Supporting an agile working environment for the Information Provision Department at Menzis

Supervisor UT:

Gayane Sedrakyan

2nd supervisor UT:

Patricia Rogetzer

Supervisor Menzis:

Berdien Zwarthoed

Researcher:
Jasper Borren, s2361000

Date: 24-09-2022

UNIVERSITY OF TWENTE.

Preface

The report you will read is “Supporting an agile working environment for the Information Provision department at Menzis.” As the title suggests, the research takes place at the health insurance company Menzis. More precisely, the research takes place at Menzis, in Enschede. This thesis is written as part of the graduation assignment for the bachelor of Industrial Engineering and Management. The research and the formulation of this thesis have taken place between April and June.

Together with my supervisor at Menzis, I was able to formulate the research problem of this thesis. The research was rather complex due to the multitude of systems and my little expertise in agile methodology. After extensive and primarily qualitative research, I could answer the research question and formulate recommendations. During this research, I had the help and full support of my supervisor at Menzis, Berdien Zwarthoed, and my supervisors at UTwente, Gayane Sedrakyan as the first supervisor and Patricia Rogetzer as the second supervisor. During the research, they were always open to answering my questions and providing feedback, which helped me write my thesis.

I want to thank my supervisors for their help and support during this research. Furthermore, I would like to thank all the colleagues at Menzis who opened themselves to answer my questions and guide me at Menzis. Thanks to the kindness of everyone at Menzis, I felt welcomed, which helped me with the thesis. Without the help provided by everyone at Menzis and my supervisors, I would not have been able to write this thesis.

I would also like to thank my parents for helping me get the internship at Menzis and helping me with writing the thesis as well.

I wish you a good reading,

Jasper Borren

Enschede, September 24th 2022

Management summary

The research of this bachelor thesis took place at Menzis in Enschede. More precisely, the research took place at the information provision department of Menzis. In Dutch, the information provision department is called “Informatie Voorziening (IV).” To enhance the feasibility of this report, the information division department will be referred to as IV. Menzis is a health insurance company focusing on giving its clients the best policies possible. Menzis encourages its clients to be healthy and focuses on maintaining good relations with its clients. Maintaining good relations relates to having high customer satisfaction by offering good health insurance packages. To offer the best health insurance packages and maintain good relationships, the organization should be able to work as well and efficiently as possible. Therefore, an organization where interruptions of daily operations are minimized. Therefore, the IV department at Menzis has adopted an agile working methodology. Adopting agile methodologies should help Menzis solve incidents such as computer outages, software outages, and bugs.

Problem context

At the moment, for Menzis, an agile way of working is not fully supported due to unclarity by having too many tools, employees not understanding/wanting to use the tools, and incomplete information within the Configuration Management DataBase (CMDB). All the mentioned reasons result in IV being unable to be as agile as possible and, therefore, adequately respond to incidents. The problem, therefore, is: *“The IV department of Menzis should be fully able to work in an agile way, while they currently cannot.”* It is about being able to quickly and smoothly solve incidents related to software and hardware. When an incident at the workplace occurs, daily operations are interrupted. For example, when one’s laptop is not working, he/she cannot complete the daily tasks. When the daily tasks are not executed, work is interrupted, and time is lost. As an employee or team, it is desired to operate smoothly without incidents interrupting these operations. Being agile aims at resolving these incidents and interruptions as quickly as possible. It is also about the software development process not being fully compatible with contemporary agile development methodologies. With the application of agile methodologies, it is possible to estimate efforts and divide tasks in a much easier and more flexible way than in a traditional development lifecycle. The following research question has been formulated: “*What can be done or altered in the current situation to better support agile working for the IV department of Menzis?”.* Finding the answer(s) to this question will help solve the core problem described by Menzis.

Current situation

At Menzis, many tools are in place, resulting in usability issues such as employees not being entirely familiar, adequately trained, etc., and thus not being able to fully operate in an agile way. These tools relate to the development of software and the resolution of incidents. Incidents resolved from January 2018 till mid-May 2022 have been analyzed to find the exact reasons for not fully operating in an agile way. Incidents take a long time to be solved due to incidents being shifted between teams too often, the lack of necessary information to resolve

incidents, the absence of an incident manager, etc. This report will explain why this is not in line with agile methodologies.

Recommendations

Multiple recommendations have been provided to Menzis as multiple root causes have been identified as to why IV cannot fully operate in an agile manner. The recommendations are based on the literature on agile methodology, technical acceptance model, persuasion routes, incident management, and organizational alignment. These recommendations can help IV to increase the level of agile functioning. Think of recommendations such as creating self-organizing teams who together solve and reflect on the process of resolving incidents. Reflecting on the process of solving incidents and adapting will help become more effective. Furthermore, it is recommended to create self-organizing groups that prioritize incidents so that the most critical incidents are resolved first.

Reader's guide

Chapter 1 – Introduction

In the introduction, the motivation of the research and the problem description is described. The introduction also includes the problem-solving approach and the research design.

Chapter 2 – Theoretical framework

Theories related to agility and the theory used for formulating recommendations are explained in the theoretical framework.

Chapter 3 – Methodology

The different research methods conducted for this report are explained in this chapter.

Chapter 4 – Description of the current situation

The description of the current situation includes a description of the organizational structure of IV, the current tools in place, and the way incidents are handled.

Chapter 5 – Problem identification

This chapter includes the results of the interviews and the results of the observations.

Chapter 6 – Validity

This chapter explains how this research is validated.

Chapter 7 – Implementation

This chapter includes the steps taken during this research, the research journey, the experienced drawbacks, the adaptations, and reflections on the intermediate findings.

Chapter 8 – Recommendations

This chapter contains the recommendations formulated for IV. The recommendations are based on the conducted research and literature studies

Chapter 9 – Conclusion

The conclusion consists of a problem description, the limitations of this study, an overview of the recommendations, and considerations for future research.

References

All the references used for the research can be found in this chapter.

Chapter 10 - Appendix

The appendix includes the results from SPSS and Excel and an overview of the literature study. The appendix also includes transcripts of the interviews.

List of Figures

List of tables

List of abbreviations

BD	BlueDolphin
BPM	Business Process Modeling
CI	Configuration Item
CMDB	Configuration Management Database
FCR	First Call Resolution
ITIL	Information Technology Infrastructure Library
IV	Informatie Voorziening (Dutch name in translation meaning information provision)
SAFe	Scaled Agile Framework
TAM	Technology Acceptance Model
TFS	Team Foundation Server
WSJF	Weighted Shortest Job First
STC	Systematic Text Condensation

Table of contents

1. Introduction

This report presents the final result for the bachelor thesis in Industrial Engineering and Management on recommendations to the company Menzis about how to support a more agile way of working which they currently lack. The research was conducted at Menzis and provides solutions and suggestions about better support of agile working at the "*informatie voorziening"* (IV) department than the current level of their work processes suggests. It has to be noted that the problem does not cover/relate to Menzis as a whole but only covers a particular department, namely the IV department of Menzis. The IV department of Menzis is responsible for developing software and handling continuously occurring incidents.

In 2019 Menzis started reshaping IV, where agile working would be the new standard. There was a restructure of IV. This reshaping was aimed at incorporating agile methodologies and SCRUM principles as much as possible. The new structure would support the continuous delivery of software and the fast solving of incidents. However, as it turned out, the process of entirely reshaping IV to support agile working completely has not been 100% successful. That is why, in this work, the current organizational structure of Menzis, the current tools, and the role of the employees will be explained. To add to this, current related problems will be explained, appropriate frameworks from the literature will be consulted, and recommendations will be derived accordingly.

1.1 Company description

Menzis is one of the largest health insurance companies in the Netherlands, with over two million clients. Menzis has three offices across the Netherlands, in Groningen, Enschede, and Wageningen, with Wageningen as the head office. Menzis counts over 2000 employees, which also makes it a large organization. Menzis strives to strengthen a "healthier lifestyle" for people and ensure excellent and affordable care. The idea behind a "healthier lifestyle" is for people to eat healthier and move more.

1.2 Problem identification

The problem found by Menzis can be described as an ***action problem***. An action problem can be defined as "*a discrepancy between the norm and the reality, as perceived by the problem owner"* (Heerkens & van Winden, 2016). There is a norm needing to be realized and a current reality of the situation. Currently, at IV, an agile way of working is not fully supported. Therefore, the problem statement is: *"The IV department of Menzis should be fully able to work in an agile way, while they currently cannot, due to several issues mentioned below."*

1.2.1 Problem cluster

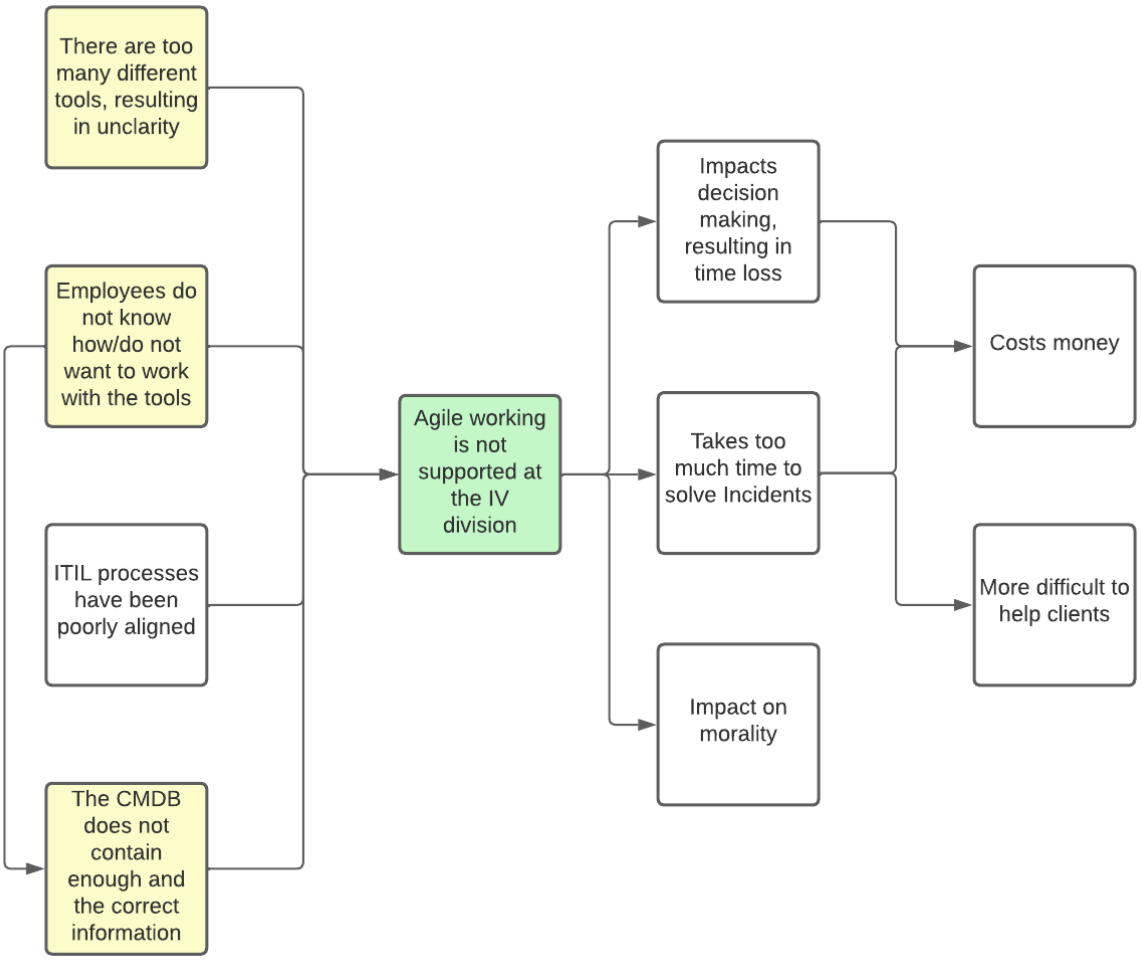

Figure 1-1 Problem Cluster

The core problem in figure 1-1 has been colored green. The current working environment of IV is not in line with agile methodologies, resulting in incidents taking too long to be solved. Four arrows are pointing towards the core problem. These four arrows can be seen as possible causes of the core problem. An in-depth analysis of the causes will be conducted to find solutions to the core problem. However, only an in-depth analysis of the issues described in the yellow squares will occur. This is because after analyzing the results of meetings with Menzis, it was found that there is a more significant chance the core problem is caused by the issues described in these three squares. Furthermore, an in-depth analysis of those three root causes will eventually give the most added value. The root causes have the highest added value for Menzis, as problems with the CMDB will give an inaccurate overview of the current situation. Therefore, it is beneficial to have a good working CMDB (a CMDB containing complete information). Furthermore, having too many tools can cause a chaotic working environment, resulting in unclarity for the employees involved. One of the critical values of agile methodology states, "*Individual and interactions over processes and tools*" (Fowler & Highsmith, 2001), as tools are merely for the support of the employees. Thus, as tools aim to

support agile working and not hinder employees, the tools are researched. Lastly, not understanding tools or seeing the added value of tools can result in less acceptance of the tools. As tools aim to support the employees, why employees are reluctant to work with the tools will be researched. Therefore, these three root causes will be the sub-problems of the research.

1.3 Research question

The core problem at the IV department of Menzis is that agile working is not fully/optimally supported. Therefore, the following research question has been formulated to formulate solutions and solve the core problem: "*What can be done or altered in the current situation to better support agile working for the IV department of Menzis?".* Finding the answer(s) to this question will help solve the core problem described by Menzis. The answers to this research question will be formulated through recommendations. The next section has formulated a problem-solving approach to structure the research.

1.4 Problem Solving Approach

Without a systematic research approach, the research can become very unclear. It is meant to ensure the stability of the research. Thus, following the steps described below will contribute to finding answers to the research question.

I. The first phase of the problem is conducting research and analyzing the current situation at Menzis. Researching whether or not the CMDB is complete. Furthermore, exploring what the current tools are used for. It was researched what tools are currently in place, what the CMDB looks like, and what it is used for. Furthermore, IV and the organizational structure of IV were researched. Business Process Modelling (BPM) shows what different processes look like from beginning to end. For example, when a printer breaks, it should be clear where this can be reported and the steps taken to fix it. It mostly concerns processes and the steps taken related to resolving incidents. BPM is used to show how incidents are solved and later analyze how it could not be in line with agile methodologies. This means that when a problem occurs within Menzis, current tools are used to let the IV division know, and IV can solve it. This way, it is possible to explain and show the current state at Menzis. During this phase, literature studies were conducted to explain the important concepts of this research.

II. The second phase is about finding and analyzing the root causes of the core problem. This is done through interviews, observations, and literature studies. This part of the research aims to find the reasons for not having complete information. It was about finding out how the different tools hinder agile working. Also, it was about understanding what steps during the incident resolution process are not in line with agile methodologies. Understanding the core problem's root causes shows how agile working is hindered.

III. Next, solutions must be generated when the root causes have been found and established. A literature study will be conducted to develop the best possible solution. "*Building your research on and relating it to existing knowledge is the building block of all academic research activities, regardless of discipline"* (Snyder, 2019). This existing knowledge can be used to find possible solutions to the problem. The solutions to the problems are formulated as recommendations in line with agile methodologies and/or in line with promoting an agile working environment. A recommendation not related

to agile methodologies but does promote an agile working environment can be found in section *8.1: CMDB*.

IV. After conducting the literature study, the possible solutions will be aligned and described. It is important to have a good overview of the solutions so that the best solutions can be chosen in the next step. The intention is not just to select the first solution found but to select the solutions best fit the context of IV. As many solutions as possible have been sought for the best solutions to be chosen.

V. The fifth step of the research approach is selecting the best solutions. For choosing the best solutions, certain criteria should be established. When comparing solutions, the advantages and disadvantages of the solutions should be considered. The eventually chosen solution should outweigh the advantages over the disadvantages. Furthermore, there should be an added value to implementing the chosen solution. The solution, for example, should decrease the time some processes take.

VI. In the last phase, recommendations have been made regarding the solutions for solving the problem. The relation with agile methodologies is explained to show the effect of the formulated recommendations. The research findings are presented to Menzis so that Menzis has an overview of possible courses of action that can be taken. Courses of action can only be taken if there is a clear overview of recommendations.

1.5 Summary chapter one

Menzis is one of the largest health insurance companies in the Netherlands, with over two million clients. Menzis strives to strengthen the "leefkracht" of people and ensure good and affordable care. To achieve this, daily operations should run smoothly. To smoothen the daily operations, IV adopted agile methodologies. Currently, agile working is not fully/optimally supported. The following main research question has been formulated to solve this problem: "*What can be done or altered in the current situation to better support agile working for the IV department of Menzis?"*. The answers to this research question will be formulated as recommendations.

2. Theoretical framework

Before conducting an in-depth analysis of the sub-problems and conducting research at Menzis, it is desirable and important to have a good understanding of the concepts of agile methodology and an initial assessment of the context at Menzis. Therefore, this section covers the conducted literature study to better understand the core problem and its root causes. The following knowledge questions have been answered in this chapter: "*What exactly is meant by tools?"* and *"What is meant with agility in the context of Menzis?" And "What is the best process modeling notation to use?"*

2.1 Literature study on Agility and corresponding concepts

Central to the problem provided by Menzis is the IV department not being compliant with agile working methodologies enough. Therefore, it is important to first describe what agility entails. Agile processes rely on "*people and their creativity rather than on processes"* (Dyba & Dingsoyr, 2008). Agility can be described as: *"stripping away as much of the heaviness, commonly associated with the traditional software-development methodologies, as much as possible to promote quick response to changing environments, changes in user requirements,*

accelerated project deadlines, and the like" (Dyba & Dingsoyr, 2008). It is about iterated change and constant feedback. To further elaborate on agile methodology, the following key values should be considered (Fowler & Highsmith, 2001):

I. *Individuals and interactions over processes and tools.*
II. *Working software over comprehensive documentation.*
III. *Customer collaboration over contract negotiation.*
IV. *Responding to change over following a plan.*

Thus, these values are at the heart of agility software development and should be considered in this research.

The first value says that tools and processes cannot add value if seen separately from the employees wanting to use them. The second value says documentation is useless if the software does not work. For customers, the most important aspect is that the software is working. The third value says that an organization's operations can be better executed when understanding the customer's needs. Lastly, the fourth value says that it is not always beneficial to a plan. Following a plan could prevent organizations from changing and innovating, even if necessary.

Next to the key values of agile software development, the agile manifesto also considers twelve principles. These twelve principles are at the core of agile software development (Fowler & Highsmith, 2001):

I. Customer satisfaction: Our highest priority is to satisfy the customer through early and continuous delivery of valuable software.
II. Welcome changes: Welcome changing requirements, even late in development. Agile processes harness change for the customer's competitive advantage.
III. Deliver frequently: Deliver working software frequently, from a couple of weeks to a couple of months, with a preference to the shorter timescale.
IV. Business people: Business people and developers must work together daily throughout the project.
V. Motivated individuals: Build projects around motivated individuals. Give them the environment and support they need, and trust them to get the job done.
VI. Face-to-face conversation: The most efficient and effective method of conveying information to and within a development team is face-to-face conversation.
VII. Working software: Working software is the primary measure of progress.
VIII. Sustainable development: Agile processes promote sustainable development. The sponsors, developers, and users should be able to maintain a constant pace indefinitely.
IX. Technical excellence: Continuous attention to technical excellence and good design enhances agility.
X. Simplicity: Simplicity–the art of maximizing the amount of work not done–is essential.
XI. Self-organizing teams: The best architectures, requirements, and designs emerge from self-organizing teams.
XII. Intervals and reflections: At regular intervals, the team reflects on how to become more effective, then tunes and adjusts its behavior accordingly.

Three of the abovementioned principles clearly show that the principles relate to the software development process: principle I: customer satisfaction, principle III: deliver frequently, and principle VII: working software. The principles are about delivering valuable software, frequently working software, and measuring performance with working software. The remaining nine principles focus on how software development culture can best be achieved: welcome changing environments, business people and developers should work together, teams should reflect, etc.

To better understand IV aiming to promote agile methodology, it is first important to understand why organizations would target promoting agile methodology in the first place. Teams or organizations choose agile as a methodology to respond to changes or problems in the marketplace. Agility is also about being able to seize opportunities when they present themselves. Thus, agile methodology can be applied to adapt to consumer needs changes.

The daily operations should run smoothly and continuously for IV to continuously deliver service to its customers. For example, when the laptop of one of the employees is not working, he/she cannot work. Therefore, the employees' tasks for that day cannot be worked on, and time is lost. This could result in a delay in the daily tasks and thus a delay in the service. For operations to run smoothly, several tools and software are in place at Menzis. The department responsible for the continuous delivery of software is IV. The software can be delivered continuously using tools such as ServiceNow, TFS, and DevOps. Therefore, IV incorporates agile methodology as a way to deliver working software.

As already mentioned, agility can be applied in other types of work instead of focusing purely on software development. Agility can also be applied to the regular business of an organization, for example, the handling of incidents by creating self-managing teams and reflecting to become more effective. This is called business agility and focuses on the culture, strategy, leadership, and governance (AgileBusinessConsortium, 2022). This way, the business should be able to add value to its customers who work in uncertain environments. In the case of IV, its customers are internal customers at Menzis. Agile teams should be able to respond to change "without going off the rails," where bugs and incidents refer to "going off the rails" (Winter, 2022). Thus, the employees at Menzis work in an uncertain environment; there are failures or incidents of software and hardware interrupting daily operations. At IV, it is also believed that agile methodology can be applied to solving incidents. Agile methodologies should solve incidents as quickly as possible to restore daily operations. Instead of just continuously delivering software, IV also aims at continuously solving intermediate incidents. This way, IV aims to satisfy internal customers through continuously restoring daily operations. Thus, for the management of incidents, agile methodologies should be applied.

Returning to the four key values of agile software development, according to the value of customer collaboration, it should be possible for customers to report incidents (Winter, 2022). At Menzis, this can be done through ServiceNow. The incidents reported through ServiceNow are well-documented. This way, the ServiceDesk and teams responsible for the incidents are aware of the incidents. Thus, the reporting and the incidents' status are clearly and quickly communicated through ServiceNow. Should all of this be done through e-mail or telephone calls, significant time can be lost.

A great way to respond to change (in this case, incidents) is through retrospectives as a way to analyze incidents. Retrospectives enable teams to see what went well and what did not work well during the resolution process. Thus, the incident can be resolved faster when the same incident occurs, or a similar one occurs. Furthermore, a better understanding of the resolution process could prevent the same incidents from happening again. As will be discussed in more detail later, retrospectives are part of agile software development, and SCRUM, however, can also be applied to manage incidents.

Thus, there are at least two values of agile methodology that can be applied to the management of incidents. Therefore, agile methodologies do not necessarily have to be applied to software development. As IV aims to resolve incidents quickly and restore the daily operations of internal customers, it makes sense to adopt agile methodologies.

With agile methodologies, it is also possible to estimate efforts and divide tasks. Furthermore, concepts such as agile coaches, SCRUM masters, sprint masters, Information Technology Infrastructure Library (ITIL) processes, and the Configuration Management Database (CMDB) will be explained in this section.

First, ITIL can be described as: *"ITIL focuses on the flow of activities that cross-organizational units, both inside and beyond the IT function. The objective is to maximize IT's ability to provide services that are cost-effective and meet the needs and expectations of the business, as manifested in the Service Level Agreement (SLA)"* (Eikebrokk & Iden, 2014).

Menzis also works according to an agreed service level. Therefore, the ITIL processes and the CMDB are designed to achieve these SLAs. SLA specifies how IT will be provided (Brenner, 2006). The SLA is a predetermined agreement about services provided and the quality of those services. However, it is also necessary to specify how this quality will be measured.

Closely related to ITIL is IT Service Management (ITSM). ITIL is an approach to support ITSM. *"ITSM is the discipline that strives to better the alignment of IT efforts to business needs and to manage the efficient providing of IT services with guaranteed quality"* (Brenner, 2006). This is what IV tries to achieve as well. With ITIL processes and tools in place, they try to align business needs with IT. ITIL processes are also used as guidelines for the CMDB, which "*is supposed to serve as a repository and information retrieval tool for a services and IT infrastructure tool – and also as a platform for information integration between the other ITSM processes"* (Brenner, 2006). In section *4.4: CMDB,* it is explained how the CMDB is related to agility and how it reduces the rate of defects.

A framework for applying agile methodologies is the SCRUM framework. At IV, SCRUM masters are in place to ensure agile principles are applied. The SCRUM master is responsible for: *"enacting the SCRUM values and practices and removing impediments"* (Cervone, 2010). SCRUM is "*a lightweight framework that helps people, teams, and organizations generate value through adaptive solutions for complex problems"* (Schwaber & Sutherland, 2020).

