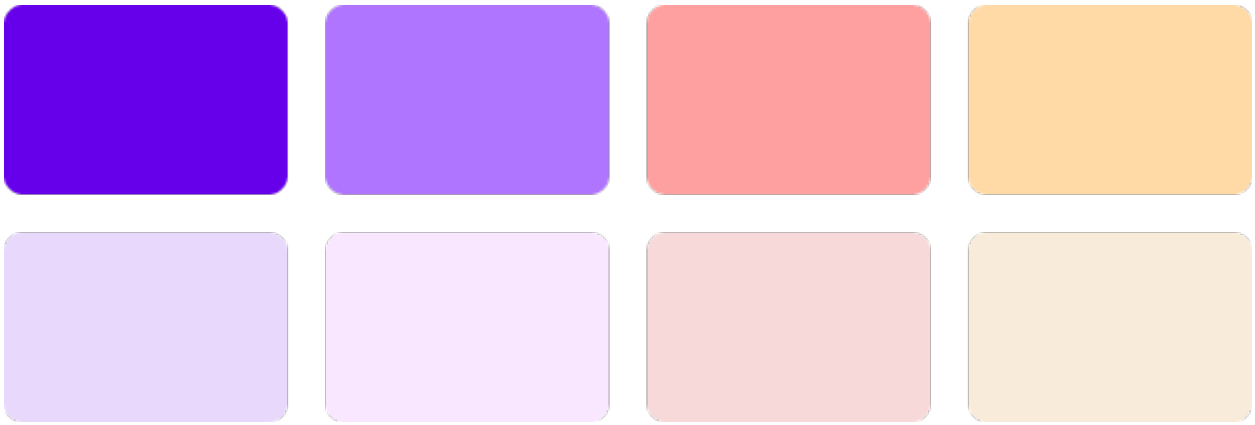


User Interface for Evaluating and Improving Software Security

Establishing Design Guidelines for a Comprehensive OWASP SAMM Tool

Master's Thesis in Computer Science and Engineering

Anna Rikardsson
Louise Tranborg

Department of Computer Science and Engineering
CHALMERS UNIVERSITY OF TECHNOLOGY
UNIVERSITY OF GOTHENBURG
Gothenburg, Sweden 2024

Master's thesis 2024

User Interface for Evaluating and Improving Software Security

Establishing Design Guidelines for a Comprehensive OWASP SAMM Tool

Anna Rikardsson
Louise Tranborg

UNIVERSITY OF GOTHENBURG

CHALMERS
UNIVERSITY OF TECHNOLOGY

Department of Computer Science and Engineering
Chalmers University of Technology
University of Gothenburg
Gothenburg, Sweden 2024

User Interface for Evaluating and Improving Software Security
Establishing Design Guidelines for a Comprehensive OWASP SAMM Tool
Anna Rikardsson
Louise Tranborg

Supervisor: Thommy Eriksson, Department of Computer Science and Engineering
Advisor: Julia Arvidsson; Roger Norrlén; Simon Wendel, Decerno AB
Examiner: Morten Fjeld, Department of Computer Science and Engineering

Master's Thesis 2024
Department of Computer Science and Engineering
Chalmers University of Technology and University of Gothenburg
SE-412 96 Gothenburg
Telephone +46 31 772 1000

Cover: Overview of design system colors incorporated into final design

Typeset in LaTeX
Gothenburg, Sweden 2024

User Interface for Evaluating and Improving Software Security
Establishing Design Guidelines for a Comprehensive OWASP SAMM Tool
Anna Rikardsson
Louise Tranborg
Department of Computer Science and Engineering
Chalmers University of Technology and University of Gothenburg

Abstract

The purpose of this project was to improve software security management by centralizing the Open Worldwide Application Security Projects Software Assurance Maturity Model (OWASP SAMM) assessment procedure. This was done by designing and implementing a user interface for SAMM which reduces complexity and improves the user experience. The thesis was done in collaboration with Decerno AB, a IT consulting firm which specializes in building tailored software systems. By integrating user-centered design principles and participatory design methodologies, the product named Salsa was developed. Salsa provides the entire SAMM process in one place making SAMM easier to work with. From careful evaluation of Salsa and the process a set of design guidelines for a comprehensive OWASP SAMM tool were established. This thesis contributes to the field by demonstrating how centralized and interactive tools can enhance the efficiency and effectiveness of software security management.

Keywords: OWASP SAMM, software security, user experience, user-centered design, participatory design.

Acknowledgements

We would like to express our gratitude to everyone who made this thesis possible, and to some in particular.

To Simon Wendel for advising, helping and supporting us throughout the project. Working with you has been a privilege.

To Julia Arvidsson and Roger Norrlén for making us feel welcome at the office. We are grateful for your commitment as advisors, and it has been really fun to work with you!

To all participants who provided feedback by participating in interviews and user tests. Your insights were essential for the project's success.

To Thommy Eriksson for the feedback and help throughout the project.

Anna & Louise, Gothenburg 2024-06-24

Contents

List of Figures **xi**

List of Tables **xiii**

1 Introduction **1**
1.1 Research Question . . . 2
1.2 Deliverables . . . 2
1.3 Delimitations . . . 2

2 Background **5**
2.1 OWASP SAMM . . . 5
2.2 Applied SAMM . . . 7
2.3 Sammy . . . 9
2.4 Stakeholders . . . 11

3 Theory and Related Work **13**
3.1 Participatory Design . . . 13
3.2 User-Centric Software Security Management . . . 13
3.3 User Engagement . . . 14

4 Methods **17**
4.1 User-Centered Design Process . . . 17
4.2 Double Diamond . . . 18
4.3 Competitive Analysis . . . 18
4.4 Interviews . . . 19
4.5 Visualizing User Behavior . . . 19
4.6 Prototypes . . . 20
4.7 MoSCoW . . . 20
4.8 Agile Software Development . . . 21
4.9 Evaluation . . . 21

5 Execution **23**
5.1 Planning . . . 24
5.2 Specifying Persona and User Journey . . . 24
5.3 Step 1: Assessment . . . 25
5.3.1 Investigate . . . 26

5.3.2 Prototyping . . . 27
5.3.3 MoSCoW . . . 29
5.4 Step 2: Report . . . 30
5.4.1 Investigate . . . 30
5.4.2 Prototyping . . . 31
5.4.3 MoSCoW . . . 34
5.5 Step 3: Phases . . . 34
5.5.1 Investigate . . . 35
5.5.2 Prototyping . . . 36
5.5.3 MoSCoW . . . 39
5.6 Step 4: Overview . . . 39
5.6.1 Investigate . . . 40
5.6.2 Prototyping . . . 40
5.6.3 MoSCoW . . . 42
5.7 Additional Features . . . 43
5.7.1 Phase Reports . . . 43
5.7.2 SAMM Question Subset: Mild Salsa . . . 44
5.8 Evaluation . . . 45
5.8.1 Continuous Informal Self Report . . . 45
5.8.2 User Testing with Primary User . . . 46
5.8.3 User Testing with Secondary Users . . . 48

6 Results **53**
6.1 Final Design . . . 53
6.2 Guidelines . . . 59

7 Discussion **63**
7.1 Project Results . . . 63
7.2 SAMM Difficulties . . . 64
7.3 Integrating Agile with Design Sprints . . . 65
7.4 Future Work . . . 65

8 Conclusion **67**

Bibliography **69**

List of Figures

2.1 Visualization of the SAMM tree structure [14] 6
2.2 An overview of business functions, security practices and streams within SAMM [15] 7
2.3 Excel template provided by OWASP [4] 8
2.4 SAMM process at Decerno AB 9
2.5 Sammy's assessment view 10
2.6 Examples of abundance of icons in Sammy 10

4.1 The four phases of the UCD process described in [25], [26] 17
4.2 Double diamond methodology [27] 18

5.1 Gantt chart of the time plan 24
5.2 User journey 25
5.3 Persona of primary user 25
5.4 Initial sketches for the assessment 28
5.5 Initial sketches for the assessment after feedback from primary user . 28
5.6 High-fidelity prototypes for the assessment 29
5.7 Report generation sketches 32
5.8 Radar chart visualizing the scores for all security practices 33
5.9 Sketches for planning and reviewing a phase 37
5.10 Sketches for the team dashboard view 38
5.11 Prototypes 38
5.12 Original view of the landing page 41
5.13 Landing page with tabs to divide teams and statistics 41
5.14 Landing page where teams and statistics are merged 42
5.15 Selection of SAMM question set when creating a new team 45

6.1 Landing page presenting all the teams and statistics for the company 54
6.2 Create new team modal 54
6.3 Assessment view 55
6.4 Finishing modal 56
6.5 Team dashboard page 57
6.6 In progress of creating a new phase 58
6.7 View for reviewing the questions selected in phase planning 58
6.8 Review ongoing phase 59

List of Tables

5.1 An example of how the scores from an assessment are visualized in the report 33
5.2 Tables from phase report showcasing the score improvements the phase will result in 44

1

Introduction

Software security is becoming increasingly crucial for keeping programs free from vulnerabilities. Moreover, security has become a necessity due to the regulation from the EU Commission in the EU Cybersecurity Act [1], [2]. However, security is rarely a main objective or task for users [3]. In addition, measuring and improving security levels takes time and resources, which organizations might be tempted to spend on other objectives. As a result, it is of interest to find a way to manage security efficiently.

There are several models for measuring, analyzing and improving software security. One of these resources is The Open Worldwide Application Security Project's Software Assurance Maturity Model (OWASP SAMM) [4]. OWASP is a nonprofit organization providing resources related to web application security [5]. SAMM helps measure and evaluate an organization's current software security practices and further improve them in an iterative program to be reassessed.

Despite SAMM being a powerful model, the practical aspects of implementing the framework can become troublesome. OWASP provides an Excel template for the SAMM process where participants in a project can assess their security maturity level. However, this template does not provide an optimal user experience for several reasons, such as having flaws in visibility, feedback and efficiency.

In many instances, there is a need to compile a report summarizing the results from the SAMM assessment, which OWASP's template does not support. This results in manual work transferring content from Excel to, for example, Word documents. In summary, SAMM as a model is highly useful but the user experience in the process of implementing better cybersecurity with SAMM requires improvements.

For a company managing plenty of projects such as a consultancy firm, the amount of Excel sheets can become increasingly difficult to regularly supervise and maintain. This in combination with the template not supporting the entire SAMM process confirms the need of a comprehensive and centralized system. Hence, such a system will be developed in collaboration with the consultant firm Decerno AB [6]. They build a variety of software projects for different stakeholders, and are therefore in need of maintaining several SAMM assessments simultaneously.

1.1 Research Question

The main purpose of this master's thesis is to improve software security management by centralizing the SAMM assessment procedure. This will be conducted by identifying and solving usability issues in the process of working with SAMM at Decerno AB. The following research question can be formulated in order to properly handle this objective:

What are the key design guidelines for a comprehensive OWASP SAMM assessment tool?

1.2 Deliverables

In order to fulfill the purpose of this thesis, a new assessment system for SAMM will be designed, implemented and evaluated. The main deliverable of the project will therefore be a software system handling the management of the SAMM process. It is also discussed to release the developed system open source for anyone to use in order to contribute towards safer software.

Through the interface, the users need to be able to perform four main tasks. First of all, an initial interview assessment needs to be performed. Secondly, a report summarizing the score results from the assessment is generated. Thirdly, the SAMM score is increased through an iterative program. The idea is to continuously improve the security by always improving the scores little by little. Finally, an overview with statistics from all the different projects is displayed. These features comprise the four steps that will be designed and implemented in this master's thesis, and they further conclude the central Minimal Viable Product (MVP) of the system.

A second deliverable will be a set of guidelines summing up the findings from the project. These will be assembled through an evaluation of the software system. In conclusion, the design guidelines, together with the product, should establish the main design choices for a successful SAMM management tool for consultant firms.

1.3 Delimitations

The main stakeholder of this project is the application security specialist of Decerno AB, who will be the primary user of the system. As a result, this user will be most considered throughout the development of the SAMM assessment tool. This comes with both benefits and difficulties. One benefit is that having one primary user will speed up the process, since close and fast communication can be held throughout the entire project. However, one disadvantage is that tailored design decisions might not be optimal for a wider audience. This will be counteracted in several ways.

First of all, it will be counteracted by having additional interviews with secondary users at Decerno AB in order to gain wider insights. In addition, all employees at Decerno AB will have access to the program via Decerno AB's network throughout its development in order to try it out and give feedback. Finally, since the primary

user is an expert on SAMM and software systems, it will be taken into account that this knowledge does not necessarily extend to other users. However, the development will still be highly focused on only one user, which can affect the suitability of the program to a wider set of users.

When it comes to the developed tool itself, plenty of features could be included. However, due to constraints such as time limitations and choice of user, not all suggestions will be implemented. One feature that will not be implemented is an authentication system controlling which user has access to which resources. Instead, data accessibility will be granted through internal firewalls and certificates within Decerno AB's internal network. On a similar note, at Decerno AB, which hosts many projects with many different customers, employees rarely have privileges to access information on projects outside of their own. However, a solution for handling privileges across several users will not be implemented.

One idea for the system was to have it teach the user how to utilize SAMM. This could be performed, for example, by adding connections between the interview questions in order to help the user understand their relations. However, this is not an area that will be included since it would put the focus on perfecting and evaluating SAMM instead of creating an interface for the original model. In addition, since the primary user is an expert on SAMM it is not prioritized to have the system teach the user about SAMM.

Furthermore, the prototype and implementation will solely be designed for desktop and therefore not be responsive to mobile screens. This is due to time constraints and the fact that the application will only be used on desktops by the primary user. However, some effort will be put into supporting different sizes of desktop screens, with a primary focus on preventing the interface from breaking.

2

Background

This master's thesis is executed in collaboration with the IT consultancy firm Decerno AB [6], which specializes in building tailored software systems. The company is in turn part of Addnode Group AB [7], which holds companies engaging in design, process or product life cycle management. Decerno AB falls in the third category, and they further focus on customers providing social benefits. Companies within industries such as healthcare, construction as well as the public sector are represented in their customer base, all requiring mature security precautions. These needs turn Decerno AB into the perfect candidate for a SAMM management system.

This chapter starts with describing the OWASP foundation and their model SAMM in detail. Afterwards, it is demonstrated how Decerno AB currently utilizes the security model. Finally, a possible solution to some of the usability issues with the current setup, Sammy by Codific [8], is presented.

2.1 OWASP SAMM

OWASP stands for Open Worldwide Application Security Project and is a nonprofit foundation focusing on improving software security [5]. The foundation was launched in 2001 and was incorporated as a nonprofit charity in the United States in 2004. They have since then worked on their mission to protect web applications from cyberattacks by providing education, tools and collaboration through their community. The OWASP community consists of tens of thousands of members working on open source projects that are free for anyone wanting to improve software security. OWASP provides multiple tools to support their cause, such as OWASP Top Ten, an awareness document of the ten most critical risks [9], and the OWASP Web Security Testing Guide (WSTG), a guide for web service and application testing [10]. However, this thesis will focus on OWASP's Software Assurance Maturity Model (SAMM), a tool for effectively analyzing and improving a software project's secure development life cycle [4].

In their paper titled *A Survey and Comparison of Secure Software Development Standards*, Aiello et al. divide security models into three different categories, Maturity Models, Industry Groups for Software Development and Standards Institutes [11]. In their case, SAMM falls into the first category, Maturity Models. Maturity Models ease the introduction of a security process, while the second category emphasizes the stages in greater detail such as coding and design standards. The third

category focuses on specific industries with detailed security needs.

SAMM was originally established in 2009 by Pravir Chandra, an expert within the field of software security [12], [13]. The model is built to be evolutive, versatile and risk-driven, in order to work well for organizations with different needs. Since its creation it has gone through minor changes, turning it into a robust and well-established model. To summarize, SAMM helps organizations evaluate their existing software security practices, build a software assurance program in iterations, display concrete improvements to the system, and define and measure security-related activities.

In order to measure and evaluate the organization's existing software security practices an initial assessment is performed. This assessment consists of a set of 90 questions divided into five different business functions, Governance, Design, Implementation, Verification and Operations [14]. Each of these levels further has three security practices, where each practice represents a security-related activity related to the function. Furthermore, each security practice consists of two streams, where a stream has an objective to be reached in three different questions. With each question, its level gets increasingly sophisticated in terms of maturity. A visual representation of the SAMM tree structure can be seen in Figure 2.1. In addition, an overview of all business functions, security practices and streams can be seen in Figure 2.2.

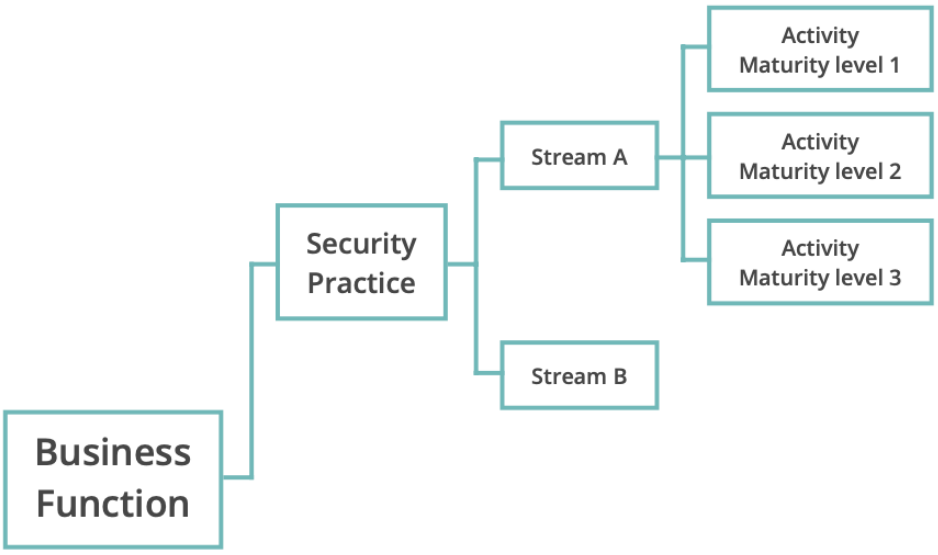


Figure 2.1: Visualization of the SAMM tree structure [14]

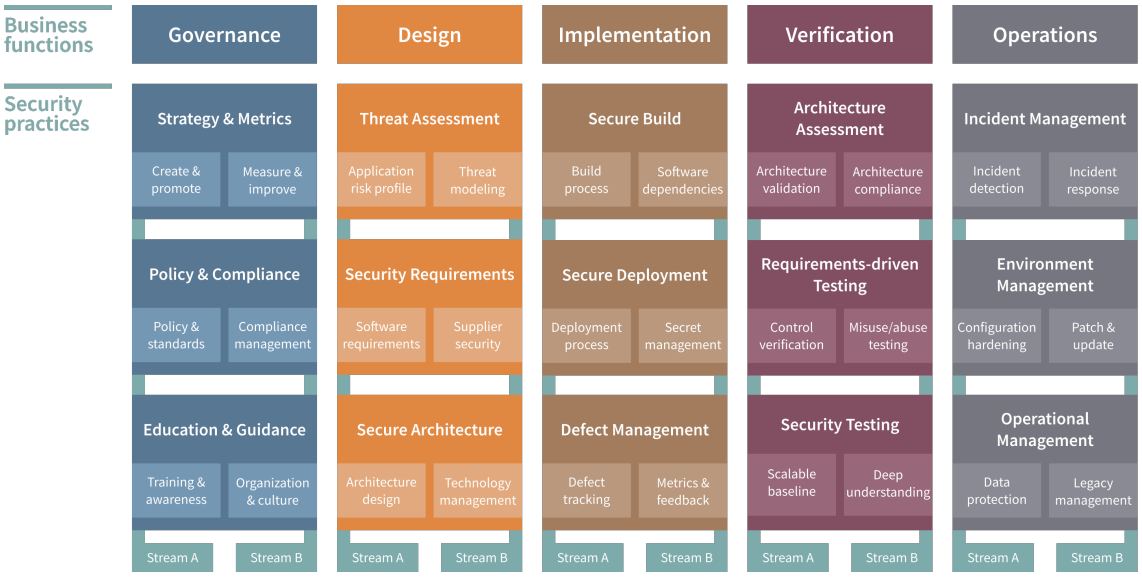


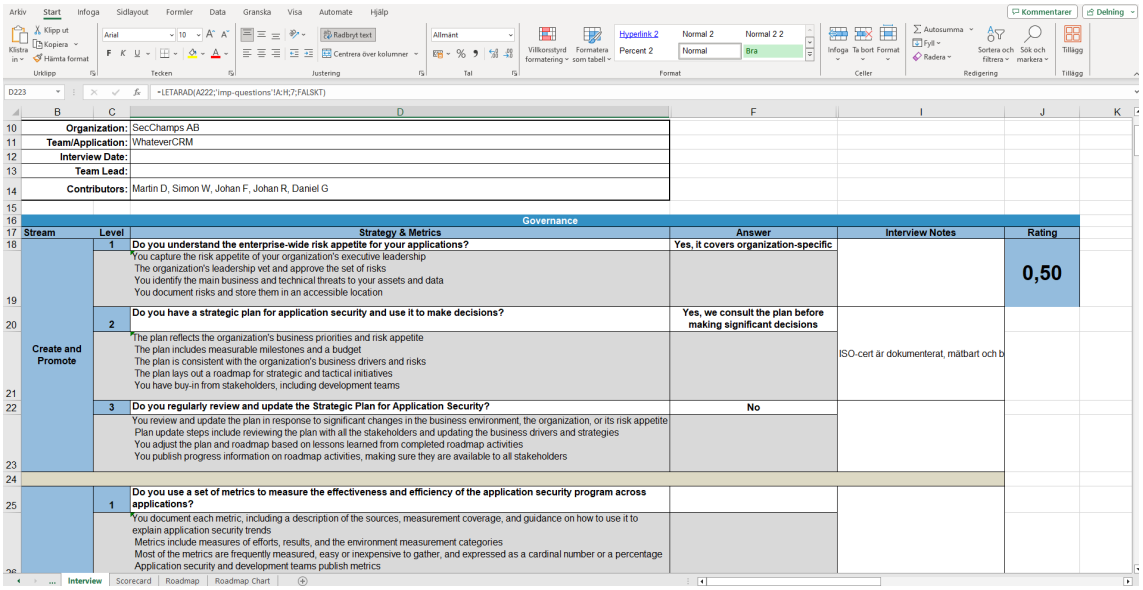
Figure 2.2: An overview of business functions, security practices and streams within SAMM [15]

2.2 Applied SAMM

The questions in the SAMM assessment can be difficult to answer, and there can in many cases be the need to clarify what is asked for. In order to assist in this issue, OWASP provides the SAMM Handbook [14], which is a 300-page long document consisting of detailed explanations of each question. Every question has an objective statement which is the motivation for the question as well as a short benefit description, summarizing why the objective is helpful. Furthermore, it also has a longer activity section providing context for the question as well as which aspects could be considered while answering. Every question also has between two and five quality criteria, which provides general guidance on how the question should be answered.

