were provided with a smartphone with the application installed, and also given the option to use their own smart phone, which some did. The participants were also given a document that was to represent a certificate. For the usability test there was a prepared a list of tasks that the participants were asked to perform in the application. During this, we observed their interactions with the application and whether or not they were able to perform the given tasks. In addition a list of questions for a post-test interview with the participants was prepared as well as a short demographic interview before the test started. The participants were informed of the application's purpose as well as the purpose of the test. The tasks were presented in as neutral a way as possible. An example of this was the uploading of documents. The participant was only instructed that the document they were given needed to be stored on the service, without providing more information. This way, we could observe how the participants solved the given task within the application, without asking leading questions such as whether or not you can upload this document with the use of the camera functionality.

4.4.2 Observation

Observation is a useful data gathering technique at any stage during product development (Rogers et al., 2016). Observation conducted late in a development in evaluation, may be used to investigate how well a developed prototype support tasks and goals(Rogers et al., 2016). In our research users was observed directly while they were performing giving tasks in the usability test and interacted with the mobile application. It was not only important for us to see if the participant was able to perform the given task, it was important to observe how they solved them, how much time they used, and which path they took to solve them. An example of from our research is that even doug the majority of the participants were not able to locate the QR code in the application, we learned for the observation that most of the participants went directly to the profile page in the application, where the QR code in fact is located.

4.4.3 Interview

The Interview in our research was a semi-structured interview, which allows room for clarifications and the possibility to add questions and to follow up interviewee comments (Lazar et al., 2017). A semi-structured interview combine features of structured and unstructured interviews and use both closed and open questions. A semi-structured interview uses a basic script for guidance, to ensure the same topics are covered with each interviewee(Rogers et al., 2016).

Our interview took place after ended usability testing with the same participants as soon as the tasks in the usability test were completed. The participants were asked prepared questions, but was made aware that this was an open interview, in order to allow them to feel free to share their thoughts at any time.The list of questions was present to help us for interview structure and to ensure that all the participants were asked the same questions to provide us with data that could be more easily analyzed. The questions in the interview concerned their experience with the application, as well as their feelings towards the solution and the application in general. Specific questions were also asked, such as if they felt the use of visual elements like icons were understandable, and if the application reminded them of any other applications. The participants were also asked what they did or did not like about the application, and if there was anything they would have changed in the application.

4.5 Reliability and Validity

Reliability or consistency of a method is how well it produces the same results on separate occasions under same circumstances (Rogers et al., 2016). Different evaluation methods will ultimately have a different degree of reliability (Rogers et al., 2016). The semi-structured interview and observations method used in our research would have a low reliability, as it would be almost impossible to repeat or reproduce the exact same discussion, answers and interactions that took place during the usability test. The survey in this thesis would have a higher level of reliability compared to the interview and observations, as there are limited ways for participants to answer, as well as due to the way the answers are collected and presented. Nevertheless, the number of responsends to the survey is too low to say that the research has a generally high reliability, and can only represent the participants we were able to reach. There is a high possibility that the statistics would change if the number of respondents was higher, and thus reaching more of the potential users. Validity is concerned with whether the evaluation method measures what it is intended to measure (Rogers et al., 2016). The low number of respondents on the survey will affect the validity of the data collected. To gain a general understanding of the potential users, who they are, their needs and requirements, one would need to reach a much larger number than we were able to do due to time limitations. However, the data collected has brought us closer to an understanding of the potential users, and served it's purpose at this stage in development and research.

5. Findings

During and following the end of the practical development period for the resources created for this thesis, described in chapter 2, action was taken to both try identify the potential users of SFCV and to assess the usability of the outsourced mobile application. A user research survey was given to potential users of SFCV, as well as a usibility test and following interviews with several individuals of various backgrounds.

5.1 User Research survey

Knowing the age of the potential users gives us a greater understanding of user needs. Our findings can confirm our assumptions that we are designing an application that have a broad audience, and that the application needs to satisfy users in a broad age group, as shown in figure 5.1. The age of the users can also tell us something about the potential technology acceptance, as age is connected with acceptance according to the the unified theory of acceptance and use of technology model figure 3.3. We can hypothesize that younger users would be more familiar with new technology and digital on-line platforms.

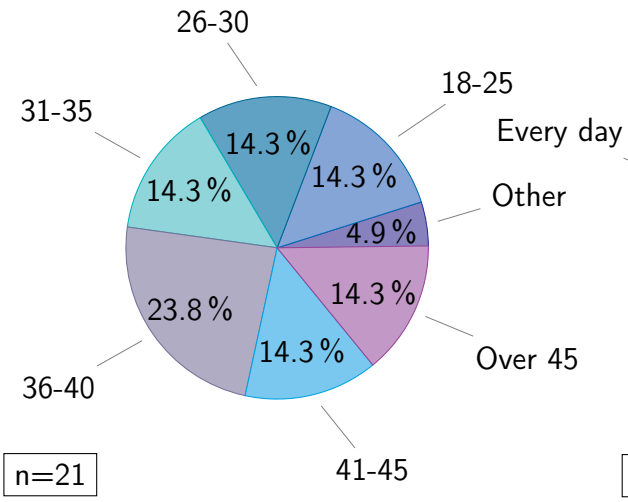

Figure 5.1: Age of users from survey

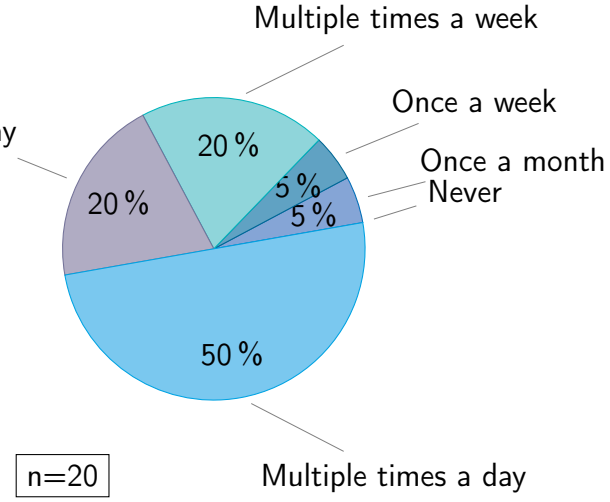

Figure 5.2: Application use frequency from survey

We know that the industry we are targeting is a male dominated industry and according to ITF Seafarers, women make up only an estimated 2% of the world's maritime workforce (ITF Seafarers, 2018). This is reflected in our findings, as all our responses were from males. The participants were asked if they felt there was a need for an application or service such as SFCV. Out of 20 respondents, 45% answered yes, 35% answered maybe and the remaining 20% answered that they were not yet familiar with SFCV. Whether users feel that technology has made their life more difficult or easier can also tell us something about the acceptance for a product like SFCV. All the participants of the survey felt that technology has made their life easier. In addition, we asked the participants how much time they typically spend on line or on mobile applications, and if they could mention some of the websites or applications they used the most. This could give us some insight to what kind of applications and on-line services the users were already familiar with. The applications and on-line services mentioned were know social media applications and web sites such as Facebook, Twitter, Instagram and Gmail. Some of the respondents also stated sites for maritime jobs and certificate verification. Figure 5.2 shows that 50% of the participants in the survey are on-line or use mobile applications every day, and that 20% do multiple times per day. As figure 5.3 shows, a majority of the participants of the survey states that they bought their smart-phone more than two years ago. Data such as this can tell us about the user's interests in technology and also help us identify technical requirements and limitations. How the users would rate their internet reliability at home and at work gives us data that can help us identify additional requirements and needs for the systems with regards to off-line and on-line usability and functionality. Figure 5.5 shows responses on how the participants rated their internet reliability at home, while figure 5.6 shows how the participants rated their internet reliability at work.

