MASTER'S ORAL EXAMINATION REPORT

Ejawara Alexander took the master's oral examination on
Student's name

10/20/2015
Month, day, year

The student's performance was:

X Satisfactory

___ Unsatisfactory

___ Other, explain

_____________________________________

_____________________________________

_____________________________________

Paper Advisor
Member
Member
Member

November 14, 2013
Best practices for risk management and their compatibility with agile methodologies

A Thesis
Presented to
The Graduate Faculty
University of Wisconsin-Platteville

In Partial Fulfillment
Of the Requirements for the Degree
Master of Science (M.Sc.) in Computer science

by
Eiermann Alexander
2015

1st supervisor: Prof. Dr. Andelfinger
2nd supervisor: Dr. Voß
3rd supervisor: Prof. Ph.D Shi

In cooperation with
Accso
Accelerated Solutions
Statement of authorship

I, Alexander Eiermann, declare that the work presented in this thesis has been performed and interpreted solely by myself, except where explicitly identified to the contrary.

All sources of information, including graphs and data sets, have been created by myself or specifically acknowledged.

I confirm that this work has not been submitted elsewhere in any other form for the fulfillment of any other degree or qualification.

Darmstadt, November 2, 2015

Eiermann Alexander
Acknowledgments

This thesis and its content is the result of hard work and the input of many different people. Therefore I would like to thank everybody who helped and contributed to it.

Thesis advisors
Before all others, I want to thank my thesis advisors Dr. Markus Voß, CEO of Acesso, Prof. Urs Andellfinger from the University of Applied Sciences Darmstadt as well as Ph.D. Yan Shi from the University of Wisconsin–Platteville for their valuable guidance and support while writing this thesis.

Interview participants
Besides my advisors, an ample amount of other people contributed their knowledge and personal experiences during the different best practice interviews. Thus, I would like to thank them for their inputs, feedback, patience and anecdotes.

- Dr. Renato Vinga-Martins
- Tim Bönsch
- Wulf Oberschulte
- Ralf Kachel
- Florian Kraus
- Markus Dechert
- Frank Dieregsweiler
- Dr. Martin Fritz
- Thore Kleischmidt
- Markus Ganß
- Dr. Markus Voß
- Gunnar Winkel
Non-disclosure note

This thesis contains information, Accso considers internal and confidential. Disclosing this thesis or parts of it, as well as creating unauthorized copy’s is forbidden. Exceptions require the written approval of Accso.
Abstract

This thesis aims to identify best practices for risk management. The investigation starts with a broad literature research. Based on the results, an empirical study is conducted which includes interviews with experienced project managers working for Accso, a German software development company.

The purpose of these interviews is to collect best practices for risk management. Afterwards the practices are evaluated to determine whether they are compatible with agile methodologies like Scrum or not. The reason for this is that agile methodologies became more and more popular over the years, but despite this fact, there is little literature on agile risk management. The different practices are then combined into a practice catalog and are integrated into Accso’s internal know-how and technology management system, the BeST framework. This catalog is one of the core deliverables of this thesis. It contains a total of 19 verified practices. From those 19 practices 6 are classified as "highly recommended", 7 are "recommended", 4 may depend on the situation and are therefore "optional" and the remaining 2 practices received the "not recommended" status.

Besides extracting best practices the interviews are also used to identify typical project risks which occurred in the past. A total of 64 risks were collected and combined into a generic risk list which can be used as a starting point for risk management in every new project. Besides the simple collection, the risks are also rated by experienced project managers with regard to their probability of occurrence and their possible impact on the project. The averaged results can therefore be used to derive a basic prioritization of the potential project risks.

After a careful evaluation of the gathered empirical data, the thesis reaches the conclusion that risk management seems to be rather independent of the chosen project management approach, at least within typical Accso projects. It recommends to combine the inherent risk management capabilities of Scrum with existing best practices from "traditional" risk management to make risk management even more effective.
Contents

Abstract .......................... V
List of figures ........................ IX
List of tables ........................ X

1 Introduction ........................ 1
1.1 Motivation .......................... 1
1.2 Research approach ..................... 2
1.3 Chapter overview .................... 3

2 State of the art .................... 4
2.1 Risk management ...................... 4
2.1.1 Risk management and its related project management areas ......... 9
2.2 Methodologies ...................... 10
2.2.1 Traditional vs. agile methodologies ......................... 10
2.2.2 Scrum overview ..................... 14
2.3 Findings from the literature research .............. 18
2.3.1 Findings related to risk management ................. 18
2.3.2 Findings from related areas .................. 20
2.3.3 Preliminary conclusion based on the literature research ....... 21

3 Research approach ................. 22
3.1 Methods of empirical research ............. 23
3.2 Initial survey ......................... 25
3.3 Best practice interviews .................. 29
3.3.1 Interview preparation ................ 29
3.3.2 Conducting the interviews ............... 32
3.4 Aggregation and analysis of interview results ............. 33
3.4.1 Aggregation of interview data ........... 33
3.4.2 Analysis of interview data ............... 35
3.5 Verification of the interview results .......... 37
3.5.1 Verification survey .................. 37
3.5.2 Verification results ................. 39
List of Figures

2.1 The risk management process ........................................... 6
2.2 Recommendation on how to manage the various risks [Mor14] ........ 7
2.3 Typical waterfall model .................................................. 11
2.4 Spiral model [Boe00] ..................................................... 11
2.5 Accso’s agility profile [Vos15] ........................................... 14
2.6 Scrum framework overview [Rub12] ................................... 15
2.7 A Scrum Task Board ....................................................... 17

3.1 Process steps of the applied research approach ........................ 22
3.2 Expert identification survey .............................................. 26
3.3 Analysis of the survey results ............................................. 28
3.4 Selected interview participants .......................................... 29
3.5 Typical project phases, based on v-model xt .......................... 31
3.6 Organization of interview data .......................................... 34
3.7 A best practice candidate ................................................ 36
3.8 Verification survey ......................................................... 38
3.9 Results of the verification survey ....................................... 39

4.1 Practice catalog ............................................................ 42
4.2 The generic risk list ....................................................... 44
4.3 Distribution of risk scores ................................................ 45
4.4 Relationship between the deliverables .................................. 46
4.5 BeST framework: Project management [Vos15] ....................... 48

5.1 Practice 1 ratings .......................................................... 52
5.2 Practice 2 ratings .......................................................... 52
5.3 Practice 3 ratings .......................................................... 53
5.4 Practice 4 ratings .......................................................... 53
5.5 Practice 5 ratings .......................................................... 54
5.6 Practice 6 ratings .......................................................... 54
5.7 Practice 7 ratings .......................................................... 55
5.8 Practice 8 ratings .......................................................... 55
5.9 Practice 9 ratings .......................................................... 56
5.10 Practice 10 ratings ......................................................... 56
5.11 Practice 11 ratings ........................................................ 57

IX
List of Tables

1.1 Process steps of the applied research approach ........................................... 2
3.1 Expert survey's rating scale ........................................................................ 28
5.1 Classification scale for practices ................................................................. 51
5.2 t-test data ................................................................................................ 62
5.3 Hypothesis tests ......................................................................................... 63
Chapter 1

Introduction

1.1 Motivation

Today's software business is highly competitive and most companies are striving for continuous improvement to gain a competitive advantage. This is also true for Aceso, a German software company which is specialized in the development of individual software products. In order to ensure continuous improvement, Aceso initiated a company internal knowledge and technology management program called the BeST framework. This program deals with various areas like analysis, architecture, development, testing, infrastructure and project management. All of them are interesting, but the most important one for this thesis is project management or to be more precise, risk management.

If you ask yourself, why risk management \(^1\), then Tom DeMarco provides a good answer, "Greater risk brings greater reward, especially in software development [DL03, cover]." Since most companies try to increase their profits, they have to take risks as well, because risks are something inherent in the nature of business.