In order to perform the assessment, OWASP provides an Excel template for SAMM, see Figure 2.3. The assessment sheet includes the questions, as well as their quality criteria and a dropdown for the alternatives. Each question also has a small commentary space.

Once the assessment is finished, scores between 0.00 and 3.00 are given on each business function. The higher score the better. The total score from the assessment will be the average of the scores from the business functions. After the assessment is completed, the team will know the strengths and weaknesses of the system concerning software security. As a result, the model encourages the team to proceed with the SAMM process by setting up a strategy from these results in order to improve the overall score of the project [16]. This process is iterative and done through a concept called phases. A phase will be a set of questions from the assessment where a higher-scoring alternative is planned to be reached. The plan is then executed by the team, and the phase is set to complete, which will increase the score. This is then repeated until a pleasing level of maturity is reached.



Organization:	SecChamps AB	
Team/Application:	WhateverCRM	
Interview Date:		
Team Lead:		
Contributors:	Martin D, Simon W, Johan F, Johan R, Daniel G	

Governance					
Stream	Level	Strategy & Metrics	Answer	Interview Notes	Rating
Create and Promote	1	**Do you understand the enterprise-wide risk appetite for your applications?** You capture the risk appetite of your organization's executive leadership The organization's leadership vet and approve the set of risks You identify the main business and technical threats to your assets and data You document risks and store them in an accessible location	Yes, it covers organization-specific		0,50
	2	**Do you have a strategic plan for application security and use it to make decisions?** The plan reflects the organization's business priorities and risk appetite The plan includes measurable milestones and a budget The plan is consistent with the organization's business drivers and risks The plan lays out a roadmap for strategic and tactical initiatives You have buy-in from stakeholders, including development teams	Yes, we consult the plan before making significant decisions	ISO-cert är dokumenterat, mättbart och b	
	3	**Do you regularly review and update the Strategic Plan for Application Security?** You review and update the plan in response to significant changes in the business environment, the organization, or its risk appetite Plan update steps include reviewing the plan with all the stakeholders and updating the business drivers and strategies You adjust the plan and roadmap based on lessons learned from completed roadmap activities You publish progress information on roadmap activities, making sure they are available to all stakeholders	No		
	1	**Do you use a set of metrics to measure the effectiveness and efficiency of the application security program across applications?** You document each metric, including a description of the sources, measurement coverage, and guidance on how to use it to explain application security trends Metrics include measures of efforts, results, and the environment measurement categories Most of the metrics are frequently measured, easy or inexpensive to gather, and expressed as a cardinal number or a percentage Application security and development teams publish metrics			

Figure 2.3: Excel template provided by OWASP [4]

When performing the assessment as well as the following phases, the setting usually includes a mediator and team members from the project that is to be evaluated. The mediator is preferably someone specialized at SAMM, and is familiar with the process. In the case of this master's thesis, the mediator is the primary user of the system to be developed. The team represents the product that is to be evaluated, has knowledge about project operations and is needed to perform the assessment as well as its improvements. The team members are seen as the secondary users of the system to be developed.

In the dynamic landscape of Decerno AB, a consultancy firm with a multitude of clients and projects, the need for an efficient and streamlined approach is paramount. With a portfolio of diverse projects, the company has a process of evaluating its projects using SAMM, which has proven to be successful. However, there are challenges in managing this multitude of projects without compromising efficiency.

Although the model is built to be versatile, the assessments are usually performed in the Excel template. This process can sometimes be sub-optimal when alternating between different sections of the assessment, and the Excel file does not contain all of the handbook guidance on how to answer the questions. In addition, when managing multiple projects, the multitude of assessments and projects can be challenging to keep track of and get an overview of. After the assessment is completed, a report is generated and sent to the team members and other stakeholders. This is currently done manually by copying and pasting the results from Excel to Word, which takes unnecessary time for the mediator. Figure 2.4 shows the current SAMM process at Decerno AB.

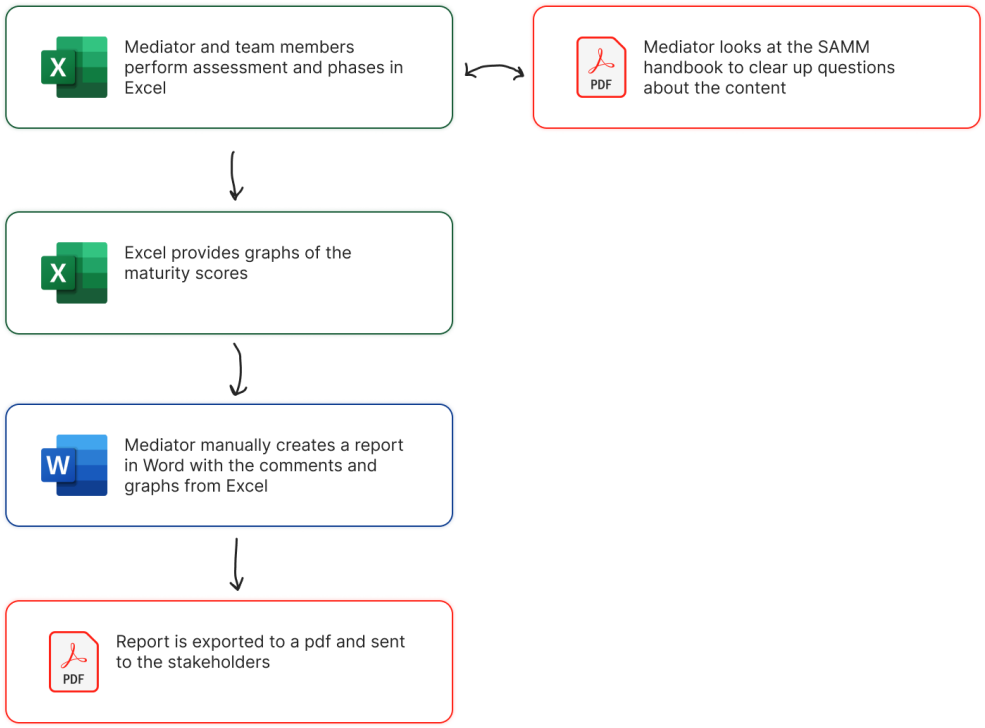


Figure 2.4: SAMM process at Decerno AB

Furthermore, it is also shown in a recent study performed by Fucci et al. that many times, employees with different roles in a project can have diverging opinions on which alternative should be selected [17]. When performing the assessment separately without the entire team gathered, the resulting scores hence become very different. Therefore, a conclusion from their study is to include many different stakeholders when performing the assessment. This is something that is of importance at Decerno AB, as seen in Figure 2.4.

2.3 Sammy

In order to tackle some of the problems with the template, the company Codific has produced and delivered the website Sammy, which is a user interface (UI) for the SAMM management process [8], see Figure 2.5. Sammy provides a way to create teams, add users, perform the assessment and phases, and generate a report of the current stage of a team. However, Sammy is not suitable for Decerno AB for several reasons.

One of them is the design of Sammy. A competitive study was performed on Sammy at the start of this thesis in order to gain wider insights. This study showed that Sammy's design breaks several of Jakob Nielsens's 10 heuristics for user interface design [18], such as visibility of system status, consistency and standards, error prevention, and aesthetic and minimalist design. In more detail, the unintuitive design decisions are for example the three hamburger menus on the top of Figure

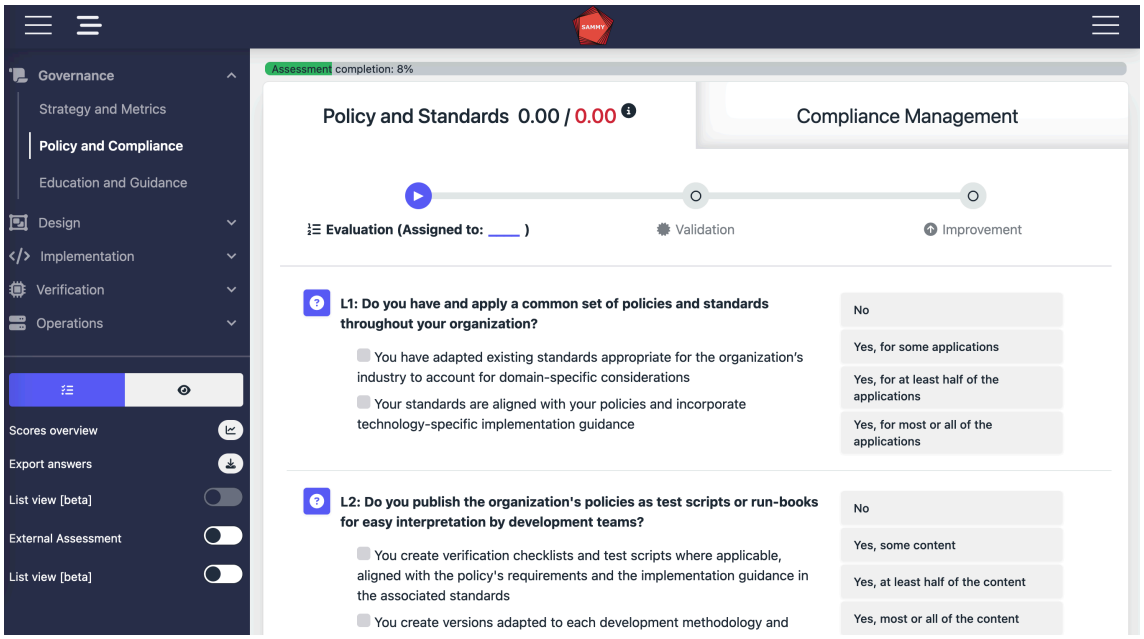


Figure 2.5: Sammy's assessment view

2.5 and the abundance of icons shown both in Figure 2.5 and in Figure 2.6. As a result, Sammy makes it difficult for the users to fulfil their goals, which discourages them from utilizing the interface.

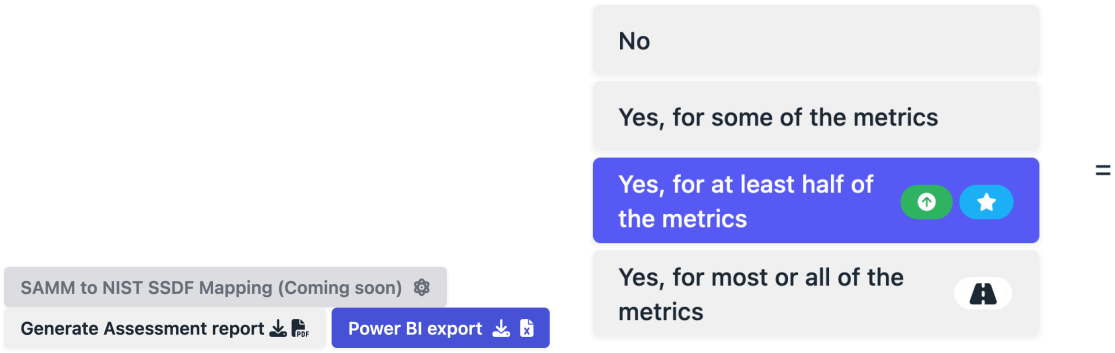


Figure 2.6: Examples of abundance of icons in Sammy

Another reason for Sammy not being suitable for Decerno AB, is the fact that many clients of Decerno AB do not approve of having their data stored by a Software-as-a-Service (SaaS) product. This prevents Decerno AB from trying Sammy even if Codific would make improvements.

This master's thesis comes in response to this pressing need, aiming to create a tailored UI solution that aligns seamlessly with the requirements of Decerno AB, and specifically their application specialist who acts as the mediator throughout the SAMM process. The goal is to create a tool that can offer an overview of a multitude of projects, alongside optimizing the processes. This involves improving efficiency by simplifying the assessment process, automating document generation and freeing up valuable time for the mediator to focus on other critical aspects of their work.

2.4 Stakeholders

The results from this thesis are expected to benefit various stakeholders. The main stakeholder taken into consideration throughout the project is the primary user, namely the mediator at Decerno AB. However secondary users, i.e. the teams working on the software that is assessed with SAMM, will also be involved throughout the project. The following list presents the relevant stakeholders to this thesis.

- **Mediator:** The mediator will be seen as the primary user of the system to be developed, and is an expert of SAMM and software security. The mediator will benefit greatly from the results from the thesis since they are in charge of performing assessments with the different teams at the company. During this thesis, only one mediator is available at Decerno AB.
- **Team members:** The team members from the projects that are assessed will be seen as the secondary users of the system to be developed. These are people with different expertise, such as developers and project managers. Compared to the mediator, who are part of several SAMM assessments, the team members are only involved in their own.
- **Decerno AB:** The top level management and leadership of Decerno AB are interested in providing high software security in the projects. Therefore, they are also benefited from the results of this thesis.
- **Clients:** The clients to Decerno AB also want high software security in their systems. The results from this thesis can help in this regard by making the process of achieving high security easier.

3

Theory and Related Work

This chapter presents the theoretical concepts employed in this thesis. Explored concepts include participatory design where users are engaged in designing the product, and an overview over how software security models is currently revised. Furthermore, the concept of user engagement is explored due to its focus in this thesis.

3.1 Participatory Design

Participatory design (PD), also called co-design, is a set of theories and practices related to having end-users as participants in activities leading to software products [19]. The field of PD is very diverse, touching on fields such as user-centered design, software engineering and architecture. Due to this diversity there is no single theory or approach to practice for PD. However, compared to user-centered design, PD stands more towards the philosophy of seeing the user as a partner than as a subject [20].

The idea with PD is that researches and participants are brought together in the context of software design and development [19], which in this thesis will be designers, developers and end-users of the SAMM assessment tool. By working together all voices can be heard and the combination of diverse knowledge can result in better products. In other words, designers and people not trained in design work together in the design process with the goal of finding better solutions than if the designers worked more independently [20].

3.2 User-Centric Software Security Management

The article *A Survey and Comparison of Secure Software Development Standards* written by Armando Ramirez, Anthony Aiello and Susan J Lincke investigates the awareness of different models within cybersecurity [21]. Ramirez et al. compare a set of standards, guidelines and certifications for software security. The goal was to provide an overview of them for people to quickly be able to gain an understanding of which they should follow to develop more secure products.

Ramirez et al. further explain that companies face immense challenges in developing secure software as well as managing security in the Software Development Life Cycle (SDLC). One reason for this is that organizations often have conflicting goals

between development speed and security. This, according to Ramirez et al., discourages the use of secure software standards. A way to tackle this challenge is to simplify software security technologies, making them require less time and expertise. However, Ramirez et al. do not further investigate how this actually could be done, since it was not within the scope of the study.

While there are plenty of papers highlighting the importance of security and available models, not as many can be seen looking into the user experience, user engagement and simplicity of methods to improve security in the SDLC.

A paper leaning more towards the latter alternative is *Designing User-Centric Information Security Management Systems in Financial Services Organisations* written by Ivano Bongiovanni [22]. This paper presents a pilot study exploring the human side of information security by looking into the potential trade-off between cybersecurity and productivity. Bongiovanni explains that research shows that employees are more likely to unsafe behaviors when they think security mechanisms give additional unnecessary workload. As a result, there is a need for more effective information security measures that are easily usable and implementable. The study focused on the use of design-inspired methods, personas and journey mapping, to improve cybersecurity management. With this, some promising findings and recommendations were identified which can lay the foundation for further work.

A finding from Bongiovanni's study which is of relevance for this project, is the importance of having information security management and processes transparent. This is useful in the case of having employees of different roles, expertise, seniority or cultural background doing the procedure, since they could have different definitions and understandings of security and its complexity. This statement is in agreement with Fucci et al., who argues that a diverse team is fundamental for achieving accurate results from a SAMM assessment [17].

A study trying to incorporate usability in the otherwise complex topic authorization is *User-friendly yet rarely read: A case study on the redesign of an online HIPAA authorization* written by Pearman et al. [23]. The authors investigate if an improved user interface when reading a consent form before sharing data with a third-party operator can help the users understand what they are agreeing to. In three iterations, they created prototypes focusing mainly on changing titles and adjusting it to fit the users' mental model. The study is relevant for this thesis due to its aim to simplify a difficult process by enhancing the UI and acknowledge the users' expectations.

While evaluating the prototypes it was shown that although the new prototypes increased the participants' understanding of the process, some still was unsure of what they signed. However, this seemed to stem mainly from the fact that they did not read the included description text.

3.3 User Engagement

Designing for usability and user engagement can be difficult since there are no clear guidelines to follow that work for every project. Renee Garett, Jason Chiu, Ly Zang

and Sean D Young are looking into this issue in *A Literature Review: Website Design and User Engagement* [24]. To tackle the problem of having no clear guidelines to design for user engagement, a literature review was done to determine website design elements that are most commonly used or suggested to increase user engagement. Based on the findings, Garett et al. defined a list of website design elements that best facilitate user engagement.

The literature review discovered 20 design elements commonly discussed in research surrounding user engagement: organization, content utility, navigation, graphical representation, purpose, memorable elements, valid links, simplicity, impartiality, credibility, consistency/reliability, accuracy, loading speed, security/privacy, interactive, strong user control, readability, efficiency, scannability and learnability. Garett et al. calculated the proportion of studies mentioning each element, and picked out them used in at least 30% of the studies, to get a list of the most relevant ones. These were, navigation (62.86%), graphics (60%), organization (42.86%), content utility (37.14%), purpose (31.43%), simplicity (31.43%) and readability (31.43%).

However, Garett et al. highlight that the relevancy of the design elements also is impacted by the industry and objectives of each individual project, and this should therefore be taken into consideration when using this list when designing websites. In other words, Garett et al. advice designers to consider their list with design elements, along with the project's unique needs, when developing for user engagement.

4

Methods

This chapter describes the methods used in this project. They are all established approaches within the fields of interaction design and software engineering, and they provide a foundation for the process of this thesis project. Specifically, the methods include user-centered design for enhancing usability, agile development for iterative progress, prototyping for testing ideas quickly and evaluation for reassessing the design results. These methodologies are used to ensure a comprehensive approach to the design, development and evaluation of the project outcomes.