Central to SCRUM is the sprint; the sprint includes Sprint planning, daily SCRUMS, sprint review, and sprint retrospective. These are the SCRUM events that are performed to execute the SCRUM framework. During the sprint planning, mentioned topics are: a proposal by the

product owner, creating a product backlog (list of what is needed for improvement), and evaluating how everything will be completed. Daily Scums take place every day and are meant to check progress. How is it going, and how can we adapt the product backlog to complete the goals? Therefore possibly altering the planned work. The sprint is ended with a retrospective of the sprint. Retrospectives are conducted to analyze what went well and what went wrong. This way, future mistakes can be avoided, and sprints can be performed better. So, the ones responsible for the infrastructure of the sprints are busy with sprint planning, daily sprints, and continuously thinking about how the sprints can be executed better. Information regarding sprints in the context of Menzis can be found in section *4.3:" Tools for sprints and software development."* However, as IV aims at incorporating agile methodologies for solving incidents, recommendations related to the SCRUM framework will be formulated. Therefore, a detailed explanation of the SCRUM methodology is given in this section.

To elaborate on the sprint retrospective, at IV, the sprint retrospective takes place after every sprint to evaluate and discuss how the sprint went. This means the following: "*The SCRUM team inspects how the last sprint went with regards to individuals, interactions, processes, tools, and their Definition of Done."* Identified improvements during the sprint retrospective will be implemented in the next sprint.

Agile project management SCRUM methodologies are iterative, including incremental processes executed on a team-based approach. One of the motives for using SCRUMs and iterative processes is to help resolve conflicting interests and needs within self-organizing teams. With iterative processes, communication, as well as cooperation, is improved. Furthermore, errors, incidents, and disruptions can be prevented or resolved as quickly as possible with iterative processes. For instance, building software starts with a prototype that is tested and validated before releasing the software. During the next loop, the software can be improved. This way, incomplete software resulting in incidents can be prevented. As explained earlier, IV aims to resolve these incidents and disruptions by adopting agile methodologies. These agile methodologies can be adopted by applying the SCRUM framework and principles. For example, part of the SCRUM framework is the retrospective phase, where it is decided how the resolution process can be improved during the next sprint. This is in line with principle XII: *Intervals and reflections*, where teams reflect on how the resolution process can be improved. Thus, with the retrospective phase of SCRUM, agile methodologies are adopted.

SCRUM has three pillars that are of high importance when applying the SCRUM framework (Schwaber & Sutherland, 2020):

I. Adaptation: adaptation is about reflecting on the processes of how incidents are solved, and it is about continuously changing and adapting processes as new problems and incidents arise. Thus, there should not be one standard approach for solving all problems; however, the same or previously in place processes can be used when incidents are similar.
II. Inspection: Inspection is about continuously reflecting on how incidents are solved. Asking oneself what can be done better next time helps improve the resolution

process. A better understanding of how incidents can be solved helps with the faster resolution of incidents.

III. Transparency: with transparency, team members should have insight into what everyone is working on and what problems they encounter.

Next to the SCRUM events and pillars, the SCRUM framework has three artifacts: the product backlog, sprint backlog, and increment. The product backlog shows the activities that need to be done to improve the delivered product. In the case of IV, backlog items currently show activities related to the continuous development of software. However, as IV wants to incorporate agile methodologies for solving incidents, these artifacts can also be considered. For example, frequently creating backlogs (typically a cycle of two to four weeks called Sprint), reviewing the resolution process, and executing retrospectives to improve the resolution process of incidents. The sprint backlog includes the product backlog items selected for that sprint, the goal of the sprint, and the plan of how the sprint will be executed. An increment can be seen as an addition to the current product or service. For example, prioritizing the features that have been requested for implementation or the issues that need to be solved within the next sprint. How some recommendations relate to SCRUM and how it improves an agile way of working has been described in section *8.2: Incident management*.

As IV uses the Definition of Done (DoD) terminology, it is important to understand this concept. Every team or person has a different understanding of what it means for a backlog item to be done. Therefore, it should be defined beforehand when a backlog item is done. "*DoD is a shared understanding within the SCRUM team on what it takes to make your product increment releasable*"[1]. This relates to the development and continuous delivery of software. However, as mentioned before, retrospectives to evaluate individuals, processes, and tools are also applicable for solving incidents.

What is used within DevOps[2], by IV, is the refinement stage. During this stage, a game is played: "*planning poker, also called SCRUM poker, is a consensus-based, gamified technique for estimating, mostly used to estimate effort or relative size of development goals in software development."* In other words, it includes a method to estimate efforts regarding the Sprint Goal. The efforts are related to a number of points, which will be further explained later in this report.

Important to SCRUM are iterative processes as a way to improve communication and cooperation. For example, one of the main pillars of SCRUM is transparency, where team members should have insight into what everyone is working on and what problems they encounter. The team is transparent about their issues and what is needed. One of the events of SCRUM is daily SCRUMs, where the team can come together and discuss their progress. During these daily SCRUMs, the self-organizing team can discuss their challenges. What, for example, the CMDB should be able to contain (related to the problem of the CMDB not containing complete information). The SCRUM team should include a SCRUM master as the SCRUM master helps with the following: supporting the team with self-management,

[1] https://www.scrum.org/

[2] Menzis intranet

supporting the team in being cross-functional, supporting SCRUM adaptation, and reducing obstacles between teams (Schwaber & Sutherland, 2020). Thus, the SCRUM master is an important factor in having self-organizing teams. Furthermore, as the SCRUM master helps with overcoming obstacles, the SCRUM master can support overcoming the obstacle of having incomplete information within the CMDB. The SCRUM master can do this by taking the lead, organizing the teams, coaching the teams, and advising the teams. Furthermore, the team should include a product owner responsible for creating a list of what is needed to be achieved. Therefore, the product owner can help the team by providing topics to be discussed.

2.2 Literature study on process analysis approach

Aguilar-Savén (2004) has been consulted to describe a business process: *"The combination of a set of activities within an enterprise with a structure describing their logical order and dependence whose objective is to produce a desired result."* Therefore, business process modeling makes it possible to describe current processes for analysis. That is why a process modeling approach will be used to understand the current processes at IV. However, there are multiple languages for business process modeling available. A decision in the business process modeling language will be made to best describe the processes at IV.

The business process modeling technique that will be used is a flow chart. The rationale is that flow charts can be used to show the sequence of processes and, therefore, the infrastructure of processes. With flowcharts, the processes related to incident handling can be depicted. This way, possible inefficiencies can be identified, after which the design can be altered. Furthermore, a commonly used standard for depicting flowcharts is Business Process Modelling Notation.

2.3 Tools and systems

This part of the literature study relates to the knowledge question, "*What exactly is meant with tools?".* This knowledge question should give a better insight into the tools currently in place at IV and why these tools are of importance to the context of the company. The knowledge question aims at finding literature that explains the tools of IV in a regular organization's context. This way, a good understanding of the tools can be conveyed and reflected on for the specific context of Menzis. The tools that will be mentioned in this report are DevOps, TFS, and ServiceNow.

DevOps

DevOps is a software development tool combining software development and the SCRUM methodology. To be more precise, DevOps is not only a way to keep track of sprints and backlogs, *"DevOps is used as an agile software development technique in operations to increase the pace of their software development process and to improve the quality of their software"* (Erich et al, 2017). DevOps relates to agility. To further elaborate on this, DevOps extends agile in the following way: *"DevOps stresses more on the communication and collaboration between developers and operators rather than tools and processes, it can achieve agile goals to reduce team working latency and extend agile principles to entire software delivery pipeline"* (Jabbari et al, 2016). It is used as "Software engineering tools and methods," where code is used for the infrastructure. So within DevOps, code is linked to

architecture and software development. It is used for the continuous delivery of software. TFS has relatively the same capabilities as DevOps, as TFS is also an agile software development process. Also, both are used for keeping product backlogs

ServiceNow

Finally, as one of the tools in place at Menzis and IV is ServiceNow, a clear description of ServiceNow should be included. Therefore, *"ServiceNow is a cloud-based company that provides software as a service for technical management support"* (Fitzgibbons, 2020). ServiceNow focuses on ITSM, which allows users to manage projects, teams, and customers. Depending on the organization, ServiceNow can be used for storing and organizing data. It contains service portals where incidents can be reported. Furthermore, within ServiceNow, the handling of incidents is organized.

For example, Menzis uses ServiceNow as a way to report incidents. To be more specific, IV uses ServiceNow to keep track of the reported incidents, such as who the incidents were assigned to and what the priorities of the incidents are. As shown in *Figure 1-1*, having an agile working environment influences the handling of incidents by having intervals to become more effective and have self-organizing teams that work on the incidents. As principle *XII: Intervals and reflections* states, intervals are used to reflect behavior and the resolution process. This way, employees can adjust their resolution process to become more effective for the next interval. As the incidents are being handled through ServiceNow, in-depth research on ServiceNow is conducted.

Furthermore, within ServiceNow, it is possible to maintain the CMDB of the company. The CMDB shows all the applications, services, computers, database clusters, database instances, databases, and servers. CMDB supports the management of incidents, as changes and root causes can be found within the CMDB. Therefore, it is possible to track incidents and see what configuration items (CI) are impacted by them. Principle XII: Intervals and reflections can be related to the CMDB. With the CMDB, it is possible to track incidents and thus see how incidents were solved. As it is possible to see how incidents are solved, teams or employees can reflect and become more effective. Therefore, ServiceNow plays a significant role in customizing company processes to fit agile methodology and is therefore researched thoroughly in this thesis.

2.4 Technology acceptance model

Often organizations assume when new technology is implemented, it solves all their problems or ensures possibilities are grabbed. However, how the technology is implemented and whether the technology is correctly implemented are often neglected. "*Thus, having the technology available is simply not enough; it must be accepted and used appropriately by its target user group in order to realize anticipated productivity gain"* (Agarwal & Prasad, 1997). To see whether technology is correctly implemented and adapted by internal customers, the technology acceptance model (TAM) is formulated.

The Technology Acceptance Model was first formulated by Fred Davis, who formulated that the use of technology was explained by the user's motivation to use the technology. Figure 2-1 nicely shows the first formulation of the TAM.

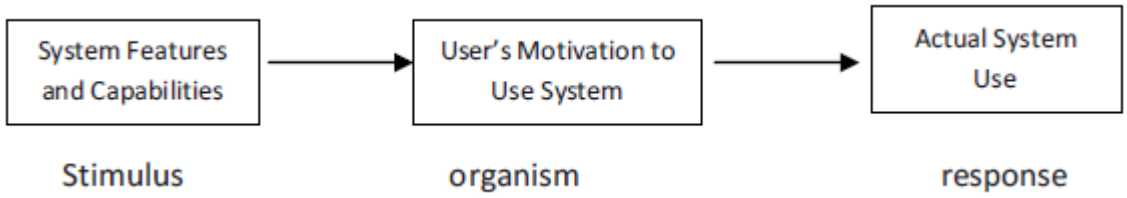

Figure 2-1: First formulated model of technology acceptance (Chuttur, 2009, p1)

However, it was believed more factors would influence the actual use of technology within an organization. The attitude towards the use of technology would be influenced by the user's perceived usefulness of the system and the perceived ease of use. The perceived usefulness and ease of use are themselves influenced by external factors such as the system design characteristics. Figure 2-2 illustrates this formulation of the TAM.

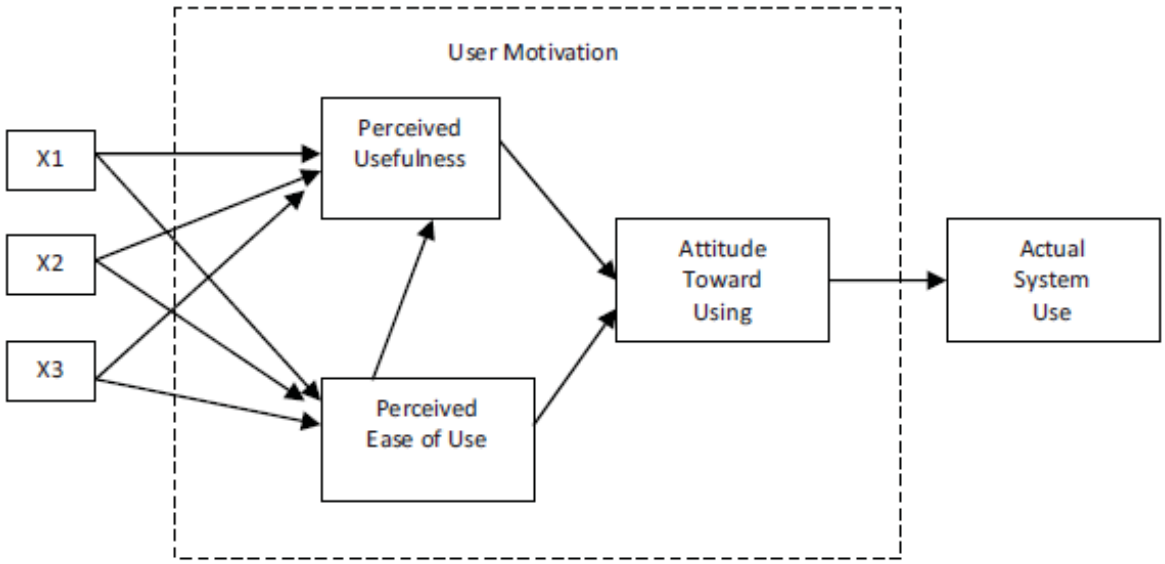

Figure 2-2: Original TAM (Chuttur, 2009, p2)

After years of research and studies, it was found that perceived usefulness and ease of use directly influenced the intent to use technology. Therefore, the "attitude toward using" would be replaced by "behavioral intention. This replacement would result in the final version of the TAM. The external variables mentioned in figure 2-3 relate to the "*characteristics of the technology, user training, user participation in design and the nature of the implementation process*" (Chuttur, 2009). This makes sense, as ones perceived ease of use depends on how and what the technology looks like and how it works. Technology is also perceived to be useful depending on the characteristics of the technology. This can be explained by the fact that technology is perceived as useful depending on the technology. Perceived usefulness is also influenced by perceived ease of use. This makes sense, as people perceive something more useful if the technology is relatively easy to use. If the technology is relatively easy to use, it makes up for not fully understanding it because it is relatively easy to use anyway.

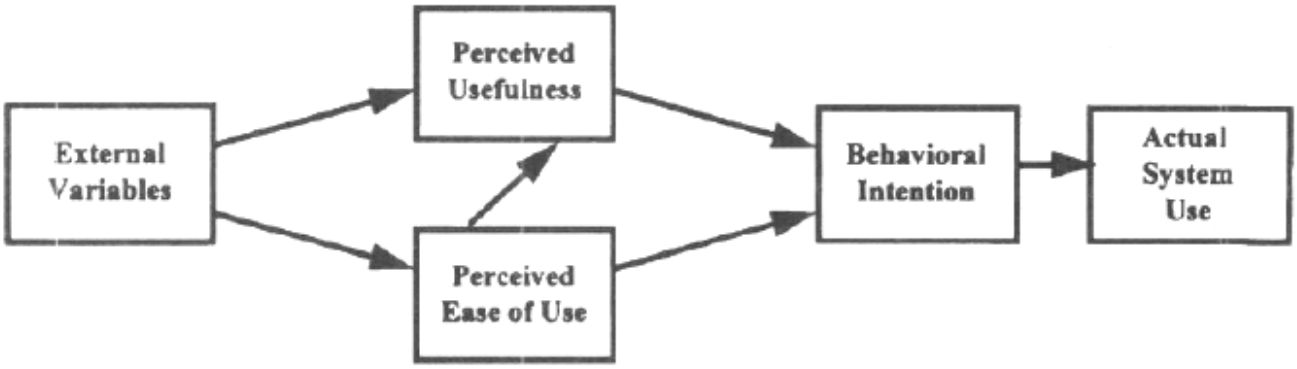

Figure 2-3: Final version TAM (Chuttur, 2009, p10)

The TAM can be used to analyze the acceptance of internal customer tools. For example, the actual use of the CMDB at Menzis. More information regarding CMDB use at Menzis can be found in section *4.4: CMDB* and section *5.1: Results of the interviews*. In these sections, the CMDB within the context of Menzis is better explained.

Furthermore, as seen in figure 2-3, the external variables influence perceived usefulness and perceived ease of use. This means external values must also be considered regarding technology acceptance at Menzis.

To elaborate on the perceived usefulness of using technology, several detailed factors influence this perceived usefulness. These factors could explain why internal customers perceive technology to be useful or not. Figure 2-4 shows the different factors that could influence the perceived usefulness of technology.

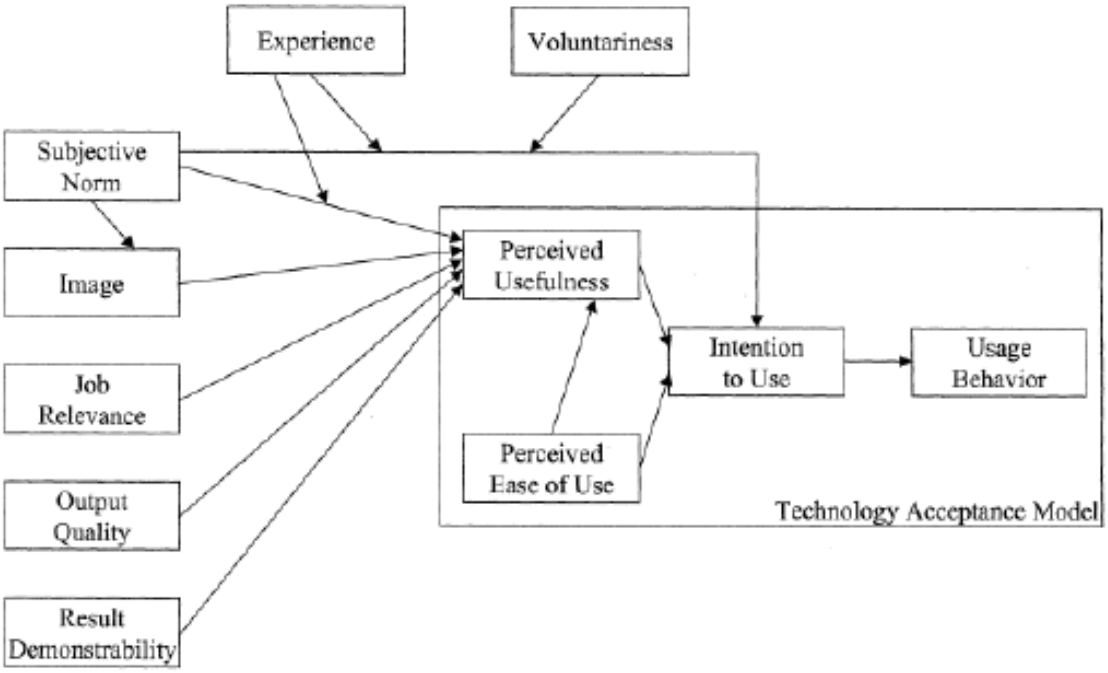

Figure 2-4: TAM 2 (Chuttur, 2009, p14)

For example, job relevance can be described as *"the message recipient's perceived relevance of an IT system their work"* (Bhattacherjee & Sanford, 2006). Also, the output quality can be described as "*the degree to which an individual believes that the system performs his or her job tasks well."* (Mlekus et al, 2020). Furthermore, voluntariness can be described as: "*the degree to which use of the innovation is perceived as being voluntary, or of free will"* (Chuttur, 2009). As will later be explained, these external variables are of most importance to IV.

Persuasion techniques

In section *8.1: CMDB,* two persuasion techniques are mentioned to encourage employees to use and accept technology. The techniques will be elaborated on here to better understand the recommendations about persuasion techniques. The two persuasion techniques are part of the Elaboration Likelihood model, which provides a framework for the processes of persuasion communications (Cacioppo & Petty, 1986).

To elaborate more on the persuasion techniques, the central route of persuasion can be described as: "*involves educating users about the potential benefits of IT acceptance by providing high-quality arguments about how the new IT can substantially improve their work"* (Bhattacherjee & Sanford, 2006). The peripheral route of persuasion can be described as: "*involves providing peripheral cues such as endorsements from reputable or preferred sources about the benefits and potential impacts of IT acceptance"* (Bhattacherjee & Sanford, 2006). The peripheral route focuses less on facts and technology quality but positive emotions and characteristics. Bhattacherjee & Sanford (2006) show that the central route has a better long-term effect on technology acceptance than the peripheral route. Therefore, later in the report, the focus will be on the central route of persuasion.

2.5 Incident management

Incidents related to technology and software can be described as: *"An incident is any event which is not part of the standard operation of the service and which causes, or may cause, an interruption or a reduction of the quality of service"* (Cusick & Ma, 2010). Therefore, it is of high importance that incidents are solved as quickly as possible; having incidents hinders the quality of service. It makes sense that it is never desirable for the quality of service to be hindered. To minimize the occurrence of incidents or the time it takes for incidents to be solved, incident management is often in place within organizations. Incident management can be described as: *"The objective of incident management is to restore normal operations as quickly as possible with the least possible impact on either the business or the user, at a cost-effective price"* (Cusick & Ma, 2010).

Part of incident management is having an incident manager who everyone can go to and helps and manages the incident resolution process. Furthermore, "*an incident manager's job is to respond to incidents when they occur and take any necessary steps to restore service and return the business to normal operations as quickly as possible"* (Feuvre et al, 2021). The incident manager has multiple responsibilities, such as prioritizing incidents, formulating expectations for resolution time, collecting data, finding root causes, and contacting incident requestors. All these responsibilities are executed to better understand the incidents, solve

the incidents faster, stay within the SLA, and prevent incidents from happening repeatedly. Thus, this results in employees better being able to conduct their business

To measure whether or not incident management is working correctly or as good as it should be, there are several KPIs (Feuvre et al, 2021) that can be measured:

i. Resolve time (mean resolve time is currently 230 hours).
ii. First call resolution (relates to awaiting supplier – and user information).
iii. Issues resolved per day (at the moment, every 29 minutes, an incident is solved, this is 49.66 incidents per day).
iv. Escalation rate (Escalation rate means the number of times incidents are shifted between teams or employees).

First call resolution (FCR) can be described as "*the percentage of the calls that does not require any further contacts or callbacks to address the same customer's reason for previously calling"* (Aliyu, 2010). This means the extent the incident can be solved at once with own knowledge and expertise.

2.6 Summary chapter two

Central to the problem provided by Menzis is that the IV department is not compliant enough with agile working methodologies. Therefore it is important to first explain what is meant by agility. Agility is explained as a methodology that can be adapted for organizations operating in a volatile and changing environment. With agile methodologies, organizations can respond to problems and seize opportunities. More precisely, it is possible to deliver software continuously with agile software development. At IV, the SCRUM methodology is incorporated to be more agile. Therefore, the principles of SCRUM, such as sprints and retrospectives, have been explained.

Literature study showed that flowcharts are an effective way to detect inefficiencies within the current composition of processes. The literature study also showed that the TAM could be used to explain possible reasons employees do not use technology. In chapter 8: *Recommendations*, the TAM is used to derive recommendations for the better use of the CMDB.

3. Methodology

To describe and understand the current situation at IV, research was conducted. Several research methods were used to best understand the current situation at IV and see how IV lacks support for agile methodologies. Therefore, this section will discuss the conducted research methods. The research methods consisted of observational studies, interviews, and literature studies.

Literature studies explained concepts such as agile methodologies, SCRUM, and TAM. It is impossible to conduct observational studies or interviews about agile methodologies without first understanding what is meant by agile methodologies. A basic understanding of agile methodologies was needed to analyze data retrieved from observational studies or interviews and to see how IV could improve in agile methodologies. Furthermore, literature studies have also been conducted for the formulation of recommendations. After root causes have been

found and formulated, literature studies were conducted to understand how these root causes can be tackled. For example, using theory about the SCRUM framework as a way to create self-organizing teams. These self-organizing teams can prioritize incidents as these groups would consist of at least one person from every team. A more detailed explanation can be found in section *8.2: Incident management.*

Furthermore, interviews were conducted to find the root causes for IV being unable to fully support agile methodologies. All the interviews were conducted online as Menzis still has a strict policy regarding Covid. Before conducting the interviews, I introduced myself and explained what I was doing at Menzis. It was also clear that the interview was meant for this research. This was done so that the interviewee would understand why their help was needed; this was meant to be an incentive for the interviewees to answer honestly. It was mentioned that the research was related to agile working. However, any specifics regarding why certain things would be researched were left out. This intends to prevent any biases.