Therefore, this thesis aims to identify best practices for risk management. Since agile methodologies like Scrum are increasingly more common, this thesis will also evaluate risk management from an agile perspective in order to learn about commonalities and differences. Unfortunately, there is not a lot of literature which looks at risk management from an agile point of view. This is especially surprising when considering the fact that agile methodologies are probably one of the most important influences in software engineering in the past years.

\(^1\)A risk is an event which is usually considered to be negative. It has a certain probability of occurrence and a certain impact on the project. Typical negative consequences of risks are additional costs or delays. Risk management is a project management area which deals with the identification and mitigation of potential project risks [IEE11].
1.2 Research approach

As a consequence, additional information is required. To get this information, an empirical study is conducted which includes interviews with experienced project managers working for Accso. The goal of these interviews is to identify best practices for risk management based on their personal experiences. The collected practices will be integrated into the BeST-framework to support and guide new project managers with less experience.

Another goal of this thesis is to collect potential project risks. Tom DeMarco wrote in his quite famous book "Waltzing with Bears" [DL03] that someone should collect the risks of past projects in order to create a good baseline for risk management. This list of potential project risks can be used like a checklist when starting a new project.

Both goals fit together quite well, because both artifacts complement each other. Moreover, both goals can be achieved by using empirical research.

The next two sections provide short summaries of the applied research approach and the structure of this thesis.

1.2 Research approach

In order to identify the previously mentioned best practices as well as the important project risks, a structured approach is necessary. This section provides a short overview over the applied approach. More details can be found in chapter 3.

Overview:

<table>
<thead>
<tr>
<th></th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Initial literature research to get an state of the art overview and to identify possible best practice candidates.</td>
</tr>
<tr>
<td>2</td>
<td>Research on empirical strategies like surveys, questionnaires and interviews to learn about their different types, advantages and disadvantages.</td>
</tr>
<tr>
<td>3</td>
<td>Company internal survey to identify appropriate subject matter experts.</td>
</tr>
<tr>
<td>4</td>
<td>Preparation and execution of expert interviews to draw from their experiences.</td>
</tr>
<tr>
<td>5</td>
<td>Analysis of aggregated expert responses to derive best practice candidates.</td>
</tr>
<tr>
<td>6</td>
<td>Verification of best practice candidates by a second group of experts.</td>
</tr>
<tr>
<td>7</td>
<td>Processing of verification results and integration into the BeST-framework.</td>
</tr>
</tbody>
</table>

Table 1.1: Process steps of the applied research approach
1.3 Chapter overview

This section provides a short description for all the chapters contained in this thesis.

- Chapter 1 introduces the motivation for this thesis, the applied research approach and explains the structure of this thesis.

- Chapter 2 provides an overview over the current state of the art, with regard to risk management, its related project management areas and agile methodologies. Since there are numerous, Scrum was chosen as their most prominent representative.

- Chapter 3 describes the different steps of the research approach in more detail. It also presents some of the resources and results which were created as a part of the process.

- Chapter 4 presents the results and the core deliverables, the relationships between them and how they can be integrated into the BeST framework.

- Chapter 5 evaluates the results of the verification survey. The gathered best practice candidates are examined to define whether they have more agile or more traditional characteristics. Moreover, it investigates the relationship of Scrum and risk management.

- Chapter 6 recapitulates the most important aspects and draws a final conclusion.
Chapter 2

State of the art

This chapter provides a state of the art overview over risk management and its related project management areas, based on the defacto standard "A Guide to the Project Management Body of Knowledge" (PMBOK) from the Project Management Institute (PMI) [IEE11]. In the next section, traditional and agile methodologies are introduced. Both of them are compared to highlight important differences and commonalities as well as resulting consequences.

Subsequently, the most prominent representative of agile methodologies, Scrum, is introduced to explain important terms and principles [Rub12][Lef07]. The chapter ends with the presentation of the different findings from the literature research.

2.1 Risk management

Before talking about risk management or anything else, let's look at the PMBOK's definition of a risk. "[...] [It is] an uncertain event or condition that, if it occurs, has a positive or negative effect on a project’s objectives. [...] [IEE11] p.446. This is a broad definition and it includes many things. To reduce the range and the resulting implications of this definition, this thesis ignores positive risks, often called opportunities.

Even when ignoring the different opportunities, the remaining negative aspects are still too broad. Negative aspects like software bugs, code which is hard to understand, missing test cases, and many more are negative outcomes but they are basically the daily work of a programmer. Therefore, they are usually not covered within risk management and are also out of the scope of this thesis. Besides, people already created many effective measures someone may describe as indirect risk management activities. Some examples
2.1 Risk management

are programming guidelines, intelligent IDE’s, automatic code checking tools, reviews or methodologies like test driven development (TDD)\(^2\).

We already defined the term risk, so let’s look at risk management and why it is necessary. By now, people have accepted that risks are ubiquitous and that it is better to heed them instead of ignoring them.

Since someone cannot ignore them, risk management is exactly what its name indicates. It is the management of potential project risks. The management of these risks is necessary to increase the probability that the project stays within the boundaries of the magic triangle. Meaning that it doesn’t exceed its scope, time or costs limitations.

In addition to those three, it may be a good idea to add quality as a fourth attribute, causing the triangle to transform into a magic square [Mo10]. Quality should be considered, because the reduction of quality is a tempting way to stay within the scope, time, cost boundaries, especially in software development.

Within the context of this thesis, risk management always refers to the management of risks within software development projects.

When reading books about risk management, someone will find many different methods to perform it. At the beginning of this thesis and during the first best practice interviews, all discovered methods were briefly evaluated, but it quickly became clear that it does not make sense to use most of the heavy weight methods in typical Access projects. The project sizes are simply too small. These heavy weight methods like statistical analysis are more useful when someone operates on the program \(^3\) level or when the project reaches a certain size. As a consequence, they are out of scope as well.

In order to return to the actual scope, a look at the PMBOK’s definition of a project helps to further describe risk management and its purpose.

\[ A \text{ project is a temporary endeavor undertaken to create a unique product, service or result. } [\text{IEE11]} \text{ p.5} \]

Based on this definition each project is unique and therefore has its own unique risks which are relative to the project’s specific context.

Of course, there are some typical project risks like: schedule delays, problems with 3rd party deliverables, accidents and many more, which occur quite often. Even so, the way to manage

---

\(^2\)Test driven development advocates that programmers should create test cases first and then write the code to make the test cases work.

\(^3\)A program usually consists of multiple projects, which support a certain strategic goal. It is basically the next higher level of abstraction. [IEE11]
2.1 Risk management

these risks may be different each time. This is one of the challenges associated with individual software development. Each customer operates within a specific context, has different goals, a different IT-landscape and different employees. These factors and many others create a very customer specific mixture someone has to understand first. It is therefore no surprise, that many interview participants stated that some creativity is necessary to be good at risk management.

Risk management itself can be separated into multiple phases which are repeated over and over again. The phases in this thesis are based on the definitions of Matthias Knoll [KB14] (see figure 2.1).

![Diagram of the risk management process]

Figure 2.1: The risk management process

**Definition of context:**
This phase deals with the preparation of the various risk management activities. It defines the projects context, risk policies and thresholds which specify which risks are acceptable, when someone has to mitigate them or other procedures which are triggered when certain events occur. This phase holds greater importance for bigger projects or on program level.

**Identification:**
This phase aims to identify potential risks which can influence the projects objectives. The identification of risks is actually the most important part of risk management, because you
can only manage risks when you know about them [Car13]. If a risk manifests unexpectedly, it is way harder to handle because someone has to act immediately. Moreover, possible counter/mitigation strategies may not be applicable anymore because the risk already occurred.

Analysis:
This phase focuses on the analysis of the identified project risks. It estimates the risks probability to occur as well as its possible consequences and implications for the project.

Evaluation:
This phase deals with the evaluation and the assessment of the previously analyzed risks. Leading to a prioritization of risks based on their risk exposure. The risk exposure is a result of a risk’s likelihood times its impact.