4.1 User-Centered Design Process

User-centered design (UCD) is an iterative process where designers focus on the users' needs [25], [26]. In order to fulfill the users' needs, they are involved throughout the entire process. This can be done through a variety of techniques, for example interviews. Some of these techniques will be further explained in this chapter.

The user-centered design process is divided into four phases, see Figure 4.1. Initially, the designers try to understand the context of how the users will use the product. Once the context is well understood, they identify the requirements of the users. Then, they design possible solutions fulfilling these requirements. Finally, they evaluate the designs against the specified requirements. This concludes one iteration of the UCD process, and the design team can make further iterations until satisfying results are reached.

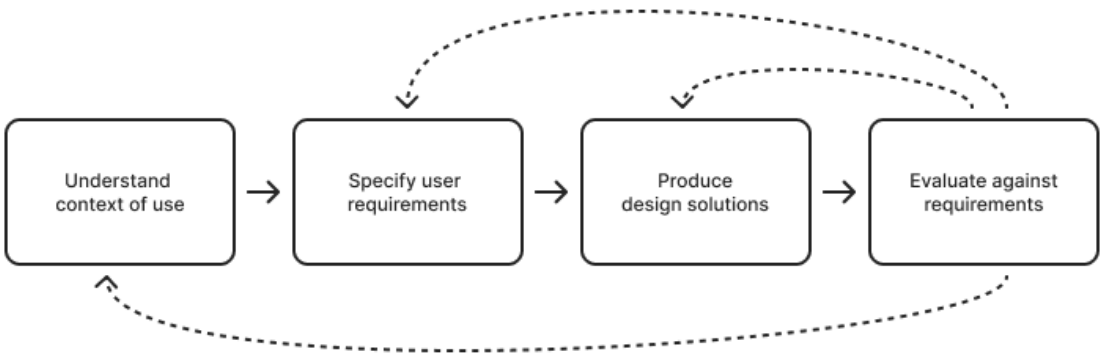


Figure 4.1: The four phases of the UCD process described in [25], [26]

4.2 Double Diamond

Double diamond is a design process framework highlighting the importance of continuously alternating between divergent- and convergent thinking when solving complex problems [27]. The framework consists of four steps: Discover, Define, Develop and Deliver. These are further divided into two diamonds, see figure 4.2.

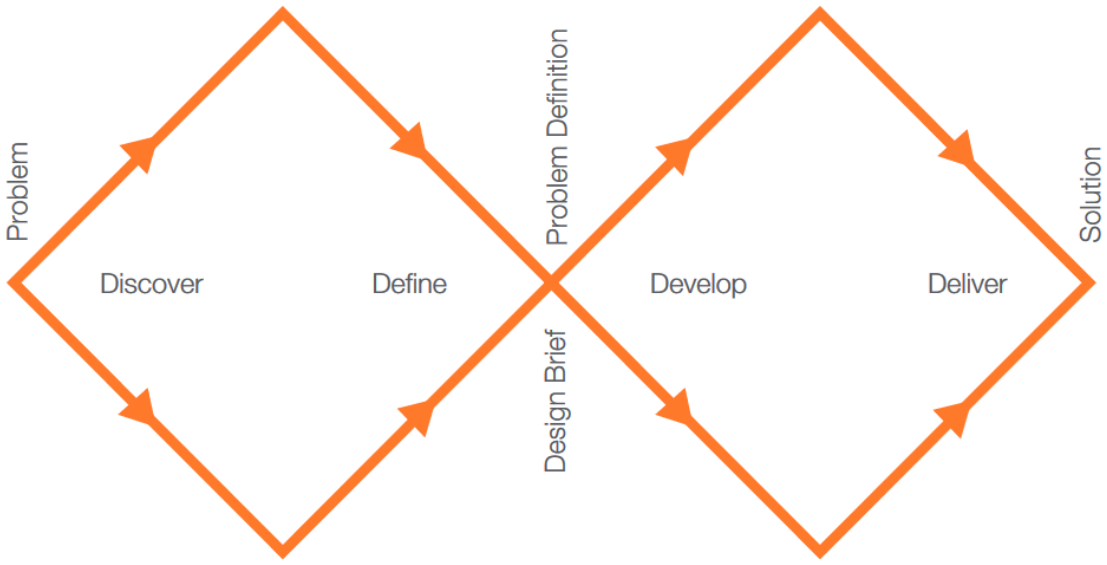


Figure 4.2: Double diamond methodology [27]

The first step, Discover, means to diverge in order to understand the challenge. This step can include methods such as interviews in order to help discover and understand the users' needs. Define is about converging to define the problem so that it can be addressed. At this stage, ideas are either accepted, rejected or merged to compose the best solution. During the Develop step, divergent thinking is utilized to explore different design solutions via for example prototypes. Finally, in the step of Deliver, convergent thinking is used to combine and reject the different alternatives in order to select a final solution.

4.3 Competitive Analysis

The goal of a competitive analysis is to gain insights into the strengths and weaknesses of competitor businesses [28]. By understanding the competitors, strategic decisions can be made to stand out from the crowd. The method can also highlight gaps in the market, which can be used to fulfill needs that are not yet met.

A competitive analysis consists of performing a heuristic evaluation [29] of the competitor businesses' user experience. A heuristic evaluation is executed to identify design issues in an interface [29]. The interface is evaluated against a set of guidelines. The guidelines could be the set of heuristics from Jakob Nielsen [18] or a customized set suited for the occasion.

A competitive analysis should be one of the initial steps in the research process [28], and it should be performed before starting the design for the new project. However, competitors can emerge at any time, increase their offerings or change form, which makes it important to continuously review and evaluate competitors' products throughout the project.

4.4 Interviews

In order to get to know the users of a system, interviews can be utilized. Successful interviews are held by a knowledgeable, clear and gentle interviewer that is also sensitive and flexible towards the interviewee [30].

There are three types of interviews: structured, semi-structured and unstructured [30]. Structured interviews follow a predetermined set of questions where there should be as little deviation as possible. This is useful when major issues are known and when detailed information is requested. A structured interview is commonly used as a follow-up interview. Unstructured interviews, on the other hand, do not follow a question protocol and are an alternative when problem areas are not well understood. They could for example be used in brainstorming sessions where open-ended exploration is required. Semi-structured interviews are a combination of the two, which can be used when there is some knowledge about the topic.

Furthermore, interviews can be used in different parts of the execution process. They can be utilized in the information gathering and discover stages as well as in the stages of evaluation. In this project, semi-structured interviews will be utilized throughout the whole process, both for specifying user requirements and during evaluation of the design.

4.5 Visualizing User Behavior

There are several ways to visualize and oversee the user behaviors of the system to be designed. The ways utilized during this thesis are personas, user journeys and user flows. Together, these tools are important for creating systems that meet the specific needs of the users.

Personas are fictional characters that are created based on research to represent users' needs to the design team [31]. It is a helpful way to keep the users' needs, experiences, behaviors and goals central in the design process. In addition, personas can help the designer to step out from themselves which limits the risk of making biased design decisions.

While a persona represents the user, a user journey is a scenario-based sequence of steps that the persona takes to achieve a high-level goal with a product [32]. The time frame for the story of a user journey can be long, following the user through different emotions and locations. User journeys provide a design team with an idea of how the users interact with the product. In addition, they help the team to reach a common understanding of the users' journey through the application. A

user journey should be memorable and easy to digest so that stakeholders such as designers and managers can relate to and understand it.

In contrast, a user flow describes a more focused set of interactions aimed at accomplishing a specific task within the product [32]. In comparison to a user journey, a user flow is more focused and narrowed down to a specific task within the product. In addition, a user flow is often short with the user staying in one physical location throughout the story. In other words, a user flow can be seen as a detailed view of one part of the user journey.

4.6 Prototypes

In order to realize a design solution, prototypes can be used as a tool [33]. Prototypes are powerful tools for starting conversations, communicating ideas and testing possibilities. When a prototype is presented, it gives users the possibility to provide feedback at an early stage in the process without wasting too many resources. With prototypes, changes can be adopted easily and early in the process, and they give an overview of potential benefits, risks and costs. Prototypes can address both broad and specific design questions. Broad such as a navigation system, or specific such as details in the design system.

Prototypes can range from low to high-fidelity, where low-fidelity prototypes are fast and cheap, and can be done with either pen and paper or digital [33]. High-fidelity prototypes are usually made in a design program such as Figma [34], and they usually contain interactive elements. They are made with great attention to detail, and are supposed to be as equal to the finished product as possible. Mid-fidelity prototypes are an in-between option. Common mid-fidelity prototypes are for example wireframes.

4.7 MoSCoW

The MoSCoW method is one of the most utilized methods for software requirements prioritization (SRP) [35]. MoSCoW stands for *Must have*, *Should have*, *Could have* and *Will not have this time*. During the development of a product, requirements are put into one of these four categories in order to keep focus on what is most important. It also helps avoid a continuously growing scope, keeping the project within its time frame, and having mutual understanding between all stakeholders of what the product will contain and not.

The category *Must have* contains requirements that must be included in the final product. The *Should have* category consists of high-priority requirements that should be included if possible within the time frame. *Could have* includes requirements that are desirable but not essential. Finally, the category *Will not have this time* include requirements that will not be considered for the current project.

4.8 Agile Software Development

Agile approaches are designed to be iterative, flexible and sustainable methods for developing software products [36, ch. 1]. There are many techniques that are under the wide category of agile software development. One of them is Scrum, which will be the one used during this project. Scrum is particularly good for developing websites and mobile applications [36, ch. 1], which makes it perfect for this project where a website application will be made.

Scrum is a way for teams to work focused for a limited period of time on a defined set of features, with the understanding that the next set of features could be unpredictable due to, for example, feedback from customers [36, ch. 1]. In addition, Scrum provides a way for each developer to work parallel on smaller tasks, also called user stories. As a result, an MVP can be established fast. Then additional features can be added through iterations, all in line with the users' needs since Scrum creates the opportunity for the team to reflect on the process and take everybody's feedback into consideration.

4.9 Evaluation

Evaluation can be performed using formative and summative methods. Formative evaluations revolve around determining which aspects of the design need to be improved and which work well, while summative evaluations usually compare a product to a benchmark such as a competitor [37].

In order to verify that a product is usable and fulfills the user requirements, user tests can be performed [38]. When performing user tests, the testers should configure a set of objectives that are to be tested, and then through a set of tasks see if the users are able to effectively reach these objectives. This can be evaluated with for example a think-aloud technique, where the user explains what they are thinking while performing a task [39]. Once the tasks are finished, a follow-up interview can be utilized to further evaluate their experience.

In order to analyse and draw conclusions from the qualitative data gathered from evaluation a thematic analysis can be used [40]. During a thematic analysis, the researchers examine the data to identify themes and patterns. The thematic analysis is normally performed by a six-step process: familiarization, coding, generating themes, reviewing themes, defining and naming themes, and writing up [40]. By following a set process the risk of confirmation bias is decreased.

During the first step, familiarization, the researchers get to know the data. During the next step, coding, the researchers highlight sections of the text and come up with short labels describing the content. Next, patterns are identified amongst the codes, and themes are created. The themes are then reviewed and defined. Finally, the results are documented.

5

Execution

This chapter describes the execution of this project. Since the process is based on user-centered design and agile methodologies, it is not entirely linear. However, the process largely follows the structure chronologically as presented and accounts to the double diamond framework, presented in Chapter 4.2. To begin with, a plan was made to set the milestones and objectives. Once a proper time plan was in place, the process of defining the users and their requirements was set in motion. This concluded the initial setup of the project, and it was thereafter time to execute the four milestones. This would result in the platform Salsa (Simon, Anna and Louise SAMM Assessments).

The first step was to design and implement the assessment which is done to receive an initial score for a software project. The score from the assessment is an indication of how securely managed the project is and how well it performs according to SAMM standards.

The second step consisted of generating a report of the results from the assessment. This also included producing several graphs to be displayed for stakeholders engaged in the project. The generation of the report should be automated in order to save time for the primary user, who is currently manually writing the report.

After receiving an initial score, the project team will have an understanding of which areas they are currently managing well and what could be further improved. The aim of SAMM is then to increment the score. Therefore, the third step was to create a process for improving the score through a phase system. In a phase, the participants select a number of questions where the score should increase.

Finally, the fourth and last step included displaying an overview of how the different teams at Decerno AB perform overall and drawing statistics from them. This could include areas which are well managed, and what could be further improved.

Once these steps were completed, the system was evaluated in order to examine the viability of Salsa for the SAMM process. The evaluation was three-fold, meaning it consisted of three different evaluations. These were a user test with the primary user, user tests with the secondary users and a continuous self-reporting evaluation following throughout the entire project.

5.1 Planning

Before initiating the design and implementation process, a plan was set up in order to validate the feasibility of the project. The aim was to arrange a plan that would enable to iteratively design, implement and evaluate the system through a continuous feedback loop. In order to achieve this, the time plan follows an agile structure, see Chapter 4.8, where small increments are made throughout the entirety of the thesis. An overview of the planned process can be seen in the Gantt chart in Figure 5.1.

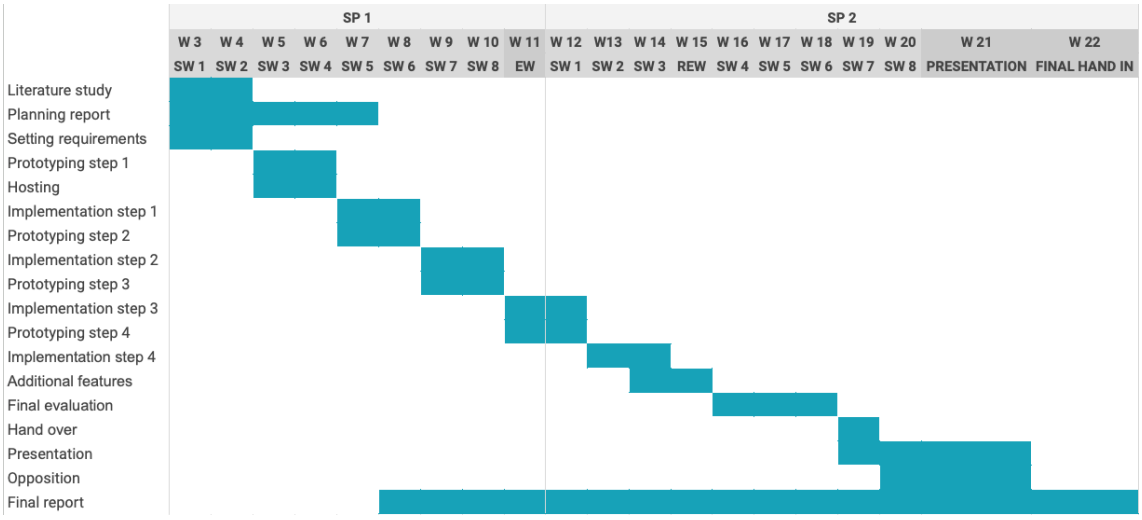


Figure 5.1: Gantt chart of the time plan

The initial weeks were spent investigating the overall expectation of the finished product, researching literature for projects with resembling challenges and understanding SAMM. Furthermore, a competitive analysis, see Chapter 4.3, was performed to receive insights into a current solution.

Time was also set aside to set up a GitHub [41] repository and initiate the client- and server infrastructure as well as decide on the tech stack. The chosen frameworks were React [42] for the client side and for the server and a .NET [43] infrastructure connected to a PostgreSQL [44] database.

The aspiration with the plan was to utilize continuous design sprints and deployments in order to ensure users' needs were met when developing the product. By doing this, continuous feedback could be gathered throughout the project and thereby employ an agile process over a waterfall procedure.

5.2 Specifying Persona and User Journey

In order to keep the project focused on the user requirements and to stay within the time frame of the thesis, it was important to set up a user journey for the product. This was done through semi-structured interviews with the application security specialist at Decerno AB to understand the current process and what features the MVP should contain. A couple of interviews with secondary users were also held to gain

insights from their perspectives. From these insights a user journey was created, see Figure 5.2.

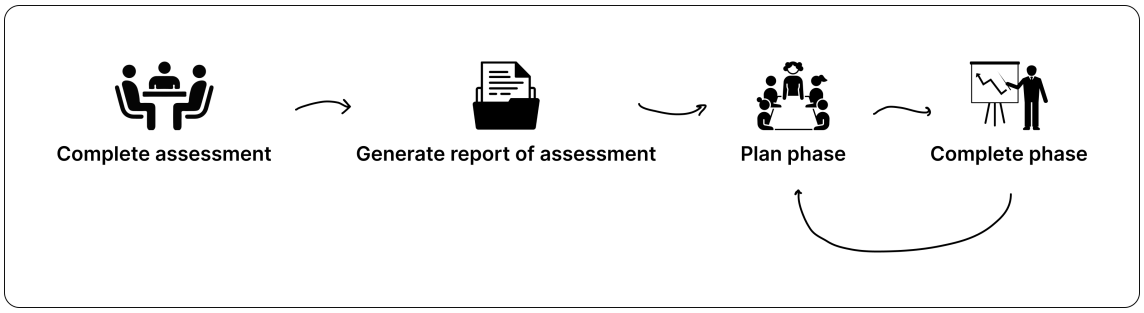


Figure 5.2: User journey

It was also established during these interviews what kind of user characteristics to keep in mind during the development of this system. It was, for example, established that the primary user, the application specialist at Decerno AB, is an expert user with a lot of knowledge of SAMM and software systems. Even though the primary user was easily reachable when questions arose during development, a persona was created to compile the characteristics, see Figure 5.3.

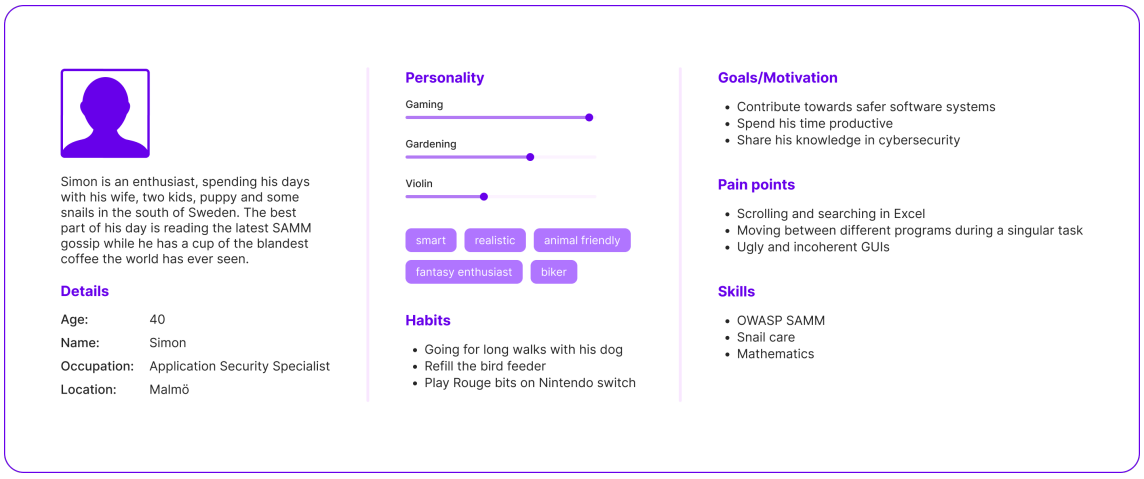


Figure 5.3: Persona of primary user

5.3 Step 1: Assessment

The first step consisted of designing and implementing a way of performing an assessment. The assessment is the first step of the SAMM process. Here, the primary user and the secondary users sit together and answer the questions throughout several sessions. This is a time-consuming process, but it is crucial to review the current security levels of the assessed system in order to know where to put in efforts for improvements. As a result, the assessment can be seen as the core feature of the MVP and was therefore tackled first.

5.3.1 Investigate

This step was initialized by having a semi-structured interview with the primary user and one with a secondary user. The goal of the interviews was to get an understanding of how the assessment currently is done in practice and specify concrete features for this step. The questions for the interviews varied in detail and were mostly used as conversation guidance. The following list presents some of the asked questions.

- How long does the assessment approximately take to complete?
- How long do you normally spend on each question?
- How often do you use the SAMM handbook?
- What content in the SAMM handbook is of relevance to the assessment?
- How long do the comments usually become?
- What do you think is working well with the current setup?
- What do you think could be improved with the current setup?
- What do you think about Sammy?

The interview with the primary user was held online and the interview with the secondary user was held in person. Two interviewers were present, one guiding the interview and the other being a secretary. The discussion notes from the interviews were filtered and grouped to highlight the main points. The interviews resulted in many findings, such as the fact that an assessment normally takes two to three sessions of two hours each to complete. In other words, each question is carefully answered which speaks to the idea of having each question represented individually on the screen.