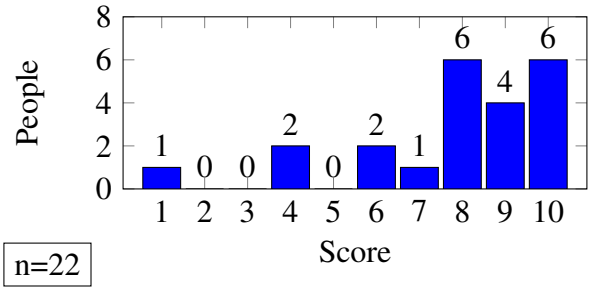

Figure 5.5: Internet reliability at home from survey

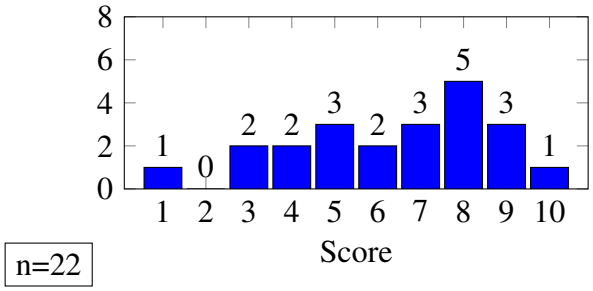

Figure 5.6: Internet reliability at work from survey

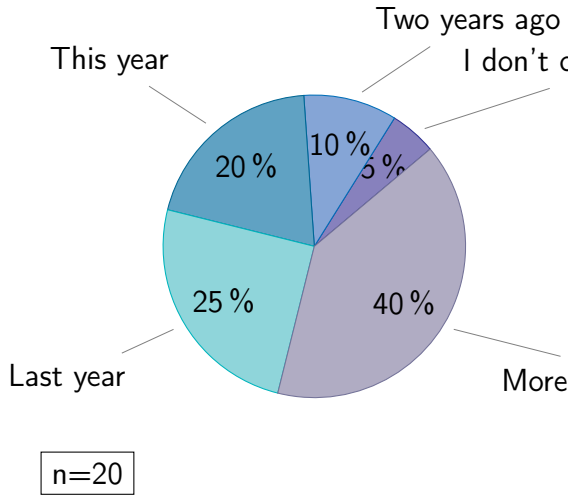

Figure 5.3: Last smart-phone purchase from survey

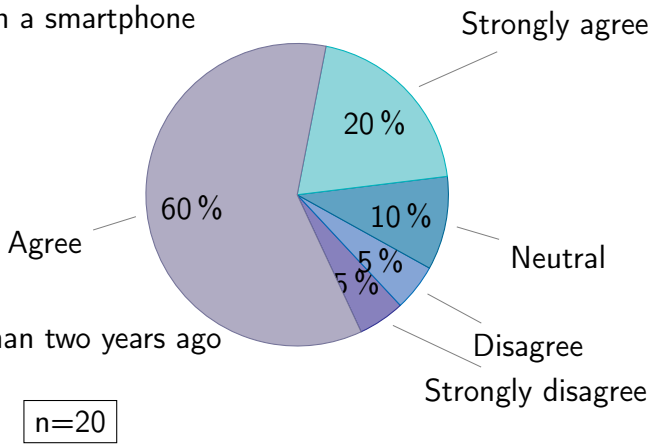

Figure 5.4: Feeling of safety from survey

One of our assumptions was that an application like SFCV not only needs to be secure, but needs to feel secure, and that this would be relevant to the overall user experience. To gain more insight into what the potential users feel about on-line security and privacy, as shown in figure 5.4 they were asked whether or not they feel safe uploading documents to an on-line service if they are guaranteed that their documents will not be used or shared with anybody without their consent.

Within the applications of SFCV, what are the most important features for a user? As we can only assume that the needs of the potential users had been identified prior to our participation in the SFCV project, it was interesting for us and for the development and design process of the application to identify what the potential users felt was the most important feature for

them. Figure 5.7 shows that 60% of the users stated that storing their documents was the most important feature, while 30% answered that the verifying of the documents was the most important feature. We can only assume that the numbers would be reversed if the survey was aimed at employers using the application rather than seafarers. Data like this can be used to confirm or disprove assumptions of user needs made earlier, and can help us with prioritizations and delege focus for the correct user group, as well as tell us about the potential expectations of users.

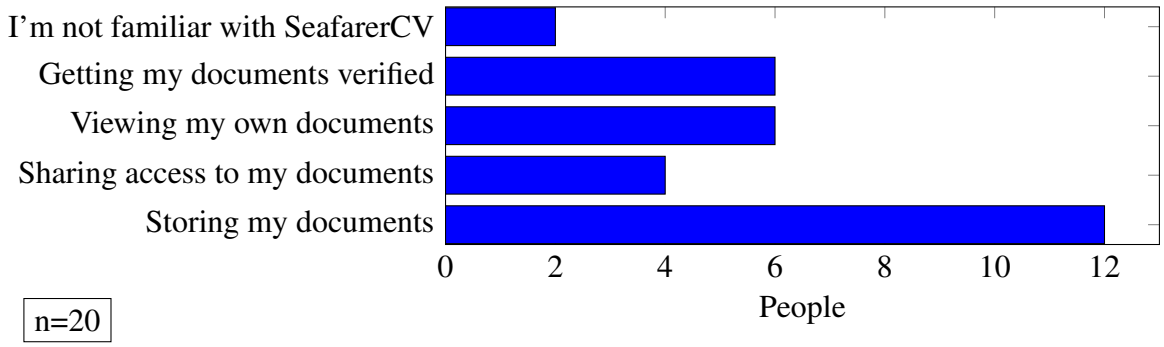

Figure 5.7: Important features from survey

5.2 Usability Test

Figure 5.1 shows how to the results for the tasks given to the participants as well as their age, gender, plattform, and where we would rank them in technology proficiency. The results show us that most of the participants were able to solve the task in given to them. It also shows in what areas people needed help to progress. This uncovers potential weaknesses of the interaction design. We were not able to identify any result where age, gender or level of technology proficiency affected their ability to use the application.