The questions asked were not based on a standard questionnaire. Rather, the interviews were open questions, with every interview having different questions. The reason for not having a standard questionnaire was that all the interviewees had different roles within IV. Furthermore, different type of information was needed from every interviewee. The questions were qualitative and were formulated in a way to create freedom for the interviewees. This resulted in much information and was meant to find root causes.

It was eventually chosen to conduct a semi-structured interview, a mix of a structured – and unstructured interview approach. In a semi-structured interview, "*the questions are pre-planned before the interview but the interviewer gives the interviewee the chance to elaborate and explain particular issues through the use of open-ended questions*" (Alsaawi, 2014). Also mentioned by Alsaawi, 2014, is that a structured format could prevent good depth and richness of the responses. An unstructured interview could lead to subjects or answers not important to the research. Therefore, the "freedom" mentioned before was also somewhat limited.

The best way to decide on what employees to interview is to determine a sampling technique. First, the target population needs to be chosen. In this case, the target population is Menzis. The sampling frame is a *"list of the actual cases from which the sample will be drawn"* (Taherdoost, 2016). The sampling frame is, therefore, the IV department. Then, the most important step in the sampling process is choosing the sampling technique. The sampling technique eventually decides which employees will be interviewed. For these interviews, non-probability sampling has been used. Non-probability sampling is often used in qualitative research, which relates to the qualitative nature of the interviews. More specifically, purposive or judgmental sampling has been used to determine what employees to interview. Purposive or judgmental sampling can be described as: "*a strategy in which particular settings persons or events are selected deliberately in order to provide important information that cannot be obtained from other choices"* (Taherdoost, 2016). Thus, the choice of employees has been made based on the fact that these employees were believed to have specific and important information that other employees do not have.

No specific sample size was determined beforehand, and the sample size solely depended on whether the required information was already found. Thus, a new interview would be appointed if more information was required after an interview.

Lastly, observational studies were conducted to find the root causes for IV's inability to fully support agile methodologies. The observation includes observations regarding organizational structures, the tools in place at Menzis, and the human factors related to the organizational structures and tools. It will be an uncontrolled observation; the situation will not be influenced. A description of the current situation is not possible when the environment is controlled or influenced. When an environment is controlled, employees could act differently as they are being watched. This research aims to understand how IV naturally operates. It is, therefore, desirable to observe in an uncontrolled environment.
To conclude this section, this use of multiple research methods is called triangulation.
Triangulation is a strategy to see whether or not research can be validated, and it focuses on getting information from multiple sources to validate the research. These sources relate to having multiple research methods, multiple researchers, multiple theories, and multiple data sources. For a more detailed explanation, see section *6:* Validity.

Part of conducting research is analyzing the data/results corresponding to the interviews and observations. The interviews and observations were qualitative to understand concepts such as agile methodologies and why IV cannot fully support agile methodologies. With numerical data, it cannot be explained why IV would not be able to fully support agile methodologies. As the interviews and observations were qualitative, qualitative analysis of the results was also performed. The analysis performed to understand the collected data was systematic text condensation (STC) (Malterud, 2012). For example, with analyzing interviews: STC starts with getting a general impression of the data; understand the themes related to the employees' answers. The second step involves finding data to help answer the research or sub-research questions. This is done as not everything that is said during interviews/ found during observations is relevant to the research question. In this case, the research question seeks things at IV that can be altered to better support an agile working environment. For example, data showed that the CMDB is incomplete. This relates to the research question as a complete CMDB supports transparency between teams. Every team can go to the CMDB and collect information about present applications, laptops, servers, etc. This way, IV can be transparent with each other. The third SCRUM pillar: transparency, states that teams and employees should be transparent with their work. Thus, an incomplete CMDB can be linked to the SCRUM framework (used at IV to apply agile methodologies) and thus the research question. This displays the importance of the second step of STC, where data related to the research question should be detected.

In the third step, the data is decontextualized and transformed into code groups. This is a way to arrange a large number of data. However, as the data with the interviews and observations were not too extensive, the third step has not been performed. The fourth step involves using the data related to the research question to formulate descriptions and concepts that can help answer the research question. These descriptions and concepts can be found in *Table 1: Overview root causes*. To conclude, STC has been performed to get as much important information as possible to answer the research question.

3.1 Summary chapter three

Chapter three explained the three research methods used for writing this report: semi-structured interviews, uncontrolled observations, and literature studies. Furthermore, chapter three introduces triangulation and an explanation of systematic text condensation. Semi-structured interviews were conducted to elicit as much important information as possible. Uncontrolled observations were conducted to find root causes as to why IV cannot fully comply with agile methodologies. Lastly, literature studies were performed to explain agile methodologies and concepts related to agile methodologies (SCRUM framework, for example).

4. Description of the current situation

To formulate solutions and make recommendations, it is desirable to have a good overview of the current situation at Menzis. To get a good overview of the current situation, the current organizational structure of IV will be described, an overview of the different tools related to agile working will be made, and a description of the different tools will be written. Business Process Models will be used to better show the steps of handling incidents. This chapter answers the following knowledge questions: *"What are the different tools in place at the moment?", "How are the different tools used and integrated?", "How and where can incidents be indicated?"* and *"What exactly is the CMDB used for? And what is its importance?"*

4.1 Organizational structure IV

In 2019 Menzis proposed redesigning the organizational structure to better support agile working within IV. Therefore, an explanation of this new organizational structure will be explained in this section. However, not everything described in the 2019 redesign of IV is already in place, and this will be taken into account in the description of the current situation.

The redesign intends to resolve incidents better and deliver software by incorporating agile methodologies. Menzis uses DevOps as an agile framework, which shows the sprints and backlogs of all the teams within Menzis. Menzis believes that these teams should consist of 6 to 9 employees to best add value. The employees have the following functionality profile (see figure 4-1).

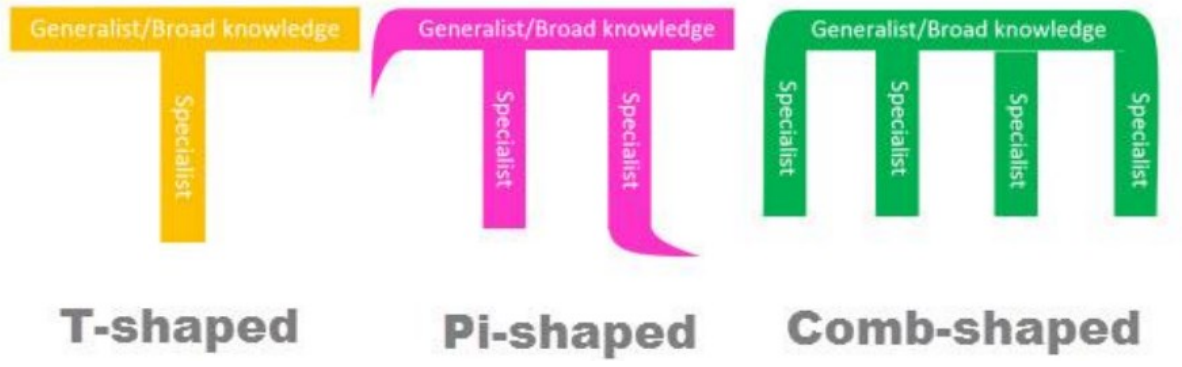

Figure 4-1: Functionality profiles (Menzis Intranet, retrieved on 20-05-2022)

Figure 4-1 shows that every employee has one or more specialties but broadens his knowledge, so he is employable in multiple places. The idea is that if an organization aims to have all different available specialists in one team, the teams would be much bigger than just nine employees. This would not be the case with a T-shaped, Pi-shaped, or Comb-shaped employee.

To elaborate on the organizational structure of IV, a visualization of the structure can be seen in figure 4-2.

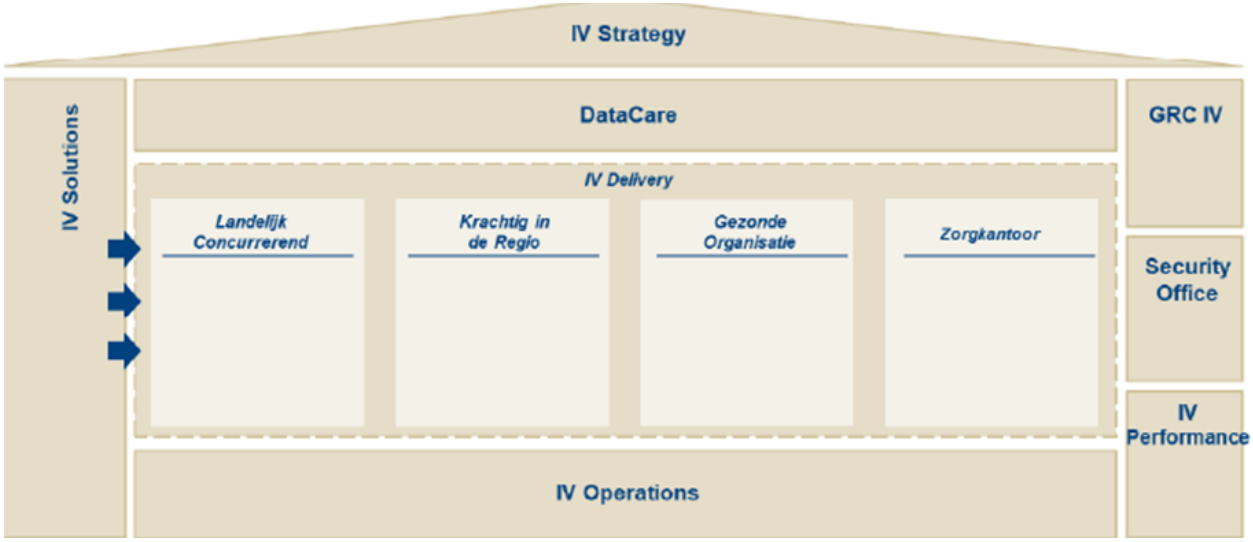

Figure 4-2: Organizational Structure IV (Menzis Intranet, retrieved on 20-05-2022)

Figure 4-2 shows seven different departments that make IV: IV strategy, IV Solutions, IV Operations, GRC IV, Security office, IV performance, and DataCare. IV delivery is not a department but is employed by different departments. IV solutions make up most of the IV delivery. Within IV solutions, several disciplines are shared by the agile teams, such as domain architects, SCRUM masters, and information analysts.

Menzis has an agile framework used as guidance for working with agile methodologies. It is called the Scaled Agile Framework (SAFe). IV incorporated SAFe together with the Hybrid Agile framework. However, the SAFe is outdated, and most teams work with the SCRUM methodology. Further explanation of the agile Frameworks in place at Menzis can be found in section *4.3*: *Tools for sprints and software development.*

To continue with the organizational structure of IV, figure 4-3 shows an overview of the IV departments currently in place.

Figure 4-3 shows there are only two departments directly communicating with the management of IV. Every department can be split into different specific departments and functions. However, this is too detailed.

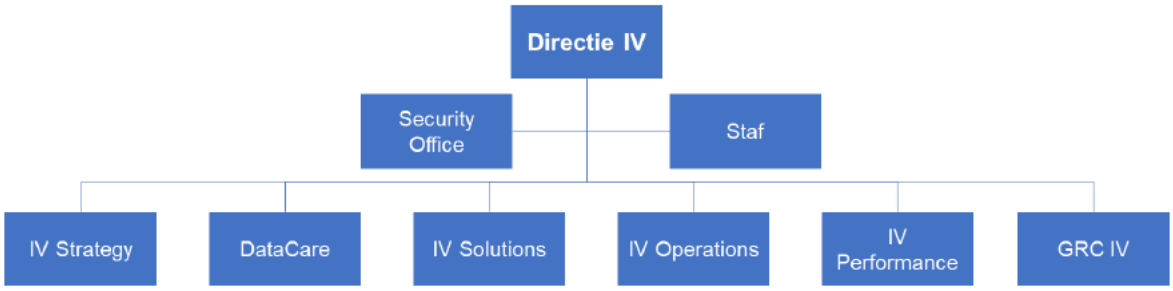

Figure 4-3: Overview of the IV departments (Menzis intranet, retrieved on 20-05-2022)

4.2 ServiceNow and Incident management

The main tool at Menzis that employees use/can use is ServiceNow[3]. For IV, agile methodologies are incorporated to handle incidents as efficiently as possible. ServiceNow is included as this research also aims to explain the tools in place. ServiceNow is incorporated to handle incidents. Thus, ServiceNow supports the employees in handling incidents. As mentioned before, one of the key values of agile methodology states, "*Individual and interactions over processes and tools"* (Fowler & Highsmith, 2001). Tools are merely incorporated to support employees in their tasks, in this case, the handling of incidents. Therefore, for IV, ServiceNow supports employees in handling incidents and is part of agile methodology.

To seek explanations for why there are so many tools and what these different tools exactly do, it became clear there are not too many tools regarding incident management. Researching ServiceNow showed that there is a large number of incidents currently handled at Menzis and showed the incidents that were already handled. There are multiple tools resulting in unclarity and not being able to fully operate in an agile way (see section *4.3*: *Tools for sprints and software development*). However, regarding incident management, there are other reasons for not being able to be agile (see *Table 8: Interview with an expert in ServiceNow* and its analysis in section *5.1: Results of the interviews*). That is why the decision was made to include these "other reasons" in the report and write recommendations related to these problems. So these problems were found after researching "*there are too many different tools*."

When an employee experiences a problem regarding hardware, software, connections, PC applications, etc., one can easily open a report via ServiceNow through the intranet. After opening ServiceNow, one can go to the service portal, where a detailed explanation of the problem can be left. It is even possible for employees to leave a problem description of incidents other employees are experiencing. After an incident has been reported and assigned to someone, the incident will be given a "priority" on a scale from 1 to 5, with 1 being the highest priority. The incident will not necessarily be assigned to the ServiceDesk at first. For example, often, when an incident occurs, employees call or e-mail the people they know will be able to handle it. Thus, e-mailing or calling employees does not go through the ServiceDesk.

[3] ServiceNow.com

This can be seen in the flowchart as well. A clear overview of the flow of incident reporting is depicted in figure 4-4.

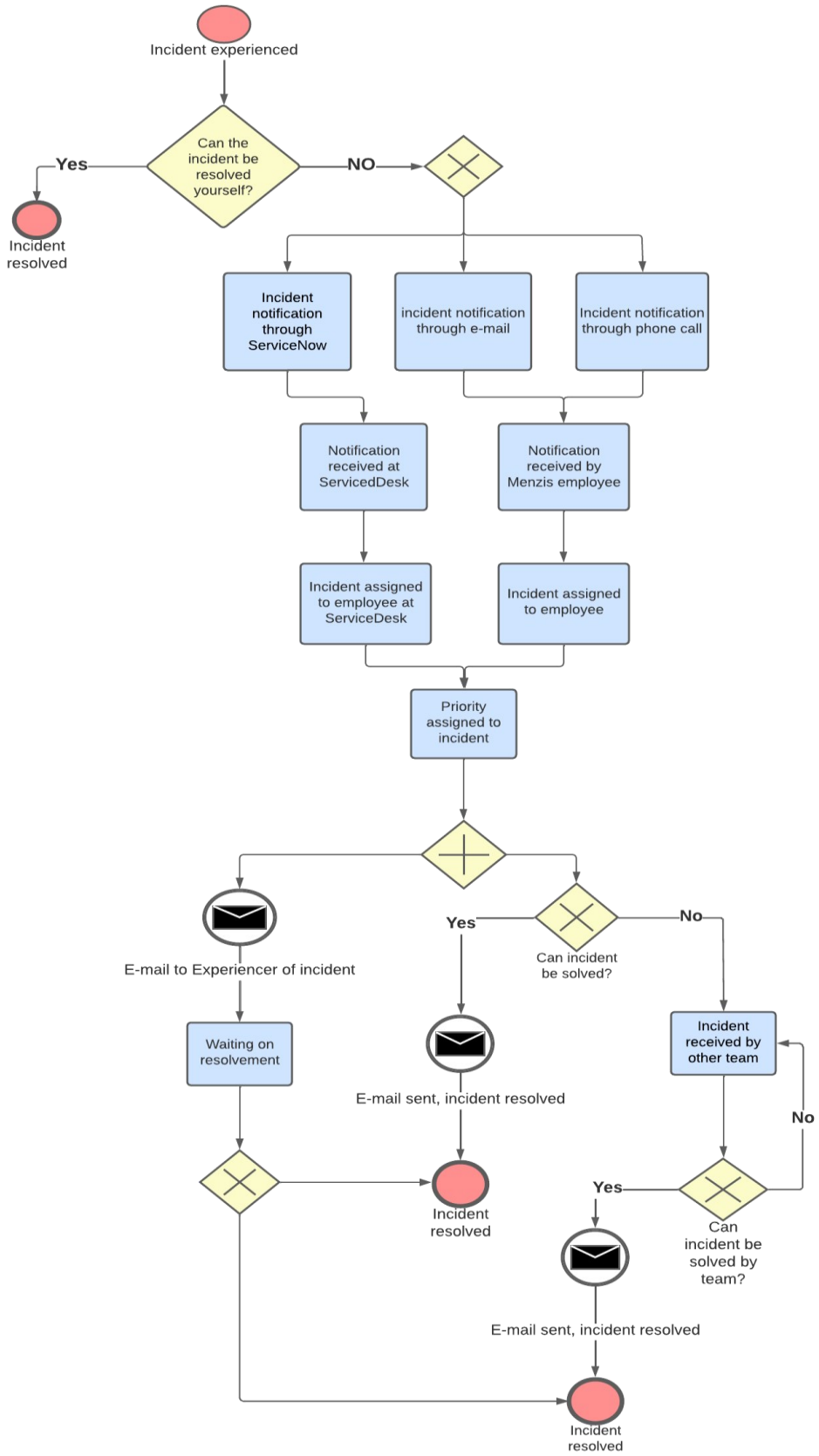

Figure 4-4: Flowchart incidents

As research needed to be conducted regarding incidents, access was granted to a dashboard with an overview of all reported incidents. The dashboard has a tab with “Open incidents” and “Closed Incidents.” Open incidents give an overview of all the incidents that were not able to be solved/still need solving. These include incidents from a few minutes ago to incidents from days or weeks ago. It can be seen who experienced the incident, what the incident is, the category, what the priority of the incident is, the status of the incident (already assigned or not), what department is experiencing the incident, and to whom the incident is assigned. It is possible to see other attributes depending on what is selected. So, this dashboard makes it very straightforward to see/understand what incidents still need handling. The tab concerning the closed incidents looks the same. However, the status is always set to “solved,” and the incident is always assigned (otherwise, it could never have been solved). Comparing the two tabs shows that more incidents are solved than unsolved (more than four times as many). This indicates how many incidents are solved compared to unsolved.

To ensure the feasibility of analysis, there is also a tab with incidents to see which incidents have not been assigned yet. This way, checking the number of unassigned incidents daily is much easier.

To further elaborate on the importance of incident management, some calculations have been made regarding mean resolve time and mean time between resolved incidents. These calculations have been made to give the researcher and the reader an indication of how many incidents are resolved and the corresponding durations. These calculations have been done with Excel. Furthermore, the data within the excel file has also been exported to SPSS to find possible outliers. These outliers can then be further investigated to explain why these incidents had such a high-resolution time. IV incorporated agile methodologies to solve incidents quickly. Researching the outliers can therefore indicate why it is not always possible for IV to quickly resolve incidents and how this is possibly related to agile methodologies. The SPSS results can be found in the Appendix.

At the moment, incidents resolved from January 2018 till the tenth of May 2022 have been included in the calculations. This means a total of 75050 incidents were solved in the period mentioned. Keeping in mind that there are also incidents that were not solved or still need solving shows that the number of incidents at Menzis is very high. The mean resolution time calculated during this period is 230 hours. Many incidents only take a few minutes to be solved. Having a mean of 230 hours could indicate multiple high outliers.

This mean of 230 hours also includes nights, weekends, and holidays, resulting in a high mean. The option “business resolve time” shows the time it takes for an incident to be solved during office hours. Office hours within ServiceNow include 8 hours daily, excluding weekends. For the user, who needs a fast resolution of the incident, the ”resolve time” is the most important.

What could also be an important calculation regarding resolved incidents is the mean time between resolved incidents. This, on average, shows the number of incidents solved every day and at what rate. The mean time between resolved incidents is 29 minutes and 18 seconds, which gives a rate of 49.66 incidents per day. Therefore, an incident is solved less than every

half hour. A screenshot of the Excel file with the corresponding numbers can be found in figure 4-5. A screenshot with the adjusted mean for outliers can be found in the appendix.

Attributes	Numbers
Mean resolve time in seconds	829629,1796
Mean resolve time in minutes	13827,15299
Mean resolve time in hours	230,4525499
Mean time between resolved incidents	00:29:18

Figure 4-5: Numbers regarding resolving incidents

Furthermore, figure 4-6 shows that more than 20,000 incidents took longer than one million seconds to be resolved. It also shows that almost 55,000 incidents took less than one hundred thousand seconds to be resolved. However, the high mean of 230 hours can be explained by the high number of incidents that took over one million seconds to resolve. This can be substantiated as around 3500 incidents took more than 5 million seconds to resolve. As one million seconds is as much as 277 hours, the 3500 incidents that took over 5 million seconds to be solved are too high.

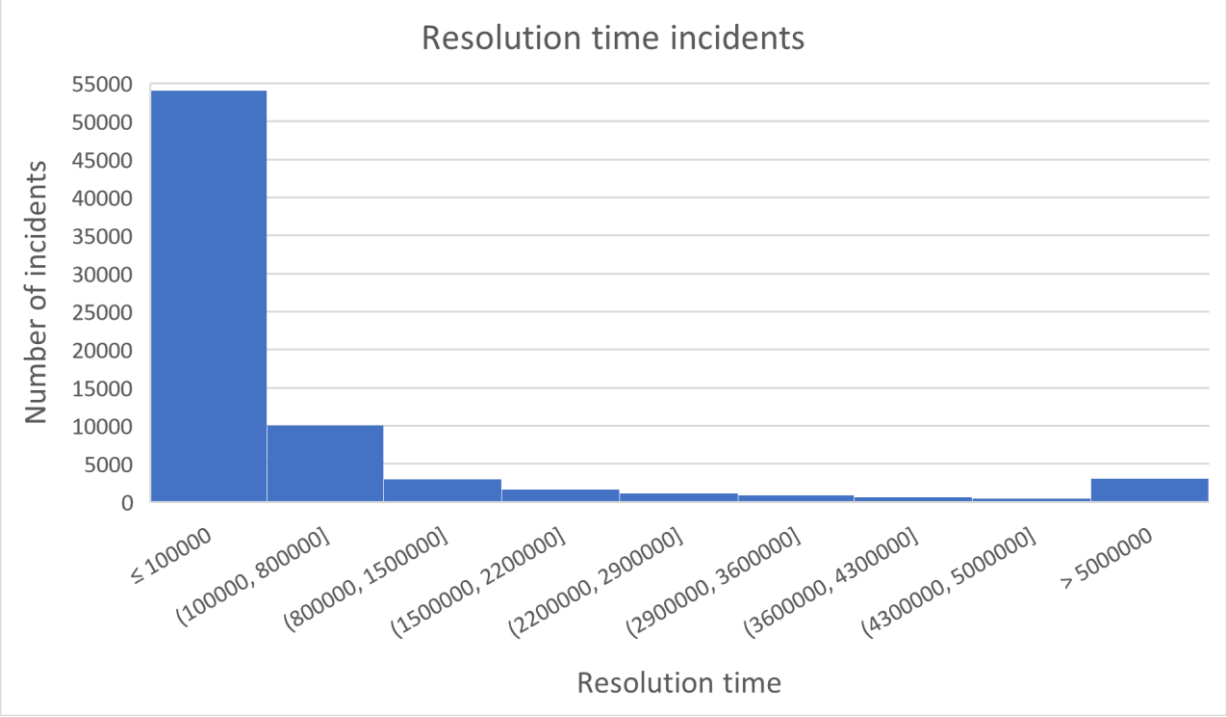

Figure 4-6 Resolution time incidents

Figure 4-6 aims to show that many incidents took over one million seconds to solve in the current situation. However, compared to the incidents that took less than a hundred thousand seconds to be solved, only a small number of incidents took over one million seconds. With agile methodologies, the aim is to reduce these resolution times. As the time to resolve incidents is currently high, agile methodologies could be applied better to reduce the resolution time, for example, iterations and retrospectives where teams can reflect on how they solve incidents and adjust their behavior. Adjusting the behavior or the process of solving

incidents is done to solve these incidents faster and more efficiently. Having high resolution times substantiates the fact that agile principles are not supported enough. It is important to determine why these resolution times are that high.