Management:
This phase deals with the actual management of risks. This includes the documentation and management of known project risks as well as the initiation and the tracking of countermeasures. Usually one of the first objectives after collecting, analyzing and evaluating the risks is to get a good overview. A common way to do this is by creating a risk map, because it results a very clear and abstract visualization of the collected risks (see figure 2.2).

![Figure 2.2: Recommendation on how to manage the various risks [Mor14]](image)

Based on their position within the coordinate system, the risk map defines a couple of generic risk management strategies, recommended by Allan Moran [Mor14]. These can be used to derive possible counter-measures or mitigation strategies.
2.1 Risk management

- **Accept:** Since these risks are less likely to occur and have a rather small impact on the project, someone should simply accept them. As a consequence any risk management efforts should be focused on the risks in the yellow or red zone. A typical example could be the risk that one of your non critical team members gets sick and is therefore unable to come to work for a couple of days. This might be unpleasant, but since it is not a critical team member it is not worth the effort.

- **Reduce:** Typically, these risks are either likely to happen and/or have a rather big impact on the project. Thus, counter-measures should be initiated to reduce their likeliness or their potential impact. A typical example could be that the project team has no experience with a certain technology. To mitigate this risk, the risk manager could hire and external expert to support the project team.

- **Share:** This category of risks is special, because these risks have a rather low likeliness but a very high impact on the project. To deal with them, someone should share them with other stakeholders or 3rd parties. Good examples are insurance policies or service level agreements \(^4\) (SLA).

- **Avoid:** Whenever possible, the risk manager should try to avoid any risks within this area, because they have a rather high likeliness and impact. Possible counter measures could be the choice of an alternative solution/approach or the removal of certain aspects from the project scope.

In addition to the 5 main phases which were explained earlier, Knoll [KB14] also defines 2 additional phases which are performed in parallel to the other phases. These are:

- **Controlling:** This phase is responsible for monitoring the other phases. Moreover, it ensures that the required steps are performed and that people adhere to the defined risk policy.

- **Reporting:** Just like the name indicates, this phase is in charge of the reporting procedures. It makes sure that everybody receives the relevant information e.g. the higher management who wants to know the project’s current risk status.

The 7 phases presented above and their various interactions are the foundation of project risk management. Other areas which are important from a risk management perspective and for this thesis are described in the next section.

\(^4\)A service level agreement is a contract which defines certain aspects of a service, e.g. the availability or the performance of a certain system. Usually it also defines consequences should these agreements be violated [Pal15].

8
2.1 Risk management

2.1.1 Risk management and its related project management areas

Risk management as described above, consists of rather basic concepts and ideas. To increase its effectiveness, it makes heavy use of the information collected in related project management areas [IEE11]. Since many of these areas also contain practices which mitigate certain risks, it would be interesting to identify those practices which provide the biggest contribution to risk management.

Since there is only a limited amount of time for this thesis, only some project management areas received a more detailed evaluation. Originally, only two areas were selected. These were scope- and stakeholder management. They were selected based on a subjective basis after some discussions with one of my advisors. The areas of communication and monitoring & control emerged during the interviews.

Scope management

A large part of the potential project risks are associated with the project’s scope. Some examples are the complexity associated with certain features or the development of features which turn out to be unnecessary.

The word scope is basically a synonym for the products functionally. Scope management invests a lot of time to understand, plan, define, manage and control the scope. While doing this, it collects large amounts of information which can help to identify possible project risks, e.g. those which are associated with certain features or their underlying technologies.

Stakeholder management

The second selected area is stakeholder management, because stakeholders and their individual expectations and motivations can have a huge influence on the project and its outcomes. Some risks are results of conflicting stakeholder expectations or goals while others are the results of un-addressed fears or antipathies. Unfortunately, their management requires a lot of empathy and experience. This information is usually difficult to collect, but they are valuable from a risk management perspective.

Communication

The communication within the project team or with the customer is always crucial. In internal Acesso workshop actually identified communication as the most important factor for a project’s success [Vos13]. Thus common communication related methods are evaluated as well.
Monitoring & Control

The project management area monitoring & control, collects all kinds of project related information. Since it is likely that some of them are also interesting for risk management this area receives closer investigation.

The previous sections covered risk management, important terms and concepts as well as some related project management areas. The next section will shed some light on agile and traditional methodologies.

2.2 Methodologies

2.2.1 Traditional vs. agile methodologies

The topic of this thesis is "Best practices for risk management and their compatibility with agile methodologies". To define the meaning of agile, another term is introduced. This term is "traditional". This section will contrast both terms in order to define agile and to answer the question "What are the differences between "traditional" and "agile" approaches with regard to risk management?". This is not a simple question to answer. In this thesis, the term "traditional" refers to the rather sequential and plan driven methodologies which were common before agile methodologies emerged. They are usually associated with extensive planning and detailed descriptions resulting in very structured and clearly defined workflows and processes [Rub12][Lef07].

The term agile on the other hand refers to lightweight processes with only a minimum set of structure and restrictions which intend to guide someone’s workflow rather than defining it. The idea is to allow room for personal creativity. Also upfront planning is very limited.

Some of the most common traditional approaches are the waterfall and the spiral model. In agile literature, the "waterfall model" a very well known software development methodology, is usually presented in its "purest" form and in its most sequential version (see figure 2.3). This is done, because it allows authors to differentiate it more easily, from agile approaches. In this sequential approach a lot of work and time is spent to develop an initial plan for the whole project and this plan is then executed in the subsequent phases. In traditional projects, the work is usually planned by a group of experts. Afterwards it is broken down into work packages which are assigned to the different employees. This is similar to master / worker processes, were a few have the main responsibility and many others are just following
their orders. A typical sequence for a waterfall based approach is: Analysis => Design => Implementation => Test => Deployment. There is theoretically no problem with this representation, because the waterfall model is a rather sequential and plan oriented approach. But it should be remembered that it is possible to use it in a more iterative way as well. Moreover, it can be reasonable to proceed in this sequential manner. Especially, when someone knows exactly what he wants and knows how to realize this idea. Or in other words, when the project and its context are very stable and predictable. Under these circumstances the waterfall approach can be superior to agile approaches, but unfortunately not all software projects are that straightforward. As a consequence many of them failed (see Standish Group: chaos report [Wiki15a]).

The second example is the Spiral model, invented by Barry Boehm [Boc86]. Even though it is rather sequential with regard to the project's life cycle, it also makes use of iterations. The iterations within the spiral model make sure that the different factors like goals, requirements, risks, development and planning are reviewed and updated constantly. Moreover, it is an approach which explicitly addresses risks and integrates them into the process (see figure 2.4).

![Figure 2.3: Typical waterfall model](image1)

![Figure 2.4: Spiral model [Boc00](image2)]

Someone could even say it is risk driven. This is especially true for the newest version of the spiral model, where the evaluation and resolution of risks is mandatory before proceeding to the next iteration [BLKT14].

Now that some of the traditional approaches have been introduced, it is time to talk about agile as well. Before introducing Scrum, the probably most common agile approach, lets
look at the historic background first. This will help to better grasp the agile core ideas and concepts.

As mentioned earlier, a lot of the software development projects which used traditional approaches failed and the resulting dissatisfaction lead to the creation of the agile manifesto. Based on the learnings and experiences collected in the past, the authors of the agile manifesto defined a small set of values and principles to guide future software development approaches.

Probably the most famous part of the agile manifesto which sums up the core ideas is shown below [[Lef07] p.9]:

- Individuals and interactions over processes and tools
- Working software over comprehensive documentation
- Customer collaboration over contract negotiation
- Responding to change over following a plan

They also noted: “That is, while there is value in the items on the right, we value the items on the left more.” [[Lef07] p.9].

Most of today’s agile methodologies like Scrum and others are based on these principles, even though some interpret them more strictly than others. As a result, there are some differences between the existing approaches. The next paragraphs describe the most important differences between traditional approaches and Scrum. A more detailed analysis can be found in chapter 5. The various information are based on [Rub12][Lef07][MS14].