Person	Tech proficiency	Platform	Sign up	Log in	Locate ID	Camera upload	File upload	Accept request	Revoke access	Locate QR	View document
Male, 34	Medium	iOS	✓	✓	✓	✓	✓*	✓	✓	✗	✓
Male, 27	High	iOS	✓	✓	✓	✓	✓*	✓	✓	✓	✓
Female, 28	Low	iOS	✓	✓	✓	✓	✓*	✓	✓	✗	✓
Male, 48	High	iOS	✓	✓	✓	✓	✓*	✓	✓	✗	✓
Male, 18	High	Android	✓	✓	✓	✗	✗	✓	✓	✗	✓*
Male, 28	Medium	Android	✓*	✓	✓	✓*	✓	✓*	✓	✓	✓

* With help

Table 5.1: Usability test results

5.3 Observation

From our direct observations of the participant's interaction with the mobile application, there were several finds. Many of the participants struggled with the password criteria, and because of this had to use several attempts in order to sign up. About half of the participants had to ask or think out loud about the definition of a non-alphanumeric character. How the participants interacted with the mobile application on a first-time log-in was interesting to observe, as many of the participants chose to explore in the application, to get familiar with its content. This was reflected in how the participants were able to perform the coming tasks. All participants quickly found the sign-up link on the sign-in page, and clicked it when is asked to sign up. Not all participants found the correct country when signing in. Some would scroll through the list while others would start typing the name of the country they were searching for. With a few exceptions, the participants located the correct page in the mobile application relevant for the given task on the first try. The plus button on certificate page seemed to be understandable and intuitive for all participants, as nobody hesitated or questioned this when they added a new document on the certificates page. Issues with bugs and errors in the application resulted both in the participants thinking he or she had done something wrong, or that the participants themselves concluded that there is something wrong with the application, and restarted it. From our observations it was noticed that many of the participants waited for some form of notification when they were asked to answer a request. This is intended, but was not implemented at the time of testing. Some of the participants visited the profile page in search of this. After some time, they went looking for the request in the notification page, and thus able to solve the given task. Observation also lead to us uncovering more bugs and system errors within the application. We also noticed that some of the participants got their confirmation mail categorised as spam. The Android version had several bugs that clearly affected the UX with the application, in some cases preventing the participant to be able to solve given task in the usability test. Receiving error messages when trying to upload documents and not able to view uploaded documents are examples of these bugs and system errors.

5.4 Interview

For the most part, the participants stated that they knew what to do in the application, with the exception being the locating of their QR code, as well as some confusion about the password criterias when signing up. Some of the participants felt that some form of notification was missing when asked to answer an access request, which caused them to wait unnecessarily, which confirmed our observation. The participant that answered that he did not always know what to do, still felt that the applications was easy to understand and navigate. Every participant answered that the icons were understandable. Some of the participants questioned the icon used on certificates, and stated that they possibly would not understand the icon had it been on its own. The participants answered for the most part that the application seemed like a good solution and was easy to use. From some of the participants, it was mentioned that the application felt purposeful and straight forward and that it did not suffer from an overflow of too much information or functionality. One participant that had some difficulties with the tasks stated that despite the fact that he did not always know what to do, he still felt secure in the knowledge that he could not do something destructive in the application. Most participants mentioned the placement of the QR-code when asked if there were anything they would change in the application. It was further suggested that there should be some symbol or indication that the QR-code is located on the profile picture. On of the participants thought that he possibly would had guessed this location if he had had his own profile picture on the profile page. It also mentioned by one of the participants that he would preferred som alert or sound to confirm his actions, specifically when confirming access to a company. It was also suggested by a participant that there should be instructions on how to upload documents, and that it would be help with a short set of instruction or a tutorial on first use. Some of the participants stated that the application reminded them of Facebook's application. This is possibly due to the use of the colors and icons. One participant felt that the function for uploading documents reminded him of dropbox. For some, even though the application did not remind them of any other application in particular, it still felt familiar. The reoccuring answer when asked about what they liked about the application, was the simplicity of the application. When the participants were asked what they didn't like about the application, the placement of the QR-code was stated by multiple participants, as well as the fact that there was not any way to upload a profile picture or customize their profile. To them this made the profile page seems unnecessary and boring. In addition, the fact that there was not any way to edit or crop the image when uploading a document with the use of the camere function, made it hard to use and to get a good copy of their document stored in the application.

6. Results and Discussion

Although our research questions are largely divided with respect to theoretical fields of research, they were heavily connected in the practical project period of our thesis. Because of this, each research question will first be attempted answered individually. Throughout, the discussion will explore how the results of one research question can be connected to others, and attempt to form a picture of the overall result of the project period, in which all of the research questions are based.

6.1 SeafarerID Implementation

During the development of the resource described in chapter 2, an authentication and authorization service was created. This service is to be used with all current and future internal services, as well as to be a OpenID provider for third-party services. This was done both at the request of SFCV and to answer our research question of how a SeafarerID can be implemented as a single authentication service for all internal and external SFCV services. To assess the result of this implementation process, as well as the resulting functionality, the choices taken must be compared to research and theory in the relevant fields.

As described in chapter 2.1, the resource was planned as many different microservices, one of which was the authentication and authorization service. In chapter 3.4.2, this microservice architecture is described. This, together with the related DevOps philosophy, allows applications to be developed as small independent services. The DevOps approach also minimizes coordination required between teams responsible for different components, as well as breaking barries between development and operations teams. Despite the fact that these concepts both primarily concern themselves with team organization, they still hold relevancy in this project where only one developer was responsible for the technical implementation. By deciding on the microservice architecture early in development, it simplified the development process by defining independent services that could be developed without deeply affecting each other. This saved both development time and potential issues regarding things such as dependency management and bugs. Similarly, the DevOps philosophy became highly relevant with only one developer, as this one developer was responsible for both the development and the server set up and operations of the services.

Although the services were developed with the microservice architecture in mind, there were some concession that were made in the development process. For instance, purists would argue that the single database is currently too connected with both the authentication and authorization service and the API service. This could arguably be resolved by splitting the database into two independent databases, where one database is used by and contains the data of the authentication and authorization service, and the other is used by and contains the data of the API. Although this would greater satisfy the microservice architecture philosophy, it could also result in a more complicated development process, which is the reason this was not focused upon in the creation of the resource.

As mentioned in chapter 2.2, the IdentityServer4 framework allowed us to implement the OAuth 2.0 and OpenID Connect protocols together with the SSO functionality described in chapter 3.5.2 quickly, securely and without major development issues. Although the necessary core functionality is in place in the finished resource, there are certain conveniences that were not implemented due to the limited time available. One such missing convenience is a UI for management of OAuth 2.0 clients. This means that currently all clients are defined and compiled along with the binaries of the application, meaning the entire application must be recompiled and republished every time a client is modified. Although this was not an issue in the development period due to frequent recompilation and republishing, it would cause unnecessary risk and complexity in a live production environment.

Although the GDPR compliance and PbD philosophy described in chapter 3.5.1 was emphasized by the stakeholders of SFCV before the initiation of the project, this became less of a focus from their point of view throughout the development process. While the resource, including the authentication and authorization service, was designed with this compliance in mind for the future, it can in no way be described as GDPR compliant. The reason for this from the development point of view was the target state of the resource as a beta-ready testing resource. Because of the target state being beta-ready, this was not focused upon as much as would be needed if there was to be a full release by the end of the development period. This was undermined half-way through the development process by the decision of SFCV to elevate the target of the resource from beta-ready to release-ready, and to immediately at the end of the process release the resource to the public, as well as releasing outsourced mobile applications using the unfinished and vulnerable authentication and authorization service. This was argued by us to present significant security and privacy vulnerabilities, both in general as well as relating to the GDPR. Despite this, SFCV decided to release these resources regardless of our concerns. In contrast to this, we would ideally have liked to see a commitment to delay this release until security and privacy vulnerabilities could be addressed, preferably through a

process similar to PbD as described in chapter 3.5.1.

Throughout the development there was some conflict between us and the stakeholders of SFCV with regard to the technicalities of the implementation of the services. In order to present our views on how to proceed clearly, a document that can be found in appendix A was produced defining our suggestions as to how to tackle certain issues that had arose. Eventually, the points argued in this document was agreed upon, and progress was continued on the resource development.