4.3 Tools for sprints and software development

It is recommended to follow a chosen agile framework to better support an agile working environment. The idea of the framework is to have a specific approach to planning and executing agility. That is why Menzis has also chosen an agile framework that needs to be followed. The chosen framework aims to continuously provide software through the tools mentioned in this section.

Dashboards can be found within ServiceNow under "Menzis Agile Framework." Figure 4-7 shows an overview of these dashboards and what they look like within ServiceNow. The name of one of the employees has been blurred out due to privacy matters. ServiceNow shows dashboards related to the SAFe framework. However, even though ServiceNow is still used for software development, the SCRUM framework is incorporated to execute agility.

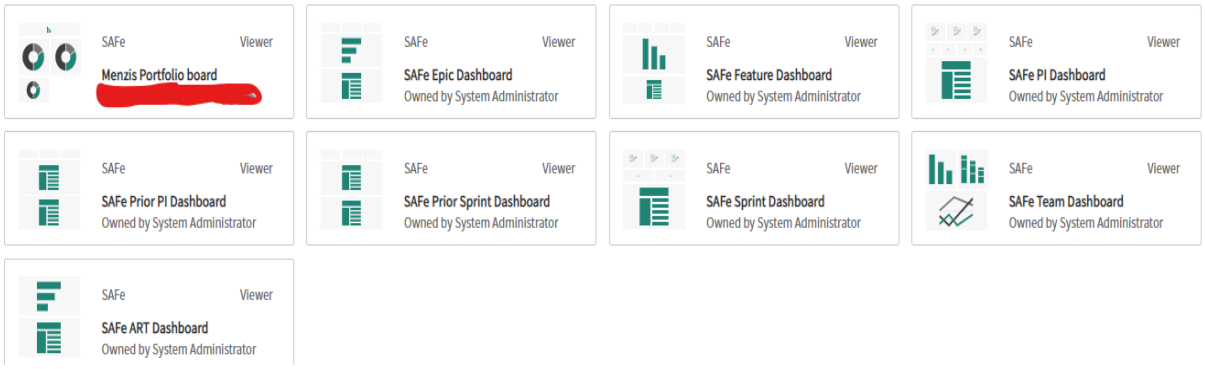

Figure 4-7: Overview Agile dashboards ServiceNow (Menzis Intranet, retrieved on 10-05-2022)

Figure 4-7 only shows the agile dashboards in place. However, within ServiceNow, related to the framework, an agile board, epics, features, defects, and stories can be seen. The IV agile board, for example, shows the current backlog of IV; the specific tasks that need to be finished for the product owner to reach its goal. This agile board currently shows 30 different epics that are finished according to a "weighted shortest job first (WSJF)" score. These epics, in their place, are broken down into specific features that need to be finished. These features describe what needs to be done, who created the feature, the feature owner, the state of the feature, and who is concerned with the feature. As mentioned earlier in this section, there is also a tab that includes all the features. This tab in its place shows what epic the feature belongs to. The epics and features are activities that must be finished within the current sprint.

Next to ServiceNow, two other tools are used for the sprints and keeping track of the backlogs: TFS and DevOps.

DevOps

Within Menzis, DevOps is used to guide the development processes through all the team's Kanban boards, sprints, backlogs, and backlog items. These sprints and backlogs relate to the SCRUMs of agile methodology. Several SCRUM masters within IV cover multiple teams within

both TFS and DevOps. The SCRUM master ensures that the SCRUM methodology is performed correctly and uses DevOps to do so. DevOps will be described as one of the causes of too many tools. As mentioned in section *2.3: Tools and systems*, there is code processed within DevOps related to software development and architecture.

Within the DevOps development methodology, it is possible to get an overview of everything development teams are working on and what has been finished. Thus, specific activities need to be performed by team members. Seven tabs help the teams understand the projects and DevOps' goals. First, there is the overview tab, which explains how SCRUM will be applied. Part of the SCRUM and what has been explained in DevOps is SCRUM Retrospective (see section *2.1*). To quote the mindset of IV regarding Retrospective: "*The purpose of the sprint retrospective is to plan ways to increase quality and effectiveness."* At IV, the sprint retrospective takes place after every sprint to evaluate and discuss how the sprint went. Next to the sprint retrospective, IV plays planning poker during the refinement stage to estimate the effort of the sprint. The efforts are related to points, where three points mean one day of work by a full team. A maximum of 8 points are assigned to a backlog item. If the backlog item exceeds points, too much effort has to be put in, and the backlog item will be split. Thus, planning poker is a fun way to estimate efforts for the next sprint. The overview tab of DevOps shows much more information regarding how things are handled and how everything works/employees have to comply with DevOps and SCRUM. Think of explanations of the SCRUM framework and security rules regarding data.

DevOps has different dashboards which show a team's statistics regarding how much time it takes for backlog items to be completed, for tasks to be completed, to make a backlog item, and to make a task. These are called lead times and cycle times. Lead time is the time between the product owner asking for a product and the time this product is delivered. Cycle time is the time it takes for the product to be made. It makes sense that, therefore, the lead time is higher than the cycle time.

DevOps also has different boards regarding the work tasks, Kanban boards, backlogs, sprints, and retrospectives. The work tasks are all the tasks that still need to be finished, tasks that are already finished, and tasks that are removed. It shows the task's state and who is assigned to it. This way, it is very clear what still needs to be finished and by whom. Tasks are part of a product backlog item that needs to be finished for finishing the backlog item. A better explanation of the hierarchy of backlog items can be seen in figure 4-8:

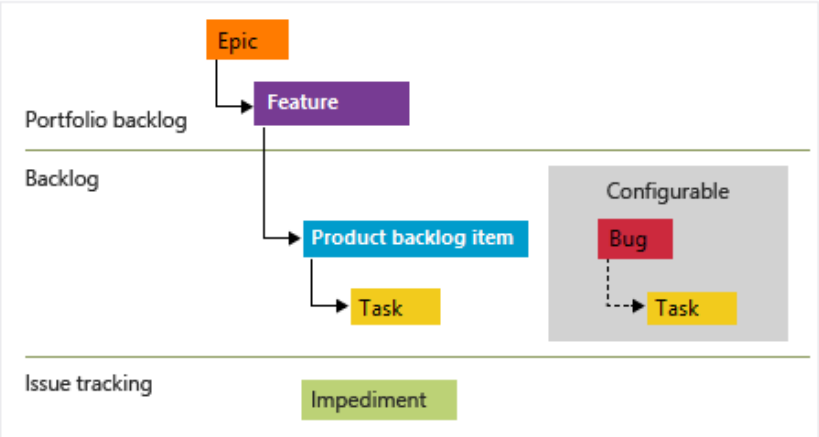

Figure 4-8: Backlog work item hierarchy (Menzis intranet, retrieved on 30-06-2022)

This shows that tasks are small activities broken down to finish the "epic."

As mentioned earlier, the SCRUM methodology uses sprints to support an agile way of working. Thus, within DevOps, it is possible to view the current sprint and the past sprints. Every sprint shows the backlog item related to that sprint. It can be seen which backlog item is done and which still needs to be finished. Clicking on a backlog item shows the tasks needed to finish the item and the tasks related to the backlog item. It also shows the feature to which the backlog item belongs. After clicking on the feature, one can see all the backlog items that need to be accomplished to complete that feature. It also shows the epic of that feature. An epic is a broad description of what a product owner wants to be finished. The epic is broken down into more specific activities to finish the epic. Thus, within DevOps, it is possible to navigate through the backlog item hierarchy to see what activities are related to what epics and features. This overview of epics, features, product backlog items, and tasks is how IV applies the SCRUM framework to support an agile way of working. IV uses Kanban boards within DevOps to also create a great overview of the tasks that need to be done. Kanban is not part of SCRUM but allows the users of DevOps to create another great overview of the tasks that need to be done.

As mentioned before, DevOps is not only used as a way to order backlog items and view sprints. DevOps also guides the development process, infrastructures, and software configurations. This software development is done by altering or adding code to existing code. Thus, there is code processed within DevOps related to software development and architecture. These codes are saved within "pipelines." So within the tab "pipelines," all the code written in DevOps can be found.

TFS

TFS is the predecessor of DevOps and thus has multiple similarities compared to DevOps. TFS enables teams to smoothly and better work together to complete projects. All the projects the teams work on are stored in TFS.

When opening TFS, there is the option to choose the sprints of all the teams. The teams have different sprints, epics, features, and backlogs. The duration of the sprints also differs. Some teams work in sprints of two weeks, and some work in sprints of four weeks, for example.

Within TFS, it is possible to see the current and past sprints. Within the sprint, the current backlog and Kanban board are shown. The Kanban board shows what tasks still need to be finished, which are in progress and completed, and by whom. The items within the backlog show a description of the created item, to whom the item is assigned, the severity of the item, and the state of the item. This shows that both ServiceNow and TFS have the same components. However, TFS is a little different because the epics are cut down into features, which are then cut down into specific backlog items. Compared to ServiceNow, where a feature is not cut down into backlog items, the state of a feature can be "backlog."

Thus, comparing TFS with DevOps shows that the two tools have relatively the same attributes. The view is different, but both tools have dashboards, epics, features, backlog items, and Kanban boards. Also, both tools have code integrated regarding the development of software. This all makes sense because TFS is the predecessor of DevOps. Thus, DevOps can be seen as the newer version of TFS. However, TFS and DevOps are currently used simultaneously at IV.

4.4 CMDB

The CMDB at Menzis is used as a repository of information regarding hardware and software. The CMDB shows all the applications, services, computers, database clusters, database instances, databases, and servers. Having an up-to-date CMDB is important as it reduces the number of outages, and having an overview of all the hardware and software helps to reduce the time it takes for outages to be restored. Furthermore, the CMDB supports incident management as changes and root causes can be found within the CMDB. Therefore, tracking incidents and seeing what CIs are impacted by them is possible. Furthermore, by understanding the root causes of outages, outages can be prevented, which reduces the number of outages. Thus, a CMDB supports responding to changes, problems, and opportunities closely related to agile methodologies.

More precisely, the following principles can be linked to the CMDB: Principle II: Welcome changes, principle V: Motivated individuals, and principle XII: Intervals and reflections. First, finding root causes with the CMDB is possible, so incidents can be prevented or solved faster. As it is possible to find root causes, teams or employees can reflect and become more effective. Therefore, the CMDB supports being responsive and simplifies the process of reacting to incidents. Furthermore, the CMDB supports reflections and becoming more effective, which aligns with the third mentioned principle. Within the CMDB, CI's are connected, and their relationships can be seen. Understanding the relations between CI's and what changes caused certain incidents can help prevent or estimate future incidents. Thus, reflecting regularly and understanding relationships and causes can help in the future with

solving incidents. With these reflections, relationships and causes can be recognized, and incidents can be solved faster in the future. Lastly, as the CMDB supports the employees in handling changing environments and incidents by reflecting and understanding relations between CIs, the CMDB should be complete to enable adequate responses/actions. Again, by reflecting and understanding what changes resulted in incidents, such incidents can be prevented or solved faster in the future. Without a complete CMDB, employees are less able to track incidents and reflect promptly. Therefore, an incomplete CMDB is not in line with having the right instruments as part of the agile environment and support employees need. As tools are part of the environment and are there to support the employees, the CMDB should be complete. This shows how the CMDB can be linked to agile methodologies and why it is important for IV to have a complete CMDB.

The CMDB, for example, shows applications registered since 2013, which indicates that the CMDB has been in place for a relatively long time. For example, compared to incident management, which has been tracked at ServiceNow since 2018.

Applications

Within the applications tab, there is an overview of the registered applications within Menzis. The overview shows the different applications, a description of the application, when it was last updated, and when it was created. It is possible to click on every different application to see further information related to that application. It shows the SAT ID, the application type, availability of the application, the integrity, confidentiality, whether or not it contains medical info, whether or not it contains specific personal data, the owner staff manager, the support group, the contract manager, and more. The SAT ID is an identification number to easily find applications within the CMDB. These applications are not automatically registered or updated within the CMDB. This is the responsibility of the employee responsible for that application. If this employee does not register or update applications, misinformation will arise.

Furthermore, not all available information may have been provided within the CMDB. For example, some applications miss the owner, staff manager, contract manager, application type, etc. This also results in misinformation/lack of information.

Within the overview of the specific information related to an application, it is possible to request an exception (*Figure 4-9*). This means the following: "*When your organization can't comply with published vulnerability management or security policy, standard, or guideline, you can request an exception. Exception management entails requesting, reviewing, approving, or rejecting exceptions for a test result group that cannot be remediated according to the policy"*[4].

[4] Menzis intranet, retrieved on 05-06-2022

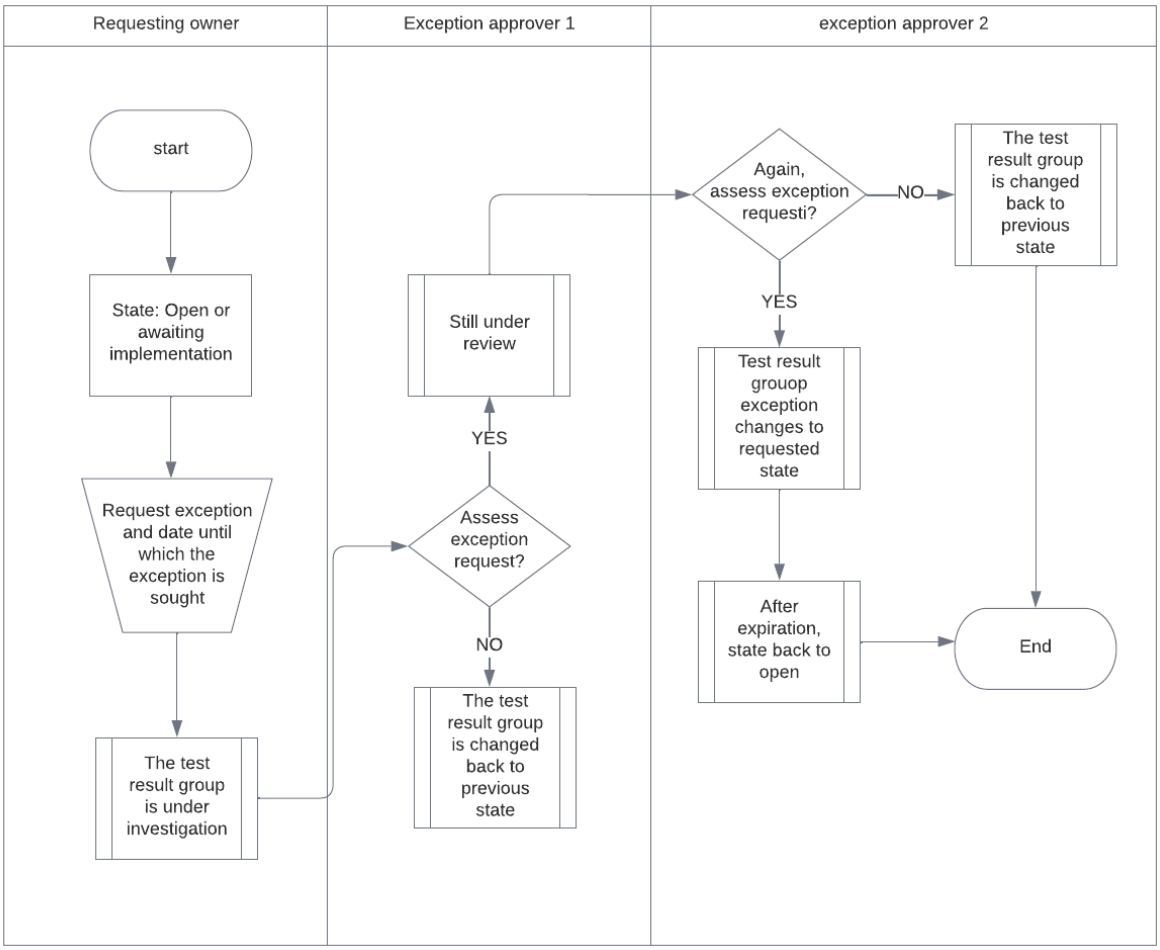

Figure 4-9: Requesting an exception (Menzis intranet, retrieved on 05-06-2022)

Within the configuration exception menu, a change number can be seen. This change number exactly shows the change that has been requested. *"A change request enables one to implement a controlled process for adding, modifying, or removing approved and supported configuration items. A change request records the detailed information about the change, such as the reason of the change, the priority, the risk, the type of change, and the change category" (ServiceNow.com).*

The sequence of steps a change request follows can be seen in figure 4-10.

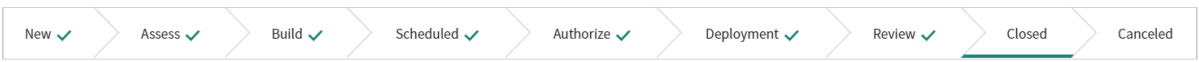

Figure 4-10: Change request sequence (Menzis intranet, retrieved on 05-06-2022)

These are the states a request goes through. The checks show the states that have already been passed/completed.

- *New*: Start describing the details of the change. It is possible to save the details without submitting the request, so one does not have to build the change simultaneously.
- *Assess*: A peer review and technical approval of the change detail.
- *Building*: finish the building of the change request.
- *Schedule:* Decide on the change's planned start date and end date.

- *Authorize:* final authorization of the change request.
- *Deployment:* the implementation of the change has started.
- *Review:* the one requesting the change checks whether or not the change has been completed successfully.
- *Closed:* the change has been reviewed and completed.
- *Canceled:* the change can be canceled in any state if it turns out unnecessary. However, when the change request has reached the "closed" state canceling the change is not possible anymore.

The many steps a change request has to go through shows the difficulty/complexity of adding/changing items within the CMDB. The many steps also show that requesting a change can be time-consuming. This is a reason for employees to pay less attention to the CMDB. It could be too difficult to understand or just take too much time. To furthermore show what is involved with requesting a change, see figure 4-11:

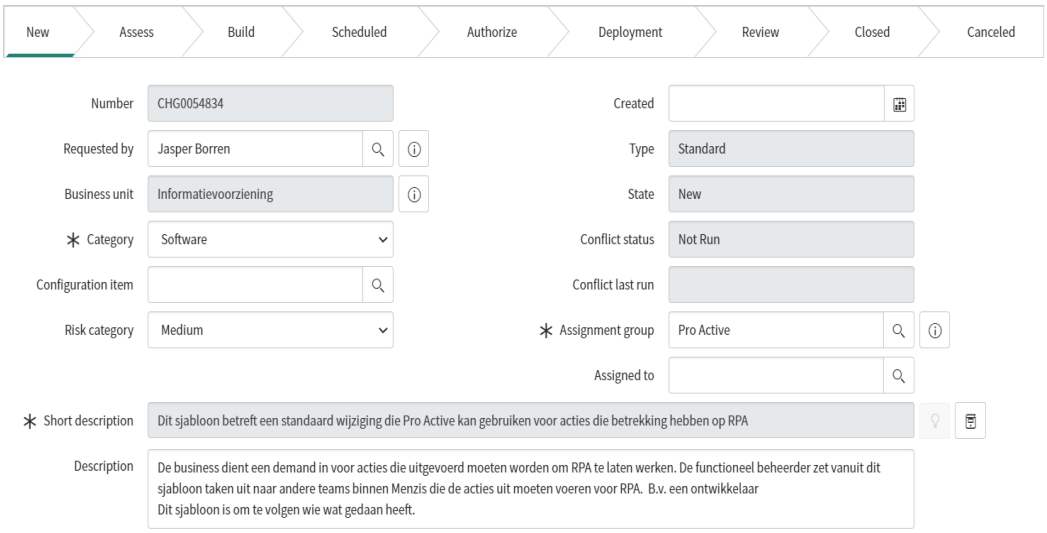

Figure 4-11: Template for requesting change (Menzis intranet, retrieved on 05-06-2022)

Figure 4-11 shows a regular template that can be used for requesting a change. Figure 4-11 shows a template that has not yet been requested and all the information that can be given. With an asterisk, mandatory fields are indicated. When much information is given, there can be a better assessment of the change, resulting in the change being processed faster. What also can be seen is that the state is "new," as the details of the change need to be described. Furthermore, what cannot be seen in figure 4-11, is that when one wants to request a change, one needs to fill in a justification of the change, risk and impact analysis, and backout plan (should the change turn out not to be needed), planning and which CI's are affected. Thus, before requesting a change, employees need much important information. If the change is small, requesting a change could not be worth it. The importance will outweigh the difficulty if the change is rather big and important/necessary to execute the work.

Computers

As mentioned before, the CMDB is a repository for hardware and software. Thus, computers are included within the CMDB as well. The computers included within the CMDB of Menzis are all-in-ones, laptops, desktops, and virtual desktops. The CMDB must be updated regularly, for example, to see what laptop is assigned to who. This way, when problems arise regarding a specific laptop, all needed information regarding that laptop can be found in the CMDB. Furthermore, when changes are made to a computer, these can be easily tracked.

For example, the laptop used for writing this thesis was a Menzis laptop named PS12576. This specific CI was created in July 2021 and had the status "in stock." This means the laptop was not assigned to anyone yet but was available to be assigned. Before the laptop got assigned on April 26th, the status changed from "in stock" to "pending install." After the user received the laptop, the status changed from "pending install" to "installed." These changes in the status can be seen in figure 4-11. It shows who the laptop got assigned to when it got assigned, and the location of the assigned laptop. The sub-status even changed, showing that the laptop is now in use and no longer available.

This example shows that Menzis is not completely reluctant to use the CMDB and keep track of changes. This enables Menzis to see whether or not laptop PS12576 is available to be used or has already been assigned. Should this not be possible, then Menzis would have no idea where this laptop would currently be. Furthermore, when the laptop is returned, the status would be changed back to "in stock," telling the company that the laptop is ready to be assigned again.

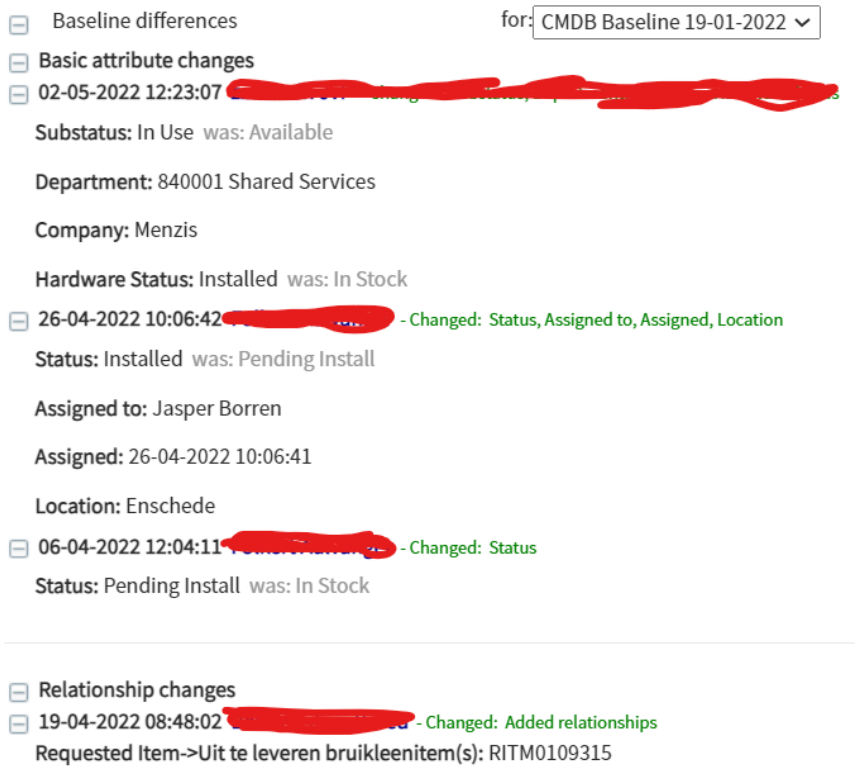

Figure 4-12: Example track of changes (Menzis Intranet, retrieved on 05-06-2022)

There are many other CIs in the CMDB, such as servers, databases, ethernet connections, routers, switchers, mobile devices, and other electronic devices. This shows how extensive and complex a CMDB can get. It also shows that when used correctly, an organization can establish a great overview of its devices.

4.5 Summary chapter four

A description of the current situation starts with describing the current organizational structure at IV. IV comprises seven departments, where IV solutions make up most of the delivery of working software. An explanation of ServiceNow is included as ServiceNow is used both for delivering software and handling incidents. Furthermore, the CMDB and its relation to IV and agile methodologies have been explained. Lastly, the different tools related to the SCRUM framework and software development have been explained. In the next chapter, the results of the observation and interviews will be discussed.