Another finding was the importance of the SAMM handbook. The primary user uses the SAMM handbook to look up more information about the question if the secondary users need more explanations in order to answer it. Specifically, the primary user looks up information beneath the titles *Benefit*, *Objective* and *Activity*, which contains more information about the question's meaning and purpose. Currently, the SAMM handbook has to be located and then searched within in order to find the right information every time it is needed. Continuously moving between the Excel sheet and the SAMM handbook takes unnecessary time and creates frustration for the primary user.

Another discussion point was the opportunity to write discussion comments to questions during the assessment. Here, the primary user requested more text fields with different purposes: one for discussion, one for improvements and one for corrections. This is to more easily structure the conversation and distinguish concrete improvement suggestions from the discussion. The text field for corrections is a specific request from the primary user. This feature is to be used when the assessment is finished and the primary user inspects it alone to ensure everything is ready for completion. During the inspection, the primary user sometimes has to change an

alternative to a lower or higher one, due to new insights and the primary user's expertise. In cases like this, it is wished to have a specified text field to use to explain these changes effectively to the secondary users.

Furthermore, the benefits and drawbacks of the current Excel setup were discussed. It was highlighted that SAMM is a powerful model, and that Excel technically works for a small number of projects, even though the user experience is flawed. However, many issues were discussed. These included the endless scrolling between questions in Excel, difficulties with managing several projects at the same time and the alternation between Excel and the SAMM handbook. Sammy was also brought up, where some doubt about their interface was discussed. The fact that many clients of Decerno AB do not accept SaaS products withdraws the opportunity to try Sammy even if they would improve their interface.

These results from the interviews were used to specify the initial features of Salsa, which were divided into the different categories in the MoSCoW document, see Chapter 5.3.3. The core features for completing an assessment were the ability to select an alternative, read *Benefit*, *Objective* and *Activity* from the SAMM handbook and write comments for discussion, improvements and corrections for each question. Another feature, being able to flag questions, was also wished for. This feature was taken into consideration early since it was highly demanded and would not require much time in design and implementation.

5.3.2 Prototyping

The next step was to create design solutions for the highlighted issues from the interviews. This was done through several iterations of ideation, sketching, and prototyping of different fidelity levels. The double diamond framework, see Chapter 4.2, was utilized throughout this process to keep track of when to diverge and converge. This helped to explore different design possibilities as well as decide on the best solution for this project's requirements and users. The primary user was kept in focus through continuous communication and participatory design, where sketches and prototypes were shown and feedback was given iteratively.

After some initial brainstorming and sketching, a number of different alternatives for representing the assessment were created, see Figure 5.4. The focus was kept on the general layout of the assessment. It was discussed how the SAMM tree structure would be visualized for navigation in order to fit the users' mental model. It was also discussed how much of the Excel template would be included in regard to color and visual hierarchy.

Another discussed aspect was how many questions would be displayed simultaneously on the screen. The two main options were to display all maturity levels within a stream at once (three questions) or one question at a time. A benefit of having three questions at once was that it eased the navigation. However, a benefit of having one question at a time was that space opened up on the screen for other content, such as information from the SAMM handbook.

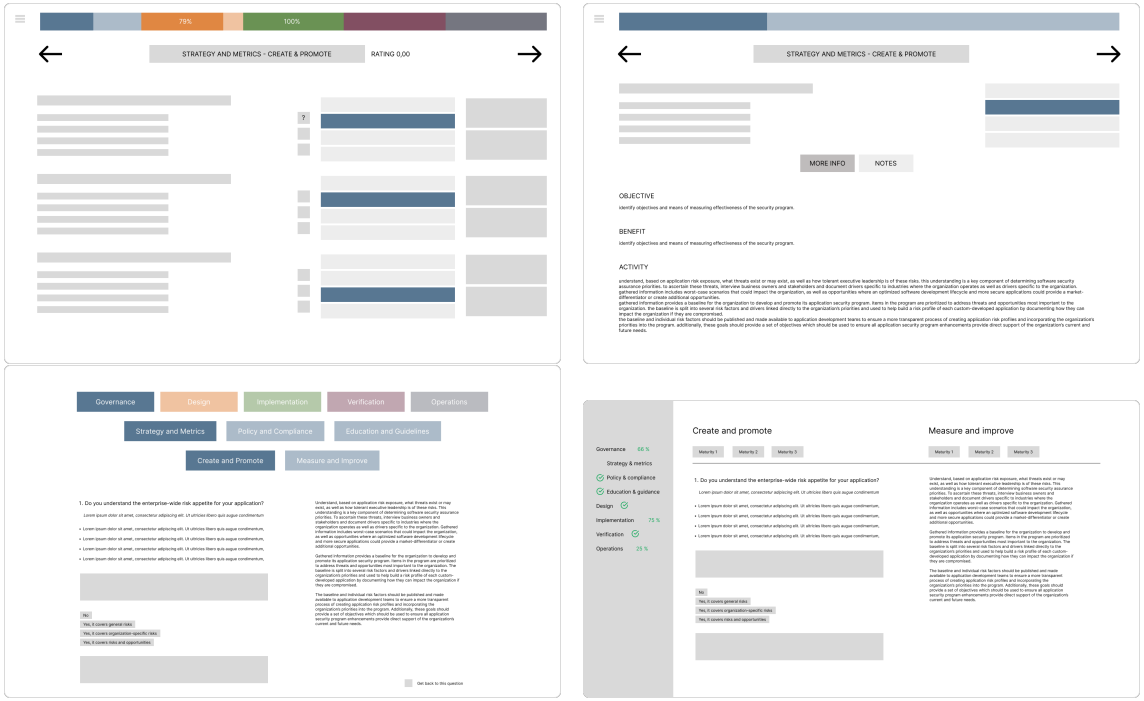


Figure 5.4: Initial sketches for the assessment

After receiving feedback from the primary user, a new set of sketches was made, see Figure 5.5. The primary user preferred having one question per page in order to focus on one at a time and to get an oversight of the *Activity*, *Objective* and *Benefit* information. Furthermore, it was decided to put this information under one tab and the discussion, improvements and correction comments under another since there was no need to see these at the same time. It was also appreciated to see a progress bar of the assessment on the top of the page, so that the users could get an overview of how far they had come in the assessment. At this stage, it was discussed how granular the progress would be, and if it would be seen at the main or sub-navigation bar with the arrows.

Figure 5.5: Initial sketches for the assessment after feedback from primary user

These sketches were then refined to high-fidelity prototypes through iterations, see

Figure 5.6. It was decided to have the progress shown in the main navigation bar thus letting the sub-navigation bar remain neutral in terms of color. When the prototypes became of higher fidelity, the focus was set on smaller details such as the *Flag this question* icon, and how that visual feedback would transfer up into the main navigation bar. It was also decided to show in the maturity tabs if the question was completed or not.

On these prototypes, the initial color scheme can be seen, which was a pastel purple color. Purple was chosen since it is one of the primary user's favorite colors. The primary color was later changed to a more saturated purple in order to obtain better contrast in regard to accessibility. Furthermore, a design system was initiated where the hover and activity behaviors were chosen.

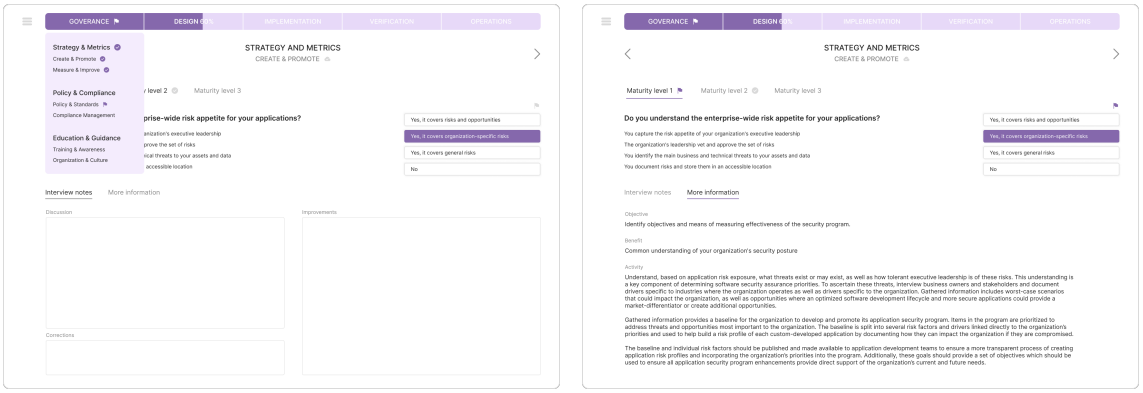


Figure 5.6: High-fidelity prototypes for the assessment

5.3.3 MoSCoW

During the process of investigating and designing an interface for the assessment, a MoSCoW model was formed. This model was used as a foundation for the acceptance criteria in the user stories for the implementation. In addition, it helped in prioritizing features, keeping the project within its time frame and improving communication with stakeholders to ensure everyone was on the same page.

Must have:

- Be able to answer all questions in the assessment.
- See *Benefit*, *Objective*, *Activity* for each question.
- Text fields for discussion, improvements, corrections.
- Auto-saving function.
- Leave assessment in *in progress* state. Be able to come back and finish it later.

Should have:

- Flag question feature.
- Low-hanging fruit feature. A feature similar to flag question but specified to highlight easily fixed improvements.

- Move between questions with the keyboard arrows.

Could have:

- Navigation with keyboard only.
- Improved low-hanging fruit feature. A feature connected to the text field for improvements where the user can add improvement items and select a degree of difficulty for them.

Won't have:

- Implement authentication system.
- Synchronize with Decerno AB AD (Active Directory).
- Have Salsa teach users about SAMM, by for example adding connections between questions to help the user understand them better.

5.4 Step 2: Report

The second step consisted of generating a report in order to summarize the assessment. This was granted its own slot in the time plan due to the uncertainties in implementation as well as the complex structure of the report. At the beginning of the project, an example of a report was given where it was shown how the information could be displayed. However, it was not certain this was the preferred way of visualizing the data. In addition to generating the report, this step also included a way of completing the assessment.

5.4.1 Investigate

In order to understand the requirements for this step, a semi-structured interview with the primary user was conducted before initiating sketching and prototyping. Initial thoughts were gathered before the interview, and from those a series of interview questions were formulated. Due to the interview being semi-structured, the discussion included but was not limited to these topics. The interview questions can be seen in the following list.

- How much of the report should be auto-generated?
- Which language should the report be in?
- Do you want the ability to edit the file after generation?
- When should the resulting scores be visible, in the assessment or report or both?
- Which file format should be used?
- How should the report be structured?
- Except for the content in the example file, do you want any other data to be presented?

The main findings from the interview were first and foremost that under ideal circumstances the primary user would prefer the report to be completely auto-generated and ready to be sent to the stakeholders immediately. In order to achieve this, solutions for generating summaries and accessing the name of the mediator would be needed. This could for example be solved by adding user accounts to Salsa and utilizing AI for the summaries. However, due to time limitations and uncertainties in implementation, this was not prioritized for the MVP.

For the MVP it was therefore decided to give the primary user the opportunity to edit the document in order to fill in the non auto-generated information. The primary user also highlighted the benefit of having the opportunity to edit the document in order to customize the content. In order to achieve this from a technical point of view, the document needed to be downloaded as a .docx file and not as a pdf.

Related to this, it was also discussed which language the report should be generated in. Decerno AB generally operates in Swedish, hence it was decided to be the default language. In a future iteration, there could be the opportunity to select a language before generation. However this was excluded in the MVP.

The structure of the report was also discussed. The primary user had an example document of how the report is usually structured. The document had a first page containing the team name, date, the name of the mediator, and contact information to Decerno AB. After this followed a table of contents. The first chapter contained a short summary of the assessment sessions and outcomes, the scores represented in different tables and graphs, improvement suggestions from the mediator and a list of corrections the mediator made in the assessment. The second chapter contained the discussion comments on each question. The questions without a discussion comment were not listed in the example document. The report was also branded with Decerno AB's logo on each page. The structure of the example document was appreciated and therefore used to a great extent for the report generation feature in Salsa.

Finally, it was discussed when and how the scores should be presented. The primary user expressed that the scores are not crucial when performing the assessment, and they could even be distracting for the secondary users if shown too early. In addition, the scores often discourage the users due to them being very low (a number between 0.00 and 3.00). However, graphs of the scores are popular amongst the users and stakeholders, which is why they cover a great part of the example report. As a result, it was decided to not show the scores during the assessment, but first when the assessment is completed and in the report.

5.4.2 Prototyping

The sketch and prototyping iterations of this step covered completing the assessment and the generation and structure of the report. These features did not require extensive design work, since many could be solved by a simple button. However, the technical aspects concerning implementation were a bit uncertain, such as generating the .docx file. As a result, this step required more time on implementation than design compared to step one.

In order to complete the assessment, a finish button was added beside the navigation bar. Once all the questions in the assessment were answered, the finish button would become active and when clicked on a modal would appear. The modal should show the total score and prompt the user to download the report.

In order to download the report, a download button needed to be added to the interface. However in order to demonstrate alternative solutions, some interfaces were sketched. One alternative was having the user insert the summary in the interface before downloading the report, see Figure 5.7. However, due to needing more insertions than the summary in order to fully auto-generate the document, the alternative was dismissed and a download button for a .docx file was used instead.

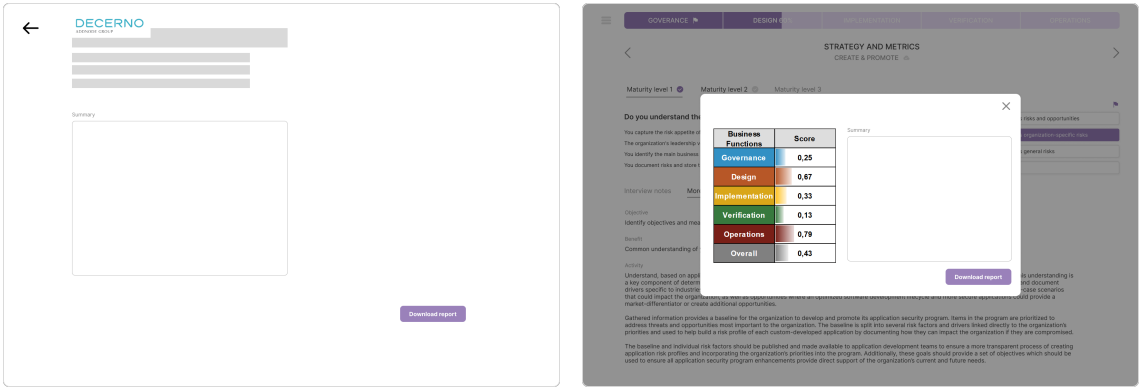


Figure 5.7: Report generation sketches

As previously mentioned, the structure of the example report was used to a great extent. It was decided to have the scores, improvements and corrections summarized in the first chapter and the questions with answers and discussions listed in the second chapter. However, a couple of improvements from the example report were made. First, the improvements and corrections could be accessed directly from Salsa due to the new text fields. Second, in chapter two all questions would be listed with their results and/or discussions, not only the questions with a discussion. The decided structure can be seen as follows:

1. Summary
    (a) Result
    (b) Improvements
    (c) Corrections
2. Question map
    (a) Governance
        i. Strategy & Metrics
            A. Create & Promote
                - M1: ...

- M2: ...
- M3: ...

B. ...

ii. ...

(b) ...

As seen in the structure, the first chapter includes a section for results containing the scores. An example of how to present the scores from an assessment can be seen in Table 5.1. This section should also include some graphs visualizing the results, for example a radar chart, see Figure 5.8.

Business Function	Score
Governance	0.79
Design	0.75
Implementation	0.67
Verification	1.17
Operations	1.13
Total Score	0.86

Table 5.1: An example of how the scores from an assessment are visualized in the report

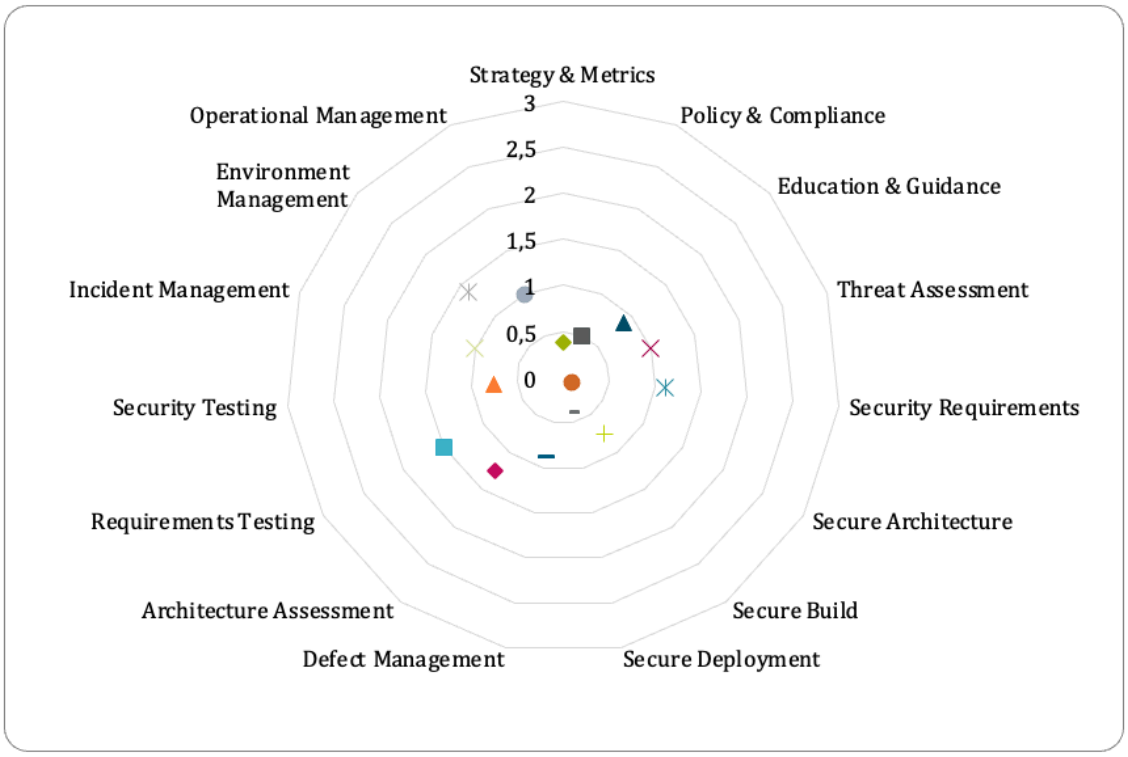


Figure 5.8: Radar chart visualizing the scores for all security practices

5.4.3 MoSCoW

During the investigating and prototyping stages a MoSCoW model was formed together with the primary user. The model was used as a foundation for the acceptance criteria in the user stories for the implementation.

Must have:

- Ability to generate a report with structure as mentioned in section 5.4.2.
- Being able to properly finish an assessment.
- Prompting the user to download a generated report after completing the assessment.
- Calculate the total score for the assessment as well as for business functions, security practices, streams and maturities.
- Summarize the discussion, improvements and corrections that were gathered during the assessment.

Should have:

- Functionality to generate the file as pdf where the summary is written in Salsa and the name of the mediator is accessed.
- Highlight questions where it is easy to improve the score.

Could have:

- Ability to upload a logo for the branding of the report.
- Choose a template in Salsa before generating the report.

Won't have:

- An auto-generated summary using AI tools.
- Save generated reports in the database.
- Ability to edit the template from Salsa.

5.5 Step 3: Phases

The third step was to design and implement a way to plan and complete phases. Phases can be seen as a core feature of SAMM since they have the purpose of iteratively improving the initial assessment score. During the planning of a phase, the primary user and the secondary users meet and select a number of questions to be improved. When the improvements are implemented, they meet again and mark the tasks in the phase as completed. Once all tasks are completed, a new phase can be started to continuously improve the software security of the system.

In contrast to the assessment, the workflow for phases is more complicated. This step can therefore be seen as the largest one in the project and it requires most

views to be designed and implemented. In addition, the phases open up much space for creativity to motivate the team members to improve their score. Concepts like gamification and user engagement are close at hand and could be explored in great depth. However, these concepts had to be explored moderately due to time limitations.

5.5.1 Investigate

As in previous steps, the investigation process started with a semi-structured interview with the application security specialist at Decerno AB. Before the first interview, some concerns were listed in regard to this step. The following list shows some of the questions that were discussed during the interview.