The resulting resource of the development process provides the desired features of serving as a SSO service providing a single point of authentication and authorization for all internal and external services of SFCV. It also acts as an OpenID provider, allowing other websites and services to allow users to sign in using their SFCV credentials. Although most of the planned functionality is in place, the security and privacy of the service is only suitable as a beta-ready resource. This leads us to question the wisdom of releasing the service to the public, and ultimately allowing real user data to be stored on a service that is both unsecure and vulnerable.

6.2 Human Centered Design Process

The human centered design process has been successful in that sense that we were able to do one cycle in the process as shown in figure 4.1, and that potential end users has been involved. There is now a better understanding of the potential users, in which further design can be based upon. There is also understanding of how people interact with the mobile application. The usability test and the following interviews, gave us useful data that can be used to improve upon the design, confirming and disconfirming assumptions of how the functions in the application should work and be used. Based on the result from the interviews and observations of people interacting with the system, the mobile application seems like a system and a design solution people would use and enjoy to use, hinting at potential good user experiences in a fully functional application. We consider that the design resides in the process, where the design needs to be further evaluated against known requirements, and that there should be further work with identifying requirements and needs for human centered design.

From our observation of the participants in the usability test, we learned more than merely whether or not the participants were able to perform given tasks or not as table 5.1 shows. The password criteria is a security measure that will not be changed in the beta-resource, but we can hypothesize that if there was an example of non alphanumeric characters in the description, the sign-up process would have been more pleasant and less time-consuming. The participants

who made themselves more familiar with the application seemed to have a better mental model of the application, and seemed to understand and solve the task in less time. This can possibly tell us that the application has good first-time learnability and ease of use. A further example of this was when one of the participants chose to use the mobile camera when asked to upload a given paper document and then subsequently tried to upload the file, which was the goal of a task given to the participants later in the usability test. When the participants again was asked to upload the document, the camera functionality had already been discovered within the application, and was chosen for the task at hand.

The version of the application used by the participants in the usability test had som limitation and some bugs. Observing their interaction gave us the possibility to hypothesize that if some of the lacking functionality were working as intended, the participant would have solved certain tasks more effectively. It could also have eliminated confusion when interacting with the application. This also emphasizes how important it is to have minimal system errors and bugs in an application.

It became apparent early in the planning phase of the project that the HCD process was perhaps not ideal for this project. It turned out to be challenging to fully incorporate the process in the project, due to the structure and limitations within the project as described in chapter 1.4. Initially in the planning phase, a document was made and uploaded to our main communication channel with SFCV. The topic of the document was UX strategy, branding, defining user needs and user research. This was done as an attempt to start the process with HCD early and with a hope that this document could function as a on-line focus group. This document was either ignored or overlooked by SFCV, resulting in us having limited understanding of requirements and needs from SFCV as a stakeholder. In addition, it resulted in us not being able to communicate the importance of these subject matters early on in the project. There seemed to be no planing of anything that could resemble an application development life cycle, nor any interest in spending time on testing and developing design for the application before starting the coding phase of the development process. This resulted in a highly asynchronous development process, with limited communications between the external developers of the mobile application and the Master students. As a consequence, we needed to wait for the sign-up page to be developed, explained in 2.2 to distribute our user survey. Because SFCV wished to start the coding phase early, we were required to start developing prototypes based on assumptions and previous experience, as explained in 2.2. A method more similar to the LeanUX approach described in chapter 3.4.1 than a traditional HCD process.

6.3 Project Success

Whether or not SFCV will be accepted and used by its intended users, in other words successful, can possibly be determined by how the system and applications meets the key features of acceptability mentioned in 3.3. The systems have not been introduced for simple economic reasons, but will possibly affect certain hiring processes and structures which can in turn affect acceptability in the community that is targeted. There can be many economic reasons for a system to be acceptable or not, and price is the obvious one (Benyon, 2014). SFCV is a free application that should never cost anything for the seafarer, and from our understanding will also save the seafarer money, due to the potential cost related to copying and verifying documents. Convenience associated with the application can be a strong factor with SFCV's acceptability. This is also closely connected with the design of an application, as a design should fit effortly into the situation (Benyon, 2014). There are however people who still find that paper documents is more convenient to carry and read, preferring physical documents over digital content. One of the key features mentioned by Benyon is usefulness, stating that it goes beyond the notions of efficiency and effectiveness and concerns usefulness in context, as something can be usable but not sufficiently useful in its context (Benyon, 2014). This is naturally difficult to answer from our standpoint. In our survey, we asked the potential users whether or not they feel there is a need for an application like SFCV. 45% of the respondents agreed that there was a need while 35% answered maybe. The raiming 20% answered that they were not yet familiar with SFCV. Data like this can say something about the perceived usefulness of SFCV from the user's perspective. Perceived usefulness together with perceived ease of use are the two external variables that affects the attitude towards using a technology and the behavioral intention to use it, according to the technology acceptance model illustrated in figure 3.2. The answers that were given by the potential users, and the fact that they are participating in a survey, can possibly illustrate that the people who replied to our survey, show a perceived usefulness towards the application. The unified theory of acceptance and use of the technology model shows that external factors such as performance expectancy, effort expectancy, social influence and facilitating conditions, are all affected by the gender, age, experience and voluntariness of use of the potential users.

At present, the application and system is completely voluntary. However, with the backing of ITF and the possible implementation as SeafarerID as a standard authentication method as discussed in chapter 6.1, the feeling of voluntariness could change. The respondents to our survey appear to have previous experience both with digital solutions and with seafarer job and verification services. With a fully functional system and product, users can possibly find it

convenient and useful in its context. This, together with the potential economic advantages, SFCV could be argued to score high in acceptance, along with a high perceived ease of use based on the findings in chapter 5.

The mobile application developed by the external developers were developed as an hybrid mobile application as described in chapter 3.6.1. In their study, Hansson and Vidhall concluded that in their case as much as 75% of the application logic could be shared between platforms (Hansson & Vidhall, 2016). This can be a major advantage, and could be a significant factor with regard to the eventual success of the development process and the resulting mobile applications.

As stated by Ruparelia, the life cycle of a development process lasts from its inception to its completion and maintenance (Ruparelia, 2010). In the theory of chapter 3.4.3, several models of a development life cycle are described, including their uses and disadvantages. These models must be compared with the actual development process of the resource described in chapter 2 in order to measure the success of the development process, and in turn the potential success of the application based on the theory of chapter 3.6.

For the success of a project, several studies suggest that proper use of project management tools can be important (Patanakul, Iewwongcharoen, & Milosevic, 2010). In addition, as mentioned in chapter 3.6.2, the project schedule and planning is one of the key factors for the success of a project life cycle. In the planning phase of the resource development described in chapter 2.1, the work done previously on the prototype was examined. Although it is not known to what extent a model was followed in this process, one can assume that it followed something similar to the Rapid Application Development model described in chapter 3.4.3 for rapid prototyping. When our involvement with the project started, we intended to follow a modal similar to the one illustrated in figure 3.5. The reason for this was the wish of SFCV to build a beta-ready resource that would have the ability to be further developed to a finished product, thus requiring some further planning and analysis than a simple prototype. This model would also have to be shortened and reduced due to our short development period.