5. Problem identification

This chapter will describe and discuss the specific problems causing the core problems. Together with observations and interviews, the problems or irregularities causing IV not to be optimally agile were found. This chapter answers the following knowledge question: "*What are the root causes for IV not being able to optimally work in an agile way?"*.

5.1 Results of the interviews

To find and explore the root causes of Menzis, interviews were conducted with employees of Menzis. These employees were the ones responsible for the tools and the CMDB. Therefore, they could perfectly explain what they think the tools/CMDB lacked. The interviews conducted are solely qualitative questions seeking qualitative answers. The interviews were conducted to explain the current situation at Menzis and find the problems they currently experience. Therefore, the questions were created, so the interviewees had much freedom in their answers. The exact transcripts of the interviews can be found in section *10.4: Interviews* in the appendix. This section reflects on the interviewees and the results of the interview.

Interview with a SCRUM Master

One of the employees interviewed is the SCRUM master of two teams that work with DevOps and TFS. As SCRUM master, he has the responsibility to encourage the teams to work in an agile manner and keeps track of the backlog. He was, therefore, a good fit with the thesis goal, namely, building an understanding of the use of these two tools in the context of Menzis's objectives. *Table 6: Interview SCRUM master* in the appendix shows the most important questions that were asked and the answers to these questions.

As a result of this interview, it became clear that it is difficult for the IV teams to be agile as multiple tools are used for the backlogs. The SCRUM Master has to shift between tools to see what tasks need to be finished, which makes it confusing. Furthermore, if teams use different tools for software development, the activities as a whole become unclear. It is difficult for teams to be transparent with each other if employees do not know who works with what tool. This contradicts the third pillar of the SCRUM framework, stating that teams and employees should be transparent. Currently, the underlying code intertwined within TFS is the root cause

of not having one uniform tool. Coding is used to develop software continuously. It is not possible to copy-paste this code from TFS to DevOps.

Interview with a domain architect

The next interview was with a domain architect. As a domain architect, he works with ServiceNow and the CMDB. As part of the research includes researching the CMDB and ServiceNow, interviewing a domain architect makes sense. The idea behind the interview was to find possible reasons for the incomplete CMDB or why incident management is not fully optimized. Also, questions were asked regarding what to look at within the CMDB, but those are not important for problem identification. These questions were asked to better understand how to work with the CMDB. These were merely some tips and tricks which helped ease the research. The interview transcript can be seen in *Table 7: Interview Domain architect* in the appendix.

As a result of this interview, it became clear that "the CMDB does not contain enough and the correct information" because not everything that needs to be stored can be stored. Some people do not store everything that needs to be stored as they do not understand the added value of filling in the CMDB. If a person does not understand the added value of the CMDB, it can be seen as too time-consuming to use the CMDB. Therefore, employees neglect the use of the CMDB. It differs per team on what is necessary to be stored, which makes it difficult to agree on what the CMDB should be able to store. Without having a complete CMDB, it becomes more difficult for IV to reflect on the handling of incidents; an explanation of the importance of the CMDB and how it relates to agile methodologies have been discussed in section *4.4: CMDB* and section *8.1: CMDB.*

Interview with an expert in ServiceNow

One interviewed employee often works with ServiceNow and previously worked at the ServiceDesk. At the ServiceDesk, the incidents are received and handled. Therefore, he could perfectly describe how incidents are received and resolved. The interview intended to determine how incidents are currently handled and why some incidents have a high resolution time. Within this interview, some comments were made by the interviewee independent of the questions but are highly important. These comments were a result of the decision to conduct a seme-structured interview. With semi-structured interviews, interviews can result in a conversation. A more detailed explanation of the difficulties of semi-structured interviews can be found in section *7.3: Difficulties of the research*. The interview transcript can be found in *Table 8: Interview with an expert in ServiceNow* in the appendix.

An elaboration of the first question and answer can be found in section *5.3: Results observation*. This interview was helpful as it helped to identify where and what to look at within ServiceNow. Furthermore, the interview showed multiple reasons for some incidents to have a high resolution time. First, as the ServiceDesk is outside the organization, priorities could differ from the priorities of Menzis. Secondly, incidents are shifted from team to team, which takes time. Finally, due to the absence of an incident manager, reaching the SLA has been less of a priority for employees. One of the incident manager's responsibilities was monitoring the incident resolution process and ensuring the SLA would be reached. Thus,

without a manager monitoring the resolution process, employees could become more negligent in reaching the SLA. This does not mean that employees do not care about reaching the SLA. They could, however, become more negligent without a manager monitoring them. Furthermore, an incident manager can also help and support the employees to reach the SLA. Thus, the absence of an incident manager can result in employees having more difficulties resolving incidents. The importance of quickly resolving incidents concerning agility has been explained in section *4.4: CMDB.*

Meeting with another Domain Architect

The goal of this meeting was merely to identify more reasons why agile working is not yet fully supported and for him to explain how certain tools within ServiceNow work. As the first interview with a Domain Architect was mainly focused on the shortcomings of the CMDB, this meeting had no particular focus on a tool to begin with. The meeting intents to find out and describe his experiences regarding agile working. Furthermore, he works with both the CMDB and the tools for software development, enabling him to have the freedom to talk about both subjects. If new reasons were to be found for IV not being able to be fully agile, new answers to the research question could be found. The meeting was initially meant to be an interview. However, after asking the first question regarding agile working, the interviewee elaborately explained all the experiences/problems the company had with the current situation. This resulted in only one follow-up question, leading to all the information that was aimed for. The interview transcript can be found in the appendix in *Table 9: Meeting with another domain architect*.

This meeting was helpful, showing multiple reasons for hindering an agile working approach. First, configurations are not always checked, resulting in possible incorrect notated information. In the past, configurations were checked to ensure no mistakes were made. Secondly, employees and teams have limited authorization to make decisions, resulting in significant time loss. Employees have to ask for permission too much to make certain decisions. Thirdly, when changes are made within TFS, these changes cannot be seen within ServiceNow. This substantiates the argument that having too many tools can result in unclear overviews. This relates to managers losing an overview of the backlog and backlog items. Furthermore, having integrated code on different tools for software development also results in unclarity for the involved employees. See section *4.3: Tools for sprints and software developmento* better understand what is meant. It could be much more effective if changes are made and these changes are aligned with the other tools.

5.2 Discussion interviews

For the interview with the SCRUM master, the goal was to find out why there were multiple tools for keeping a framework and backlog. Furthermore, determining whether having multiple tools for keeping a framework and backlog would be a problem regarding agile working. That is why the third question, "*Is this use of three different tools not confusing? Is it not preferable to have a uniform tool everyone can use?".* The answer to this question indeed revealed the information that was searched for. Even though it was a biased question where the interviewee tended to relate agile working to having too many tools, it did show that having multiple tools is not ideal. The answer to this question shows that the SCRUM master

has to shift between backlogs between tools. This results in unclear sprints and makes it unclear what tasks are stated where. Therefore, it becomes difficult to see what tasks are already completed and which need yet to be completed. This results in being less flexible and the loss of time for employees involved and the managers of the teams. Principle V: motivated individuals of agile methodology states employees should be provided with an environment they need. However, an environment where tasks and code are divided over multiple tools is an unclear and undesirable environment. Therefore, seeking solutions for creating a better environment where limiting the number of tools is desirable is in line with agile methodologies.

For the interview with the ServiceNow expert, the goal was to find out why it could be possible for incidents to take a long time to be resolved. The research questions seek answers to what can be done or altered to better support agile working. As mentioned before, part of being agile is responding quickly to problems/incidents. Therefore, finding reasons why some incidents take a long time to be resolved can be altered to reduce this "time to be resolved." As this "time to be resolved" relates to being agile, it relates to the research question. Principle I: "Customer satisfaction" of agile methodology states the highest priority is to satisfy the customer through software delivery. For IV and the handling of incidents, the highest priority is to satisfy the customer through restoring daily operations as quickly as possible. Therefore, as the problems mentioned in the interviews (transferring of incidents, for example) hinder the fast restoration of daily operations, an agile working environment is hindered.

The questions regarding the SLA were of importance, as it turned out the SLA is partly the reason for employees to quickly resolve incidents. This is because the SLA is the maximum time for an incident to be resolved. Breaching the SLA would have consequences for the business side of Menzis, but it also serves as underlying pressure to solve an incident in time. The business side of Menzis is out of the scope of this research. However, it should be mentioned that not complying with the SLA would have consequences for the business side of Menzis. Therefore, understanding the determined SLA could explain why some incidents are quickly or slowly resolved.

As mentioned before, the interviews were conducted to answer the following knowledge question: *"How can information regarding the root causes be used to help IV support an agile working environment ?"*. From the interviews, reasons as to why IV is not able to be fully agile were found. Think of findings such as employees not understanding the use of the CMDB and being unable to store everything that needs to be stored. As mentioned earlier, the CMDB helps with retrospectives and welcomes changing environments. Therefore, the findings of why the CMDB is incomplete are causes for IV not being fully agile. Now, related to these findings, recommendations have been formulated. The literature used to formulate the recommendations is mentioned in section *2: Theoretical framework* and section *8: Recommendations*. How the existing literature was found has been explained in section *10.3: Systematic literature review*. For example, recommendations have been made about incidents being shifted between teams. This shifting of the incidents results in a longer resolution time, which can be reduced by agile processes. Now, recommendations can be written to prevent the incidents from being shifted or reduce the number of shifts.

5.3 Results observation

Observational studies have also been conducted to find the root causes of the core problem. The observation includes observations regarding organizational structures, the tools in place at Menzis, and the human factors related to the organizational structures and tools. The research question seeks factors within IV that can be altered so IV can work as agile as possible. For example, adjusting people's attitudes toward the CMDB, how employees handle incidents, and the IV structure will help solve incidents with agile methodologies. The goal of the observation was to find and analyze factors and conclude whether or not the factors influenced agile working. Thus, by observing IV and the tools in place, why IV cannot be fully agile should be found. In this section, the results relevant to agile working will be stated.

All the observations regarding the current situation at Menzis and agile working are explained in section *4*: *Description of the current situation*. This intends to get an overview of important data and simplify the interpretation of data. This is data about the current organizational structure at IV, such as the present IV departments and the structure of the teams. Also, data about the CMDB, the complexity of making changes, and the amount of missing information. The complexity of making changes is why the CMDB is not being used, and the missing information shows the CMDB is not being used. As mentioned before, the CMDB helps with retrospectives and is, therefore, important for IV. This section shows the results of the analysis and interpretation of the current situation.

Tools for agile framework and software development

When observing the SAFe dashboards at ServiceNow, it can be seen that it is not updated that often. Many features and epics are from 2020 and 2021, with only a few stemming from 2022 (when looking at the dashboards). The dashboards themselves also show few and outdated data. These observations indicate that the agile Framework is not used often or well enough. It also substantiates that other tools are in place for executing and keeping track of epics and features. The features often lack information about the feature owner and who is involved. This missing information can confuse.

Observations show that code is integrated and stored within TFS and DevOps for software development. As TFS is the predecessor of DevOps, many teams within IV started using TFS. Therefore, many teams have code related to software development stored within TFS. This makes it difficult to switch from TFS to DevOps, even though this is desirable. As mentioned earlier, it can get difficult for SCRUM masters when backlogs are spread over multiple tools.

CMDB

The CMDB lacks significant information, and many options for describing the hardware/software are left out. This results in misinformation, miscommunication, and an unclear overview of the current hardware/software available at Menzis. As can be seen in the description of the current situation, much complexity and time are involved when requesting a change. When a change takes too much time, employees are reluctant to request this change in the CMDB.

As mentioned earlier, the complexity of making changes is a reason for the CMDB not being used, and the missing information shows that the CMDB is indeed not being used. The CMDB helps with retrospectives and is therefore of high importance for IV. A more up-to-date CMDB can simplify retrospectives, and agile methodologies can be better supported. Employees can better understand what CIs are connected to and what changes resulted in incidents. Understanding what items are impacted by certain incidents can also help estimate future incidents' priorities. Therefore, recommendations will be formulated related to TAM and agile methodologies. Thus an up-to-date CMDB helps the following principles: Principle II: "Welcome changes," Principle V: "Motivated individuals," and principle XII: "intervals and reflections." So, the CMDB greatly impacts IV, agility and handling incidents. To understand TAM, see section *2.4: Technology acceptance model.* To further understand the CMDB and the relation between its recommendations and agility, see section *8.1: CMDB*.

When an incident occurs to a certain laptop, the one handling the incident can go through the CMDB and quickly find out whom the laptop is assigned to, what operating system the laptop has, what the version of the operating system is, how old the laptop is, etc. When an incident occurs with the software of a particular laptop, it could be necessary to know what operating system is in place or what CI's are related to that software. It could be possible that the incident occurred due to other CI's. It is, therefore, necessary for the CMDB to provide as complete information as possible. Should certain related CI's not be added to the CMDB, it becomes more difficult for employees to understand and solve the incident. Thus, a great overview of all the CI's helps with understanding incidents, finding root causes of incidents, and responding to continuously occurring incidents. As mentioned earlier, finding root causes helps with the retrospectives of agile methodologies. As information within the CMDB is missing, IV has more difficulty conforming to changing environments (responding to incidents), creating needed environments for employees to get their job done, and reflecting.

Incident Management ServiceNow

After the interview with the expert in ServiceNow, there was a better understanding of ServiceNow. This enhanced the quality of the observational studies explained in this section. Observations within ServiceNow have been conducted for incident management and seeking reasons as to why some incidents have high resolution times. These observations relate to the high outliers found by using SPSS. The five highest outliers will be described and analyzed to see whether the resolution time could be shorter and to seek explanations to prevent future incidents from being solved slowly.

Case 2177

As seen in appendix section *10.1: SPSS* (*Figure 10-3*), case 2177 is the highest outlier of the whole sample size. The highest outlier means that this case has the highest resolution time. ServiceNow shows that the start date of this incident is 03-07-2018, and the stop date of this incident is 23-02-2022. The start date is when an incident is reported, and the stop date is when an incident is closed. More specifically, the total elapsed time is 1323 days, 22 hours, and 37 minutes. The specific incident also shows that the incident has not been solved, as it

was too costly. However, after the conclusion that the solution to the problem was too costly, the state of the incident shows “resolved.”

After contacting the person responsible for eventually resolving the incident, it turned out that it was not an incident. The “subject” was wrongfully submitted as an incident. This is because the subject was a “change,” not an incident. Therefore, the incident was never solved; there was no incident to be solved. However, it did not necessarily take a long time to be resolved. It was related to wrongfully submitting an incident. The decision was made to keep it as an incident. This results in the incident just staying as an incident and absorbing time. It was decided in 2022 to remove the incident as it also lost its priority after all those years. The decision to keep the incident was due to keeping the “customer service” high. It intends to keep morale high among employees. IV decides not to complain too often to other employees about wrongfully submitting incidents. Should IV complain too often to other employees, morale could take a hit. Therefore, it was decided to keep it as an incident.

Case 9454

This incident relates to the incident with the second-highest resolution time. The actual elapsed time is 1189 days, 23 hours, and 33 minutes. In other words, the actual elapsed percentage is 7911.24%. The actual elapsed percentage shows, in percentages, whether or not the SLA has been reached. When the 100% mark has been hit, it means the total time of the agreed SLA has been reached. Thus, having an elapsed percentage of 7911.24% means the time has exceeded the agreed SLA.

Sometimes, when there is an incident regarding software or hardware, it is too difficult to solve the problem, or there is missing information to solve the incident. When there is too little information about the problem, the status of the incident could change to “Awaiting Supplier,” as is the case for this incident. The supplier of the software/hardware can have the information needed for solving the problem. So, waiting for information from the supplier is only done when there is too little information to correctly solve the incident. Eventually, this incident was not solved but was closed because the incident had been open for so long. Thus, waiting for the supplier took too long, and the incident was closed. There is no predetermined agreement about when the decision to close such an incident should be. When the incident is considered to be unsolvable or highly time consuming/costly, the incident is closed.

Case 9944

The actual elapsed time of this incident is 838 days, 1 hour, and 53 minutes. In other words, the actual elapsed percentage is 5571.73%.

During the interviews, it became clear that some incidents take a long to be solved because incidents are transferred between teams. When one team cannot solve the incident, the incident will be transferred to a team/person who is believed to solve the problem. This is one of the reasons why this incident has a high time to be resolved. The incident was transferred between teams and employees seven times. This takes time, as every time the incident is transferred, the incident must need to be taken a look at, analyzed, and decided on. After eight months, the incident was yet to be solved and was already transferred between

employees and teams six times. Then, it became clear that there was missing information. No information was missing from the supplier but the "user." Thus, the state changed from "Active" to "Awaiting User Info." This, as a result, took another year. However, the incident was still unable to be solved, and thus after 16 months, the incident was again transferred to another team. Eventually, it was decided that the incident took too long to be solved and that the employees had no time to work on it anymore, resulting in the incident being "resolved" and then "closed."

Case 2159

After reading the description of this incident, it becomes clear that this "incident" should not have been reported as an incident but as a "change." Just as with case 2177, the incident could not be solved. However, it never was an incident. A request for a change should have been made.

With a change, a user wants to change the current situation to the desired state. Thus, no incident needs to be solved. There is just a desired state that one wants to be reached. In this case, the user wanted the daily reporting to be more streamlined, as it now takes two hours daily. This, however, is not an incident regarding some software or hardware not working. Therefore, it should have never been reported as an incident.

Case 8981

The actual elapsed time of this incident was 1005 days, 19 hours, and 48 minutes. In other words, the actual elapsed percentage is 6420.16%. Thus, having an elapsed percentage of 6420.16% means the time has exceeded the agreed SLA. With this incident, the incident was routed to the wrong team. Therefore, the incident got picked up only after a few years. That is why it took so long.

Analyzing the highest outliers showed that almost all these incidents were not resolved. The incidents were closed as the incidents took too long to be solved, which cost time and money. Therefore, as it became too costly to further work on these incidents, it was decided to close them.

The following findings explain the cause for incidents taking a long time to be "resolved": wrongfully reporting an incident and keeping it due to "customer service," having to wait for supplier info, having to wait for user info, and incidents being transferred between teams. These findings help explain the high resolution times of incidents and can help formulate recommendations to prevent high resolution times. For example, by finding existing literature about the findings and adapting it to the case of IV.

Furthermore, the flowchart (*Figure 4-4*), seen earlier in this report, substantiates what was found during the interviews and other observations.

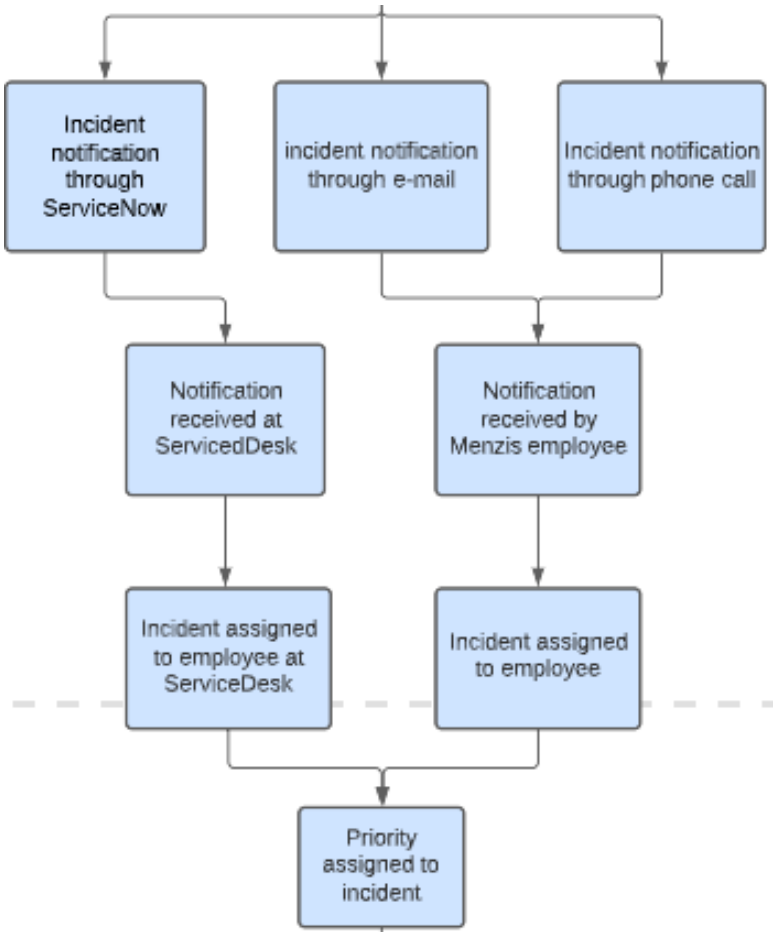

Figure 5-1: Incident notifications

Figure 5-1 shows employees do not only report incidents through ServiceNow. However, incidents are always supposed to be reported through ServiceNow. Employees at IV are already working on incidents and other activities. It is, therefore, not desirable for them to be called to solve other incidents; they do not have time for this. This way, incidents often get forgotten or neglected. Therefore, employees at IV should be assigned incidents through ServiceNow, not through e-mail or phone. It shows that employees outside of IV do not understand how incidents should be reported.

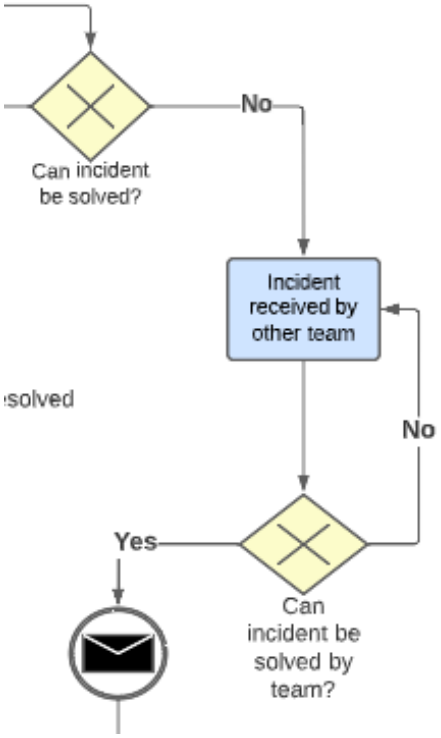

Figure 5-2: transferring of incidents

Figure 5-2 shows if an incident cannot be solved, the incident is shifted to another team. This visualizes that, theoretically, incidents can be shifted between teams infinitely. The figure is cut out from *Figure 4-4*. The obsolete word “solved” on the middle left of the figure has no importance.

As mentioned in “discussion interviews,” for IV and handling incidents, the highest priority is to satisfy the customer through restoring daily operations as quickly as possible. Therefore, as the problems mentioned in this section (waiting for supplier- and user information, for example) hinder the fast restoring of daily operations, an agile working environment is hindered. Furthermore, figure 5-2 shows the possibility of incidents being transferred infinitely. However, according to agile methodologies, in the case of IV, the highest priority is to satisfy customers through restoring daily operations. When incidents are often transferred, daily operations take longer to be restored, and agile working is not supported.

5.4 Summary chapter five

To give a good overview of the root causes and an explanation of the root causes, table 1 has been created:

Root cause	*Explanation of root cause*
There are too many different tools, resulting in unclarity.	Some underlying code within TFS prevents one unified tool for sprints and backlogs.
The CMDB does not contain enough and the correct information.	Not everyone sees their personal added value for fully using the CMDB.
	Not everything that needs to be stored currently can be stored. And the teams have different priorities regarding the needs of what should be able to be stored.
	Due to complexity, it takes much time to make changes within the CMDB.
	Configurations are not always checked, which results in incorrect notated information.
Employees do not know how/do not want to work with the tools.	Regarding the CMDB, explanations can be found in the previous rows.
	Regarding the tool for incidents, employees often do not know whether an incident or a change request should be reported. Furthermore, employees sometimes leave out important information when reporting an incident.
	Employees and teams have little authorization.
	Employees do not understand how to report an incident.
Incident management	ServiceDesk is outside the organization. Priorities could differ from the priorities of Menzis
	Problems are shifted from team to team, which takes much time
	Due to the absence of an incident manager, incidents take longer to be resolved.

	Awaiting user – and awaiting supplier information increase the resolution time of incidents.
	Employees often call or e-mail to report incidents. This result in unclarity of the number of incidents and incidents being assigned to the wrong people.