- Please explain how you currently work with phases.
- How long does the planning of a phase approximately take?
- How many questions are normally included in a phase?
- In addition to the selection of questions, what else is included in a phase?
- How long does a phase approximately take to complete?
- Do you want to generate a report of the plan to the client?
- If yes, how do you want the report to be structured?
- How is the team's progress currently visualized?
- What happens if the team does not complete the phase?
- What is the normal increment in the score for a phase?
- Around what score is normal to be around for the teams?

It was found that currently at Decerno AB, the phases are executed in the Excel template given by OWASP. A phase normally takes one session to plan and the meeting is around one hour. The mediator and team members meet up again when the phase content has been completed. This time frame is very open, and can vary between one week to several months. The idea is that the team meet the mediator again when they are ready. The primary user also expressed that it is not important to set deadlines in Salsa. Instead, the importance concerns continuously working with software security by always having an ongoing phase. This removed the need to design and implement features to handle deadlines.

The user further explained that a phase in general contains one to five questions, and normally stays on the lower side of the interval. This is drastically less than the maximum of 90 questions the assessment contains. In addition to select a better alternative to reach during the phase, it is of interest to select a person responsible for the increment and write comments on how to proceed. When selecting questions it was also requested to see the comments from the assessment as well as the information from the SAMM handbook.

The user also showed interest in generating a report of the plan in order to summarize it for the stakeholders. The phase report should be similar to the report for the assessment, just in a shorter format. However, due to the size of the phases step the generation of a phase report was put into the *Should have* category of the MoSCoW, see Chapter 5.5.3, and would be tackled as an extra feature if there was time.

Another finding from the interviews concerns the score increments from phases, being that the increases are in general quite low. This is due to several reasons such as a phase usually contains a small set of questions, and that the scores between the alternatives are not linear. Furthermore, the total score is calculated based on an average, which further diminishes the increment. As a result, the scores take a long time to improve which does not encourage user engagement.

5.5.2 Prototyping

After the interview, possible design solutions for phases were tried out through iterations of ideation, sketching and prototyping. In this step user flows were also utilized in order to visualize how the user moves between the different stages of phases.

The first views were created for planning and reviewing phases, see Figure 5.9. The idea was to reuse the layout for the assessment on the plan a phase view. This is because the assessment view already solves many needs for the phase planning, such as navigation between the questions and visualization of relevant information. In addition, this decision would increase consistency across the interface. However, the idea of not reusing the assessment was also tried out in sketches in this part of the process. In more detail, it was tested to visualize the questions in list form instead, especially during the reviewing of the phase.

Another relevant view when introducing phases was a team dashboard to handle the navigation between phases and the creation of new phases for a team. This view was therefore sketched, see Figure 5.10. A team dashboard could contain a lot of different information, visualizations and features. However, to stay on the timeline, aspects connected to the phases were prioritized.

The sketches were shown to the primary user several times for feedback and to further understand the phase process. The idea of reusing aspects from the assessment view was highly appreciated. The primary user also highlighted the insignificance of the additional content on the team dashboard view and that focus should be on the phases. However, the user liked the security quote added to the top of the page. The user appreciated the playfulness and wanted to keep it. From several iterations high-fidelity prototypes were formed, see Figure 5.11.

On these prototypes, the new color scheme can be seen, which was set to a more saturated purple color. This decision was made to achieve better contrast in order to fulfill accessibility guidelines. In this stage, the purple color was used a lot and the overall look became very purple. To tackle this, black and grey were tried out in some places. In addition, more colors like yellow, orange and red were tested to make it more visually appealing.

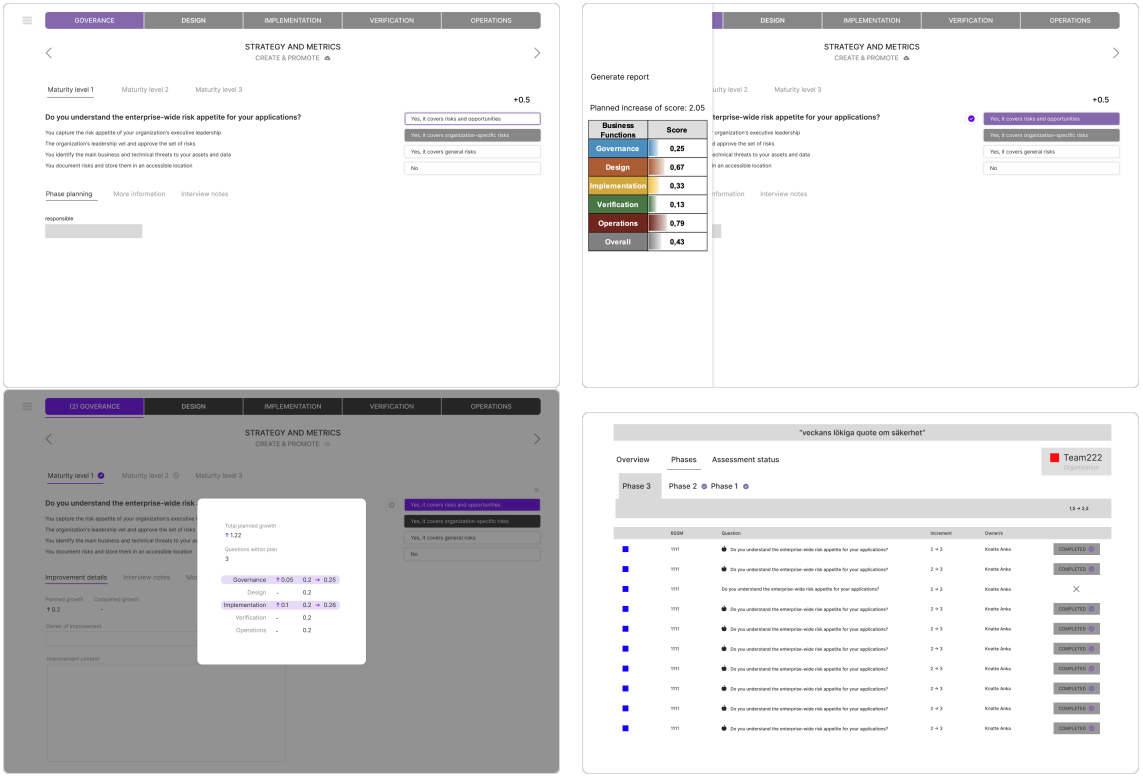


Figure 5.9: Sketches for planning and reviewing a phase

During the transformation from sketches to prototypes a checkout view was introduced to the phase planning, see Figure 5.11. The idea was to think of planning a phase as shopping. The mediator and team members shop questions, put them in their shopping cart, and then checkout the plan to actually start the phase. This was an effective way to restructure the questions from the assessment view of 90 questions to a list form of only a small portion of them.

The list could then be used for the review. This was beneficial since using the assessment view to review up to five questions would result in a lot of unnecessary navigation and non-interactive elements. The checkout view also had the benefit of acting as a summary of the phase with all the content in one place, which was not possible to achieve with the original assessment view.

The view for reviewing, see Figure 5.11, would then contain the list of the questions in the phase with the addition of a done button to mark each question as complete. The team dashboard would visualize the team's current score, their strongest and lowest categories, a radar chart and a card showing the phase progress or encouraging the user to start a phase.

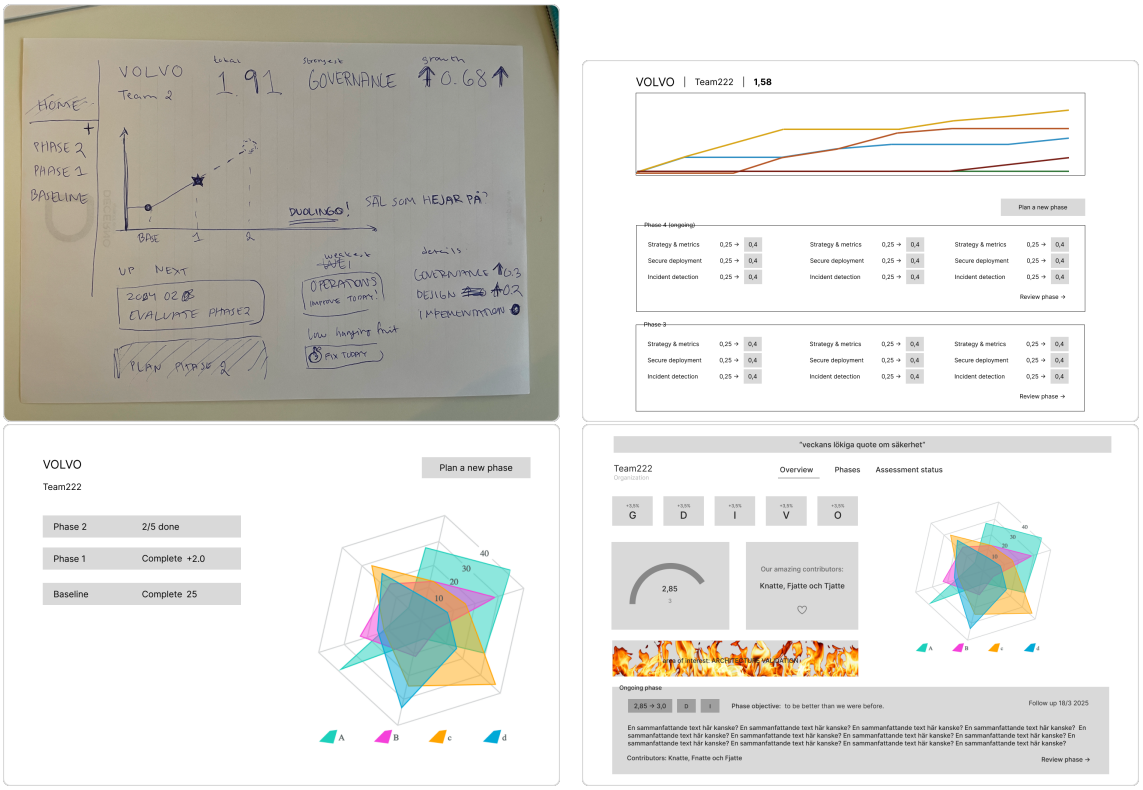


Figure 5.10: Sketches for the team dashboard view

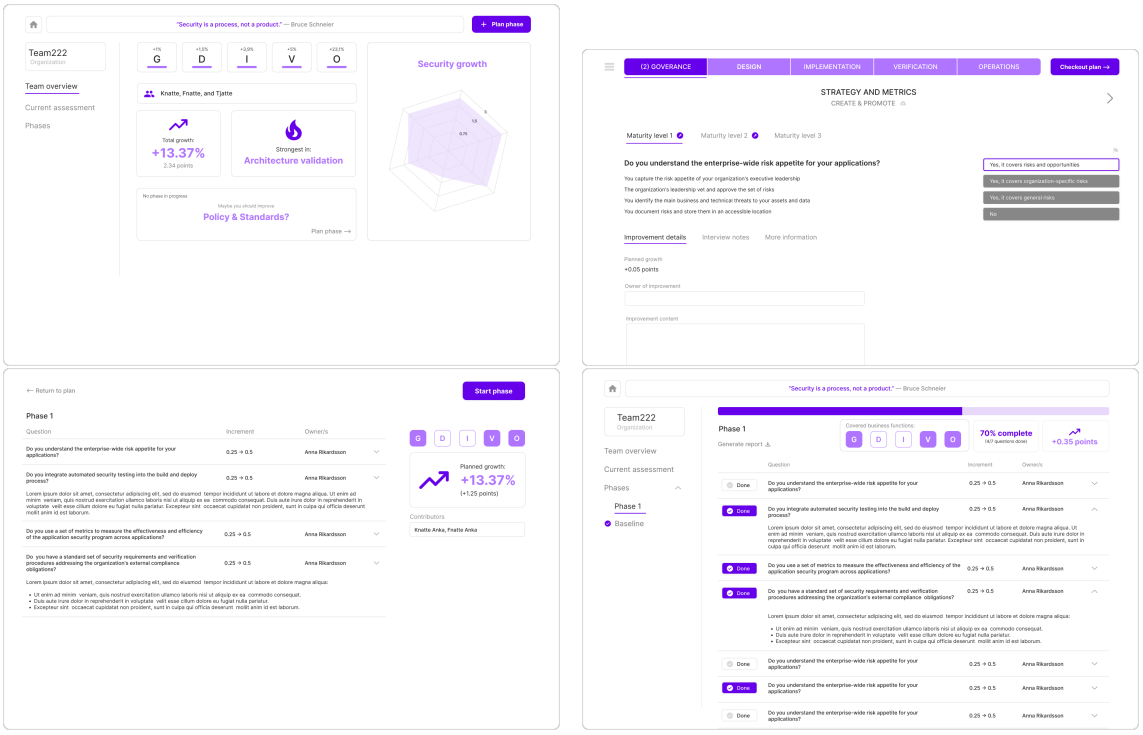


Figure 5.11: Prototypes

5.5.3 MoSCoW

The following procedure resulted in a MoSCoW model for the phases step. This model was used as a foundation for the acceptance criteria in the user stories for the implementation.

Must have:

- Plan a phase: select what to improve, add a responsible person for the task, write a comment.
- Finish the phase: select the plan as done, update each question and get a new score.
- Be able to see both current and previous phase content.

Should have:

- Generate phase report.
- Delete phase.

Could have:

- Auto selection of low-hanging fruit, i.e. improvements that are easy to perform.

Won't have:

- Edit phase content. Be able to go back and add/remove questions from the latest phase.
- Connect phase item with user story.

5.6 Step 4: Overview

The final step was to summarize the collected data and draw conclusions on strengths and weaknesses across the company, as well as to display the different teams and navigate to them. This iteration was the least predetermined since it is not present in the original version of SAMM, and it was therefore much creative freedom during this step.

5.6.1 Investigate

In order to uncover which statistics were the most interesting to display, semi-structured interviews were once again utilized. The interviews were performed with the main stakeholders. This included the primary user, secondary users, as well as management members.

During the interviews, thorough discussions were held regarding which data would be displayed. None of the stakeholders had strong opinions on what type of statistics to display and were very open for suggestions. Therefore, a list of suggestions was made which helped to decide upon statistics valuable for the stakeholders. The following list shows some of the statistics which were discussed.

- Strongest and weakest business function.
- Strongest and weakest security practice.
- Strongest and weakest stream.
- Average score across teams.
- Top three best teams.
- Total growth in points and percent.
- Amount of completed assessments.
- Amount of completed phases.

It was decided to focus on the strongest and weakest security practice since the others seemed either too broad or too granular. The security practice level seemed to be a reasonable measurement of how the teams performed in general. Furthermore, the total amount of completed assessments and phases could be interesting to visualize and act as a motivational factor. The same reasoning goes for the teams' top list, it could be interesting to see which teams scored the best and have them act as an inspiration. It was also decided to calculate the average score across teams in order to see what the general SAMM score standard was.

5.6.2 Prototyping

Previously, a simple landing page had been implemented due to the need for the user to navigate between teams, see Figure 5.12. However, for the sketching and prototyping of this step, the idea was to replace this view with a better one, containing statistics discussed during the interviews.

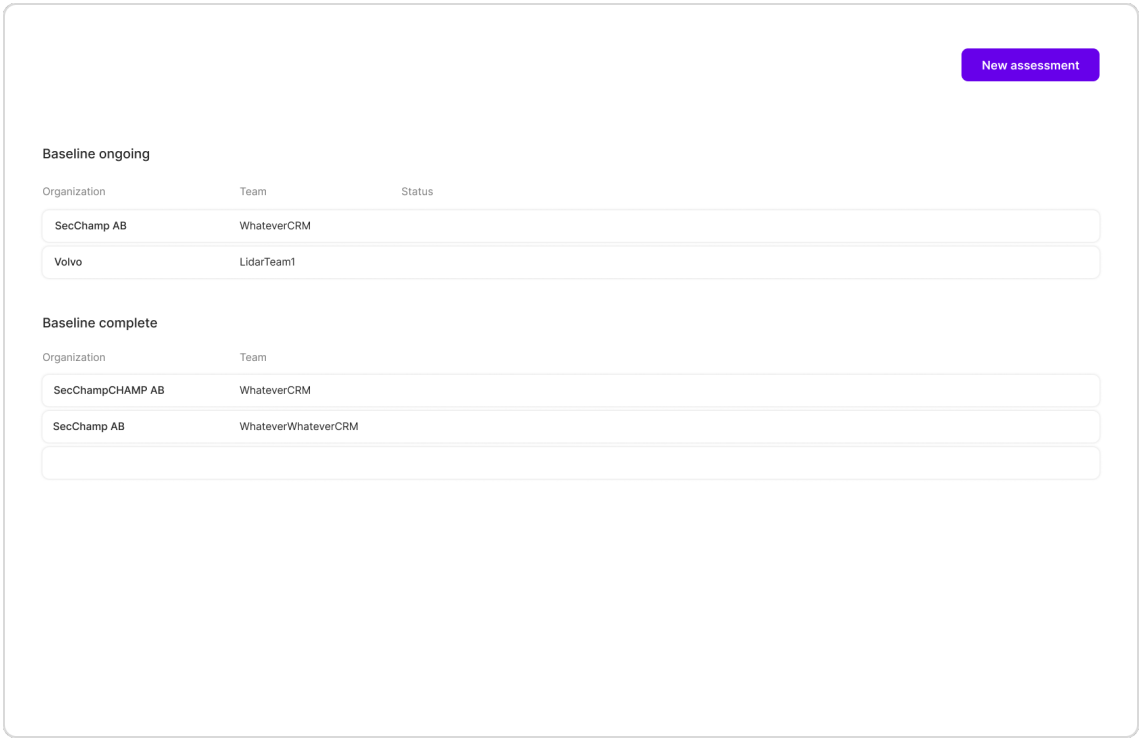


Figure 5.12: Original view of the landing page

An idea tested during the sketching stage was to add tabs to the landing page, one for the list of teams and one for the statistics, as seen in Figure 5.13. When the user would enter the program they would be met with the statistics, and they would then be able to navigate to the tab with the teams listed. An initial idea was also to gather data abstracted onto the business function level, but the data was later presented at a more granular level as mentioned in the previous chapter.

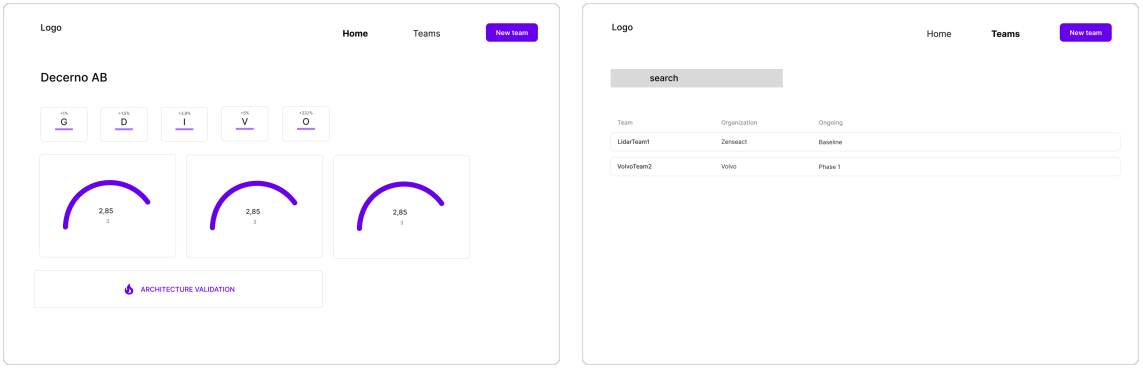


Figure 5.13: Landing page with tabs to divide teams and statistics

Another presented idea, visualized to the left in Figure 5.14, was to have the teams and the statistics on the same page. This idea also explored letting the calculated data take up relatively little space on the screen. The statistics for this view focused on the score increments that had been made throughout the phases and also displayed a card of the team with the highest scores.

In one of the final iterations, displayed to the right in Figure 5.14, the statistics were moved to the right side of the screen. The statistics contained a team top list, both the strongest and weakest security practices, the number of assessments and phases as well as the average score. Two new colors, red and yellow, were also incorporated as well as the concept of taco branding. In addition, an emoji of a taco and one of a chilli were added to make it more fun and to strengthen the branding of Salsa.