Due to the existing prototype, it was assumed that the system requirements and design requirements had already been given some consideration before our involvement in the process. We later discovered that this was only partly the case, which would later lead both to drastic change of direction of the development as well as minor conflicts between us as the developers and SFCV as the stakeholders. One such conflict arose from the sudden change in direction as described in chapter 6.1, namely to change the target state of the resource from beta-ready to release-ready. This change significantly altered the first critical factor for success as described in chapter 3.6.2, namely the project mission.

In the model described in figure 3.5, program design was in reality the first step we

undertook at the beginning of our involvement with the process. This process is explored in detail in chapter 2.1 and the result is illustrated in figure 2.1. Although this period was merely a week in length, it was an important period for the remainder of the development process. The next step in the model as well as in our development process was coding. This process was conducted throughout the remainder of the project period, with occasional regression to the program design process when necessary. Operations was a process that was started for the purposes of beta testing, but not a process that was followed through to completion in any meaningful way. Testing was a process that remained largely untouched, due both to the fact that this would be performed later before the final release, as well as the fact that Bentsen and Holte would be performing security testing on the authentication and authorization resource.

As can be seen, some semblance of the waterfall model of figure 3.5 was followed, although in a disconnected and often disorganized way. Much of this can be attributed to the short development period, while some can also be attributed to the change of direction and change of target from beta to release for the resource as described earlier in this chapter.

7. Conclusion and Further Work

The goal of this thesis was to take a previously developed prototype through a development process to reach beta-ready functionality. During and after this process, we would attempt to answer the research questions stated in chapter 1.2. The first question was how SeafarerID could be implemented as a single authentication service for all internal and external SFCV services, with the second being to what degree HCD-based methodology could be followed in the development of an outsourced mobile application.

With regard to the first research question, we would conclude that the result at the end of the process described in chapter 6.1 satisfies most of the functionality desired by the stakeholders of SFCV at the beginning of the project as well as what is stated in the research question. Despite this, there are some concerns and related future work with regard to this functionality. While this resource meets the targets specified at the beginning of the development period, these targets were shifted during the development process from being a beta-ready resource to being release-ready. Because of this, the resource does not meet the needs of the current target with regard to privacy and security. Therefore, further work is needed on both securing the production environment of the service, as well as ensuring the system is compliant with the GDPR. In addition, further work must be done on convenient functionality such as identity management and OpenID Connect client management for it to function on a large scale as a stable SSO identity service.

While the conclusion of first research question is based mostly on the resulting resource and its functionality, the second research question involved questionnaires and interviews to gather relevant data. This data was used in chapter 6.2 to look at the HCD process undertaken, if any, by the external team developing the mobile application, and see whether or not the data reflects the process compared to the theory of chapter 3.1. In conclusion, although the HCD process was initiated by us during the development of the resource as described in chapter 2.2, this process was not undertaken to a significant degree by the external team of the project. Despite this, our findings presented in chapter 5.2 and table 5.1 show that the mobile application could largely be operated successfully by most participants, barring some usability issues and crashing. An interesting finding as shown in figure 5.7 was that despite the perceived focus of SFCV being on the verification of seafarer documents, twice as many people stated the simple storage of documents was the most important feature of the application. Because of no HCD methodology being followed in the application development, this can be one example of how priority of

development could have been focused differently had the proper procedures been followed in the correct order, resulting in a product that was more developed for its intended audience and a design that was more human-centered.In conclusion, the lack of a HCD-based procedure in the outsourced development does little to definitely answer the research question of this thesis. However, we deem this a result of a rushed and mismanaged development strategy on the part of SFCV, as despite our initiation of this HCD methodology early in development, it was not a focus of SFCV to proceed with this with the team of the outsourced mobile application.

Through our discussion and theory on the subject matter in research question three, we have looked at known criteria for successful applications, and what metrics developers should measure to evaluate their potential application success. As demonstrated in chapter 6.3, it could be suggested that users felt a perceived usefulness towards the solution. This, along with other concepts, can influence how technology will be accepted in a community, and can be significant for the success of a product. The development process itself can also have a big impact on potential success. As discussed in chapter 6.3, a project life-cycle model was followed to a certain extent, but with some flaws. The results of these flaws can also be observed in the HCD process discussed in chapter 6.2, where a lack of a defined process resulted in a disorganized development period for the external developers with little to no theoretical foundations beyond the High-fidelity prototype resource discussed in chapter 2. For the continuation of the development process, we would suggest more closely following a development life-cycle model as described in chapter 3.4.3, as well as more careful consideration of potential critical success factors such as in the ones described in chapter 3.6.2, as concepts such as these seem to have a significant impact on the final result of a project.

Our experience regarding the overall development of the resources described in chapter 2 was that SFCV wished to start the coding phase of the mobile application as soon as possible. From our observation there was no interest from SFCV to await design based on knowledge of the end users and results from usability testing on a High-fidelity prototype. Because of this, the design is largely based on assumptions on what we considered known interactions patterns from known popular applications. Examples of this are that the uploading of documents was based on how you would upload a document on Google Disk, the camera functionality can visually resemble Snapchat, and that the overall design with the notifications and profile pages can resemble multiple social media applications. We could assume that the potential users were familiar with the interaction in applications such as these. The result is a design based merely on assumptions, considered by us to be a resource for usability testing to confirm or disprove said assumptions. From our experience with the project, and in a project where larger parts of the development is being performed by external developers, we would propose to use

alternative ways of implementation and development of design and interaction design than the traditional HCD methodology. The way the process worked in our project could be similar to the concept of lean UX described in chapter 3.4.1, in that the design was based on assumptions and then subsequently tested to see if our assumptions where true or false. However, we believe it was performed on too large a part of the application and should therefore have been divided into smaller parts that could be tested along the way, during development.

Although most of the internal and outsourced resources discussed in this thesis are now released to the public, it is clear to us that much work still remain on this project for it to reach a stage ready for the public state it now inhabits. Although the research of Bentsen and Holte on the resources created for this thesis suggest that the software implementations are solid, as mentioned above and by Bentsen and Holte the production environment that is currently used is only meant as a beta-testing platform (Bentsen & Holte, 2018). Because of this, it is safe to assume that vulnerabilities and insecurities of the developed back-end production environment, as well as the fragile and buggy front-end applications, will in our opinion at best detract potential customers, and at worst result in data breaches and potential catastrophic data theft and loss.

REFERENCES

Abras, C., Maloney-Krichmar, D., & Preece, J. (2004). User-centered design. In W. S. Bainbridge (Ed.), (Vol. 37, pp. 445–456). Sage Publications. Retrieved from http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.94.381&rep=rep1&type=pdf

appdynamics. (2015). *16 metrics to ensure mobile app success.* Retrieved 20.05.2018, from https://www.appdynamics.com/media/uploaded-files/1432066155/white-paper-16-metrics-every-mobile-team-should-monitor.pdf

Balalaie, A., Heydarnoori, A., & Jamshidi, P. (2016). Microservices architecture enables devops: Migration to a cloud-native architecture. *IEEE Software*, *33*(3), 42–52. Retrieved from http://ieeexplore.ieee.org/abstract/document/7436659/

Bentsen, E. E., & Holte, W. (2018). Seafarercv penetration test.

Benyon, D. (2014). *Designing interactive systems: A comprehensive guide to hci, ux and interaction design, 3/e.* United Kingdom: Pearson.