Table 1: Overview root causes

6. Validity

Validity checks whether or not a measure accomplishes its claims. There are two types of validity: internal validity and external validity. Internal validity can be described as: "*do the conclusions we draw about a demonstrated experimental relationship truly imply cause?".* External validity can be described as: *"does an observed causal relationship generalize across persons, settings, and times?"* (Cooper and Schindler, 2013). To test validity, triangulation can be referred to. As mentioned before, triangulation can be described as: *"the use of multiple methods or data sources in qualitative research to develop a comprehensive understanding of phenomena"* (Carter et al, 2014). Furthermore, triangulation is a strategy to determine whether research can be validated. It focuses on getting information from multiple sources to validate the research. These sources relate to having multiple research methods, multiple researchers, multiple theories, and multiple data sources.

The conducted research for this report consisted of multiple research methods. Thus, this relates to method triangulation for validating research. The research methods applied for this report are: interviews and observations. Multiple research methods are often used to prevent biases as much as possible and to combine the strengths of multiple research methods (Thurmond, 2001). Both the interviews and the observations of this study were qualitative. For example, observations showed that incidents were often shifted between teams/employees. This was backed by interviews stating that incidents often shift between teams/employees. Furthermore, as multiple methods combine the strengths of both methods, different findings from the different methods were detected. For example, observational studies showed that time was lost due to often having to wait for the user – or supplier information, which was not mentioned during the interviews. Thus, should only interviews have been conducted, "awaiting supplier info" and "awaiting user info" would have been problems that were not found. This shows the strength of method triangulation and why this research can be validated.

7. Implementation

This section explains the steps taken, research methods, the drawbacks, and how the research was adapted after the encountered drawbacks.

7.1 Research methods

The research included three methods necessary for finding important information and data. As mentioned earlier in this report, triangulation has been applied to gather as much information as possible at IV. In the case of this research, triangulation is the use of multiple

data gathering techniques. Furthermore, the goal of triangulation is to find reliable information. For example, gathering the same data from multiple research techniques can ensure the reliability of the gathered data. For the gathering of data at IV, observations and interviews have been conducted. As the definitions of these research methods and the reasons for the use of these research methods have already been covered in earlier sections, here, a summary has been written.

- Observations: Observations have been conducted to determine why IV cannot comply with fully agile work processes. It was an uncontrolled observation; the situation was not influenced. A description of the current situation is not possible when the environment is controlled or influenced. When an environment is controlled, employees could act differently as they are being watched.
- Interviews: semi-structured interviews have been conducted to find answers to why IV cannot comply with fully agile work processes. With the semi-structured interviews, freedom was lent to the interviewees to get as much information as possible.

Literature studies research has been conducted to better understand employee behavior, agile methodology, and the formulation of recommendations. Literature studies include research that uses already existing data. The findings of the literature studies can be found in section *2: Theoretical framework*. The literature studies were a knowledge problem and aimed to better understand the agile methodology. By obtaining a detailed understanding of agile methodology, it was possible to explain why IV would want to incorporate agile methodologies to resolve incidents. Furthermore, understanding agile methodology enabled to see what parts of agile methodology IV lack, e.g., the lack of communication, face-to-face conversations, intervals, and reflections. Without the literature studies, it would not have been possible to understand agile methodology correctly.

Section *7.3*: *Difficulties of the research* explains the difficulties encountered during the execution of the discussed research methods.

7.2 Steps taken

This part of the report includes a more detailed explanation of the steps taken during the research. Thus, this part of the report provides more insight into the research.

Interviews

I. As mentioned earlier, the conducted interviews were semi-structured. The decision was made to interview the employees who were thought to have important information for the research. Thus, in choosing the samples for the interviews, the first step was to decide on what information was necessary to be collected. This was information regarding what the interviewees thought was hindering an agile working environment.
II. The sampling technique of choosing interviewees based on their expertise and available knowledge is called purposive sampling. For example, SCRUM is a framework for agile methodologies. It was decided to interview a SCRUM master. Furthermore, as IV aims to adopt agile methodologies for solving incidents, two domain architects and a ServiceNow expert were interviewed. The domain architects work closely with the

CMDB, which is important for handling incidents. For example, relations between CIs and changes that affect CIs can be found. Therefore, changes that result in an incident can be more easily traced. Furthermore, reporting the incidents goes through ServiceNow, which is why a ServiceNow expert was interviewed.

III. After a decision was made on the interviewees, the questions were formulated. These questions were qualitative and open to ensure gathering as much information as possible.

IV. All the answers were noted during the interviews to ensure the answers could be analyzed afterward. Thus, the answers were analyzed after the interviews, and the most important quotes were included in the report. As the interviews were semi-structured and there was much freedom for the interviewees, much-unstructured information was gathered. Therefore, it was rather difficult to structure the answers. However, this will be discussed in section *7.3*: *Difficulties of the research*. The information was afterward used to formulate root causes, such as shifted incidents from team to team and the absence of an incident manager.

Observations & literature study

I. The research's first step was creating the first part of the theoretical framework. Before conducting any research, a basic understanding of the tools in place, agile methodology, and SCRUM framework was needed. Otherwise, explaining the tools and formulating recommendations would have been more difficult.

II. The second step involved observing the tools and what they were used for. Thus, during this step, an understanding and description of ServiceNow, the CMDB, TFS, and DevOps was formulated. This step also included looking for reasons why IV is not as agile as they can be and seeing if the formulated factors during the interviews could be justified. During the observations, it was, for example, found that incidents are shifted between teams too often, and employees often have to wait for supplier – and user information.

III. The third step included more literature studies to explain the 1. acceptance factors such as TAM, 2. incident management, and 3. agile methodologies and connected this with the context of Menzis. Thus, in this step, the problems and recommendations were linked to agile methodologies. How agile methodologies and SCRUM relate to the recommendations can be seen in section *8: Recommendations*.

7.3 Difficulties of the research

During the research, multiple difficulties were encountered. The difficulties discussed in this part of the report focus on the research. The difficulties were eventually solved and only resulted in the loss of time.

Interviews

Before the start of the research, a list of names who had ties with DevOps, ServiceNow, TFS, and the CMDB was provided. These employees received notifications from Menzis that there could be a possibility that they would get interviewed. However, due to IV making a list, it was decided to conduct research to assess which employees would be best to interview.

Eventually, three of the four interviewees were not on the list. Thus, a better assessment of the employees was made by a non-probability technique (see section *3: Methodology*). It was difficult to find the correct employees to interview. This resulted in time loss, as interviewing the employees on the list would have been much easier. Thus, one of the interviews was conducted only after four weeks instead of the expected two weeks.

As mentioned before, the conducted interviews were semi-structured. As the questions asked were open and the interview was semi-structured, a lot of freedom was offered to the interviewees. However, during the interviews, some problems were encountered.

- One of the difficulties with semi-structured interviews is that it is possible to formulate biased questions (Fuelcycle, 2019). With these open questions, it is possible to elicit answers from a certain direction. The questions should be as open and unbiased as possible instead of questions that already lead in a certain direction. This way, answers are much more reliable. However, I had difficulties formulating unbiased questions for the first interview. For example, the third question with the SCRUM master: "*Is this use of three different tools not confusing? Is it not preferable to have a uniform tool everyone can use?".* This question, for example, already suggests to the interviewee that multiple tools are not ideal.
- Secondly, semi-structured interviews make it possible for the interviewer to lose track of what is important (Indeed, 2022). Semi-structured interviews could result too much in a conversation where the important questions are skipped/ are followed up insufficiently. That is the problem with giving the interviewees too much freedom. This is also what was noticed during the interviews. The interviews sometimes looked like a conversation where the interviewee talked most of the time.
- Thirdly, as the interviews were semi-structured and there was much freedom for the interviewees, much information was gathered. Therefore, it was rather difficult to structure the answers. The interviews needed to be analyzed thoroughly afterward to ensure all important information was found. This, however, costs much time. Systematic text condensation has been performed (qualitative analysis) for analyzing the data. A detailed explanation of this data analysis technique can be found in section *3: Methodology.*

Research on the tools (Observation)

During the research, there were difficulties in deciding the scope of the tools. To be more precise, it was planned to analyze ServiceNow, DevOps, TFS, and BD before the research. It was decided to research BD as an overview of the processes and the architecture that can be stored within BD. Getting access to BD was rather difficult; the employees at Menzis did not know who was authorized to grant access to BD. Furthermore, after a conversation with the supervisor at Menzis and an employee working with BD, it was decided not to research BD. IV is still in the early stages of building within BD, so it was not yet possible to conduct good research. However, this was only decided after a few weeks, so time was lost in trying to understand BD and getting access to BD.

Agile methodology (Literature studies)

During the research, there were some difficulties understanding the actual definition of agile working. Agile methodologies originate and are meant for consistently developing and delivering software. However, IV wants to adopt agile methodologies for solving incidents. It was, therefore, rather difficult to apply the agility principles to the specific context of IV. However, by applying agile principles such as better face-to-face communication, better self-managing teams, and better/more retrospectives, current problems can be solved, and an agile workplace is supported. Think of problems such as many escalations of incidents, low FCR, and difficulties prioritizing incidents. So, the difficulty of understanding agility and applying it to the context of IV was underestimated and cost much time. So, most literature about agility was about software development. This made it difficult for me to adapt to the literature and fit it in the context of IV. However, IV could improve after understanding the principles of face-to-face communication and self-organizing teams. I was able to link these principles with the problems IV was experiencing. For example, IV should conduct more root-cause analyses. Part of agile methodology is retrospectives; part of retrospectives are root-cause analyses.

7.4 Adjustments

Despite the difficulties encountered, important and useful information has been found. This is also the result of adjustments made during the research. These are adjustments related to the different research methods.

Interviews

- One of the difficulties encountered was the formulation of unbiased questions. This has been prevented as much as possible by taking a good look at the questions. After understanding why the before-mentioned question was biased ("*Is this use of three different tools not confusing? Is it not preferable to have a uniform tool everyone can use?")*, it was possible to prevent this in later interviews. Understanding how certain questions can be biased helps with formulating unbiased questions. However, as mentioned in section *5.2*: *Discussion interviews,* the mentioned question did elicit needed information.
- Secondly, one of the difficulties was preventing the interviews to too much turn into conversations. It was not possible to completely prevent the interviews from looking like conversations. One of the advantages was that a lot of information could be gathered through semi-structured interviews. That is why it was decided to keep the questions open and let the interviewees do much of the talking. Furthermore, all the questions were formulated before the interviews. There was always enough time for all the questions to be asked. Otherwise, if needed information was missing, contacting the interviewees through the Teams video conferencing tool was very easy. All the interviewees allowed follow-up questions to be asked through Teams if necessary. Due to the decision to keep the open questions, it did cost a lot of time to analyze the interviews. However, as this did not result in too much disturbance, it was decided to keep the open questions for all the interviews and not skip any extra useful information.

Literature studies

One of the difficulties with the conducted literature studies was understanding agile methodology in the context of Menzis. Much literature about agile methodology was related to software development instead of agile outside of that context. Therefore, it costs a lot of time to find relevant literature. Eventually, it was decided to link the main principles of agile methodology to the context of IV to find specific gaps and measurable directions for improvement. Another step was comparing the findings from existing literature that already linked these principles with a context similar to IV. This was done by understanding the agile principles represented in agile methodology documentation and tracing whether or not IV lacks in any of those principles. This process revealed that it is possible to link agile principles with the management of incidents.

7.5 Reflections on intermediate findings

During the observations and interviews, information was retrieved and reflected upon. This part of the report reflects on the intermediate findings of the observations, interviews, and literature research. This section explains how I adapted to information regarding data and the scope of this research.

Observations

Observations showed the long resolution times of incidents. The data relating to the resolution times were then exported to excel to calculate mean resolution times. The data was also exported to SPSS to determine outliers and calculate an adjusted mean for outliers. As it was noticed that there were a lot of incidents resolved within a few seconds or minutes, the realization that there would be many high outliers also occurred. It was decided to search for the high outliers and analyze them. The findings showed some incidents were not able to be solved, incidents were escalated, and IV had to wait for supplier or user info. *Section 5.1: Results observation* contains an in-depth analysis of the outliers.

Observations also showed incomplete information within the CMDB. Within the CMDB itself, it was not possible to see why the CMDB was incomplete. In the first week, it was already noticed that the CMDB missed a lot of information. Many options for explaining details of CIs were often left out. However, it was only possible to see missing information, not why the information was missing. That is why an interview with a domain architect who works with the CMDB was scheduled. He explained the CMDB has been incomplete for ten years already and explained the attitude of the employees toward the CMDB. Employees do not understand the added value of the CMDB and how it affects their work.

Interviews

Interviews showed that at IV, BD was in the beginning stages. As mentioned before, there was not much to be researched. The focus and scope of the research were therefore limited to TFS, DevOps, and ServiceNow. This relates to the finding, " There are too many different tools, resulting in unclarity."

Interviews together with observations also showed that the problem of having too many tools were already being handled. This, however, was after the description of the current situation

was finished. However, as it currently still shows agile working is not fully supported, it was decided to keep it in the report. During one of the interviews, it was mentioned by one of the interviewees said that they were already planning on integrating ServiceNow and DevOps, for example.

The interviews showed many more underlying reasons for IV not fully supporting an agile working environment. For example, the ServiceDesk being a different organization was not expected. This enhanced the difficulty of formulating recommendations; it is impossible to formulate recommendations relating to prioritizing and escalating for other organizations. However, prioritizing and escalating can still be improved and minimized at IV. For example, creating self-organizing groups that perform daily SCRUMs and prioritize incidents.

Literature studies

For the formulation of recommendations, it was necessary to search for literature. This is done to substantiate the recommendations and make them more reliable. As IV adopted agile methodologies to solve incidents quickly, literature about solving incidents was sought. The found literature, which is mentioned in the recommendations section, was all about solving incidents quickly. However, the literature did not mention any agile principles or key values. So, the literature needed to be adapted to the context of IV. Thus, the literature about solving incidents was connected to agile principles. This has also been mentioned in the "difficulties" section and the recommendations. For example, with internal communication, the FCR rate could be increased, and the number of escalations decreased. Part of internal communication is face-to-face communication, which is an agile principle. Currently, IV lacks internal communication, and agile working is not fully supported.

7.6 Summary chapter seven

Chapter seven aimed to explain the implementation process of the research methods. These research methods included semi-structured interviews, uncontrolled observations, and literature studies. Difficulties such as interviews turning into conversations or deciding on the scope of the research have been discussed. As a result, adaptations were made to tackle these difficulties. Lastly, reflections on the intermediate findings have been explained. For example, during the research, it was discovered that BD was only in the beginning stages and was therefore decided to leave out of the scope of this research.

8. Recommendations

In this chapter, multiple solutions and corresponding recommendations are formulated regarding the problems discussed in section 1.2: "*Problem identification*." The recommendations answer the following question: "*What can be done or altered in the current situation to better support agile working for the IV department of Menzis?".* The recommendations are based on the following knowledge questions: "*How can the findings from the previous phases be used for recommendations?*" and "*Has Menzis already tried the chosen recommendation in the past?"*. The recommendations provided in this chapter are based on my knowledge and expertise and a thorough literature study. Furthermore, the recommendations relate to the root causes found during this research. The root causes found in this research can be found in table 1.

8.1 CMDB

One of the problems IV experienced was that it was not possible to store everything that needed to be stored within the CMDB. Configurations are not always checked to ensure complete information. Secondly, employees/teams do not fully understand the added value of the CMDB. Thirdly, it is rather difficult to change CIs within the CMDB. It takes some time to correctly request a change and implement it. One needs to fully understand how it works to make changes.

Currently, the CMDB is not completely and correctly used by the internal customers at Menzis. As mentioned before, it was found that the CMDB is perceived as too difficult for some employees. Furthermore, some employees do not see the added value for themselves as individuals to correctly fill in the CMDB. This was all the result of conducting interviews and observations.

Thus, the before-mentioned problems are closely related to the “Perceived Usefulness” and “Perceived Ease of Use” of the technology acceptance model. Therefore, the TAM explains the issues described in this report. The external variables mentioned in the TAM or factors related to these external variables are possible recommendations to increase perceived usefulness and perceived ease of use. For example: implementing training sessions or informative lectures could increase the perceived ease of use and perceived usefulness. This, in turn, increases the intention of internal customers to use the technology. In the case of the CMDB, it could be that “not understanding” is not necessarily the problem. The CMDB cannot be filled correctly, or due to complexity, it takes much time to fill the CMDB with the correct information. If it takes much time, “ease of use” is low, but user training will not necessarily help.

After the interviews and research, it appeared that the most important factors mentioned in the TAM 2 are: Job relevance, output quality, voluntariness, and result demonstrability. At least, these are the factors that currently affect the perceived usefulness of the technology in a negative way. Currently, regarding the CMDB, filling in the CMDB is done voluntarily and is not mandatory. Furthermore, some employees do not see the relevance of using the CMDB related to their job or why it could help others. Also, the quality of the CMDB is perceived as not optimal, as multiple options for different teams cannot be filled in. So, complete information within the CMDB is not even possible, or not as much as some would want. Lastly, there is little demonstrability as employees see little direct results from using the CMDB. Therefore, recommendations that relate to the issues described in this report and agile methodology will be formulated. The recommendations described below will be based on existing literature about TAM, the specific problems, and agile principles. The TAM framework explains employee behavior and why the CMDB is not used correctly. Literature about the agile principles will then help explain how the recommendations are connected with agile methodologies.

I. Job relevance, output quality, and result demonstrability

For a technology to be accepted by employees, the technology needs to be relevant to the user. If employees see no relation between doing their jobs and using the technology, there is no relevance for their jobs. This would result in low perceived usefulness of the technology and thus low usage of the technology. Thus, showing employees that the system performs the job tasks well would increase the perceived usefulness of the CMDB.

There are several ways to encourage and persuade employees to accept technology. The two ways this can be done are the central route of persuasion and the peripheral route of persuasion. These two routes are called persuasion routes and are used to improve the external variables of the TAM 2. See section *2.4: Technology acceptance model* for a more detailed explanation of the two routes of persuasion.

The central route focuses on statistics and facts to encourage employees to use a certain technology. For example, managers could show how much time it currently costs to resolve incidents and how much time it could save if the CMDB is used correctly. Furthermore, for example, making the employees aware that having a complete and up-to-date CMDB could result in IV better supporting employees as an individual. With these kinds of statistics, Menzis could encourage employees to use the CMDB as much and correctly as possible

Thus, to increase job relevance, output quality, result demonstrability, and perceived usefulness, it is recommended to follow the central route and show the users how the CMDB can substantially improve their work. As mentioned earlier in this report, the CMDB supports tracking changes, reduces the number of outages, and ensures that outages are solved faster. Therefore, following the central route, educating the employees about these potential benefits is recommended and showing them that it would also improve their work. This would show employees the relevance of the CMDB and also increase the perceived demonstrability of the employees. Also, having an up-to-date CMDB increases the output quality of the CMDB. In the case of this research, the output quality of the CMDB refers to the extent the CMDB can show the needed information. In turn, high output quality results in more acceptance of the technology. Thus, showing employees that using the CMDB would increase the output quality could result in more acceptance. If the CMDB has complete information, the CMDB is most useful. When the influence of employees on the CMDB is shown, acceptance could increase. This is because complete information is only possible when the CMDB is filled in correctly. So, output quality is highest when employees fill in the CMDB correctly. When employees understand their influence on the output quality, the CMDB will be better used.

Following the central route of persuasion could also show why it would be worth making, for example, changes in the CMDB, despite the time it takes to make changes. If it can be shown to employees with facts and statistics, using the CMDB would be benefactory. Obstacles such as complexity and waiting long can be overwon. Thus, it would not increase the ease of use, but the positive results of using the CMDB could outweigh the complexity of the CMDB.

To conclude, it is recommended to use the central route of persuasion to show employees the importance and the added value of the CMDB. Using the central route of persuasion results in employees understanding the added value of the CMDB and overcoming the obstacle of

complexity and long processing times of changes. Better use of the CMDB would result in better tracking of changes, reduction in the number of outages, and certainty of outages being solved faster. Furthermore, the CMDB supports retrospective processes. How the CMDB can be linked to agile methodologies has been explained in section *4.4: CMDB*.

II. Voluntariness

Voluntariness is believed to include two sides. This means that adopters of new technology can perceive choice in using technology at different levels (Agarwal & Prasad, 1997). The study shows that a superior mandate would be necessary for the initial use of technology. That is, making it mandatory to use and correctly fill in the CMDB. However, "*people will use the system continuously only if they are able to view its benefits unequivocally"* (Agarwal & Prasad, 1997). Thus, just as TAM2 depicts, the intention to use can be affected by the voluntariness of the technology. Therefore, it could be possible to mandate the use of the CMDB. When it is mandatory to correctly fill in and use the CMDB, employees could see the positive effects of using the CMDB, which would increase the 'intention to use" even more. However, the CMDB will be used continuously if employees can see the benefits of using the CMDB. In that case, following the central route of persuasion would be best.

Therefore, to be in line with the previous recommendation, it is recommended to follow the central route of persuasion to increase the voluntary use of the CMDB when employees see the benefits of the CMDB, voluntariness, and t perceived usefulness increase.

III. *Perceived ease of use*

One of the problems was that it was impossible to store everything that must be stored within the CMDB. Another finding was that different teams have other priorities as to what needs to be stored, thus making it difficult to agree on what needs to be enabled to be stored within the CMDB. This hinders the perceived ease of use and output quality. Also, filling in the CMDB is rather complex and time-consuming, hindering the perceived ease of use. This part of the report focuses on the perceived ease of use.

Perceived ease of use can be defined as "*the degree to which a person believes that using a particular system would be free of effort"* (Venkatesh, 2000). Therefore, one of the recommendations would be to think of ways to improve communication and collaboration between teams, e.g., offering regular meetings for managers of different teams to come together and discuss what needs to be stored. After an agreement, the CMDB should be able to adapt to the agreed terms. This adaptation to agreed terms would reduce the effort that needs to be put in to use the CMDB. Having agreed on terms, every employee will be able to use the CMDB as would fit them best. So, this tackles the issue of not being able to store everything that needs to be stored.

Furthermore, it is recommended to provide documentation on the intranet on how certain things need to be completed, and employees could be more inclined to use the CMDB. The documentation can be used to increase understanding of the CMDB and thus tackle complexity.

IV. Conclusion

Most recommendations related to the CMDB are about the central route of persuasion. With the central route of persuasion, employees can be conveyed to use the CMDB. Why the CMDB is not correctly used can be explained by the Technology Acceptance Model. The TAM states that the use of technology can be explained by the perceived usefulness and the perceived ease of use. With the use of the central route of persuasion, employees are persuaded to use the CMDB. However, the central route of persuasion is apart from agile processes. So, nothing is altered to answer the research question, "What can be done or altered in the current situation to better support agile working for the IV department of Menzis?". A persuasion technique is applied to persuade employees to use the CMDB. No altercation to processes or the organizational structure has been made. However, to persuade people to use the CMDB, they need to be shown data and information about the benefits of using the CMDB. Therefore, for IV, it is recommended to provide users of the CMDB with documentation about the benefits of the CMDB. As mentioned earlier, by showing documentation with statistics and numbers about time that can be saved by using the CMDB, employees can get more inclined to use the CMDB. Time saved with solving incidents is of interest for employees as their future incidents can be solved better. Thus, the information and the use of the CMDB are relevant for the employees. According to Cacioppo & petty (1986), the most important factor in conveying information to the employees is showing the personal relevance of the message. Showing through documentation or presentations time can be saved by using the CMDB, employees will understand their relevance and be more inclined to use the CMDB correctly. Also, after the CMDB is better used, retrospectives are simplified, and agile working is better supported.

The recommendation to increase the perceived ease of use is to plan meetings where a group comes together frequently. The group consists of managers or employees from different teams and frequently discusses what the CMDB should be able to do/contain. Principle XI: Self-organizing teams states the importance of self-organizing teams. Therefore, to increase the use of the CMDB and be in line with agile principles, it is recommended to offer agile instruments that support self-organization. This can be done by applying the SCRUM framework and principles. A more detailed explanation of creating multidisciplinary self-organizing teams can be found in section *2.1: Literature study on Agility and corresponding concepts*. Creating a new self-organizing team is a way to increase transparency derived from SCRUM. It also increases team communication, which is recommended by principle VI: "Face-to-face conversation." In this case, the self-organizing team communicates the preferences of the different teams regarding the use of the CMDB. These self-organizing teams can also frequently check the CMDB to ensure complete information. Thus, the recommendations formulated above will result in employees seeing the value of the CMDB, employees agreeing on what should be able to be stored, employees checking the CMDB, and employees overcoming the complexity of the CMDB. This will result in better use of the CMDB and more support for an agile working environment. Creating such a self-organizing team is a change in the current situation, which the research question asks for. To summarize, it is recommended to 1. create a multidisciplinary self-organizing team, 2. include a SCRUM master in the newly created team, and 3. include a product owner in the newly created team.