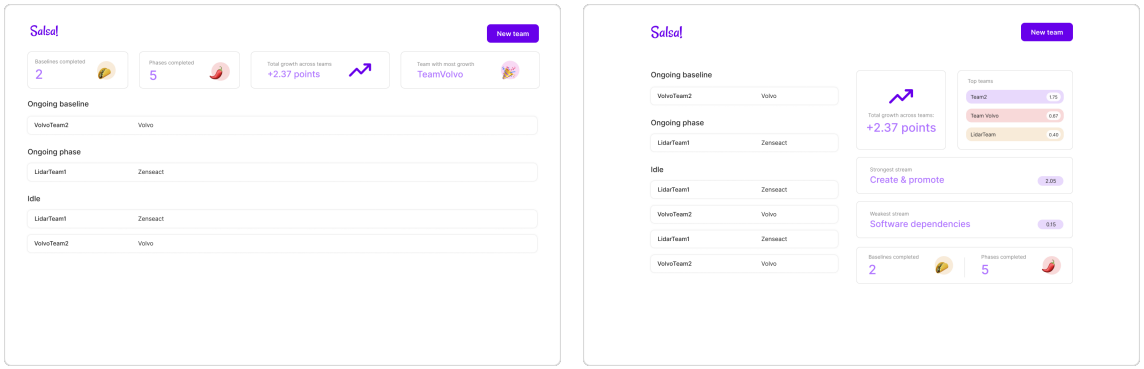


Figure 5.14: Landing page where teams and statistics are merged

5.6.3 MoSCoW

To sum up the findings from the investigate and prototyping stages, the MoSCoW list below was established.

Must have:

- Display all teams

Should have:

- Visualize the average scores of the teams.
- Show the top three performing teams.
- Calculate and visualize the strongest and weakest security practices.
- Show the number of completed assessments and phases.

Could have:

- Dates on when the team information was last updated.
- Tab to About page, in which the SAMM handbook is linked and where Salsa is explained.

Won't have:

- A tutorial on how to use the system.
- Ability to search and filter teams.

5.7 Additional Features

The overview concluded the four steps of this project. However, there was remaining time to implement some additional features to Salsa. The items under the category *Should have* in the MoSCoW were possible options for implementation. The stakeholders also came with their own suggestions on what they would like to prioritize. After some discussion, the additional features that were decided to be implemented were generation of phase reports and Mild Salsa: the option to do the SAMM assessment only with a subset of the 90 questions.

5.7.1 Phase Reports

The report generation, discussed in Chapter 5.4, came to be highly appreciated amongst both primary and secondary users as well as other stakeholders. As further mentioned in Chapter 5.5, these reports were requested also after planning a phase. Because of this, it was decided to implement phase reports in order to efficiently summarize what has been decided to be incremented during the phase.

As for the assessment report, an example of a phase report was available. The phase report was considerably shorter than the assessment report, and since the structure was appreciated it remained largely unchanged. Below is the final structure of the phase report:

1. Phase 1
    (a) Improvements
        - (Governance ⇒ Strategy & Metrics ⇒ Create & Promote) Do you understand the enterprise-wide risk appetite for your applications?
            - Goal: Yes, it covers organization-specific risks
            - Owner: Jane Doe
            - Notes: Here's the plan for what to improve ...

    (b) Changes in score
        - Tables displaying score changes

The first chapter of the phase report lists the questions included in the phase. Connected to each question is the level it is to be incremented to and the person responsible along with a discussion note (if there is one). The second chapter contains two tables, one with the current scores and the other one showing the scores after the phase has been completed, see Table 5.2.

Business Function	Score
Governance	1.04
Design	1.08
Implementation	1.08
Verification	1.16
Operations	1.00
Total Score	1.07

(a)

Business Function	Score
Governance	1.09
Design	1.08
Implementation	1.08
Verification	1.21
Operations	1.00
Total Score	1.09

(b)

Table 5.2: Tables from phase report showcasing the score improvements the phase will result in

Furthermore, the phase report can be generated at any time from when the phase has started. However, it only represents a state stuck in time from when the phase was planned. It will not state whether a task has been completed or not. This feature was not prioritized for implementation due to the report mostly being used to show the plan to stakeholders. In addition, a phase only contains a few questions which are often selected as done at the same time. In other words, a phase is mostly either not completed or fully completed, not something in between. However, this could be something added in future iterations, since Salsa supports selecting each question as done individually during the phase review.

5.7.2 SAMM Question Subset: Mild Salsa

The greatest disadvantage and obstacle when initiating a SAMM assessment, is the heaviness of the original model. The 90 questions can take more than one day of work to complete, and many of them can be difficult to answer. Many software teams do not have a need to fill out the entire form in order for SAMM to still provide value. In addition, the process can be very costly especially for smaller projects.

Due to versatility and a modular mindset when designing the data model, it was possible to customize the question set for different teams. Therefore, it was decided to extend Salsa with an option to choose a smaller set of questions. This variant of the system came to be called Mild Salsa, and it required only minor changes in the interface. Except for the differ in content in the assessment, the mediator could now select question set when creating a new team, see Figure 5.15.

The mediator could now choose to perform the assessment with a smaller question set, and hence introduce a simpler version of SAMM. The data model makes it possible to add the entire set of questions later if the team wants to continue with more questions. This feature will lower the barrier for completing an assessment, increasing the possibility for teams improving the software security of their systems.

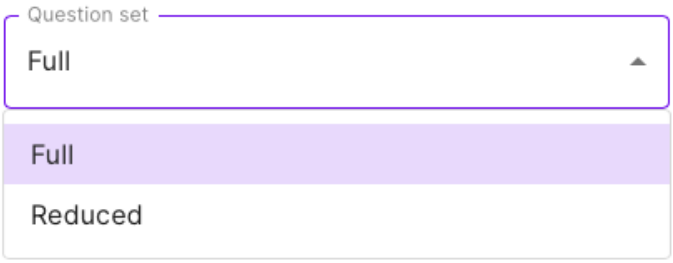


Figure 5.15: Selection of SAMM question set when creating a new team

5.8 Evaluation

The data collected during evaluation was qualitative, which is non-numeric data [45, ch. 9]. Two types of evaluation were utilized for this thesis: continuous evaluation throughout the project and user testing.

Formative evaluation was carried out throughout the whole development process. The users gave feedback on what worked well and what did not after using the current state of Salsa in production. This approach was to be referred as *Continuous Informal Self Report*, due to its implementation. By utilizing this approach, a continuous improvement of the design was ensured.

Towards the end of the project, summative evaluation was carried out where the final result was tested. This was performed through user testing utilizing the think-aloud method and semi-structured interviews. These were performed on both secondary users as well as the primary user. After the testing, thematic analysis was utilized in order to gain insights from the data.

5.8.1 Continuous Informal Self Report

Since the project was utilizing agile methodology, there was an opportunity to always have the current state of Salsa available for the employees at Decerno AB to test out. This made it possible for them to follow the development of Salsa and give feedback. This type of evaluation was informal and little time was spent on actually asking the employees for feedback. Instead, a Microsoft Teams channel [46] was available for them to take initiative and give feedback. However, sometimes feedback was received during times like lunch breaks, coffee breaks or the weekly stand-up meeting with the main stakeholders. These characteristics of the formative evaluation gave it its name *Continuous Informal Self Report*.

During the project, feedback was given from several different people at Decerno AB. Early after the first step, one change that was made due to this evaluation type was the behavior of the navigation arrows on the assessment page. In an early version, the arrows navigated between streams but the user suggested having them navigate through questions instead. This was implemented shortly after due to this being a much more intuitive iteration behavior.

Another change due to feedback from this evaluation method was the performance of the auto-saving feature. One of the testers highlighted the fact that this feature was making unnecessarily many calls to the server. In addition, some testers suggested changing the auto-save icon to something more intuitive like, for example, a text saying saved instead. Since some users had doubts about the meaning of the cloud icon, it was changed to a checkmark icon with a saving text next to it.

The report structure was also improved several times thanks to this evaluation style. Changes were for example the naming of some titles, font styles and visualizing of data. In more detail, there was a request to visualize the scores with two decimals instead of one and to capitalize the text instead of using uppercase.

Additionally, improvements to the team dashboard and the statistics on the landing page were implemented to improve the visualizations of the data. Specifically, changes included adding tooltips for the business function cards on the team dashboard, using security practices across both dashboards and focusing on score in points rather than percent.

Towards the end of the development process, improvements were made to the list view on the landing page. It was suggested to order the teams based on their status: ongoing assessment, ongoing phase and nothing active. In the first iteration, not much time was spent on the list view since other features were prioritized. However, users gave feedback highlighting the importance of having a better list view. A total redesign was implemented and the teams were also alphabetically ordered by team name. The new design proved to be appreciated by the users.

By utilizing this evaluation method, Salsa could continuously be improved throughout the development process, since missteps could quickly be found and fixed. In addition, the employees at Decerno AB could all be involved in the development of Salsa to the degree they wanted to.

5.8.2 User Testing with Primary User

In order to evaluate the final design of Salsa, a user test with the primary user was performed. The purpose of this test was to examine if Salsa is a viable system for the SAMM process. The purpose was also to examine if Salsa fulfills the expectations of the primary user.

The data collected during the user test was qualitative. The test was summative and utilized a semi-structured interview and the think-aloud method. The test was done online, with one participant and two facilitators. One of the facilitators mediated the test and the other one was responsible for recording and observing the test. The timeline of the study was as follows:

1. Prep: The participant is informed about the experiment and its structure. A consent form is sent to the participant and filled in before continuing.

2. Task: The participant completes a number of tasks in Salsa while utilizing the think-aloud method.

3. Interview: The participant takes part in an interview regarding their experience of the tasks.

The tasks for this test covered the whole SAMM process, from completing an assessment to finalizing a phase. The tasks for the test with the primary user were as follows:

1. Here, you can see the landing page. Look around a bit on the statistics for Decerno AB. When you are ready, navigate to the team Car-2.
2. Car-2 is almost done with their assessment. There is also a flagged question since they did not know its answer last session. However, now they know that the answer should be *No*. Find the flagged question and respond to it.
3. Answer the remaining questions and write some comments on them as well.
4. Finish the assessment and download a report.
5. On the team dashboard page, you can see the statistics for the team. Check the teams current score and navigate to plan a new phase.
6. Add two questions to the phase, with a goal, owner and improvement comment.
7. Start the phase.
8. Download a report of the phase.
9. Finalize the phase and check the team score again.
10. Navigate back to the landing page.

Once the tasks were completed, some follow-up interview questions were asked to the participant. The interview questions were as follows:

- How would you describe the experience you just had?
- Was there anything frustrating or challenging during this interaction?
- Would you feel comfortable using a tool like this in real scenarios?
- How do you think this tool could be improved to better support your needs?
- What was the greatest difference between Salsa and Excel?
- Do you want to add anything that we have not covered?

After the test, the recording was transcribed and a thematic analysis was performed. The results from the thematic analysis show that the primary user overall was very satisfied with Salsa. The primary user experienced safe exploration by pointing out the natural flow of moving between the questions and being able to click around freely without crucial consequences. These experiences can be seen in the following two quotes (translated from Swedish).

> *It is a very natural way to navigate between the questions — primary user*
>
> *Salsa beats Excel by a Mile — primary user*

However, one issue for the primary user was the small size of the text and some of the elements. The interface had room for making elements bigger which should be utilized in order to solve this problem. This change would result in a scroll on many views in order to see all content. However, it is more important to properly fulfill the accessibility needs of the user than to avoid a scroll.

The primary user had some improvement suggestions concerning the phase feature. The user requested more flexibility in alternative selection in phases. As of now, the interface limits the user to only select a higher alternative to the questions added to a phase. However, the user would like to only see which alternative is currently reached and then be able to select any alternative, higher or lower, when adding the question to the phase.

In addition, the primary user would like the behavior of alternative selection in a phase to be the same as in the assessment. Currently, a purple border is added to the selected alternative in phase planning, but is filled out completely when selected during an assessment. The idea behind this was to visualize that the alternative is not actually reached during the phase planning, which it is when performing the assessment. However, the primary user was not in need of this and was more positive towards consistency in this regard.

Another request for the phases was to give more feedback when a phase had been completed when selecting the last question as done. The primary user suggested having confetti show up similar to when the last question is answered in the assessment. Furthermore, the page needs to be reloaded in order to show all indications that a phase is completed. A solution for this is simply to add instant feedback instead of requiring a page reload.

The primary user also gave additional ideas, such as changing the numerical visualization of the alternatives (1/4, 2/4, 3/4, 4/4) to the alternative text instead. However, there is too little space for this in the accordion. Solutions could be to add tooltips or try different approaches to numerical visualize the alternative, for example use the score (0.00, 0.13, 0.25, 0.50). The user also gave ideas for more features, such as a feature for removing and editing teams and phases.

However, the main thoughts from the primary user were that Salsa is really pleasant to work with. The user also highlighted that Salsa is much better than Excel, providing a smoother experience with everything in one place. In addition, the user expressed that Salsa will be used with real teams in the near future.

5.8.3 User Testing with Secondary Users

Three user tests with secondary users were also performed in order to evaluate the final design of Salsa. The purpose of these tests was to examine if Salsa is a viable system for the SAMM process. Furthermore, how well Salsa aligned with the expectations of the secondary users would be examined.

The data collected during these user tests was qualitative. The tests were summative and utilized semi-structured interviews as well as the think-aloud method. The tests

were done both in person and online depending on the convenience of the participant. As the user test with the primary user, the tests were done with one participant and two facilitators. One of the facilitators mediated the test and the other one recorded and observed the test. The timeline for the study was as follows:

1. Prep: The participant is informed about the experiment and its structure. A consent form is sent to the participant and filled in before continuing.
2. Task: The participant completes a number of tasks in Salsa while utilizing the think-aloud method.
3. Interview: The participant takes part in an interview regarding their experience of the tasks.

The tasks for the secondary users were not as comprehensive as the tasks for the primary user. This was because the secondary users will not have access to all views of Salsa, since they will mostly be present during the assessment and phases. The tasks for the tests with the secondary users were as follows:

1. Your team is almost done with the assessment. Answer the remaining questions and write some discussion comments on them.
2. Finish the assessment and download a report.
3. Now it is time to plan a phase to increase the score. Navigate to start a new phase.
4. Add two questions to the phase, with a goal, owner and improvement comment.
5. Start the phase.
6. Navigate to the phase that you just created and select it as completed.

After the tasks, some interview questions were asked. The interview questions were as follows:

- How would you describe the experience you just had?
- Was there anything frustrating or challenging during this interaction?
- How do you think this tool could be improved to better support your needs?
- What was the greatest difference between Salsa and Excel?
- Do you want to add anything that we have not covered?

After the tests, the recordings were transcribed and a thematic analysis was performed. The results from the thematic analysis show that the secondary users overall were very satisfied with Salsa. As the primary user, they experienced safe exploration and found it easy to enter data. One participant highlighted that it felt harmless and non-destructive to click around and explore the assessment. They also found Salsa to be better than Excel, as seen in the following quotes (translated from Swedish).

> *Much nicer than the Excel sheet — secondary user*

> *If I had come across this as a tool and it was open source, I would have definitely looked into it — secondary user*
>
> *It's obviously much more enjoyable to manage something in this than in Excel — secondary user*

One highly appreciated feature was the report generation. The feature was described as good, cool and useful. The participants found it easy to find the content in the reports and were satisfied with their structure. The report feature was seen as something new and fun, providing much value to Salsa. In addition, it was highlighted that the report generation feature could save time since it is auto-generated to a great extent. The following quotes show the appreciation of the report generation feature (translated from Swedish).

> *A definite plus is that you can generate this report — secondary user*
>
> *This (the report) is something new and fun — secondary user*
>
> *Wow, this (the report) is really awesome — secondary user*
>
> *This (the report) is useful — secondary user*

There were some external factors influencing the tests, which mostly was the difficulty of fully understanding SAMM. All participants had completed a SAMM assessment before. However, most of them were not recent, which made it difficult to remember the SAMM process. In addition, the complex questions and concepts in SAMM could be distracting for the participants, taking away from the experience. This obstacle was tackled by explaining the SAMM process before the tests and highlighting the purpose of the study not being about SAMM itself but the interface of Salsa.

The thematic analysis showed that the secondary users experienced some issues with feedback, affordance, inconsistency and navigation. Firstly, more feedback was requested when a phase had been completed. In addition, the secondary users wanted it to be more visually clear on the team dashboard view if a phase was ongoing or not. Secondly, some users thought the information cards on the team dashboard looked like buttons and were clickable. However, another participant thought some of the cards looked inactive due to their grey color, and thought they needed to be unlocked with a paid version of Salsa. Thirdly, some confusion arose from the inconsistency of the saving feature. The assessment utilizes auto-save but the phase planning only saves locally and is properly saved first when the phase starts. Finally, there was some confusion concerning the navigation. This was due to the absence of top navigation on the landing page and the difference in navigation between the assessment/phase view and the team dashboard view.

The secondary users also had trouble understanding the text field for corrections. However, they understood the purpose of the corrections text field after an explanation and that it was only for the mediator. One of the testers also expressed concerns about their score being very low when seeing it after finishing the assessment. However, the score was quite high, a bit over 1.00. This speaks towards SAMM's scoring system not being optimal for user engagement.

The thematic analysis showed that the secondary users appreciated the playful aspects of Salsa such as the confetti and security quote. They made suggestions to add confetti when finishing a phase, not just when completing the assessment. In addition, they requested more security-related quotes. Another aspect visual from the thematic analysis was the appreciation of graphs. The secondary users found the radar chart beautiful and wanted it to have additional functionalities, such as moving through the history of previous scores. In addition, the secondary users requested more graphs for Salsa, for example on the checkout plan view.

Finally, the secondary users highlighted the benefit of Salsa in regards of it centralizing the SAMM process. The secondary users said that the assessment is often filled out online, which requires the mediator to share their screen. This is made much easier with Salsa, since now only one program needs to be shared instead of the entire screen. More positive aspects of the comprehensive solution of Salsa were time-saving and frustration elimination of not having to move between different programs and files.

6

Results

This chapter presents the final design of the system as well as the key design guidelines, which are the deliverables of this thesis. The images of the final design are all screenshots from the implemented system instead of high-fidelity Figma prototypes. The guidelines stem from a comprehensive overview of all performed evaluations, reflections on the process as well as discoveries during the implementation.

6.1 Final Design

This section will contain a walk-through from a user perspective of how they will navigate through the application. As mentioned earlier, all images come from an up-and-running web application. All data covered in these images are non-commercial and are made for demonstration purposes only.

The first screen that the user meets is the landing page, see Figure 6.1. The logo in the upper left corner is designed by the application security specialist at Decerno AB. It displays the name Salsa (Simon, Anna, Louise SAMM Assessments) along with its mascot, a seal. The landing page contains statistics on how the teams are performing generally. The statistics present the average score of all teams, the strongest and weakest security practice of the company as well as a top list containing the top three scoring teams. In addition, this view shows how many assessments and phases have been completed.

The main feature of the landing page is however to see all teams in the company and their current status. The teams can be in one of three states: they could be in the process of completing the assessment, they could have a phase in progress or they could simply be finished with the assessment but not have a phase in progress. The status of each team is displayed under the *In progress* title in the list view. Teams with an ongoing phase will have a light red *Phase* tag, teams still working on the initial assessment have a yellow *Assessment* tag and the idle teams have no status indicator.

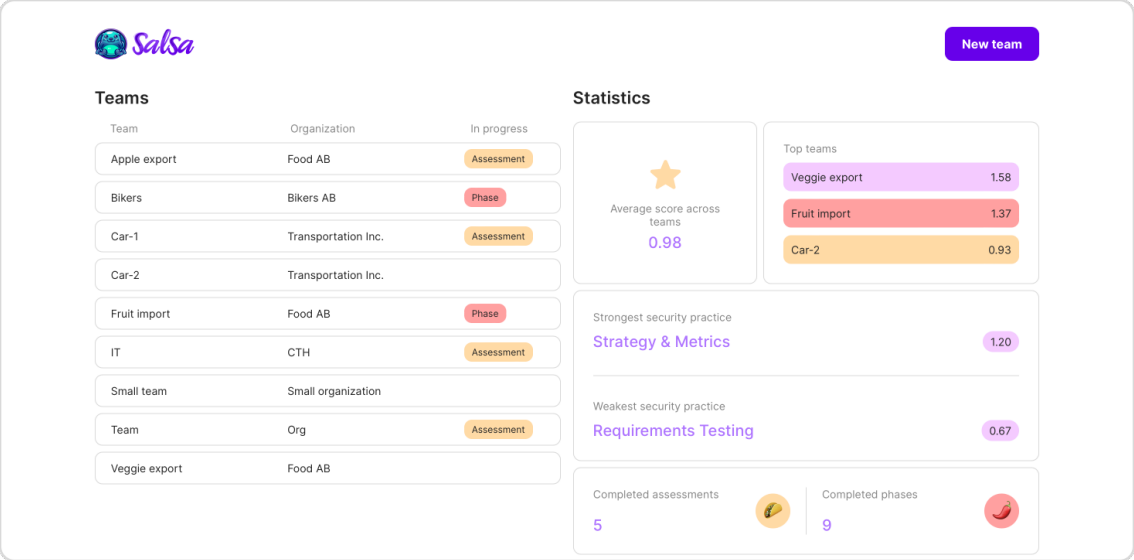


Figure 6.1: Landing page presenting all the teams and statistics for the company

From the landing page, the user can also choose to create a new team, which is done through the *New team* button in the upper right corner. When the user clicks it, a modal prompting the user to create a new team shows up, as seen in Figure 6.2. The modal has input fields for the team name, organization name and contributors to the assessment. In addition, a drop-down is included in order to select the question set. The user can choose between the full set or a subset of the questions.