The most important difference between both approaches is project organization. Traditional approaches are usually very sequential and have a long cycle time. Agile ones on the other hand have multiple nested loops which have shorter cycle times. The most important one is the iteration. As a consequence, all the steps like planning, development, testing, etc. are repeated over and over again during each iteration. This increases the feedback speed, due to problems being detected and reported earlier.

Another key insight from the past is that it is very difficult to plan projects 100% perfectly in advance, because expectations and requirements tend to change during the course of the

---

5The cycle time is basically the time from the beginning to the end of a process.
2.2 Methodologies

project. Thus, with growing size and project complexity, the gap between plans and reality increases as well. This is one of the reasons why agile emphasizes the importance of iterative and explorative methods to adapt to unforeseen changes. This also includes the reduction of extensive upfront planning. This doesn’t mean that agile skips planning, the distribution of planning tasks is just a bit different. Agile favors a very coarse grained initial plan which is refined overtime and especially at the beginning of a new iteration.

Another big problem agile addresses is the delivery of product increments. Back then, when many projects followed the traditional waterfall model, it was common to deliver everything in one big batch. When the customer did not like the final product, it was too late to incorporate any changes. The incremental delivery ensures that small parts of the final product are delivered frequently. This enables the customer to test and evaluate the new features and to provide feedback to guide future development.

Finally, the major difference introduced by agile methodologies is the value system, which advocates transparency and more responsibility for individuals. These values are expected to lead to a more open and trusting atmosphere which also improves internal communication. Thus, someone could say that traditional methods are rather top-down, while agile ones are bottom-up.

To sum everything up, this section described some common traditional approaches, the core idea of agile approaches and some of the important differences. Even though both methodologies are different, they have the same goal, a successful project. They just use different ways to achieve it. Unfortunately, not all approaches are suited for all projects, because of their inherent structure and organization.

The words of Jim Highsmith perfectly describe the resulting conflict: "Too much structure stifles creativity. Too little structure breeds inefficiency." [Hig04] p.17. As a result, someone has to find the right balance between both, structure and freedom for creativity.

These thoughts can be summarized with a single word: adequacy. In order to manage a project as best as possible, someone has to find the correct balance between agile and traditional characteristics. This idea is reflected in Acceso’s "agility profile" which is depicted in figure 2.5. The left side represents the most extreme agile ideas while the right side represents the most extreme traditional ones.
2.2 Methodologies

![Image of figure 2.5: Accso’s agility profile [Vos15]](image)

Based on the configuration of the different sliders, the project manager can select an appropriate management strategy. It can be an agile one, a traditional one or what is more likely in typical Accso projects, something in-between. From now on, the term gray area will be used to refer to this in-between state.

While this section defined the terms "agile" and "traditional" and highlighted important differences. The next section will introduce Scrum to familiarize the reader with the necessary terms and concepts.

### 2.2.2 Scrum overview

Scrum is nowadays the most common agile methodology and was mainly developed by Jeff Sutherland and Ken Schwaber [SS13]. It is a lightweight framework which helps someone with the organization and the management of software development projects. Even though it is possible to apply it to other areas.

Dean Leffingwell summarizes Scrum with the following words: "*Scrum is a lightweight agile project management method based on small, empowered, self-organizing teams; complete visibility; and rapid adaption.*" [[Lef07] p.41]

Since there is already a lot of literature on Scrum, this thesis will only provide a rather short introduction about it. Its main intention is to familiarize the reader with its basic structure and to introduce the various terms which will be needed later. Additional information about Scrum can be found in the book "*Essential Scrum - A practical guide to the most popular agile process [Rub12]*". If someone is looking for a good overview over the topic of agility in general, the book "*Scaling software agility - best practices for large enterprises [Lef07]*" is highly recommended.
2.2 Methodologies

Figure 2.6 which is shown below, depicts the basic structure of the Scrum framework and indicates how the various parts are connected with each other. The things depicted in the image can be grouped into three categories [Rub12].

Figure 2.6: Scrum framework overview [Rub12]

- **Roles:** Product Owner, Scrum Master, Development Team
  
  - The **Product Owner** represents the customer, he decides what will be developed and in what order.
  
  - The **Scrum Master** is responsible for setting up the Scrum processes and to guide and assist the team with regard to the Scrum practices. The **Scrum Master** is also responsible for impediments removal.
  
  - The **Development Team** is in charge of writing the source code in order to implement the software requested by the **Product Owner**.

- **Artifacts:** Product Backlog, Sprint Backlog, Potentially Shippable Product Increment
  
  - The **Product Backlog** is a prioritized list of tasks which need to be completed.
  
  - The **Sprint Backlog** is a refined subset of the **Product Backlog** and contains the workload for the next **Sprint**.
  
  - The **Potentially Shippable Product Increment** is a part of the final product, which is created during the **Sprint Execution**.
• **Activities:** Sprint (the main circle), Product Backlog Grooming, Sprint Planning, Daily Scrum, Sprint Execution, Sprint Review, Sprint Retrospective

  - The **Sprint** is the Scrum word for short iteration. All the other activities are performed within the Sprint.

  - The **Product Backlog Grooming** is performed in parallel to the other activities. The **Product Owner** splits the big project into smaller parts e.g. tasks or user stories\(^6\) and prioritizes them. The outcome is a prioritized list of tasks which is stored in the **Product Backlog**. This list is refined on a regular basis.

  - Before the actual **Sprint** can start, the **Sprint Planning** is performed to define the work packages for the next Sprint. The workload is usually a subset of the tasks stored in the **Product Backlog**. The selected tasks are refined with regard to design, integration, testing and other required information.

  - The **Sprint Backlog** contains the selected and refined tasks.

  - During the **Sprint Execution**, the tasks stored in the **Sprint Backlog** are implemented by the development team. The tasks the team is working on, are often visualized using a **Task Board** (see 2.2.2).

  - The **Daily Scrum** is a daily and very short stand-up meeting where every participant tells the others what they have completed the previous day, what they are doing today and whether or not there any problems. The main purpose is to keep everyone up to date.

  - At the end of the **Sprint Execution**, the development team has created a **Potentially Shippable Product Increment** which represents a subset of the overall product.

  - To ensure that the created product increment fits its purpose, a **Sprint Review** is performed. During this review, the stakeholders and the team members inspect the product and provide feedback. Potential adjustments are feed back into the **Product Backlog** to be tackled in future Sprints.

  - The last activity is the **Sprint Retrospective**. This is a continuous improvement process for Scrum itself. The team members analyze the overall Sprint, identify weak or strong points and suggest process improvements. (Note: When this activity has been completed, the cycle starts over again.)

\(^6\)A user story is similar to a use case. It describes and activity a user can perform when using the finished product.
2.2 Methodologies

Scrum core values

*Scrum* also defines a set of core values similar to those defined in the agile manifesto. These core values are: *Focus, Courage, Openness, Commitment* and *Respect* [Scr14]. The *Scrum* framework is intentionally kept simple to allow room for creativity. Since not everything is specified, these core values can be viewed as rules to guide the team’s mindset and behavior.

The Task Board

Another thing which is usually associated with *Scrum* is the Task Board. It is *Scrum’s* way to make the current project or *Sprint* status visible to everyone. A typical Task Board is depicted in figure 2.7.