These efforts to embrace the use of the CMDB also influence the following principles: Principle II: "Welcome changes," Principle V: "Motivated individuals," and principle XII: "intervals and reflections." How these three principles are related to the CMDB is explained in section *4.4: CMDB*. Furthermore, it is in line with the SCRUM framework as a new SCRUM team will be formed, including a SCRUM master, product owner, and the employees responsible for deciding on what information the CMDB should be able to contain. The team will perform daily SCRUMs where the new self-organizing team can discuss their challenges and discuss what the CMDB should be able to contain.

8.2 Incident management

It should be noted that solving incidents not only focuses on pace but also the quality of the solutions. When the quality of the solutions is high, the recurrence of similar problems will decrease. In this section, these "problems" relate to incidents that need to be solved.

The research showed several reasons why some incidents need much time to be solved. There are always incidents that take a long time to be solved due to the complexity of the problem and thus have no other underlying reason for the high resolution time. However, some incidents have other explanations for the high resolution time. These "other explanations" are: the absence of an incident manager, the ServiceDesk exists outside of the organization, incidents are shifted between teams and employees, wrongfully reporting an incident and keeping it due to "customer service," lack of authorization of teams, having to wait for supplier info and having to wait for user info, and employees do not understand how to report incidents. The correct formulation of recommendations could prevent these findings from happening too often.

The KPIs related to the root causes mentioned in the interviews and the observations are explained in the literature study part of this report. The recommendations mentioned below reduce the mean resolve time to 230 or 226 hours after adjusting for outliers. Furthermore, the recommendations mentioned below relate to the issues described in the previous paragraph and/or the before-mentioned KPIs.

With the central route of persuasion discussed earlier, it is also possible to show employees the benefits of always reporting incidents through ServiceNow. With statistics and facts, it can be shown to employees that it is better for both IV and the experiencers of the incidents that incidents should be reported through ServiceNow. This way, IV only receives notifications of incidents through ServiceNow. Employees would then pick up incidents with the available time to solve the incident.

I. Incident manager

Secondly, as the interviewee explained (third interview), the absence of an incident manager results in less pressure to handle incidents fast. It would therefore make sense to again hire an incident manager. The interviewee explained that the incident manager was responsible for monitoring, managing, and reviewing the incident resolution process to reach the SLA. Thus, the incident manager helped with keeping a short "resolve time." Furthermore, *section 2.5: Incident management* substantiates the benefits of having an incident manager. This

section shows that having an incident manager can help with shortening the resolution time. It is therefore recommended to hire an incident manager.

II. First call resolution

FCR has been explained earlier in this report and questions if the one handling the incident can handle the incident without needing further information. So, the extent the one handling the incident needs to "call" users or suppliers to solve the incident. In the context of Menzis, it is important to understand how much time is lost due to having to wait for the user - or supplier information. Therefore, a recommendation that stimulates the resolution after the first call should be written.

The recommendations related to FCR seek to prevent the status of incidents from changing to "awaiting supplier info" and "awaiting user info." So, the recommendations that will be discussed are not recommendations that seek to shorten the waiting process of the one handling the incident. Of course, there will still be occasions where information is lacking, and there will be no other option than to wait for further information. However, the following recommendations seek to minimize those situations.

First, it is recommended for every team to conduct root-cause analyses (Geraghty, 2014). Identifying, understanding, and tracing the root causes of reported incidents, can help future incidents to be solved quickly. It would mean that when incidents similar to previous incidents occur, it will be possible to solve them without waiting for further information. Also, understanding why incidents happen could prevent incidents from happening as well. This would decrease the rate of incidents. However, as mentioned in section *4.4: CMDB*, the CMDB needs to be as complete as possible to conduct accurate root-cause analyses.

Secondly, a way to increase FCR is to enhance internal communication (Geraghty, 2014). This internal communication relates to also having a manager monitoring everything. Thus, the recommendation of hiring an incident manager also increases the FCR. The manager will monitor everything related to FCR and can help and support the employees working on the incident. Thus, the manager can help employees and provide feedback about how the incident can be solved. This would increase the FCR. Thus, hiring an incident manager who focuses on internal communication between employees is recommended.

III. Escalation rate

It is possible to count the number of times an incident is escalated to another team/employee when an incident occurs. So, for example, 50% of the incidents cannot be solved by the first employee that got the incident assigned. However, in the context of IV and finding explanations as to why some incidents take a long time to be resolved, the total number of shifts per incident should be counted because the more an incident is shifted between teams/employees, the more time it takes for the incident to be solved. Therefore, a recommendation that reduces the escalation rate is written.

First, as with FCR, a root-cause analysis can help reduce escalations between teams and employees. Identifying, understanding, and tracing the root causes of reported incidents, can help future incidents to be solved quickly. When understanding root causes and incidents,

future similar incidents can be solved without having to escalate the incident to other employees or teams (Geraghty, 2014).

Secondly, cross-training employees to handle different types of incidents would ensure that when one gets assigned a task, he or she can handle the incident without having to escalate the incident (Geraghty, 2014). However, as the ServiceDesk is outsourced to a different organization, cross-training is not applicable for the ServiceDesk. It is possible to cross-train employees within IV to ensure when an incident gets picked up, it can be solved without escalating the problem. To quote: *"Cross-training can enable shorter lead time quotes and more reliable delivery by reducing the mean and variance of the cycle time to produce a product or service"* (Hopp & Oyen, 2010). It does cost time and money to cross-train employees, which relates to "training efficiency." Training efficiency is about finding the best tradeoff between the cost of training and the skill level employees need. In other words, *"cost of skill acquisition captures the average expense of training workers to cover and to retain new task types"* (Hopp & Oyen, 2010). Also, "*skill level variation considers the range of difficulty of acquiring different types of skills"* (Hopp & Oyen, 2010). As it would be too difficult and timely to train employees to become specialized in multiple fields, it is recommended to train specialized employees to understand lower-level tasks in other fields. This way, incidents will be escalated less, which in its turn saves time. Schwaber & Sutherland (2020) stated that SCRUM masters could support multifunctional teams. Therefore, current SCRUM masters should support employees in following cross-training to become multifunctional.

Cross-training would fit with the IV transition plan suggested in 2019, where employees should broaden their knowledge and expertise. The idea of broadening everyone's knowledge has not been done as much as was intended. Therefore, to be in line with the proposed ideas in 2019, it is recommended to cross-train employees as much as possible.

Thirdly, total contact ownership would also be a great way to reduce the number of escalations. "*It is when the agent who took the initial call follows the call from start to finish"* (Geraghty, 2014). This way, employees are forced to stay with the incident they got assigned to. This could reduce escalation, as employees are less likely to escalate the incident to others because they have to stay with the incident anyways. Also, with total contact ownership, when an incident is escalated anyways, there are now at least two employees working on the incident, which could result in the faster finding of a resolution and a better quality of the resolution (and lowers the chance of another escalation).

IV. Authorization

As mentioned in one of the interviews with a domain architect, teams lack the authorization to make decisions. As a consequence, when decisions need to be made, it takes more time, as one has to first seek authorization for the decision. If employees of IV have to wait to get authorization for making certain decisions, resolve time will increase. Therefore, it is recommended to give employees more authorization regarding making decisions. Otherwise, it should be made clear to employees who to go to for authorization, depending on the nature of the decision.

This lack of authorization was experienced multiple times during the research as well. It, of course, makes sense for an intern to have only a little authorization when it comes to access to tools and use data. However, contacting the right people to access the tools was rather difficult. It, therefore, took some time to get the right access and conduct the research. It was not an issue due to the earliness of the authorization requests. But, it shows the difficulty of getting the needed authorization. It would have been much easier if, for example, there was an overview of who is authorized to give access to what tools and who is authorized to make certain decisions for employees. This way, time can be saved, and thus the resolution time is shortened.

V. ServiceDesk

As the ServiceDesk is outsourced to an outside organization, it is impossible to write recommendations regarding changes within that company. Things such as root-cause analyses, cross-training of employees, and total contact ownership cannot be applied at the ServiceDesk. However, just as with the teams at Menzis, it is possible to use incentives to reduce the number of escalations or improve the FCR at the ServiceDesk.

During the interviews, it was mentioned that the staff of Menzis was located at the ServiceDesk in the past. This resulted in an alignment of the goals between the ServiceDesk and Menzis. The following quote can explain the importance of organizational alignment: "*the concept of alignment lends itself to the creation of high-performance work systems by explaining how the interdependent elements of the organization can achieve greater individual and collective efficiency and effectiveness"* (Semler, 1997). This shows the importance of having organizational alignment. As the ServiceDesk and Menzis are two different companies, there is no complete organizational alignment. Even though the ServiceDesk needs to serve Menzis, they do not have the incentives to minimize escalations and improve first-call resolutions. Therefore, to get better organizational alignment, it is recommended to again propose to have Menzis employees work at the ServiceDesk. This would also mean that the ServiceDesk can now better estimate priorities better.

Another recommendation is to create groups to discuss priorities and decide who will work on what incident. This way, multidisciplinary self-organizing teams are created. These teams can best prioritize incidents (compared to someone at the ServiceDesk) and estimate what team can best work on the incident. These groups would consist of at least one person from every team. These teams are best able to add value and are best able to restore daily operations as fast as possible. Value XI: Self-organizing teams states the importance of self-organizing teams. Thus, looking at agile methodology in combination with prioritizing incidents, it is recommended to create groups that frequently discuss priorities. This way, the most important incidents are handled first, and the daily operations with the most impact will be restored quickly. An optimal way of creating such self-organizing teams is by adopting the SCRUM framework. This is a different team than the team discussed in section *8.1: CMDB*. During the daily sprints of this SCRUM team, incidents will be prioritized, and difficulties will be discussed. A SCRUM team includes a SCRUM master, the SCRUM master can help with supporting the team with self-management, supporting the team in being cross-functional, supporting SCRUM adaptation, and reducing obstacles between teams (Schwaber & Sutherland, 2020). The team is constructed in such a

way that the team consists of people from multiple teams. As the team is multidisciplinary and the SCRUM master helps multidisciplinary/cross-functional teams, the SCRUM master should be included in the new team. Furthermore, SCRUM teams normally include a product owner that states what should be achieved during a sprint. In the case of solving incidents, a product owner could state the goal of prioritizing incidents (restore daily operations as fast as possible), decide on what should be done to achieve the goal, ensure the needs of the stakeholders are fulfilled, and ensure transparency between employees. Sprints often take one month. However, in the case of prioritizing incidents, the duration should be less. A product owner, for example, can't decide on specific backlog items when the sprint is one month long. Too many incidents occur daily for the product owner to accurately describe specific backlog items or plan. Backlog items should state what is needed to improve the resolution process of incidents. With sprints of 1 week, for example, the team can come together every week to discuss and prioritize incidents. The product owner can decide on backlog items for that week, and the SCRUM master can guide the team in self-organizing.

All the mentioned recommendations in this section focus on supporting agile working to shorten the resolution time of incidents. The recommendations are in line with agile methodologies or enhance working in an agile manner. See the conclusion to understand the relationship between the recommendations and agile methodologies.

VI. Conclusion

To start with the recommendation of hiring an incident manager. Hiring an incident manager is not necessarily in line with agile methodologies. However, it is in line with incident management. IV aims to restore daily operations as quickly as possible by adopting an agile methodology. However, to quickly restore daily operations, it is recommended to hire an incident manager. To resolve incidents as quickly as possible, it is recommended to include principles from incident management and agile methodology. The combination of having clear rules and processes together with agile methodology is best for IV. To answer the research question, "*What can be done or altered in the current situation to better support agile working for the IV department of Menzis?"* IV should also consider altercations apart from Agile methodology. IV should consider this altercation, as it does contribute to faster resolution of incidents.

For both FCR and the escalation rate, it is recommended to conduct root-cause analyses of the incidents. It is about understanding the incidents, reflecting on how they were handled, and what can be altered during future handling of incidents. This is called the retrospective phase and is very common within agile methodology. Principle XII: "intervals and reflections" substantiates that. This principle is about adjusting behavior to become more effective by reflecting regularly. So, it is best for IV to frequently reflect on how incidents are handled to become more effective and resolve incidents faster and better. So, one thing IV should incorporate more when it comes to the management of incidents is retrospectives. This is also in line with the SCRUM framework. For example, one of the SCRUM events is the sprint retrospective, aiming to improve the quality and effectiveness of the incident resolution process.

Regarding internal communication, value VI: Face-to-face conversation states the importance of face-to-face communication. Thus, recommending IV to enhance internal communication is in line with agile methodology. So, increasing FCR using agile methodology would increase internal face-to-face communication. This means communication not only through documentation, e-mail, or texts. As better face-to-face communication is in line with agile methodologies, a more agile working place is supported. One of the main pillars of SCRUM methodology states that teams and employees should be transparent. Therefore, by increasing face-to-face communication, teams become more transparent, and agile methodology is better supported.

Cross-training creates teams that can better handle incidents themselves without having to escalate the incident. Cross-training intends to broaden the knowledge of employees to enable them to solve more incidents themselves. Cross-training is a way to create teams that can better manage themselves, which is in line with principle XI: Self-organizing teams. Therefore, cross-training is a way to create a better agile working environment. Furthermore, increasing the authorization of employees and enabling them to make more decisions themselves is also a way of creating self-organizing teams. Increasing authorization, therefore, supports a more agile working environment. Again, a SCRUM master can help multidisciplinary/cross-functional teams and would be a good addition to these teams. However, as Menzis already has SCRUM masters in place at IV, no new SCRUM masters are necessary in this case. The SCRUM masters already in place should help the employees become more cross-functional. The SCRUM master can ensure employees follow certain training programs and become more cross-functional. In previous recommendations, SCRUM masters were also mentioned. As these are related to new self-organizing teams, it can be possible to hire new SCRUM masters. However, as these teams consist of employees from different teams, it is possible to appoint current SCRUM masters.

One of the mentioned recommendations was total contact ownership to reduce the number of escalations. With total contact ownership, it is possible to enhance internal communication. When an incident is escalated, the person receiving the incident first is forced to stay with the incident. When The employee decides to escalate the incident, he works with the other employee and has to communicate where he got stuck. This communication method is better than escalating the incident and leaving a short description within ServiceNow. Furthermore, as two employees are working on the incident, a new (temporary) self-organizing team has been created. Thus, total contact ownership can help create self-organizing teams and increase face-to-face communication. Total contact ownership thus supports an agile working environment for IV.

Lastly, lack of authorization or not knowing who is responsible for authorizations is not the most supportive and best environment. Secondly, giving employees more authorization or creating a clear overview would increase simplicity. Thirdly, having to wait to get authorization for making certain decisions, teams are less able to self-organize. This relates to the following three principles: Principle V: “motivated individuals,” principle X: “simplicity,” and principle XI: “self-organizing teams.” Thus, increasing employees' authorization will help support an agile working environment.

8.3 Tools regarding agile Framework and software development

One of the problems described in this report was that there are too many tools at Menzis, which would hinder agile working. For example, the interview with the SCRUM master provided the following quote: *"Yes, that is why we aim for that shift from TFS to DevOps. Because it indeed can get confusing for me; I have to work with two different tools as my teams work with both tools".* Furthermore, transparency is lost when people depend on multiple teams and tools. This is why a shift from one tool to another is recommended. Having one tool results in transparency and clarity. There will be a better overview for managers about what tasks are finished and still need to be finished. However, this shift from tools is not as easy as it seems, as the tools are used for software development. As mentioned in section 2.3: *Tools and systems*, there is code processed within DevOps related to software development and the architecture.

Therefore, the idea was to recommend IV to ask employees to take part of their time to synchronize the code between the tools. However, after looking at the Kanban board of the Azure Fundament Team (part of IV), one of the tasks is the synchronization between ServiceNow and DevOps. This procedure has been going on for almost a year already. This shows that IV is already working on being more transparent. Microsoft provides clear documentation explaining how this synchronization can be done (which IV also uses). There will be no further recommendations related to the synchronization of tools, as IV is already taking care of this problem.

8.4 Summary chapter eight

To conclude the recommendations part of the report, it is important to take a look at whether or not the research and knowledge questions have been answered. First, the findings of the previous chapters aimed at finding reasons why IV could not be fully agile. Therefore, the results of these chapters were used to formulate the recommendations for this section. Furthermore, the theoretical framework chapter researched and explained agility, TAM, and incident management concepts. These concepts were used to explain and support the derived recommendations. Thus, information and data discussed in the previous chapters were necessary to formulate recommendations.

Secondly, the other knowledge question related to this chapter aimed to ensure that there were no already existing or earlier tested recommendations. During the research and before writing the recommendations, it was found that there were already projects in place for synchronizing code between different tools. Therefore, instead of recommending actions already taken care of, it was shown how IV is already dealing with the problem. Thus, the question "*Has Menzis already tried the chosen recommendation in the past?"* ensures there were no guidelines already used by IV to address the existing problems fully or partially. There are no benefits to recommending actions that are already taken care of.

Thus, to answer the research question "*What can be done or altered in the current situation to better support agile working for the IV department of Menzis?"* the following changes were recommended: conduct more root-cause-analyses/retrospectives, hire an incident manager, create self-organizing teams (in accordance with the SCRUM framework, including SCRUM masters and product owners) for deciding on what the CMDB should be able to store, increase

internal communication, cross-train employees, introduce total contact ownership, increase authorization, and increase organizational alignment between ServiceDesk and Menzis. Furthermore, the central route of persuasion to increase the use of the CMDB is not a change. However, it does result in better use of the CMDB. Therefore, following all these recommendations will result in better support of agile working.

9. Conclusion

The research conducted for this report aimed at formulating recommendations related to IV being able to work as agile as possible. The current state at IV was researched and described to find problems related to being unable to be fully agile. This research consisted of interviews and observations. Finally, as a result of the research, recommendations were written. In section 9.1: *Core problem*, the core problem has been summarized, and in section *9.2*: *Research question*, the research question has been summarized. Finally, in section *9.3*: *Limitations,* the limitations of this study have been stated, and in section *9.4*: *Future research*, suggestions for future research related to the problem of this report have been stated.

9.1 Core problem

The problem found by Menzis can be described as an action problem. Currently, an agile way of working is not fully supported for IV. Therefore, the problem statement is: *"The IV department of Menzis should be fully able to work in an agile way, while they currently cannot, due to several issues mentioned earlier in this report."* A discrepancy can be found where IV is currently unable to fully work in an agile way, where fully working in an agile way is desired. To get to the state of fully being able to work in an agile way, root causes were researched and analyzed. It turned out there were several reasons that resulted in not being fully agile: The CMDB is not updated regularly/correctly, incidents take too much time to be solved, and the different tools for agile frameworks and team collaboration resulted in unclarity.

9.2 Research question

As mentioned earlier, the goal of this report was to formulate recommendations for IV related to IV being able to work as agile as possible. Therefore, the research question stated:

"*What can be done or altered in the current situation to better support agile working for the IV department of Menzis?".*

The research question aimed at finding the "what" to formulate recommendations for the IV department of Menzis.

9.3 Limitations

To describe and evaluate the limitations of this study, it should first be clear what " limitations " means. Limitations of a study can be described as "*potential weaknesses that are usually out of the researcher's control, and are closely associated with the chosen research design, statistical model constraints, or other factors"* (Theofanidis & Fountouki, 2018). This research shows a few limitations that should be considered for future research and additions related to this subject.

Firstly, more information could have been found due to the limited number of interviewed employees. In addition, when having only a few interviews, the researcher cannot always be

sure that the interviewees are being honest. For example, interviewing a hundred people with the same questions and receiving overlapping answers could ensure the reliability of those answers. As this study conducted only a few interviews, reliability could be affected.

This report applies the TAM to evaluate the current situation and formulate recommendations. However, the TAM has its limitations which are to be mentioned here and considered in future research. The TAM shows that before actual system use, there needs to be the intention to use the system. However, *"intention may not be representative enough of actual use because the time period between intention and adoption could be full of uncertainties and other factors that might influence an individual's decision to adopt a technology"* (Chuttur, 2009). Furthermore, there is a belief that other contributions related to perceived ease of use and perceived usefulness are not mentioned here. In addition, the way human behavior is interpreted with TAM might not be completely correct. Finally, a person's intention to use technology can be affected by the fact that people evaluate and reflect on technology. This would change the person's attitude to that technology in a possibly negative way. Thus, the intention to use would again be affected by factors not explained in the TAM. Therefore, explaining employee behavior related to accepting the CMDB at Menzis and TAM is limited.

9.4 Future research

Firstly, as mentioned in the limitations section, only a few employees were interviewed to gather information. Thus for future research, it should be considered to interview/speak with more employees to ensure gathering as much information as possible. Also, increasing the number of interviews would increase the reliability of the gathered information. Therefore, to increase the amount of information gathered and to increase reliability, it is suggested to interview more employees than the number of employees interviewed for this research. However, the four interviewees were experts in their respective fields and could therefore provide reliable answers.

Secondly, the wrong interpretation of observations and interviews can be prevented by using multiple sources of information. To ensure that information given during interviews is not wrongfully interpreted, it is suggested to seek corresponding information that confirms or denies the interpretations. Therefore, finding as many data sources as possible for future research is suggested.

Furthermore, this research can be used in future research to solve problems related to agility. The findings of this research do not necessarily need to be applied to health insurance companies. The findings of this research can be applied to all organizations that seek to optimize an agile working environment. It can be applied to organizations seeking to faster and better resolve incidents. The theory in this research can also be applied to organizations trying to convey to employees to accept technology.

Lastly, little literature was available about agile methodologies in relation to solving incidents. This enhanced the difficulty of adapting agile methodologies to the context of Menzis. Therefore, future research on agile methodologies and incident management should be

conducted. When more research on this topic has been conducted, agile methodologies can be better applied to managing incidents.

9.5 Contribution to existing literature

This research aimed at using existing literature about agile methodologies and the SCRUM framework and applying it to a health insurance company. Many concepts about incorporating self-organizing multidiscipline teams, for example, can already be found in many other existing pieces of literature. However, little literature exists about incorporating agile methodologies for prioritizing incidents. For example, this report explains how self-organizing teams can help with prioritizing incidents to get daily operations back running as quickly as possible. Little literature about this can be found.

Furthermore, this report discusses how technology such as the CMDB can be more accepted. In the context of IV, one way to increase the perceived ease of use is to better communicate and decide on what the CMDB should be able to contain. This is done by incorporating the following principles: Principle XI: "Self-organizing teams" and principle VI: "Face-to-face conversation." Principle XI states that the best architectures and designs come from self-organizing teams. Principle VI states that face-to-face communication is the best way to convey information. With the recommendation to create self-organizing teams consisting of members from multiple different teams, face-to-face communication between teams is enhanced. The new multidisciplinary team will meet frequently and decide on what the CMDB should be able to contain. Thus, agile methodologies are used to increase perceived ease of use and, thus, user acceptance. This could be a good contribution for companies struggling with a similar problem of the CMDB not being able to contain what it should contain.

Furthermore, this report introduces agile methodologies and the SCRUM framework to enhance the first call resolution rate or reduce the escalation rate. This report links retrospectives with root-cause analyses to enhance FCR and reduce the escalation rate. With the SCRUM retrospective, teams reflect on their past sprint and decide what went well and what can be improved. This reasoning can also be applied to the incident resolution process. Employees should frequently reflect on their incident resolution process. Employees need to understand the incidents they handle to improve the resolution process. If employees conduct root-cause analyses and understand the incidents, they can best reflect on what could have gone better. Furthermore, should similar incidents occur in the future, the incidents can be resolved better/faster. This way, agile methodologies and the SCRUM framework can be applied to the incident resolution process. It could be important for other companies seeking to improve their incident resolution process.

Furthermore, little literature exists about creating self-organizing teams using SCRUM masters and cross-training to resolve incidents. Literature about SCRUM masters supporting cross-training does already exist. However, applying this to resolve incidents is new. Thus, this report brings new insights into how agile methodologies and the SCRUM framework do not have to be limited to software development.

References

Agarwal, R., & Prasad, J. (1997). The Role of Innovation Characteristics and Perceived Voluntariness in the Acceptance of Information Technologies. *Decision Sciences*, *28*(3), 557–582. https://doi.org/10.1111/j.1540-5915.1997.tb01322.x

Agilebusiness consortium. (2022). *What is business agility?* agilebusiness.org. Retrieved from https://www.agilebusiness.org/page/WhatisBusinessAgility, on 21-07-2022.