Breach Level Index. (2018). *Breach level index.* Retrieved 20.03.2018, from http://breachlevelindex.com

Cavoukian, A. (2009). Privacy by design. *Take the challenge. Information and privacy commissioner of Ontario, Canada*. Retrieved from https://www.ipc.on.ca/wp-content/uploads/Resources/pbd-implement-7found-principles.pdf

Cavoukian, A. (2012a). Operationalizing privacy by design: A guide to implementing strong privacy practices. *Information and Privacy Commissioner, Ontario, Canada*. Retrieved from http://www.cil.cnrs.fr/CIL/IMG/pdf/operationalizing-pbd-guide.pdf

Cavoukian, A. (2012b). Privacy by design [leading edge]. *IEEE Technology and Society Magazine*, *31*(4), 18–19. Retrieved from http://ieeexplore.ieee.org/abstract/document/6387956/

Charland, A., & Leroux, B. (2011). Mobile application development: web vs. native. *Communications of the ACM*, *54*(5), 49–53. Retrieved from https://dl.acm.org/citation.cfm?id=1941504

Davis, F. D. (1989). Perceived usefulness, perceived ease of use, and user acceptance of information technology. *MIS quarterly*, 319–340. Retrieved from https://www.jstor.org/stable/249008

De Clercq, J. (2002). Single sign-on architectures. In *Infrastructure security* (pp. 40–58). Springer. Retrieved from https://link.springer.com/chapter/10.1007/3-540-45831-X_4

Dmitry, N., & Manfred, S.-S. (2014). On micro-services architecture. *International Journal of Open Information Technologies*, *2*(9). Retrieved from https://cyberleninka.ru/article/n/on-micro-services-architecture

Dyck, A., Penners, R., & Lichter, H. (2015). Towards definitions for release engineering and devops. In *Release engineering (releng), 2015 ieee/acm 3rd international workshop on* (pp. 3–3). Retrieved from http://ieeexplore.ieee.org/abstract/document/7169442/

Glass, S., Hiller, T., Jacobs, S., & Perkins, C. (2000, October). *Mobile ip authentication, authorization, and accounting requirements* (RFC No. 6749). RFC Editor. Internet Requests for Comments. Retrieved from https://www.rfc-editor.org/info/rfc2977

Gothelf, J. (2013). *Lean ux: Applying lean principles to improve user experience*. O'Reilly Media, Inc.

A guide to the project management body of knowledge (pmbok guide). (2008). Newtown Square, Pa: Project Management Institute.

Hansson, N., & Vidhall, T. (2016). *Effects on performance and usability for cross-platform application development using react native.* Retrieved from http://www.diva-portal.org/smash/record.jsf?pid=diva2%3A946127&dswid=-8962

Highsmith, J., & Cockburn, A. (2001). Agile software development: The business of innovation. *Computer*, *34*(9), 120–127. Retrieved from https://ieeexplore.ieee.org/abstract/document/947100/

Hursti, J. (1997). Single sign-on. In *Proc. helsinki uiniversity of technology seminar on network security.* Retrieved from http://www.tml.tkk.fi/Opinnot/Tik-110.501/1997/single_sign-on.html

International Transport Workers' Federation. (2018). *About itf.* Retrieved 12.04.2018, from http://www.itfglobal.org/en/about-itf/

Ionic Framework. (2017). *2017 developer survey.* Retrieved 25.04.2018, from https://ionicframework.com/survey/2017

ISO. (2010, March). *Human-centred design for interactive systems* (Standard No. 9241-210:2010). International Organization for Standardization. Retrieved from https://www.iso.org/standard/52075.html

ITF Seafarers. (2018). *Women seafarers.* Retrieved 20.04.2018, from http://www.itfseafarers.org/ITI-women-seafarers.cfm

Johnson, R. B., & Onwuegbuzie, A. J. (2004). Mixed methods research: A research paradigm whose time has come. *Educational researcher*, *33*(7), 14–26. Retrieved from http://journals.sagepub.com/doi/abs/10.3102/0013189X033007014

Lazar, J., Feng, J. H., & Hochheiser, H. (2017). *Research methods in human-computer interaction*. Morgan Kaufmann.

Localytics. (2016). *Overall App Benchmarks H2 2016.* Retrieved 15.04.2018, from https://www.localytics.com/lp/cheat-sheet-overall-app-benchmarks-h2-2016/

Malavolta, I., Ruberto, S., Soru, T., & Terragni, V. (2015). Hybrid mobile apps in the google play store: An exploratory investigation. In *Proceedings of the second acm international conference on mobile software engineering and systems* (pp. 56–59). Retrieved from https://dl.acm.org/citation.cfm?id=2825051

Mladenov, V., Mainka, C., Krautwald, J., Feldmann, F., & Schwenk, J. (2015). n the security of modern single sign-on protocols – openid connect 1.0. *CoRR abs/1508.04324*. Retrieved from https://pdfs.semanticscholar.org/60bf/6404ebc12a97e45290ad63b3c66176ed5ad4.pdf

MobiDev. (2016). *11 key features of a successful mobile app.* Retrieved 10.04.2018, from https://mobidev.biz/blog/11_key_features_of_a_successful_mobile_app

Patanakul, P., Iewwongcharoen, B., & Milosevic, D. (2010). An empirical study on the use of project management tools and techniques across project life-cycle and their impact on project success. *Journal of General Management*, *35*(3), 41–66.

Pinto, J. K., & Slevin, D. P. (1988). Critical success factors across the project life cycle.. Retrieved from https://www.pmi.org/learning/library/critical-success-factors-project-life-cycle-2131

Regulation (EU) 2016/679 of the European Parliament and of the Council of 27 April 2016 on the protection of natural persons with regard to the processing of personal data and on the free movement of such data, and repealing Directive 95/46/EC (General Data Protection Regulation). (2016, May 4). *Official Journal of the European Union*, *L119*, 1–88. Retrieved from http://eur-lex.europa.eu/legal-content/EN/TXT/?uri=OJ:L:2016:119:TOC

Rogers, Y., Sharp, H., & Preece, J. (2016). *Interaction design: beyond human-computer interaction fourth edition*. Wiley.

Royce, W. W. (1987). Managing the development of large software systems: concepts and techniques. In *Proceedings of the 9th international conference on software engineering* (pp. 328–338). Retrieved from http://leadinganswers.typepad.com/leading_answers/files/original_waterfall_paper_winston_royce.pdf

Rubinstein, I. S., & Good, N. (2013). Privacy by design: A counterfactual analysis of google and facebook privacy incidents. *Berkeley Tech. LJ*, *28*, 1333. Retrieved from http://heinonline.org/HOL/LandingPage?handle=hein.journals/berktech28&div=40&id=&page=

Ruparelia, N. B. (2010). Software development lifecycle models. *ACM SIGSOFT Software Engineering Notes*, *35*(3), 8–13. Retrieved from https://dl.acm.org/citation.cfm?id=1764814

Sakimura, N., Jones, M., de Medeiros, B., & Mortimore, C. (2013, November). Openid connect basic client profile 1.0 - draft 26. *OpenID Connect Specs http://openid. net/connect*. Retrieved from https://openid.net/specs/openid-connect-basic-1_0-26.html

SeafarerCV. (2018). *Seafarercv.org.* Retrieved 09.05.2018, from https://seafarercv.org/