<table>
<thead>
<tr>
<th>User story</th>
<th>To Do</th>
<th>In Progress</th>
<th>Testing</th>
<th>Done</th>
</tr>
</thead>
<tbody>
<tr>
<td>Story A</td>
<td>Task 3</td>
<td>Task 2</td>
<td></td>
<td>Task 1</td>
</tr>
<tr>
<td></td>
<td>Task 4</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Story B</td>
<td></td>
<td></td>
<td>Task 8</td>
<td>Task 5</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Task 7</td>
<td>Task 6</td>
</tr>
<tr>
<td>Story C</td>
<td>Task 10</td>
<td>Task 9</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>...</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

*Figure 2.7: A Scrum Task Board*

This *Task Board*, which is sometimes also called an “information radiator [PP03]” is especially effective, because *Scrum* favors co-location. This means that all team members share the same office space and work rather close to each other. The *Task Board* is usually in the room where the *Daily Scrum* is held, as a consequence most team members look at it quite often. Thus, the *Task Board* and the *Daily Scrum* ensure that everyone knows the current project status and the remaining work. Another great aspect is that the team members can usually decide for themselves on which task they would like to work next.
2.3 Findings from the literature research

After introducing traditional and agile approaches as well as the most important terms and foundations, the next section will summarize the findings from literature research.

2.3 Findings from the literature research

This section completes the state of the art overview by providing a short summary of the findings from the literature research. The presented findings are mainly techniques and artifacts which can be used in risk management. They are taken from standards like the PMBOK, the “Business Analysis Body of Knowledge” (BABOK), from books on project- or risk management or even from the web (see cited sources).

The first section contains all findings which are directly related to risk management, while the second section contains findings from other project management areas which may be useful from the risk management perspective. These related areas were already introduced in section 2.1.1.

Since not everyone is familiar with all these terms and techniques, short descriptions are provided where necessary. Additional information can be found in the cited sources.

2.3.1 Findings related to risk management

Techniques

- Simulation with Monte Carlo: An algorithm with can be feed with probability distributions to simulate the combination of multiple risks and their possible outcomes [DL03].
- Early project cancellation: This practice advocates the early cancellation of a project based on certain indicators. Its main purpose is to avoid wasting more money than necessary [DL03].
- Test the emergency: This is about the tentative escalation of project risks, to see how the other party handles the escalation process [DL03].
- Create the right atmosphere: This practice emphasizes the importance of an open and risk aware atmosphere. In this atmosphere, people can talk freely about potential risks without having to fear the “shoot the messenger” scenario” [DL03].
- Frequent customer feedback: This refers to the importance of customer feedback in order to develop the right product [Lef07].
2.3 Findings from the literature research

- Decide as late as possible: This technique recommends to delay decisions as long as possible to increase their quality. It argues that the longer the delay, the more information is available, thus enabling people to make better decisions [PP03].
- Fault- / Event tree analysis: A technique used to identify as much causes and consequences as possible for a given risk [KB14].
- Failure mode and effects analysis (FMEA): This technique uses logical reasoning to analyze and to evaluate the possible consequences of a given risk [KB14].
- Three point estimation: A estimation technique which combines best-, probable- and worst case estimations to create a more reliable result [DL03].
- Human Reliability Analysis (IIRA): A technique which focuses on people instead of objects. It analyzes the risks a human may cause e.g. while performing a certain process [KB14].
- Hazard and operability study (HAZOP): A technique which is based on the comparison of desired and actual states to determine possible risks and problems [KB14].
- Post-mortem analysis: A method which analyzes completed projects to identify the things which went especially good or bad. The gained insights are used for continuous improvement [DL03].
- Scoring methods: A very subjective method where someone simply assigns scores to things in order to make them comparable [KB14].
- Sensitivity analysis: This is basically a cost/benefit analysis where the stakeholders have to estimate the expected utility value for each component [DL03].
- Strengths, Weaknesses, Opportunities and Threats (SWOT) analysis: Used to identify strengths, weaknesses, opportunities and threats [KB14].
- Incident sequence analysis: A method used to analyze cause effect relationships [KB14].
- Stress test: A method which examines very unlikely scenarios to increase sensitization for potential risks [KB14].
- Scenario planning: A technique which uses scenarios to identify causes for potential risks or to evaluate known risks in more detail [KB14].
- Root cause analysis: Used to analyze cause effect relationships [KB14].
- Business impact analysis: This method is used to determine the business impact of potential risks [KB14].

Artifacts

- Risk list / risk inventory: This artifact contains the known project risks and is used to manage them [IEE11].
2.3 Findings from the literature research

- Decision- / risk diagrams: Diagrams which are used to analyze possible decisions in order to identify the most reasonable one [ÓCi12].
- Risk map: This artifact is used to display risks on a very abstract and graphical level [KB14].
- Enhance the Task board: This is about adding additional information to the Task board [Mor14].
- Integrate risks as tasks into the Product Backlog [Mor14].

Misc

- Risk attitudes: Describe someone’s tendency towards risks from a national and personal perspective [Mor14].
- Key risk indicators: Use proactive indicators/metrics to learn about risks before they occur [KB14].

2.3.2 Findings from related areas

Techniques

- MoSCoW: A technique for prioritization which uses 4 categories (Must, Should, Could, Won’t) [OW08].
- ABC categorization: A way to prioritize features by using 3 categories [OW08].
- Set based planning: A planning approach based on constraints [OW08].
- Make parts or features of the project scope optional [DL03].
- Product backlog grooming: An agile activity which is concerned with frequent refining and the re-prioritization of scope elements [Lcf07].
- Stakeholder analysis: A method to identify possible stakeholders and their motives [IEE11].
- Ask 5x Why: A technique to make people reflect on their answers in order to identify underlying root causes [OW08].

Artifacts

- Work Breakdown Structure (WBS): A way to structure a project’s work packages and to visualize the dependencies between them [DL03].
- Product vision box: A technique to visualize the future product by creating its packaging [Hig04].
2.3 Findings from the literature research

- Project data sheet: Similar to the product vision box, but this time the product is just described while limiting oneself to 1 or 2 pages [Hig04].
- Domain models: A way to visualize a specific domain, the most important elements and their relationships [OW08].
- Common glossary: A collection of the most important terms and definitions to avoid misunderstandings [OW08].
- Burn down/up chart: A rather agile method to visualize the projects progress [Lef07].
- Communication plan: A practice which deals with the definition and organization of stakeholder communication [IEE11].
- Personas: A method which helps to put oneself into somebody else’s position to better understand this person. [Vos15].

2.3.3 Preliminary conclusion based on the literature research

The initial literature research revealed that there is not a lot of literature on agile risk management. In most cases, the literature does not differentiate between traditional and agile approaches. This leads to the question, whether it is not only possible but reasonable to use traditional risk management practices in agile methodologies as well. The next chapters will answer this question by evaluating the interview feedback and the best practice candidates.
Chapter 3

Research approach

This chapter describes the applied research approach in more detail. The different steps are presented in sequential order. Each section describes a step, how it was performed and why certain decisions were made. Figure 3.1 illustrates the different steps. The findings of the literature research were already presented in section 2.3 in the state of the art chapter. As a consequence this chapter starts with step 2 and an introduction to empirical research and possible methods.

Figure 3.1: Process steps of the applied research approach
3.1 Methods of empirical research

The third step describes the survey which was used to identify appropriate subject matter experts (see section 3.2). The preparation and execution of the expert interviews is described in section 3.3. Section 3.4 deals with the analysis and aggregation of the various results from which the best practice candidates were derived. The process to verify the derived best practices is explained in section 3.5. The resulting practice catalog and the generic risk list are presented in chapter 4. This chapter concludes with learnings on the applied approach.

3.1 Methods of empirical research

This thesis started with an extensive literature research on risk management (see section 2.3). This research yielded many information on risk management and its related areas, but after a sufficient amount of knowledge was gathered, a new question came up. How to draw from the know-how and experiences of seasoned Accso employees in order to identify best practices in risk management as well as potential project risks?

As a consequence, empirical research came to mind. It is a type of research which uses observation or experimentation, either directly or indirectly, to gain new knowledge [Wik15b]. It uses either quantitative or qualitative methods to analyze the data. Quantitative research is focused on numerical data which is analyzed using statistical, mathematical or computational methods. Qualitative research on the other hand is based on observation and is more focused on people and their opinions.

A currently popular method of empirical research is grounded theory which is a qualitative approach [WNA15]. It is usually inductive, meaning that it tries to comes up with new theories based on the analyzed research data. As a consequence, someone does not really know the outcome of this method, because the theories are supposed to emerge over time. The opposite would be deductive research, which tries to prove or to disprove a certain hypothesis.