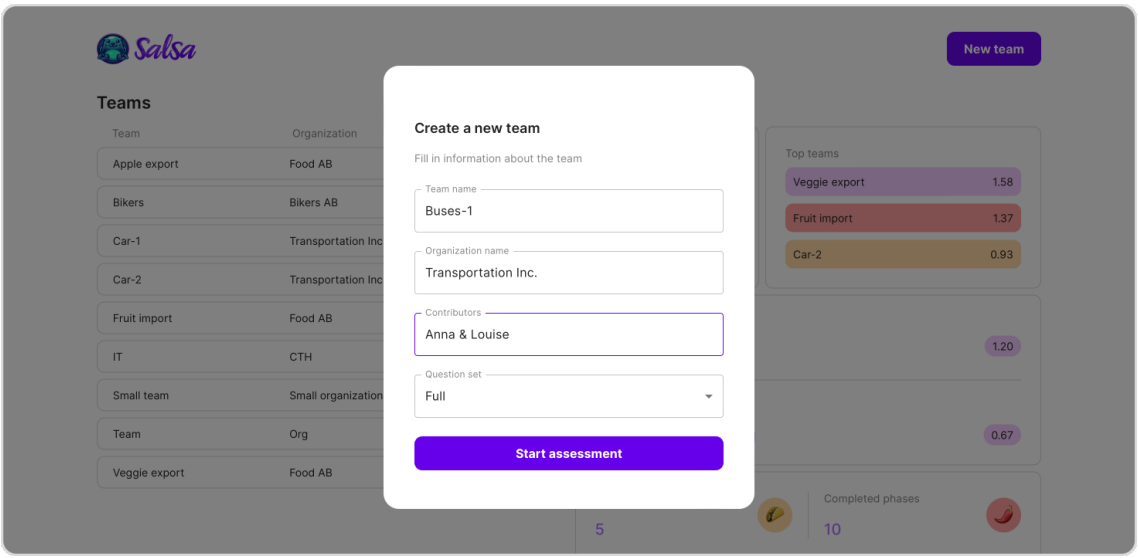


Figure 6.2: Create new team modal

When initiating a new assessment, the user is met with the assessment view, see Figure 6.3. The top navigation bar displays the business function categories of SAMM. When hovering over them a menu folds out, displaying each business function's three security practices and furthermore the two streams of them.

The navigation bar provides an overview of which streams, security practices and business functions are completed through various indices. The streams have checkboxes in front of them if they are completed, and each business function section fills up as the questions are completed one by one. In other words, the navigation bar also acts as a progress bar.

Furthermore, the user can safely return to the landing page through the home button in the upper left. Once all questions are answered, the user can finalize the assessment by clicking *Finish* in the upper right corner.

Below the top navigation bar, the user can see the current security practice and stream. Along with this, the current maturity level is visualized by a purple underline. Each maturity, as mentioned earlier, has its question, and the user can navigate between them by using the arrows or clicking on one of the other maturity levels tabs.

The rest of the page is dedicated to the question, starting with the actual question and its criteria. Next to it, the user can select one of the four alternatives to answer it as well as a flag to mark the question for later if the team is unsure of their current status. In the two tabs below, the user can alter between *More information* and *Interview notes*.

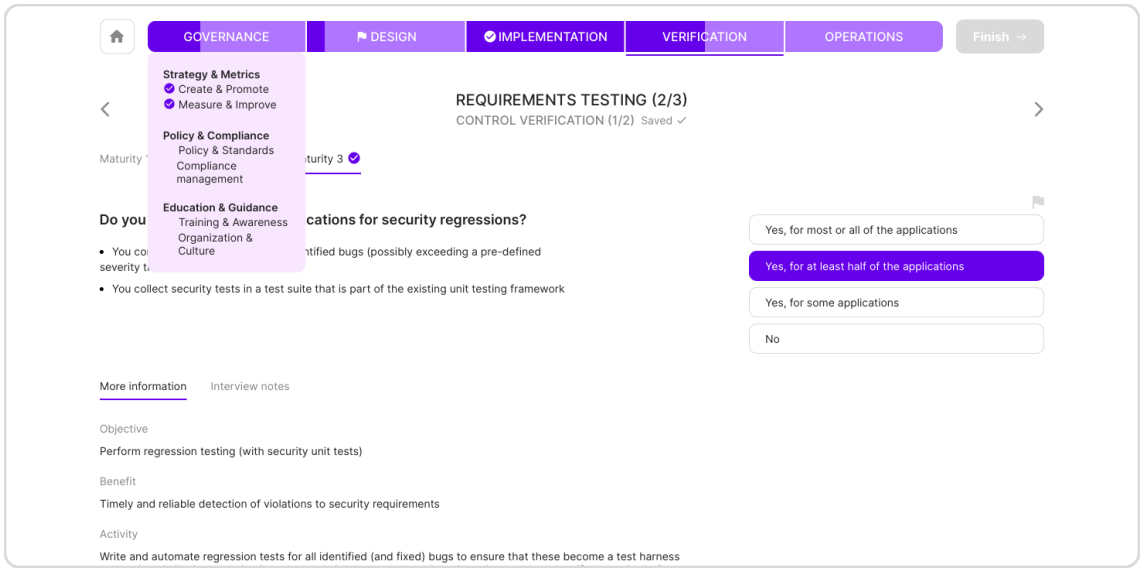


Figure 6.3: Assessment view

Under the first one, the user receives additional context to the question such as the *Benefit*, *Objective* and *Activity* sections. Under the second tab, the user has the option to write comments in the text fields *Discussion*, *Improvements* and *Corrections*. The user is supposed to write down the discussions that were held under the discussion field, and to put proposed improvement suggestions under *Improvements*. The *Corrections* text field is meant for the mediator, and its purpose is to contain an explanation if a score is changed by the mediator after the assessment meetings are over.

Once the team is finished with the assessment, a splash of confetti shows up to congratulate the team. The mediator can then click on the finish button and finalize the assessment, see Figure 6.4. They can choose to download the report and see their results within the different categories. In the report, the summary of the scores in maturities, streams, security practices, business functions as well as the total score are visualized in different graphs. Furthermore, the report contains all questions with answers, as well as all discussions, improvements and corrections.

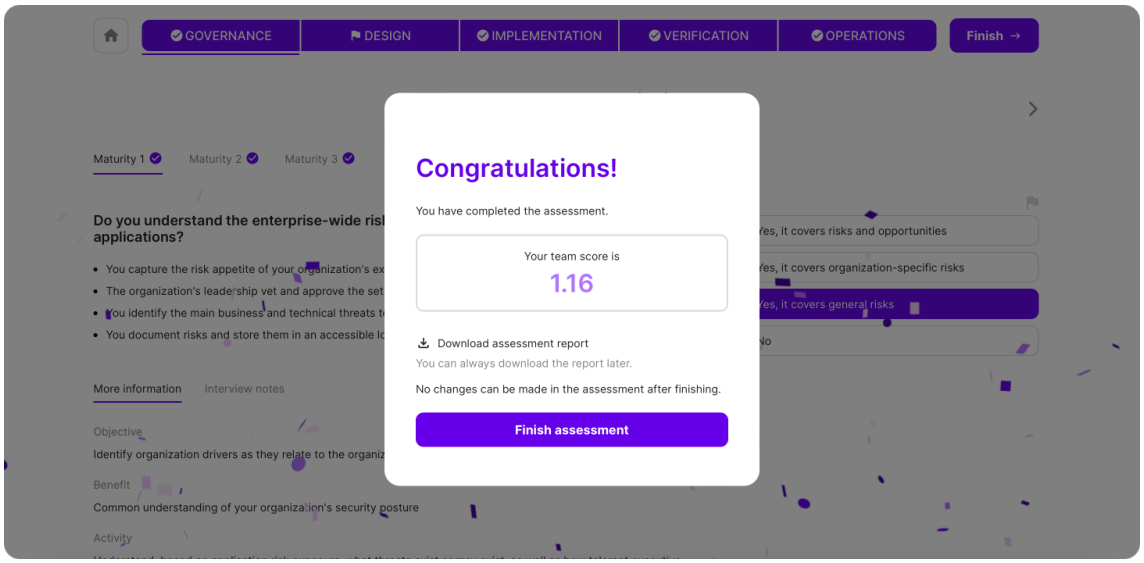


Figure 6.4: Finishing modal

When the team is finished with the assessment, they are navigated to their team's dashboard page, seen in Figure 6.5. The team dashboard displays the current state of the team in the SAMM assessment process. The user can navigate back to the landing page by clicking on the home button in the top left corner. The top of the page also displays a security quote and to the right is a button for the user to start a new phase from.

The sidebar to the left contains navigation to one of the previously performed phases or the initial assessment. It also displays with a checkmark icon if a phase is completed or in progress. On top of the sidebar is the name of the team and organization visible.

The remainder of the team dashboard is occupied by statistics and visualization of how the team is currently performing according to SAMM. It displays the team's score in the different business functions, the contributors, the current total score and the team's strongest security practice. Next to these cards, a radar chart displays the team's score in all 15 security practices.

Finally, a card intended for phase visualization is presented at the bottom of the page. If a phase is in progress it shows how many of the tasks have been completed and a button to review the phase. If a phase is not in progress, it instead shows the team's weakest security practice and prompts the team to increase it.

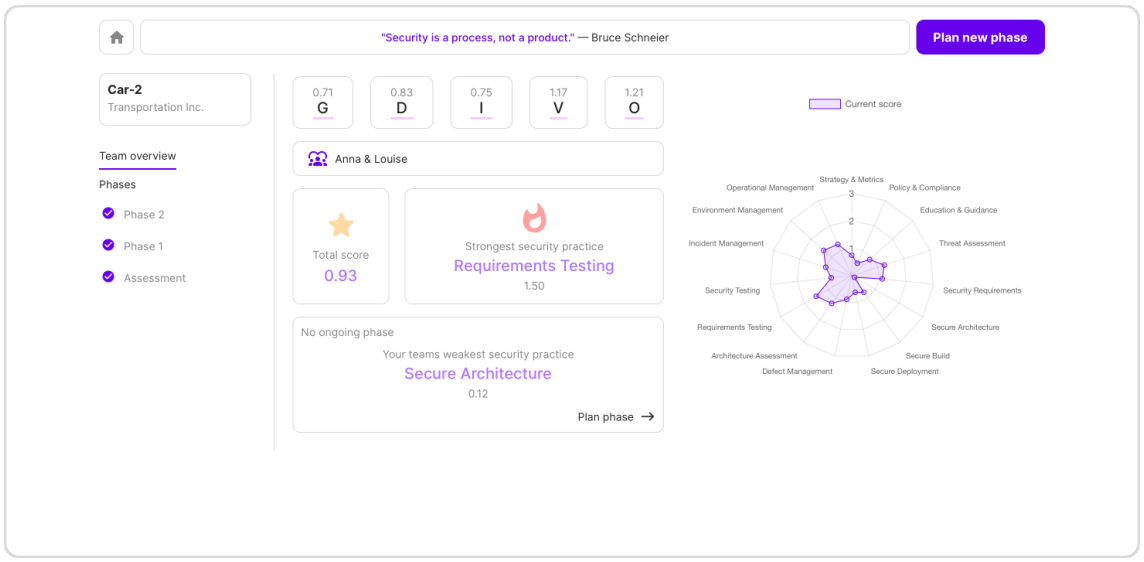


Figure 6.5: Team dashboard page

If the team decides to plan a new phase, they are met with the view displayed in Figure 6.6. The view is similar to the one where the assessment is carried out, but instead of the progress indication at the top navigation bar, each business function displays how many questions are chosen to be improved within that category. When a question has been selected for improvement within a business function, the entire section gets filled. The finish button is also replaced with a button named *Checkout plan*, and the home button to the left is replaced with a back button to indicate that it leads back to the team dashboard page instead of the landing page. All other navigation remains the same.

Furthermore, the currently reached alternative and the alternatives below are greyed out to show that they cannot be selected in order to reach a higher score. The option to flag a question is also gone since there should be no need to return later to update something. To visualize that this is a plan and not an implemented change, the selected alternative receives a border instead of changing color when selected.

In addition to the previously mentioned tabs *More information* and *Interview notes*, a new tab has appeared named *Improvement plan*. Under this tab, the user can insert the name of the person responsible for implementing the change, as well as notes from the plan meeting. The growth in points is also presented here.

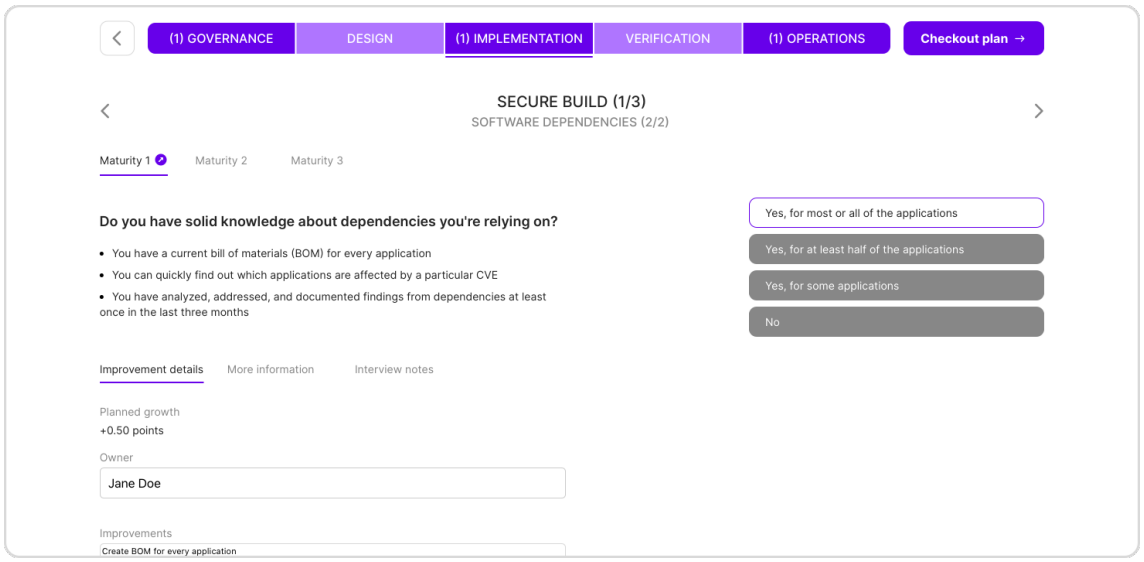


Figure 6.6: In progress of creating a new phase

When the team feels satisfied with the questions they have selected for the phase, they go to the checkout plan page which can be seen in Figure 6.7. This view presents an overview of the questions that have been selected from the previous view. A table lists the chosen questions, their owner as well as the goal that is to be reached. If clicking on the accordion arrow, the team can also see the phase plan notes.

To the right, some statistics are shown over how many points the team will gain in total, and how many percent the increase will result in. If needed, the team can return to the previous view and update the questions and their comments or owners. Once satisfied with the plan, the team finishes the planning stages by clicking *Start phase*.

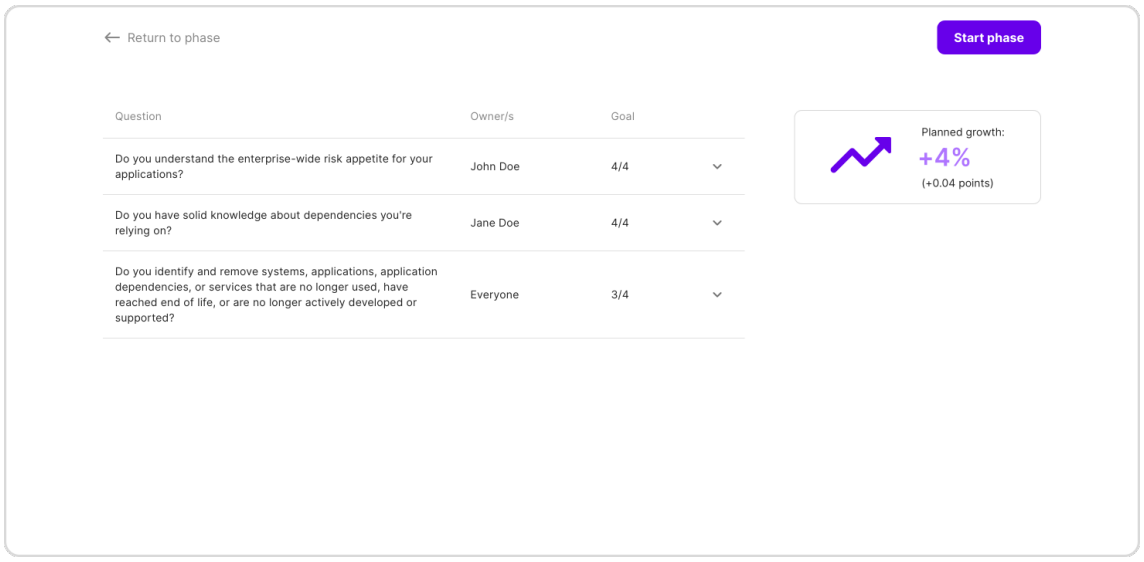


Figure 6.7: View for reviewing the questions selected in phase planning

When a team has a phase in progress, their next step is to review it once tasks in the phase have been completed, see Figure 6.8. At the top, a progress indicator displays how far the team has come in the process of completing the tasks in the phase. The user can also see the increment in points and percent on this page. In addition, the user can choose to download a report summarizing the phase. The report contains a summary of all the questions to improve, their owners and notes that were taken during the meeting session. Once a phase is completed, a new one can be started. Phases created before the newest one cannot be edited, however their report can still be downloaded.

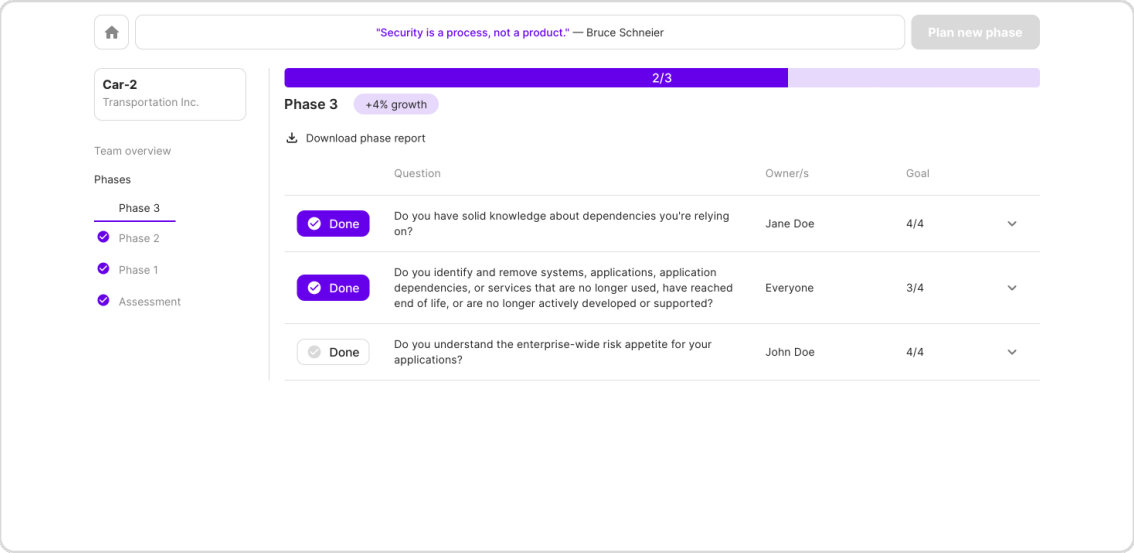


Figure 6.8: Review ongoing phase

These screens conclude the features of Salsa. Salsa provides a platform for managing SAMM assessments and phases. With its report generation features and by gathering the whole process in one place Salsa can contribute towards improving software security management.