Statista. (2017). *Mobile app usage.* Retrieved 15.04.2018, from https://www.statista.com/topics/1002/mobile-app-usage/

Sun, S.-T., & Beznosov, K. (2012). The devil is in the (implementation) details: an empirical analysis of oauth sso systems. In *Proceedings of the 2012 acm conference on computer and communications security* (pp. 378–390). Retrieved from https://dl.acm.org/citation.cfm?id=2382238

Venkatesh, V., Morris, M. G., Davis, G. B., & Davis, F. D. (2003). User acceptance of information technology: Toward a unified view. *MIS quarterly*, 425–478. Retrieved from https://www.jstor.org/stable/30036540

Wasserman, A. I. (2010). Software engineering issues for mobile application development. In *Proceedings of the fse/sdp workshop on future of software engineering research* (pp. 397–400). Retrieved from https://dl.acm.org/citation.cfm?id=1882443

Yauch, C. A., & Steudel, H. J. (2003). Complementary use of qualitative and quantitative cultural assessment methods. *Organizational Research Methods*, *6*(4), 465–481. Retrieved from http://journals.sagepub.com/doi/abs/10.1177/1094428103257362

Zhang, Y. (2015). *16 mobile mistakes that plummet user retention rates.* Retrieved 14.04.2018, from https://apptimize.com/blog/2015/10/16-mobile-mistakes-that-plummet-user-retention-rates/

APPENDICES

Appendix A: Technical reflections by master's students regarding SeafarerCV's plans

We, the master's students, believe in the concept of SeafarerCV. We have considered the plans and concepts of how this service is planned to function in detail, to then be able to provide a detailed overview concerning our thoughts regarding the technical implementations based on these ideas

Below follows our thoughts concerning each topic that we have reviewed, as well as our suggestions on how to improve these solutions, as well as alternative approaches

Identity verification

In an ideal world this would be something to strive for, after careful review however we conclude that to attempt to guarantee the validity of an identity or user account is not desirable, based on the following reasons

- Passports can be faked, and unless these documents are screened by an official government service or similar, one can not be sure of the document validity
- Even though some select countries have services rendering such a screening processes possible, they are vastly outnumbered by those who do not, especially in many of the regions SeafarerCV are targeting
- It creates a false sense of security concerning our systems. By conversely assuming that any user account could be fake, which in reality is possible whether or not we try to check passports or not, then we can strengthen the rest of the system based on that

In addition, we also believe that a guarantee of user account validity is unnecessary in SeafarerCV, based on the following core service principles

- The seafarer will directly provide an employer with his or her ID. Because of this concept, fake user accounts will be invisible both to other users and employers

- Related to the previous point; SeafarerCV is not meant as a recruitment service, and fake user accounts will therefore have no negative effect on the rest of the system
- It increases the effort and difficulty for the users, and in this case effort that we think is unnecessary

Based on this we recommend a different approach than attempting to guarantee user account validity, and instead shift the focus over to increase security in other areas based on the assumption that some user accounts may be fake. It can also be observed that as a consequence of many of the points listed here, there are very few services on the internet attempting to guarantee user identity, with the exceptions being some national services in countries where such screening is available as well as some who check documents manually

Gathering of documents

Even though we understand the thought behind the idea, we have spotted a couple of issues regarding this

- As a result of the insecurities concerning identity, regardless if one decides to try to screen passports or not, it is problematic to pair documents gathered from external sources to user accounts in our system
  - With a large number of documents in numerous different formats, it is a doubtful prospect to trust optical character recognition to pair documents with user accounts based on names found in each document
  - If an attacker manages to create a user account in our system with a fake name, he or she now has access to all gathered documents of the victim, which obviously creates massive privacy concerns
- None of the gathered documents can be thought of as verified, as they are gathered from unsafe sources, not from document issuers themselves

This leads us to believe this is too problematic to pursue, as well as being superficial to the services SeafarerCV aims to provide

User roles

When we were presented with the idea, it emerged that each user account should have one role, and that in order for a user to carry out multiple roles, he or she must have several accounts. We suggest an alternative approach based on the following

- By having multiple user accounts, the user no longer has an identity, which becomes problematic with SeafarerCV as an OpenID-provider ("Log in as SeafarerCV")
- Related to the previous point; we as a service can no longer deal with one person as one identity, but must go through multiple potential user accounts that may or may not be the one the user considers their primary one
- The user is required to remember more user names and passwords than necessary

Fortunately, there are many who have walked this path before us, and there are therefore many examples of how this can be solved in a more elegant way. In our case the solution will look similar to the following

- Each user has one singular user account (identity) in SeafarerCV
- Each user account can be the owner, administrator or user of one or more employers
- Each user is able to store documents regardless of whether or not he or she is involved with employers

As a result of this approach, roles are not defined by your user account, but rather by the memberships you have to employers

An example of a service that has a very similar approach is GitHub. Each GitHub user can own, administrate or be a member of one or more organizations, without affecting how the rest of the service functions. As an example of the consequences of this, Facebook CEO Mark Zuckerberg has the exact same kind of GitHub user account as anyone else, yet he is presumably an administrator of the Facebook organization on GitHub, and therefore has roles according to his membership.

Most other large services on the internet today have similar account principles, for instance Google and Facebook.

Conclusion

Based on the reflections above, we have the following suggestions

- Move away from attempting to guarantee user account validity, but instead shift the focus to strengthen the security of the rest of the service
- Focus on verifying employers rather than users
- Move away from gathering documents from unsafe sources in advance to pair with user accounts
- Instead of defining the user's roles in advance, let it be defined by its membership to employers

Ronny Reinhardtsen Nikolai Robstad

Appendix B: Tasks and Interview

Tasks for the participant

- Sign-up
- Login
- Can you tell me your SeafarerID (send him or her an access request)
- Upload document
  - Use the camera
  - From phone hd
- Can you locate the QR code
- Accept a access request
- Remove access to “blank” company
- View one of your documents

Questions after test interview

- Did you always knew what do?
- Were the icons used understandable?
- Does the application feel like a nice solution and was it pleasant to use?
- Can you think about anything you would change in this application?
- Does the application remind you of any other applications?
- What did you like about the application?
- What didn’t you like about the application?

Appendix C: Usability Test Results

Participant number:	1
Gender	male/~~female/other~~
Level of technology proficiency	~~low~~/medium/~~high~~
Platform	iOS/~~Android~~
Age:	34

Question	Yes	No	With help	Time
Was the participant able to sign-up?	✓			
Was the participant able to log in to his or her account?	✓			
Was the participant able to upload a paper document, with use of the camera?	✓			
Was the participant able to upload a "document / image" from their smartphone?			✓	
Was the participant able to locate and accept an access request?	✓			

Was the participant able to locate and remove access from a company?	X 3			
Was the participant able to locate his or her QR code?		X	X	
Was the participant able to locate their documents and view them?	X			

Participant number:	2
Gender	male/~~female/other~~
Level of technology proficiency	~~low/medium~~/high
Platform	iOS/~~Android~~
Age:	27

Question	Yes	No	With help	Time
Was the participant able to sign-up?	X			
Was the participant able to log in to his or her account?	X			
Was the participant able to upload a paper document, with use of the camera?	X			
Was the participant able to upload a "document / image" from their smartphone?	X2			
Was the participant able to locate and accept an access request?	X			