Just like in grounded theory the outcome of this thesis was unclear in the beginning. It was not known how many or which kinds of best practices would emerge, but it became clearer and clearer while talking to the different experts. Since grounded theory is more concerned with the analysis of data, which is not collected yet, the method will be elaborated later. Section 3.4.2 describes it and explains how it has been used in this thesis.

In order to collect the necessary data other methods are required. A little research using sources like [IC109] indicated that the most reasonable choices to extract knowledge from people are questionnaires and interviews. Even though questionnaires are considered quan-
3.1 Methods of empirical research

titative research. Before deciding which method to use, it is only logical to evaluate them, their purposes and their associated strengths and weaknesses.

Questionnaires:

Advantages:

- It is cheap to collect data from huge groups.
- Results can be quantified more easily.
- Results can be compared more easily because they were collected in a standardized manner.
- Participants can complete them whenever they have time.

Disadvantages:

- Someone does not know how much thought was put into the answers.
- Someone might misinterpret the question.
- The amount of questions is fixed.
- People need a lot of time to answer the questionnaire, especially when it contains rather open questions where people have to elaborate their thoughts.
- Open questions can result in a lot of data.
- It is harder to receive valuable feedback or to identify improvement opportunities.

Interviews:

Advantages:

- More flexible than questionnaires! Someone can specifically ask for certain aspects or more details.
- Someone can better judge the quality of the answer.
- More content can be covered in less time.
- Misunderstandings can be clarified.

Disadvantages:

- Interview process and results are likely to vary, because they depend on the received answers.
- Requires a certain skill and a good preparation.
- Evaluation requires a lot of time and is rather subjective.
After carefully considering the different pros and cons, the decision was made to use both methods. Questionnaires are used to identify subject matter experts because it is much easier to distribute them in order to collect the required information from all Acces employees. When the experts are known, interviews are used to draw from their experiences because they will likely yield better results. Moreover, only a small number of experts is selected, thus the additional workload caused by the interviews is acceptable. Since interviews require participants first, the next section describes how these participants were selected.

3.2 Initial survey

The purpose of this phase was to identify suited interview participants. Suited means they needed to have the required knowledge and were willing to participate. Since this was a companywide survey, it was necessary to find the right balance between the amount of gained information and the time required to complete the survey. An extensive survey may have yielded more information, but it also would have cost the company more money. Besides, the planned interviews should be enough to satisfy all information needs. Moreover, the more time the completion of the survey would require, the less people would probably participate in it.

To allow for an easy creation and distribution of the survey, the cloud based service Google Forms was used. It offers predefined and easy to use drag and drop elements like text fields, sliders, check- and radio boxes, matrices and others to create the survey questions. Data privacy concerns were the only drawback of the selected Google service. As a consequence, people were asked to enter a pseudonym instead of their real name. This was also done to adhere to the German laws on data protection.

To save time, most of the survey questions were rather short and closed questions. Most of the time, people had to either enter a number, select options from a provided list or rate their personal level of know-how with regard to certain methods or project management areas. Figure 3.2 shows the final survey and its structure. The complete survey can be found in the appendix A.1.

The survey questions are grouped into 5 major areas: general, methodologies, risk-, scope- and stakeholder management.
Figure 3.2: Expert identification survey

The general questions in the beginning are designed to collect information about the participants, e.g.

- Their years of experience in software development.
- Their years of traditional and agile project experience.
- Their ratio of traditional and agile projects in the past 5 years.
- The different project roles they have experience with.

The second section about methodologies is intended to learn about the preferences and familiarities of the participants. The survey asks about the most common methodologies, e.g. PMBOK, PRINCE2, XP, Scrum, Lean, Kanban, Dynamic Systems Development Method (DSDM), Rational Unified Process (RUP), Scaled Agile Framework (SAFe) and Disciplined Agile Delivery (DAD).

The third section is focused on risk management. Participants were asked to rate their knowledge with regard to:

- Risk management in general
- Risk identification
- Risk analysis
- Risk assessment
- Risk reduction/mitigation

They were also asked, whether they are familiar with the terms of technical debt and cost of delay. At the end, they were asked about typical risk management artifacts like risk lists, decision diagrams or risk probability distributions. Afterwards they had the opportunity to name other risk management artifacts.
3.2 Initial survey

The fourth section is focused on scope management. Again, the participants had to rate their knowledge about scope management, related sub-activities and artifacts like:

- Scope definition
- Scope tracking
- Managing change requests
- Scope control
- Creating a work breakdown structure
- Creating domain models
- Creating an maintaining a product backlog
- Creating product vision documents
- Creating burn-down charts

Finally, the fifth and last section covers stakeholder management. The participants were asked to rate their knowledge with regard to:

- Stakeholder identification
- Stakeholder prioritization
- Negotiating with stakeholders
- Conflict solving
- Creating a stakeholder list
- Creating personas
- Creating a communication plan

After a complete version of the survey was created, the required time to complete the survey was measured. The completion time of approximately 5 minutes seemed reasonable, thus an email was created, explaining the surveys background and how everything works. The survey’s access link was included into the email and the company’s internal mailing list was used to distribute it.

A total of 19 Accso employees completed the survey. This may not seem a lot, but the survey explicitly addressed people with project management experience. Their results were collected in an Excel sheet. Excel was also used to analyze the results and to identify suited participants.

In order to visualize interesting patterns and to get a better overview, Excels conditional formatting capability was used. This step revealed that detailed knowledge about the most common project management approaches like PMBOK, RUP and others are less common
than expected. Most of the participants stated that they have no or only little knowledge about most of them. From the 10 approaches presented in the survey, Scrum was the only exception. Most people stated that they are at least moderately or even very familiar with it. This outcome also supports the claim that Scrum is the most common agile approach (see section Scrum overview).

To allow for a more detailed analysis of the collected answers, all ordinal results were transformed into numerical ones to enable simple calculations (see table 3.1).

<table>
<thead>
<tr>
<th>Rating</th>
<th>Score</th>
</tr>
</thead>
<tbody>
<tr>
<td>Not at all &amp; Don't know</td>
<td>&gt; 0</td>
</tr>
<tr>
<td>A little &amp; Low</td>
<td>&gt; 1.5</td>
</tr>
<tr>
<td>Moderately</td>
<td>&gt; 3</td>
</tr>
<tr>
<td>Familiar &amp; Good</td>
<td>&gt; 4.5</td>
</tr>
<tr>
<td>Very familiar &amp; Very good</td>
<td>&gt; 6</td>
</tr>
</tbody>
</table>

Table 3.1: Expert survey’s rating scale

Based on this mapping, scores for each major category (methodologies, risk, scope, stakeholder) were computed to allow for a more differentiated selection of interview candidates. The application of additional filters enabled easy sorting. Figure 3.3 shows the resulting Excel sheet.

![Excel Sheet Analysis](image)

Figure 3.3: Analysis of the survey results

Based on the Excel sheet analysis, a total of 9 experts were selected. The major criteria for selection were:

- Their years of experience in software development.
- Their years of agile and traditional experience.
- Their individual scores in the 3 major categories (risk-, scope-, stakeholder management).
- Their cumulative score summing up all categories.

A list of the selected experts and their results is depicted in figure 3.4. For the sake of data protection, the list only contains the experts pseudonyms. The previously mentioned
color coding enables an easy comparison of the participants with regard to their years of experience and their various scores. Note that the applied color coding is relative to the specific range of values within each category. A green value indicates that the expert’s score is at the top of the range, while a red value indicates that the score is at the bottom of the range. With this knowledge in mind, someone can easily identify the experts strengths and weaknesses.