6.2 Guidelines

The following design guidelines provide recommendations in how to properly design for a comprehensive OWASP SAMM tool.

1. Provide clear navigation for the large tree structure of SAMM
2. Design with careful consideration to which type of user will utilize the view
3. Provide visual feedback to display the current state
4. Have a modular mindset and design for change when utilizing a large data set
5. Do not underestimate data-driven visualizations
6. Employ fun design elements

1. Provide clear navigation for the large tree structure of SAMM

During the assessment, it is important to be able to properly navigate between the questions. However, the SAMM tree structure is large and it can be difficult to understand the naming of categories. It is therefore important to take extra care when designing the navigation.

For Salsa, the top navigation bar visualizes all categories in their place in the tree. Therefore, apart from providing navigation it also helps the user to understand the tree structure. Salsa also provides several ways to navigate: navigation bar, arrows and tabs. They have different strengths and offer the user a variety of navigation options.

2. Design with careful consideration to which type of user will utilize the view

Different people are involved throughout the SAMM process. In addition, some tasks are done by one person while other tasks, such as the assessment, are done as a group. As a result, the views of Salsa are relevant for different users. In more detail, the assessment view will be used by the primary user and secondary users together. On the other hand, the landing page and report generation will often only be used by the primary user. It is therefore important to specify the user of each view before designing and to think of their specific needs when prototyping the view.

3. Provide visual feedback to display the current state

A result of the evaluation was that the users felt uncertainty concerning the current state of the application. For example, users failed to recognize the saving icon or when a phase was completed. This guideline therefore highlights the importance of providing visual feedback to the user. As designers, it is taken for granted what happens upon action. However, this is not obvious to users and should hence be presented to them.

4. Have a modular mindset and design for change when utilizing a large data set

One of the main issues with SAMM is the assessment's long process of completion. Throughout this project, it was realized that many teams would benefit from iterating over a subset of the questions. This was easily adjusted in Salsa due to a modular mindset when designing the user interface as well as the data model. In addition, it is simple to make changes to individual questions or the tree structures due to this mindset.

To design for change is therefore a main takeaway from this thesis. This includes designing dynamic list options to ensure a comprehensive user flow if the amount of items changes, and to allow different concatenations of questions and answers in the database.

5. Do not underestimate data-driven visualizations

One of the insights from the user tests was that the included graphs often resulted in positive exclamations. Furthermore, the participants usually asked if the graphs could be further extended. This implied that conventional data-driven design such as graphs can provide plenty of value for users, and is therefore included as a guideline for future design projects.

6. Employ fun design elements

This guideline highlights the importance of incorporating fun elements in the design. Software security is a serious and difficult subject, which can affect user engagement negatively during the SAMM process. It is therefore beneficial to add fun elements, such as the security quote on the team dashboard or the confetti when finishing an assessment, to ease the seriousness of the occasion.

In addition, fun elements can help in inviting the user and making them more comfortable. This is beneficial for the SAMM process since SAMM has a high threshold for users to understand and work with. Making the process more inviting and comfortable is therefore important and can be made by incorporating fun design elements.

7

Discussion

This chapter discusses the project results which are the platform Salsa and design guidelines for a comprehensive OWASP SAMM tool. In more detail, it is reviewed how these contribute towards the goal of improving software security. This chapter also discusses some aspects of SAMM that could be affecting the results. This could for example be the score system, which can be perceived as low. Furthermore, the combination of agile methodologies and design sprints is discussed as well as some possible future work topics.

7.1 Project Results

After analysing the evaluation, it is shown that the developed product, Salsa, successfully covers the important features of a comprehensive SAMM assessment tool. This was achieved by interviewing the primary user and some secondary users to gain insights into the SAMM process early in the process. The selection of additional features was also highly successful. When users tried out a well-functioning version of Salsa, they could easily concretize which additional features they wanted. The Mild Salsa feature is a perfect example of this. This idea came later in the project and was proposed by the primary user. This highlights the importance of continuously involving the user since the initial interviews can miss valuable insights that will arise when the product starts to form. Furthermore, this insight highlights the benefit of not only designing prototypes but also implementing a functional product.

A system for the SAMM process could contain several features with different goals, users and complexities. In order to be able to design and implement a successful product covering the whole SAMM process during the timeframe of this thesis, delimitations had to be set. It was therefore decided to divide the product into four steps which were to be successful. The additional features were also highly appreciated by the users. However, more features could be added to Salsa. Therefore, in order to continuously improve Salsa after this project it will be further implemented at Decerno AB. This will also contribute to the purpose of the thesis to encourage secure software.

From the evaluation, it was seen that the users were satisfied with Salsa. The primary user was very satisfied which might be due to the utilization of participatory design. A fear was that the primary user's perspective had been prioritized over secondary users'. However, the user tests showed that the decision to mainly focusing

on one user through participatory design did not significantly impact the secondary users negatively. The positive results can also be explained by the involvement of some secondary users throughout the project. They were for example from the start included to try out the current stage from Salsa. They were also involved in initial interviews and in some of the steps, for example, assessment, phases and overview. However, secondary users could sometimes experience difficulties with understanding SAMM due to its complexity. They were also more prone to ask for more graphs in Salsa and improvements in visibility and feedback.

The guidelines were carefully selected from the project process, evaluation results and Salsa. The guidelines are specific to SAMM, but many of them would probably be relevant for similar situations as well, such as other security models or interview processes. The guideline *Employ fun design elements* would for example be relevant for other situations in order to increase user engagement.

Navigation is always important to get right, however, SAMM is unique with its tree structure. Other linear security models would not need to consider navigation to the same degree as SAMM. From the evaluation, it was seen that users liked graphs, which probably is true for other situations as well. Having a modular mindset will be important in other situations where large datasets are handled. Visible feedback on current states will also be relevant where a complex process with different states is handled, such as the phases in SAMM.

Knowing which type of user will use the design is always important to know. However, SAMM provides a unique case where some features are handled individually and some in groups, between different types of users. Even though it can be argued that the guidelines are relevant for similar situations as for this thesis, there are no guarantees that they will fit perfectly.

7.2 SAMM Difficulties

An issue noticed during the project was the high threshold to understand SAMM and its process. This was not a problem for the primary user, but for the secondary users. They requested to have Salsa teach them SAMM, such as explaining the relations between questions. This could for example be implemented with a note connected to each question presenting its similarities and differences to other questions. During the evaluation with the secondary users, it was noticed that some put a lot of effort in trying to understand the questions and the SAMM process. This indicates that SAMM itself could affect user satisfaction with Salsa negatively.

Another problem connected to SAMM was the scoring system. It was seen that the score being so low, between 0.00 and 3.00, could disengage users. In addition, SAMM calculates the average of all categories to obtain the total score, instead of adding them. As a result, the growth of phases becomes very low. Another problem with using the average is that it causes the scores to be very difficult to predict and understand. This is especially true when working with phases since team members can not easily see if the score is accurately calculated. It is also very disengaging to see a growth of 0.5 on a question converting to a growth of 0.1 in the total score.

A possible solution for the scores to increase user engagement could be to simply make the scores higher, for example, an interval of 0 to 100 instead. Also to obtain the total score, addition should be used instead of average in order to make it easier to predict and understand. The average could be used on some of the lower categories, but at least the business function scores should be added in order to obtain the total score. This would make it more fun and easy to follow the score throughout phases.

7.3 Integrating Agile with Design Sprints

During this project, both agile methodologies and design sprint methods such as the double diamond framework were utilized. While the design sprint methods are meant to suit the prototyping stage, agile methodologies are intended for software development. As previously mentioned, the agile framework presents a process where small iterations are progressively implemented in comparison to a waterfall strategy where the final product is delivered at once. Furthermore, design sprints are performed in iterations as well, but each iteration increases the fidelity of the product.

The result of these two commonly used frameworks is that the design sprints are usually conducted before the agile sprint. One of the main benefits of agile methodologies is that changes are easily adopted and grant architectural freedom to the developers. However, with a process where prototypes are produced before initiating an agile sprint, a large part of the process is carried out through a waterfall mindset.

To partly counteract this, the design part of this process was divided into four design iterations instead of prototyping the entire system before initiating implementation. This made the process more agile and iterative. It also simplified change by breaking the design process into smaller parts.

However, once a prototype is implemented and one wants to revise it, it can be difficult. Due to this, it is understandable that a large part of the design work is completed and agreed upon before initiating the agile development process. Nevertheless, during this project, it was not possible to merge the design sprint and agile methods in order to make the entire process agile.

7.4 Future Work

For future work, effort could be put into further developing and improving Salsa. First, the SAMM scoring system should be updated to a more intuitive and fun system. In addition, gamification should be researched in order to make a proper decision. Gamification should also be investigated to see how it can be added to other aspects of Salsa to enhance user engagement. Finally, more features could be added to Salsa. These features could for example be from the MoSCoW documents that were not implemented during this project. The most prioritized features includes an

authorization system for accessing the different resources and functions for deleting and and editing phases. Other aspects of Salsa could be upgraded, such as adding templates and language selection to the report generation feature.

As mentioned before mostly one user was taken into consideration throughout this project. For future work, more users could involved, both other primary users and more secondary users. This would strengthen the research and perhaps even highlight additional guidelines and features for Salsa.

Finally, another interesting idea for future work would be to investigate other established security models. Both investigate how well the guidelines from this project apply to other security models, but also bring forth new guidelines for them. This would effectively work towards the purpose of this thesis, making software security more accessible which as a result will lead to more secure products.

8

Conclusion

The purpose of this project was to improve software security management by centralizing the SAMM assessment procedure. This was done by identifying and solving usability issues in the process of working with SAMM at Decerno AB. In order to handle these objectives, a research question was formulated:

What are the key design guidelines for a comprehensive OWASP SAMM assessment tool?

By utilizing concepts and methods such as participatory design, user-centered design, agile software development the SAMM assessment tool Salsa was developed. Salsa is a website which provides the entire SAMM process to be done in one place. The design guidelines diverged from the evaluations of Salsa with its users, reflections from the process and discoveries during implementation. Six key design guidelines, as can be seen in Chapter 6.2, were set:

1. Provide clear navigation for the large tree structure of SAMM
2. Design with careful consideration to which type of user will utilize the view
3. Provide visual feedback to display current state
4. Have a modular mindset and design for change when utilizing a large data set
5. Do not underestimate data driven visualizations
6. Employ fun design elements

Even though these guidelines are specific to the SAMM process, they could be relevant for similar situations as well. As for future work, improvements could be made to Salsa such as exploring gamification. SAMM could also be changed to further enhance user engagement. Furthermore, other security models could be investigated to broaden the scope. However, this thesis presents concrete guidelines about what is important for a SAMM assessment tool regarding usability. In the future, the developed system Salsa might also be available open source for anyone who wants to improve software security in their projects.

Bibliography

[1] European Commission, *The eu cybersecurity act*, 2023. [Online]. Available: https://digital-strategy.ec.europa.eu/en/policies/cybersecurity-act (visited on 12/01/2023).

[2] O. Hanka, *What the eus cyber resilience act means for companies producing / selling digital products and why companies from outside the eu are affected as well*, 2023. [Online]. Available: https://www.linkedin.com/pulse/what-eus-cyber-resilience-act-means-companies-producing-hanka/ (visited on 12/01/2023).

[3] M. Harbach, S. Fahl, and M. Smith, "Who's afraid of which bad wolf? a survey of it security risk awareness.," *2014 IEEE 27th Computer Security Foundations Symposium, Computer Security Foundations Symposium (CSF), 2014 IEEE 27th, Computer Security Foundations Symposium (CSF), 2012 IEEE 25th*, pp. 97–110, 2014, ISSN: 978-1-4799-4290-9.

[4] OWASP, *Owasp samm*, 2022. [Online]. Available: https://owasp.org/www-project-samm/ (visited on 12/01/2023).

[5] OWASP, *About the owasp foundation*, 2024. [Online]. Available: https://owasp.org/about/ (visited on 02/05/2024).

[6] Decerno, *Decerno company website*. [Online]. Available: https://www.decerno.se/ (visited on 12/12/2023).

[7] Addnode Group, *Addnode group website*. [Online]. Available: https://www.addnodegroup.com (visited on 04/24/2024).

[8] Codific, *Welcome to sammy*, https://sammy.codific.com/. (visited on 02/12/2024).

[9] OWASP, *Owasp top ten web application security risks*, 2024. [Online]. Available: https://owasp.org/www-project-top-ten/ (visited on 02/05/2024).

[10] OWASP, *Web security testing guide*, 2024. [Online]. Available: https://owasp.org/www-project-web-security-testing-guide/ (visited on 02/05/2024).

[11] A. Ramirez, A. Aiello, and S. J. Lincke, "A survey and comparison of secure software development standards," in *13th CM Conference on Cybersecurity and Privacy (CM) - Digital Transformation - Potentials and Challenges*, Kenosha, WI, USA: IEEE, 2020, pp. 1–10. DOI: 10.1109/CMI51275.2020.9322704. [Online]. Available: https://ieeexplore.ieee.org/document/9322704.

[12] P. Chandra, *Author page at opensamm*, 2024. [Online]. Available: https://www.opensamm.org/author/chandra/ (visited on 02/05/2024).

[13] OWASP SAMM, *Software assurance maturity model*, 2024. [Online]. Available: https://owaspsamm.org/model/ (visited on 02/05/2024).

[14] OWASP, *Owasp samm version 2*. [Online]. Available: https://drive.google.com/file/d/1cI3Qzfrly_X89z7StLWI5p_Jfqs0-OZv/view?usp=sharing (visited on 02/05/2024).

[15] OWASP SAMM, *Quick start guide - owasp software assurance maturity model*, 2024. [Online]. Available: https://owaspsamm.org/release-notes-v2/ (visited on 04/03/2024).

[16] OWASP SAMM, *Quick start guide - owasp software assurance maturity model*, 2024. [Online]. Available: https://owaspsamm.org/guidance/quick-start-guide/ (visited on 02/05/2024).

[17] D. Fucci, E. Alégroth, M. Felderer, and C. Johannesson, "Evaluating software security maturity using owasp samm: Different approaches and stakeholders perceptions," *The Journal of Systems and Software*, vol. 214, no. 112062, 2024, ISSN: 0164-1212. DOI: 10.1016/j.jss.2024.112062. [Online]. Available: https://doi.org/10.1016/j.jss.2024.112062.

[18] J. Nielsen, *10 usability heuristics for user interface design*, 2024. [Online]. Available: https://www.nngroup.com/articles/ten-usability-heuristics/ (visited on 04/24/2024).

[19] M. J. Muller, "Participatory design: The third space in hci," in *The human-computer interaction handbook: fundamentals, evolving technologies and emerging applications*, J. A. Jacko and A. Sears, Eds., L. Erlbaum Associates Inc., 2002, pp. 1051–1068.

[20] E. Sanders and P. J. Stappers, "Co-creation and the new landscapes of design," *CoDesign*, vol. 4, pp. 5–18, Mar. 2008. DOI: 10.1080/15710880701875068.

[21] A. Ramirez, A. Aiello, and S. J. Lincke, "A survey and comparison of secure software development standards.," *2020 13th CMI Conference on Cybersecurity and Privacy (CMI) - Digital Transformation - Potentials and Challenges(51275), Cybersecurity and Privacy (CMI) - Digital Transformation - Potentials and Challenges, 2020 13th CMI Conference on*, pp. 1–6, 2020, ISSN: 978-1-7281-9056-3.

[22] I. Bongiovanni, "Designing user-centric information security management systems in financial services organisations," in *2020 IEEE 6th International Conference on Collaboration and Internet Computing (CIC)*, 2020, pp. 192–199. DOI: 10.1109/CIC50333.2020.9492732.

[23] S. Pearman, E. Young, and L. F. Cranor, "User-friendly yet rarely read: A case study on the redesign of an online hipaa authorization," *Proceedings on Privacy Enhancing Technologies*, vol. 2022, no. 3, pp. 558–581, 2022. DOI: 10.56553/popets-2022-0086.

[24] R. Garett, J. Chiu, L. Zhang, and S. D. Young, "A literature review: Website design and user engagement," *Online J. Commun. Media Technol.*, vol. 6, no. 3, pp. 1–14, 2016.

[25] Interaction Design Foundation, *What is user centered design (ucd)?* 2016. [Online]. Available: https://www.interaction-design.org/literature/topics/user-centered-design (visited on 02/03/2024).

[26] M. Maguire, "Methods to support human-centred design," *International Journal of Human-Computer Studies*, vol. 55, no. 4, pp. 587–634, 2001, ISSN: 1071-5819. DOI: https://doi.org/10.1006/ijhc.2001.0503.

[27] P. Debney, "2.2 the design process.," in *Computational Engineering*. Institution of Structural Engineers (ISTRUCTE), 2020, ISBN: 978-1-5231-3543-1.

[28] J. DaSilva, *A guide to competitive analysis for ux design*, 2023. [Online]. Available: https://bootcamp.uxdesign.cc/a-guide-to-competitive-analysis-for-ux-design-1ddafeb9a3e7 (visited on 03/14/2024).

[29] K. Moran and K. Gordon, *How to conduct a heuristic evaluation*, 2023. [Online]. Available: https://www.nngroup.com/articles/how-to-conduct-a-heuristic-evaluation/ (visited on 04/26/2024).

[30] C. Wilson, *Interview Techniques for UX Practitioners*. Elsevier Inc., 2014, ISBN: 9780124103931.

[31] R. F. Dam and T. Y. Siang, *Personas a simple introduction*, 2024. [Online]. Available: https://www.interaction-design.org/literature/article/personas-why-and-how-you-should-use-them (visited on 03/14/2024).

[32] K. Kaplan, *User journeys vs. user flows*, 2023. [Online]. Available: https://www.nngroup.com/articles/user-journeys-vs-user-flows/ (visited on 03/14/2024).

[33] Interaction Design Foundation, *Prototyping*, 2024. [Online]. Available: https://www.interaction-design.org/literature/topics/prototyping (visited on 04/24/2024).

[34] Figma, Inc., *Figma: The collaborative interface design tool*, 2024. [Online]. Available: https://www.figma.com (visited on 04/24/2024).

[35] K. S. Ahmad, N. Ahmad, H. Tahir, and S. Khan, "Fuzzy moscow: A fuzzy based moscow method for the prioritization of software requirements.," *2017 International Conference on Intelligent Computing, Instrumentation and Control Technologies (ICICICT)*, pp. 433–437, 2017, ISSN: 978-1-5090-6106-8.

[36] M. D. Green, *Scrum : Methods for Agile, Powerful Development.* SitePoint, 2016, ISBN: 9781457199486.

[37] Alita Joyce, *Formative vs. summative evaluations*, 2019. [Online]. Available: https://www.nngroup.com/articles/formative-vs-summative-evaluations/ (visited on 02/05/2024).

[38] UserTesting, Inc., *User testing guidelines*, 2024. [Online]. Available: https://www.usertesting.com/resources/topics/user-testing-guidelines (visited on 04/24/2024).

[39] J. Nielsen, *Thinking aloud: The #1 usability tool*, 2024. [Online]. Available: https://www.nngroup.com/articles/thinking-aloud-the-1-usability-tool/ (visited on 05/20/2024).

[40] Scribbr, *Thematic analysis*, https://www.scribbr.com/methodology/thematic-analysis/, 2019. (visited on 02/05/2024).

[41] GitHub Inc., *Github: A platform for version control and collaboration*. [Online]. Available: https://github.com (visited on 04/29/2024).

[42] Facebook, *React: A javascript library for building user interfaces*. [Online]. Available: https://reactjs.org (visited on 04/29/2024).

[43] Microsoft Corporation, *.net: A free, cross-platform, open source developer platform*. [Online]. Available: https://dotnet.microsoft.com (visited on 04/29/2024).

[44] PostgreSQL Global Development Group, *Postgresql: The world's most advanced open source relational database*. [Online]. Available: `https://www.postgresql.org` (visited on 04/29/2024).

[45] H. Sharp, J. Preece, and Y. Rogers, *Interaction Design : Beyond Human-Computer Interaction.* John Wiley & Sons, Incorporated, 2019, ISBN: 9781119547358.

[46] Microsoft, *Get ready for the future of work with microsoft teams*, 2024. [Online]. Available: `https://www.microsoft.com/en/microsoft-teams/group-chat-software/` (visited on 06/02/2024).