Aguilar-Savén, R. S. (2004). Business process modelling: Review and framework. *International Journal of Production Economics*, *90*(2), 129–149. https://doi.org/10.1016/s0925-5273(03)00102-6

Aliyu, O. A. (2010). The impact of CRM Dimensions on Call center Performance. *Academia*, *2010*(10), 184–194.

Alsaawi, A. (2014). A Critical Review of Qualitative Interviews. *SSRN Electronic Journal*. https://doi.org/10.2139/ssrn.2819536

Atlassian, & Winter, S. (2022). *Agile for when things go wrong: the missing piece of your incident response plan*. Atlassian. Retrieved from https://www.atlassian.com/, on 21-07-2022.

Bhattacherjee, & Sanford. (2006). Influence Processes for Information Technology Acceptance: An Elaboration Likelihood Model. *MIS Quarterly*, *30*(4), 805. https://doi.org/10.2307/25148755

Brenner, M. (2006). *Classifying ITIL Processes - A Taxonomy under Tool Support Aspects*.

Carter, N., Bryant-Lukosius, D., DiCenso, A., Blythe, J., & Neville, A. J. (2014). The Use of Triangulation in Qualitative Research. *Oncology Nursing Forum*, *41*(5), 545–547. https://doi.org/10.1188/14.onf.545-547

Cervone, H. F. (2010). Understanding agile project management methods using Scrum. *OCLC Systems & Services: International digital library perspectives*, *27*(1), 18–22. https://doi.org/10.1108/10650751111106528

Chuttur M.Y. (2009). "Overview of the Technology Acceptance Model: Origins, Developments and Future Directions ," Indiana University, USA . *Sprouts: Working Papers on Information Systems*, 9(37). http://sprouts.aisnet.org/9-37

Cusick, J. J., & Ma, G. (2010). Creating an ITIL inspired Incident Management approach: Roots, response, and results. *IEEE*. https://doi.org/10.1109/NOMSW.2010.5486589

Cooper, D. R., & Schindler, P. S. (2013). *Business Research Methods, 12th Edition* (12th ed.). McGraw-Hill Education.

Dybå, T., & Dingsøyr, T. (2008). Empirical studies of agile software development: A systematic review. *Information and Software Technology*, *50*(9–10), 833–859. https://doi.org/10.1016/j.infsof.2008.01.006

Erich, F. M. A., Amrit, C., & Daneva, M. (2017). A qualitative study of DevOps usage in practice. *Journal of Software: Evolution and Process*, *29*(6). https://doi.org/10.1002/smr.1885

Feuvre, A. L., Peach-Geraghty, T., Muratet, B., Bell, C., & Letourneau, A. (2021). *Incident Manager Outlining the value and responsibilities of an Incident Manager in an organization*. Freshservice. Retrieved from https://freshservice.com/, on 29-06-2022.

Fitzgibbons, L. (2020, 22 april). *ServiceNow*. SearchITOperations. Retrieved from https://www.techtarget.com/searchitoperations/definition/ServiceNow, date retrieved: 06-07-2022,

Fowler, M., & Highsmith, J. (2001). The agile manifesto. *Software development*, *9*(8), 28-35.

Geraghty, S. (2014, 22 January). *11 Ways to Help You Improve First Call Resolution*. Talkdesk. Retrieved from https://www.talkdesk.com/, date retrieved: 29-06-2022.

Heerkens, H., & Van Winden, A. (2016). *Solving Managerial Problems Systematically* (1st edition). Wolters-Noordhoff.

HOPP, W. J., & OYEN, M. P. (2004). Agile workforce evaluation: a framework for cross-training and coordination. *IIE Transactions*, *36*(10), 919–940. https://doi.org/10.1080/07408170490487759

Iden, J., & Eikebrokk, T. R. (2014). Using the ITIL Process Reference Model for Realizing IT Governance: An Empirical Investigation. *Information Systems Management*, *31*(1), 37–58. https://doi.org/10.1080/10580530.2014.854089

Indeed. *Overview of Semi-Structured Interviews (With Pros and Cons)*. Indeed Career Guide. Retrieved from https://ca.indeed.com/career-advice/interviewing/semi-structured-interviews, date retrieved: 23-08-2022.

Jabbari, R., Bin Ali, N., Petersen, K., & Tanveer, B. (2016). What is DevOps? A systematic mapping study on definitions and practices. *ACM Digital Library*, *2016*(12), 1–11. https://doi.org/10.1145/2962695.2962707

Malterud, K. (2012, december). Systematic text condensation: A strategy for qualitative analysis. *Scandinavian Journal of Public Health*, *40*(8), 795–805. https://doi.org/10.1177/1403494812465030

Mlekus, L., Bentler, D., Paruzel, A., Kato-Beiderwieden, A. L., & Maier, G. W. (2020). How to raise technology acceptance: user experience characteristics as technology-inherent determinants. *Gruppe. Interaktion. Organisation. Zeitschrift für Angewandte Organisationspsychologie (GIO)*, *51*(3), 273–283. https://doi.org/10.1007/s11612-020-00529-7

Petty, R. E., & Cacioppo, J. T. (1986). The Elaboration Likelihood Model of Persuasion. *Advances in Experimental Social Psychology*, 123–205. https://doi.org/10.1016/s0065-2601(08)60214-2

Schwaber, K., & Sutherland, J. (2020). *Manifesto for Agile Software Development*.

Schwaber, K., & Sutherland, J. *Scrum Guide | Scrum Guides*. Scrumguides.Org. Retrieved from: https://scrumguides.org/scrum-guide.html, date retrieved: 23-08-2022.

SCRUM methodology, retrieved from https://www.scrum.org/, date retrieved: 30-06-2022

Semler, S. W. (1997). Systematic agreement: A theory of organizational alignment. *Human Resource Development Quarterly*, *8*(1), 23–40. https://doi.org/10.1002/hrdq.3920080105

Snyder, H. (2019). Literature review as a research methodology: An overview and guidelines. *Journal of Business Research*, *104*, 333–339. https://doi.org/10.1016/j.jbusres.2019.07.039

Taherdoost, H. (2016). Sampling Methods in Research Methodology; How to Choose a Sampling Technique for Research. *SSRN Electronic Journal*. https://doi.org/10.2139/ssrn.3205035

Theofanidis, D., & Fountouki, A. (2018). LIMITATIONS AND DELIMITATIONS IN THE RESEARCH PROCESS. *PERIOPERATIVE NURSING*, *7*(3), 155–163. https://doi.org/10.5281/zenodo.2552022

Thurmond, V. A. (2001). The Point of Triangulation. *Journal of Nursing Scholarship*, *33*(3), 253–258. https://doi.org/10.1111/j.1547-5069.2001.00253.x

Fuel Cycle. (2019, 29 august). *A Quick Guide to Semi-Structured Interviews*. Fuel Cycle. Retrieved from: https://fuelcycle.com/blog/a-quick-guide-to-semi-structured-interviews/#:%7E:text=Semi%2Dstructured%20interviews%20are%20among,the%20respondent%20through%20the%20interview, date retrieved: 23-08-2022.

Venkatesh, V. (2000). Determinants of Perceived Ease of Use: Integrating Control, Intrinsic Motivation, and Emotion into the Technology Acceptance Model. *SSRN Electronic Journal*. https://doi.org/10.2139/ssrn.4062395

Yamada, H., Yada, T., & Nomura, H. (2013). Developing network configuration management database system and its application - Data federation for network management. *Telecommunication Systems*, *52*(2), 993–1000. https://doi.org/10.1007/s11235-011-9607-0

10. Appendix

10.1 SPSS

Case Processing Summary

	Cases					
	Valid		Missing		Total	
	N	Percent	N	Percent	N	Percent
Resolve time in seconds	75050	100.0%	0	0.0%	75050	100.0%

Figure 10-1: Case Summary resolved incidents

Descriptives

			Statistic	Std. Error
Resolve time in seconds	Mean		829629.18	12938.805
	95% Confidence Interval for Mean	Lower Bound	804269.19	
		Upper Bound	854989.17	
	5% Trimmed Mean		271300.13	
	Median		2468.00	
	Variance		1.256E+13	
	Std. Deviation		3544618.684	
	Minimum		0	
	Maximum		114388621	
	Range		114388621	
	Interquartile Range		187874	
	Skewness		10.367	.009
	Kurtosis		158.699	.018

Figure 10-2: Descriptives resolved incidents

The mean of 829629.18 seconds confirms the calculations made in the excel file, where a mean of 829629.18 seconds was calculated.

Extreme Values

			Case Number	Value
Resolve time in seconds	Highest	1	2177	114388621
		2	9454	102814411
		3	9944	100749117
		4	2159	96997176
		5	8981	86903542
	Lowest	1	34394	0
		2	34076	0
		3	33299	0
		4	32890	0
		5	31613	0[a]

a. Only a partial list of cases with the value 0 are shown in the table of lower extremes.

Figure 10-3: Extreme values resolved incidents

Figure 10-3 shows the five highest and five lowest cases out of all resolved incidents. Thus, these show the incidents that took the most time to be resolved and those that took the least time to resolve.

Figure 10-4 shows the incidents plotted as outliers case numbers. For example, it can be seen that case number 2177 is the highest outlier of all the plotted outliers. Furthermore, it can be seen that multiple outliers have “0” as a value. For the research, mainly the high outliers are of importance. The goal of using SPSS was to find high outliers and seek explanations as to why these outliers were that high. Namely, taking much time to solve problems contradicts being agile

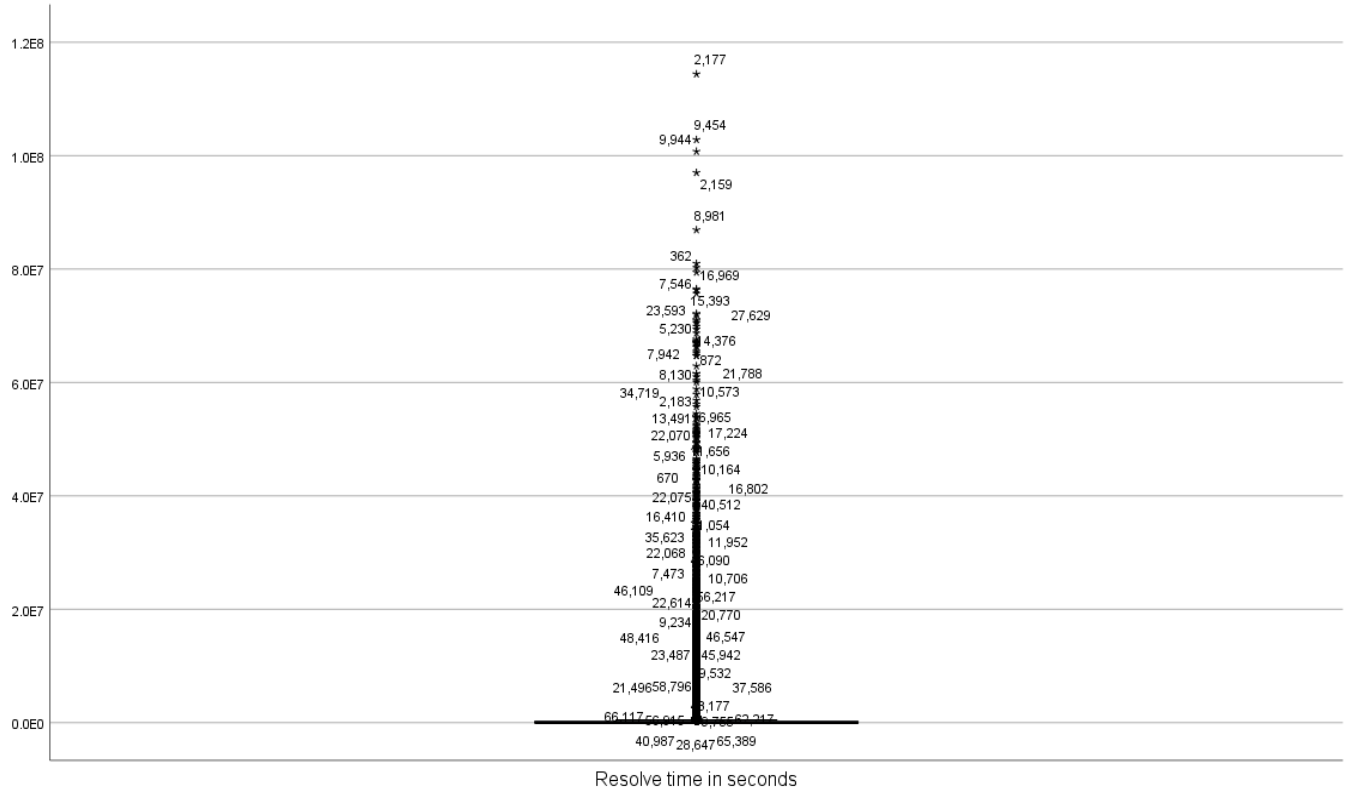

Figure 10-4: Box plot resolved incidents

In figure 10-5, the mean resolve time excluding the 15 biggest outliers can be seen. It shows that the new mean resolve time would result in 226 hours, compared to 230 hours which can be seen in *Figure 4-5*. This means that excluding the 15 highest outliers in the calculation of the mean results in a difference of 4 hours. Thus, 15 cases in a total of 75050 cases have a relatively big impact on the calculated mean. However, as the mean is still very high, it shows that even excluding high outliers, many incidents take a long time to resolve. Every extra outlier that would have been removed resulted in only a smaller difference. Thus, all incidents with high resolution time should be removed to get a small mean. This, however, would result in an inaccurate estimation of the mean.

Mean resolve time in seconds excluding high outliers	814271.265
Mean resolve time in minutes excluding high outliers	**13571.1877**
Mean resolve time in hours excluding high outliers	226.186462

Figure 10-5: Mean resolve time, corrected for outliers

10.2 Overview Recommendations

This section provides a quick overview of the recommendations; these can be seen in table 2.

Subject	**Recommendation**
CMDB	Show employees the relevance of using the CMDB by showing facts and statistics.
	Schedule meetings between teams to agree on what the CMDB should be able to store.
Incident Management	Hire an incident manager.
	Conduct root-cause analyses.
	Improve internal communication through the new incident manager.
	Offer incentives
	Cross-train employees to broaden their knowledge and skills.
	Introduce total-contact ownership of incidents.
	Give employees more authorization or create an overview of who is authorized to make what decisions.
	Propose to have Menzis employees again work at the ServiceDesk to promote organizational alignment.

Table 2: Overview recommendations

10.3 Systematic literature review

This part of the report provides insights into how relevant sources for formulating recommendations were found. Thus, what concepts were most important to answer the research question, "*What can be done or altered in the current situation to better support agile working for the IV department of Menzis?".* This section focuses on the most important concepts of this research, the most used search terms, and databases. Furthermore, this section explains why sources are picked or left out.

Inclusion and exclusion criteria

First, the key concepts of the research question are: *"agility," "agile software development," "incidents," "Incident management," "TAM," "Perceived usefulness,"* and *"Perceived ease of*

use." These key concepts would help understand agility in the IV context and formulate recommendations. These concepts will serve as search terms for finding useful literature. To ensure good literature, synonyms or alternatives of the key concepts are formulated.

Key concepts	***Synonyms/alternatives***
Agility	Agile manifesto, agile principles, agile key values, SCRUM methodology
Agile software development	Business agility
Incidents	Errors, outages
Incident management	Management of incidents, incident manager
TAM	Acceptance of technology, acceptance of CMDB
Perceived usefulness	-

Table 3: Search terms

Databases

The above-mentioned key concepts and synonyms/alternatives will be used in search of the literature to answer the research question. The literature review strategy will involve combining the key concepts of table 3 and using multiple databases. The databases used for finding literature are Scopus, Google Scholar, and books. Table 4 shows the combinations used, what database is used, and what the importance of the shown literature is.

Search terms	***Database***	***Importance***
Agile software development	Google Scholar	Explains what agile software development entails.
Agile manifesto	Google Scholar	The agile manifesto explains the key values and principles of agile software development.
Incident Management	Google Scholar	As IV aims to handle incidents by incorporating agile methodologies, it should first be clear what the management of incidents entails.
(Agility) AND (incident management)	Google	Literature can help with connecting agile methodologies to the solving of incidents

TAM	Google Scholar	Explanation of the Technology Acceptance Model to explain why employees would not use the CMDB.
Increasing Perceived Usefulness	Google Scholar	Aims to explain how the factors influencing the use of the CMDB can be prevented or improved.
(How to increase job relevance) AND (TAM)	Google Scholar	Aims to explain how the factors influencing the use of the CMDB can be prevented or improved.
Incident Manager	Google	Aims to explain the importance of an incident manager.

Table 4: Search log

The first four search entries, as seen in table 4, aim to explain agility in IV and incident management. Furthermore, the TAM framework was explored to find explanations for why employees do not see the added value of a CMDB. As can be seen, the search entry "incident management" has a lot of hits. This is because many articles are about national incidents and disasters instead of little incidents within an organization. However, after turning a few pages, one title stood out as it mentioned ITIL, roots, response, and results. After describing the technology acceptance model and explaining why employees do not want to work with CMDB, I searched for ways to increase acceptance. Therefore, "increasing perceived usefulness" was searched for. However, I found and read articles about the factors influencing perceived usefulness. No concrete explanation of how perceived usefulness can be increased could be found. Therefore, I started using more concrete search terms. Using the search terms "how to increase job relevance and technology acceptance model" resulted in a useful article about influence processes written by Bhattacherjee and Sanford.

As one of the problems mentioned during the interview was the absence of an incident manager, I started using the search term "incident manager." I wanted to better understand and explain the importance of an incident manager. The first article that I read mentioned first-call resolution and escalation rate. As mentioned in section 2.5, first call resolution means the extent to which the incident is solved at one without needing further information. Therefore, I was able to connect FCR with "waiting for supplier – or user information." Furthermore, the escalations rate could be connected to the transferring of incidents as the two are synonyms. Thus, to reduce the time employees have to wait for supplier – or user info, solutions to increase FCR were sought. Furthermore, solutions to decrease the number of transfers were sought to decrease the escalation rate. Solutions to increase FCR and decrease the escalation rate were easy to find and resulted in one source mentioned in table

5 (row 8). This source aimed at finding solutions for increasing FCR. However, it was possible to link some of the mentioned solutions to reducing the escalation rate as well: conducting root-cause analyses, cross-training, and total-contact ownership. Thus, understanding the importance of an incident manager resulted in formulating most of the recommendations related to handling incidents.

Hit	**Paper/book/site**	**Title**	**Importance**
1	Paper	*Empirical studies of agile software development: a systematic review.*	Explains what agile software development entails
2	Paper	*The agile manifesto*.	The agile manifesto explains the key values and principles of agile software development.
3	Site	*Agile for when things go wrong: the missing piece of your incident response plan*.	Literature can help with connecting agile methodologies to the solving of incidents.
4	Paper	*Creating an ITIL inspired Incident Management approach: Roots, response, and results.*	As IV aims to handle incidents by incorporating agile methodologies, it should first be clear what the management of incidents entails.
5	Paper	*Overview of the technology acceptance model: origins, developments, and future directions.*	Explanation of the Technology Acceptance Model to explain why employees would not use the CMDB.
6	Paper	*Influence processes for Information Technology Acceptance: An Elaboration likelihood model.*	Aims to explain how the factors influencing the use of the CMDB can be prevented or improved.
7	Site	*Incident manager: outlining the value and responsibilities of an incident*	Aims to explain the importance of an incident manager

		manager in an organization	
8	Site	*11 ways to help you improve first call resolution.*	Aims to improve the FCR.

Table 5: Found literature

As mentioned above, the literature on Agile methodologies, the TAM framework, and incident management have been used to derive recommendations for IV.

10.4 Interviews

In this section, the tables of the interview questions and answers can be found. Both the questions and answers are quoted from the interviews. Thus, these tables have been used to analyze the interviews and find the core problem's root causes.

Interview SCRUM master

Question	**Answer**
" What goals do you use these tools (TFS, DevOps, and ServiceNow) for?"	*"We use these tools to keep track of our sprints. It can be seen what sprint we are in now and what the product owner needs to finish. Furthermore, a product backlog shows the tasks that need to be finished."*
"What is the reason for having three different tools that have an agile framework?"	*"It just depends on the team. Every team uses another tool. Some teams even use two or three tools at the same time. For example at the moment, we aim for a shift of the teams from TFS to DevOps".*
"Is this use of three different tools not confusing? Is it not preferable to have a uniform tool, everyone can use?"	*"Yes, that is why we aim for that shift from TFS to DevOps. Because it indeed can get confusing for me; I have to work with two different tools as my teams work with both tools. However, there is some underlying code intertwined within TFS, which makes it difficult for teams to shift from TFS to DevOps, this will take some time."*
"Are there other factors you noticed or experienced that have a negative impact on agile working? Related and unrelated to TFS and DevOps."	*"What I experienced is that when someone has dependencies on multiple teams and wants to link product backlog items and have insight into other product backlogs and sprint backlogs, it loses transparency.*

Table 6: Interview SCRUM master

Interview Domain architect

Question	**Answer**
"Why do you think there is incomplete information regarding the CMDB?"	"*There are teams within Menzis that do not fill in needed information or do not know how to fill in this information correctly. Also, employees are not able to fill in the CMDB in the correct way as some options within the CMDB are just missing. The options for describing software/tools etc. are just too few. Therefore, some information in the CMDB is not correct."*
"*Why would employees not correctly update the CMDB?"*	*"Firstly, they do not understand the added value of completely and correctly filling in the CMDB. Secondly, as said before, they are just not able to fully fill in the CMDB due to the lack of options of which can be chosen".*
"How do you then keep an overview of your hardware and software?"	*"As for now, as my teams and I are not able to correctly and completely store everything in the CMDB, we use our own system for storing information regarding hardware and software."*
"Do other teams also have their own way of storing information?"	*"Yes"*
Comments	*"As an example of why it is important to have a good structured and full CMDB: We have a contract with Centric, which offers us hardware and software. However, as we do not correctly use our CMDB, there is not a good overview of the software and hardware offered. For most of these "products," it is important to have those stored in the CMDB for a good complete overview. Now, it costs me a lot of time to make a clear overview. It turns out that a lot of offered products are not even needed, which cost us unnecessary money".*

Table 7: Interview Domain architect

Interview with an expert in ServiceNow

Question	**Answer**
"What is the reason that some incidents are transferred from team to team to be resolved?"	*"First, when an incident occurs, one can call the ServiceDesk or insert a description of the incident in the Service Portal of ServiceNow. The ServiceDesk will receive the incident and sees whether or not they can resolve the incident. If they are not able to resolve the incident, the status of the incident will change, and another team will pick up the incident. However, it is possible they are not able to solve the incident as well, and they can transfer the incident back to the ServiceDesk or yet another team".*
"How exactly has the SLA been determined?"	*"There are standard values of the SLA within ServiceNow. So these standard values have been chosen".*
"Are there other aspects regarding incident management I should take a look at?"	*"Yes, within ServiceNow, it is possible to use filters. For example, you can, at the moment, see how many incidents are still open and are picked up by the ServiceDesk. At the moment, 25 incidents need resolving by the ServiceDesk. As you can see, some are a few weeks old, and their state is 'active.' This means something goes wrong: it should have been solved already by the ServiceDesk, or they should have transferred the incident to another team, etc.*
Comments	*"The ServiceDesk is part of an organization outside of Menzis. So, solving incidents has not been as high a priority as a few years back. A few years back, there were Menzis employees included within the ServiceDesk who felt a higher need to solve problems within the SLA. Furthermore, people of the ServiceDesk are responsible for assigning a 'prio' to the incidents. This is rather difficult for them as they do not work for Menzis".*
	"There is also no incident manager anymore, who ensured the SLA would be achieved. Therefore, at the moment, people less feel the need to solve problems within the SLA".
"Then why is there no incident manager anymore?"	*"When IV had the organizational restructure, the vacancy of Incident manager expired. It was believed that this*

	best aligned with the new 'agile working' methodology".
"And what exactly is the importance for the SLA to be achieved?"	*"The SLA is linked to our business performance. So it is very important to stick to the SLA."*

Table 8: Interview with an expert in ServiceNow

Meeting with another domain architect

Question	**Answer**
"What do you currently experience that would hinder agile working?"	*"Making alterations within the CMDB is not going correctly. For example, when a change takes place, the configuration is not always checked. In the past, during meetings, it was always checked to make sure everything was done correctly."*
Further Comments	*"Employees have too little authorization for making decisions. This costs way too much time. There should be much more authorizations for the teams and employees. It is currently also unclear who can get authorization from who."*
"Do you also currently experience problems that would hinder agile working related to TFS or DevOps?"	*"It is also not possible that when changes are made within TFS, it will be shown within ServiceNow."*

Table 9: Meeting with another domain architect