![Table showing interview participants]

Figure 3.4: Selected interview participants

Note: some experts listed above were selected, even though others had better overall scores. This was done to ensure a better coverage of the different areas and to balance traditional and agile work experience. Besides the 9 experts which were selected based on the survey results, 3 additional experts were selected based on personal recommendations. My thesis advisor and CEO of the company recommended 2 of them. The third one was recommended by another interview participant.

The next section describes the interview process and how it evolved over time.

### 3.3 Best practice interviews

#### 3.3.1 Interview preparation

The best practice interviews are one of the core activities within this thesis, because the whole thesis is built on their results. To increase the likeliness of good results, certain preparations had to be completed beforehand. This section explains the most important steps and the related artifacts.

1. Identify the points the interviews should focus on.
2. Define a possible structure to organize the results.
3. Define an interview structure to guide the interviews.
The first step was to provide a goal definition, to clarify the interview’s focus and to guide all subsequent activities. As explained in the chapter Motivation, this thesis strives to provide best practices for employees who just started with project management and have little experience with regard to risk management.

Thus, the interviews must focus on important influences in risk management. Furthermore, they need to yield best practices which are used by the more experienced project managers. Since this thesis also wants to collect potential project risks, they must be identified as well.

In the second step, a general structure was defined. Since interviews can vary from person to person, the results are likely to vary as well. In order to increase the uniformity of the various results, a best practice template was derived. It is based on the one defined in the BeST framework [Vos15]. It defines important aspects which are considered to be important for all practices. These aspects can also be used to derive interview questions.

- **Practice:** What’s the name of the activity?
- **Benefit:** Why is the activity performed? What does someone gain from doing it?
- **Category:** Which project management area is addressed by this practice and how does it refer to risk management?
- **Player(s):** Who is performing the activity?
- **Point in time:** When is the activity performed? (Within the course of the project)
- **Important influences (+/-):** Which influences are important for the activity? When is it especially effective or ineffective?
- **Constraints:** Which requirements must be fulfilled before the activity can be performed?
- **Handling:** Which tasks and steps are performed within the activity?
- **Optionalities:** Which non mandatory steps can be performed within the activity?
- **Artifacts:** Which artifacts are created within the activity?
- **Assessment:** Is this practice more efficient when it is used in an agile or in a traditional environment? Is it independent of the chosen approach? How useful is the activity with regard to applicability, effectiveness and efficiency? *(Note: This criterion is based on the results of the verification phase)*

The third step is based on the information from step 1 and 2. It is easier to guide and to structure the interviews, when participants know what they are dealing with. As a consequence, a document was created which explained the interviews context and the goals. The best practice template was included to familiarize people with the questions and the
"expected" results. Since it is more difficult to answer questions spontaneously a catalog of possible interview questions was prepared and included into this document. The example questions addressed risk-, scope- and stakeholder management, but people were instructed to view all areas from the risk management perspective.

All interview participants received this document one week before the interview appointment. The email to which it was attached, explained the content and asked people to use it in order to prepare for the interview.

In addition to the previously mentioned document, a schematic was developed to help people with the identification of past project risks. This scheme is based on the V-model XT, another software development methodology. The main reason for deriving it from the V-model XT is the way it is structured. It contains most of the important project phases and it is rather self-explanatory. Since the original version is too big and therefore not readable, figure 3.5 only depicts a condensed version.

**Figure 3.5: Typical project phases, based on v-model xt**

It basically shows the typical phases of a project's life-cycle as well as some additional information like common steps which are performed at each phase. Since the main focus of this thesis is project-/ risk management, the light gray area contains the most important phases. The darker area focuses on the implementation related phases and is therefore less important for this thesis.
3.3 Best practice interviews

Some example risks are written next to each phase to help participants to get started. By discussing each phase during the interview, more and more risks should emerge. This section described the preparation process for the interviews. The next section focuses on the execution of the interviews.

3.3.2 Conducting the interviews

This section focuses on the actual interviews and how they were performed. All interviews were scheduled to last approximately 90 minutes. Since risk management is a rather broad topic, it would have been easy to make them even longer. But again, the 90 minutes are another trade-off between content coverage and time consumption. During all interviews, risk management was the main focus. Depending on the participants background and survey score, additional aspects like agile or traditional approaches and scope or stakeholder management received more attention.

Further investigation of possible interview types lead to the conclusion that semi structured interviews are probably the best choice [IC109]. Semi-structured interviews consist of open- and closed questions. They are a compromise between getting as much information as possible and getting the right information. Open questions on the one hand result in a lot of information, because the participant is free to elaborate his or her thoughts. As a consequence, someone might receive duplicate or less relevant information. The closed questions on the other hand are very specific, hence they usually yield a smaller amount of information, but the information may have a higher value because they are likely to fill existing gaps. During the interview both types of questions were used alternately.

Open questions like: “When performing risk management how do you proceed?” or “Based on your personal experience what are important influences in risk management?” were used to get a first overview.

The more specific or closed questions such as: “At which point in time do you perform this practice?” or “Are there any constraints with regard to this practice?” were used to get a more specific answer or to get missing information e.g. with regard to the previously defined best practice template.

All interviews were performed in German, since all participants are German native speakers. Most of the interviews were conducted in Darmstadt, Accso’s headquarter. The remaining interviews were performed in Frankfurt or Cologne because some participants were too busy with project work and weren’t able to come to Darmstadt. From the 12 interviews, 10 were
performed using face-to-face communication, the remaining 2 interviews relied on Skype, since meeting up in person was not possible.

To case analysis and to document the results, the audio streams of all interviews were recorded. Each interview was post-processed shortly after the interview was conducted, resulting in a written summary of the important information.

Since new insights were gathered while conducting the interviews, the interview structure and content evolved over time. As a result, the interview preparation document was modified after almost every interview. Most modifications were related to adding, clarifying or deleting questions.

The first 5 interviews used a breadth before depth approach to gather as much information as possible. While more information was gathered, the amount of new information decreased. As a consequence it wasn’t enough to simply modify the preparation document. It also became necessary to restructure the interviews. To prepare for this restructuring, the information collected so far were analyzed and aggregated. Thus, the most common practices, artifacts and project risks were identified and refined.

The aggregated information were used to split the interviews 6 to 11 into 2 parts. The first part (60-45min) still consisted of semi structured questions to gather new information, even though the share of closed question increased. The second part (30-45min) focused on the verification of gathered data. A large portion of the 2nd part was used to estimate the collected project risks with regard to their probability of occurrence and impact. The above time intervals indicate that the time used for verification increased while more and more interviews were completed.

A couple of weeks passed between interview 11 and 12, hence a lot happened in-between. Thus, interview 12 was completely verification based. The participant received a draft version of the best practice candidates and only provided comments based on his own experiences. Section 3.4 describes how the various information were aggregated and analyzed to derive the best practice candidates.

3.4 Aggregation and analysis of interview results

3.4.1 Aggregation of interview data

As mentioned earlier, all interviews respectively their audio streams were recorded. During post processing, the recordings were transformed into written summaries reflecting the most
3.4 Aggregation and analysis of interview results

important information. Since the amount of data grew quickly, it became harder to keep an overview over the collected information. To maintain traceability, a color was assigned to each interview participant. By first converting each individual summary into the assigned color, before combining them into a single document, maintained traceability.

Unfortunately, this was of little help, because the different summaries were still individual and unrelated "sections". To improve the general overview, the information contained in the individual summaries were grouped by defining simple and rather general questions. Examples are: "Which are important influences in risk management?" or "How to analyze and evaluate risks?". These steps improved the overview a lot, but as the interviews continued, more and more questions were created to organize all the different information.

As a consequence, it was necessary to add additional structure. In the first step, the different questions were categorized into their different project management areas, e.g. risk management, stakeholder management, communication and others.

To avoid any additional overview problems, each category was subdivided into the different risk management phases (see section 2.1). By using the risk management phases as subcategories, it became possible to determine a precise "position" for each piece of information (see figure 3.6).

The different categories also relate to certain contexts, hence everyone can understand the gathered information more easily.