2

	locat yes			
Was the participant able to locate and remove access from a company?	X			
Was the participant able to locate his or her QR code?	X			
Was the participant able to locate their documents and view them?	X			

Participant number:	3
Gender	~~male~~/female/~~other~~
Level of technology proficiency	~~low~~/medium/~~high~~
Platform	iOS/~~Android~~
Age:	28

Question	Yes	No	With help	Time
Was the participant able to sign-up?	+			
Was the participant able to log in to his or her account?	+			
Was the participant able to upload a paper document, with use of the camera?	+			
Was the participant able to upload a "document / image" from their smartphone?	+			
Was the participant able to locate and accept an access request?	+	+		

Was the participant able to locate and remove access from a company?	X			
Was the participant able to locate his or her QR code?			X	
Was the participant able to locate their documents and view them?	X			

Participant number:	4
Gender	male/~~female~~/~~other~~
Level of technology proficiency	~~low~~/medium/~~high~~
Platform	iOS/~~Android~~
Age:	46

Question	Yes	No	With help	Time
Was the participant able to sign-up?	X			
Was the participant able to log in to his or her account?	X			
Was the participant able to upload a paper document, with use of the camera?	X			
Was the participant able to upload a "document / image" from their smartphone?	X			
Was the participant able to locate and accept an access request?	X			

4

Was the participant able to locate and remove access from a company?	X			
Was the participant able to locate his or her QR code?			X	
Was the participant able to locate their documents and view them?	X			

SeafarerCV usability test

Participant number:	5
Gender	male/~~female/other~~
Level of technology proficiency	~~low~~/~~medium~~/high
Platform	~~iOS~~/Android
Age:	18

Question	Yes	No	With help	Time
Was the participant able to sign-up?			X	
Was the participant able to log in to his or her account?	X			
Was the participant able to upload a paper document, with use of the camera?			X	
Was the participant able to upload a "document / image" from their smartphone?	X			
Was the participant able to locate and accept an access request?			X	

Id ch

Was the participant able to locate and remove access from a company?	X			
Was the participant able to locate his or her QR code?		X		X
Was the participant able to locate their documents and view them?	X			

Participant number:	6
Gender	male/~~female/other~~
Level of technology proficiency	~~low/medium~~/high
Platform	~~iOS/Android~~ Android Moto Lapa
Age:	26

Question	Yes	No	With help	Time
Was the participant able to sign-up?			X	
Was the participant able to log in to his or her account?	X			
Was the participant able to upload a paper document, with use of the camera?	X		X	
Was the participant able to upload a "document / image" from their smartphone?	X			
Was the participant able to locate and accept an access request?			\	

Was the participant able to locate and remove access from a company?	X			
Was the participant able to locate his or her QR code?	X			
Was the participant able to locate their documents and view them?	X			

Appendix D: Observation Notes

Participant 1

Observation notes

Countries is perhaps not in alphabetical order when registering, use some time to find correct country, he doesn't try to type it in. Types password that cannot be used, get information of password criteria, ask about definition of nun alphanumeric (perhaps we need an example). Confirmation email was in spam mail.
Locates certificates quickly.
Goes to profile too look for QR code when he is asked to locate his QR code, he is on right path, but cant locate the QR code on profile picture.
The participant was not able to accept the request, because of known system error, but could locate it easily, and did everything correctly.

Interview

1. The participant knew what to do, but didn't find the qr code, he thinks he maybe would have guest that if there was a image of himself there
2. To some degrees, there was only four so there wasn't to many alternatives so yes. When he was asked about uploading a document, that was the only one that would be a document. he thinks a straight forward document icon would be a better alternative,
3. He would have changed the profile, looks boring. Again mention the possibility of adding a profile picture.
4. Can't think of anything
5. It was simple, functions, not too many choices. He mentions that he would prefer some sound or similar to confirm actions.
6. Nothing in particular .

Participant 2

Observation notes

Asked about nun alphanumeric definition, wonders what was wrong with the password he used. After he has confirmed his password he starts to login to the web application,he is stopped and asked to use the mobile application.
Looks around in the applications as soon as he has logged in.

The app needs a restart to update his confirmation for access, but the participants locates the request and he himself concludes that there is something wrong with the application not that he has done anything wrong.

Interview

1. I didn't really know where to go, but the app was easy to understand and brows.
2. The icons was understandable, access and notifications was straight forward, certificats was then the last one standing. The application was in his opinion setefactoris.
3. He was able to locate the qr code, but think it would be a good idea to show this function.
4. He gets some facebook feels, can be the colors. The app feels familiar
5. It was easy to upload the certificates, the camera function was smooth “AF”.
6. Profile is boring, but he doesn't really feel that is has to be that, the profile feels a bit unnecessary when the only thing you need there is the id and perhaps gq.

Participant 3

Observation notes

The participant finds where to sign up as the others. Types to find country. Tries three passwords two times before finding a password that has the right criteria, she is know confused about what her password was. She asks if the seafarer id is the id i asked for.
Interview. Goes directly to documents page, after short while she presses the plus button and is able to upload the document. She find the correct way to upload document that is on the phone. She browses around trying to locate the QR code. She is able to locate the the documents, click the document, think something is wrong because of the time it takes to load, start clicking around. She locates the acces and removes access without any problem. She makes me aware that the email says seaferer not seafarer

Interview

1. She was confused about the password, but gets the security issue. She mention the qr code was impossible to locate.
2. She finds the icons understandable, not the certificate icon, she notes that there is only four, so use elimination method to get to the correct one. She notes that the upload from hd button was not understandable.

3. The application is pleasant to use, seems easy to use, she again mention the qr code problem.
4. She feels the applications is missing some information when it comes to uploading documents, she proposes a. Tutorial first time login.
5. It feels simple, straightforward, not to much information or functions
6. She feels it reminds her of facebook, mostly because of the color and the notification icon.
7. The QR code placement.

Participant 4

Observation notes

The participant finds the sign up link, the participant types to find country. The participant tries a password that don't have the right criteria, gets the criteria. The participant seems goes thru every page, doesn't seem to pay so much attention to the icons, Don't get email, wonders if he perhaps, typed the wrong email, looks at app email says link link. The participant was not able to find the QR code.
Tries to swipe to remove access.

Interview

1. Not with the qr code, when asked about answering an access request I was looking for some form of notification.
2. The buttons and icons was understandable, in notifications, missing a notification. And comments on the known bug, that it doesn't update
3. He feels the application is purposeful and easy to use, and dos what is supposed to do.
4.
5. Certificates reminds him of dropbox, with the plus button.
6. Profile pages, no image or personalisation, there something wrong with size of screen colon on its own line. There is a search field

Participant 6

Observation notes

Finds the sign up link
Scrolls for country
Uses a password with wrong criterias, asks for definition of non alphanumeric. Email was in spam. When the participant wants to upload he started with his mobile camera, the participant found the function when he was trying to upload the participant, and chose to use this instead. Likes it. The participant went straight for the profile page when was asked for his QR code.

Interview

1. Some, find my id, upload document, remove access and find QR code and sign up
2. Yesm classic notification icon, they seem simple.
3. Seems simple, there doesn't seem like you can do anything wrong.
4. Simple uploading.
5. Doesn't remind my of any thing, but still feels familiar.
6. That i can't crop the image of the certificate, missing functionality in the uploading from camera process, but still å solution.