During all the "refactoring" measures, duplicates were removed from the general summary. Since the frequency with which things were mentioned by the participants indirectly reflects their importance, short notes were added. Thus, even if multiple responses were merged into a single paragraph or enumeration, the names of the original contributors and the frequency were preserved.

<table>
<thead>
<tr>
<th>Risk management</th>
</tr>
</thead>
<tbody>
<tr>
<td>Preparation</td>
</tr>
<tr>
<td>Identification</td>
</tr>
<tr>
<td>Evaluation</td>
</tr>
<tr>
<td>Controlling</td>
</tr>
<tr>
<td>Reporting</td>
</tr>
<tr>
<td>Miscellaneous</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Scope management</th>
</tr>
</thead>
<tbody>
<tr>
<td>---</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Stakeholder management</th>
</tr>
</thead>
<tbody>
<tr>
<td>---</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Communication management</th>
</tr>
</thead>
<tbody>
<tr>
<td>---</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Monitoring &amp; control</th>
</tr>
</thead>
<tbody>
<tr>
<td>---</td>
</tr>
</tbody>
</table>

Figure 3.6: Organization of interview data
3.4.2 Analysis of interview data

While aggregating the interview results, they were also analyzed, even though this was on a rather implicit basis. The frequency with which things were mentioned was the first indicator for their importance. The use of questions and subcategories uncovered hidden relationships and made them more visible.

As mentioned earlier, some principles of grounded theory were used to help with the aggregation and analysis of the interview data.

Grounded theory begins with an activity called open coding [WNA15]. It begins as soon as data is available and assigns codes, which are basically tags or short descriptions for the collected information. While more and more data becomes available new and old codes are compared. If they are related, they are grouped into a higher level of abstraction, called concepts. This is called constant comparison. This process is also repeated on concept level, where similar concepts are grouped together. These grouped concepts are called categories and are the highest level of abstraction in grounded theory [WNA15].

When applying grounded theory to the interview data, the different activities, risks and influences resemble codes. The different codes are aggregated into best practice candidates which resemble concepts and the different project management areas containing the practices are the categories.

In the beginning, the aggregated and structured data was scanned and all activities mentioned in the text were extracted. By grouping the various activities into logical packages the best practices emerged.

For example the practice "Prepare important decisions". It is composed of the following extracted activities:

- Address stakeholders based on their individual level.
- Keep your stakeholders informed.
- Ask stakeholders about their opinions.
- Make sure everyone has the required information.
- Always provide explanations when you do something.

Even though all activities are different, they all support the same goal. The resulting best practice candidate is depicted in figure 3.7.
### 3.4 Aggregation and analysis of interview results

![Image](image_url)

<table>
<thead>
<tr>
<th>Practice:</th>
<th>Prepare Important decisions:</th>
</tr>
</thead>
<tbody>
<tr>
<td>Benefits:</td>
<td>This practice reduces the risk to encounter unexpected decisions, outcomes or delays: Moreover, it increases the chances that the decision maker will follow your recommendations.</td>
</tr>
<tr>
<td>Category:</td>
<td>Project management area: Stakeholder management</td>
</tr>
<tr>
<td>Purpose:</td>
<td>Treatment</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Player(s):</th>
<th>Main: Project manager</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Secondary: Decision maker</td>
</tr>
</tbody>
</table>

| Point in time: | Before important decisions will be made, e.g. before steering committee meetings. |

<table>
<thead>
<tr>
<th>Comments:</th>
</tr>
</thead>
<tbody>
<tr>
<td>Important:</td>
</tr>
<tr>
<td>Influences:</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Constraints:</th>
</tr>
</thead>
<tbody>
<tr>
<td>Handling:</td>
</tr>
<tr>
<td>1) Meet with the important decision makers who will have a say in the final decision beforehand. This can be done individually or even group wise.</td>
</tr>
<tr>
<td>2) &quot;Pick them up&quot; and explain the current project situation.</td>
</tr>
<tr>
<td>3) Ask them for their current opinion on the project.</td>
</tr>
<tr>
<td>4) Present and explain the decisions which will be addressed in the next meeting.</td>
</tr>
<tr>
<td>5) Argue which solution you prefer (in case there are multiple alternatives) and tell them why.</td>
</tr>
<tr>
<td>6) Ask them for their opinion and information needs and address them if needed.</td>
</tr>
</tbody>
</table>

| Further options: |
| Artifacts: |

**Figure 3.7: A best practice candidate**

Repeating this process over and over again resulted in the 19 best practice candidates. These are the foundation for the best practice catalog, but before calling them best practices, they must be verified first. Therefore, the next section focuses on the verification of the collected practices.
3.5 Verification of the interview results

The data creating the foundation for the best practices candidates was extracted from 12 experienced Acceo experts through semi structured interviews. This section will therefore deal with the verification of the collected interview results.

During the last phase the different results were aggregated and mapped onto the best practice structure see figure 3.7. Even though this structure was derived to make sure that all the important data was collected during the interviews, it makes sense to reuse it for the verification as well. The uniform structure simplifies the verification process because it eases navigation and someone can compare the practices more easily.

The verification phase addresses two issues. First of all, the data for the best practices was collected during multiple interviews, hence a lot of mixing and merging took place before the practices emerged. It is therefore necessary to make sure that the meaning of the original statements was not modified and that the resulting practices do represent the prevailing opinions of the interview participants. The second issue is whether the results are also valid in other contexts or not. To cover both issues, the decision was made to form a mixed verification group. The first half consists of Acceo employees and the other half consists of external experts. Unfortunately, the inclusion of the external experts made the verification more difficult, because it needed to be more flexible and less time consuming. Therefore, instead of simply printing the interim best practice catalog, a more sophisticated verification process was necessary.

At first, the Google Forms service came to mind, but because of data privacy concerns, my advisor recommended to use the company’s internal survey tool instead. It was not as comfortable as the Google Forms service but it did its job, even though some initial training was necessary to comprehend it.

3.5.1 Verification survey

Since time is always an issue, the verification survey is designed to require around 60 minutes to complete. The participants have to rate a total of 19 practices, hence they have approximately 3 minutes per practice. To increase the expert’s focus, only one practice is presented at a time. For each practice the expert has to answer 4 questions.

1. The expert has to rate the proposed practice based on its applicability.
2. The expert has to rate the proposed practice based on its effectiveness.
3. The expert has to rate the proposed practice based on its efficiency.
4. The expert has to specify which context (agile <-> traditional) supports this practice best. When an approach supports a practice, either its effectiveness or efficiency should increase, but it is also possible that a practice is context independent.

Besides those 4 mandatory questions, the expert can leave an additional comment using the text-field provided below. To get a better idea of how the survey looked like, the survey's first practice is displayed as an example below (see figure 3.8).

```
<table>
<thead>
<tr>
<th>Practice</th>
<th>Question</th>
<th>Code</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.5 Verification of the interview results</td>
<td>3.5 Verification of the interview results</td>
<td>3.5</td>
</tr>
</tbody>
</table>

| 4. The expert has to specify which context (agile <-> traditional) supports this practice best. When an approach supports a practice, either its effectiveness or efficiency should increase, but it is also possible that a practice is context independent. |

Besides those 4 mandatory questions, the expert can leave an additional comment using the text-field provided below. To get a better idea of how the survey looked like, the survey's first practice is displayed as an example below (see figure 3.8).

```

Figure 3.8: Verification survey

38
3.5 Verification of the interview results

3.5.2 Verification results

The original plan was to have 5 internal and 5 external experts to evaluate the collected best practices. To give them enough time to participate, the survey was online for almost 5 weeks. Unfortunately, some participants went on vacation before completing it or lacked the time or the interest to participate. In the end a total of 8 experts participated in the survey. Sadly, 3 of the 8 received responses were incomplete. To ensure the surveys consistency, only the completed responses were considered. Thus a total of 5 datasets was used to verify the collected practices.

The results of the verification survey are presented in figure 3.9. Note that the number in the methodology column refers to question 4 which asked which approach supports a practice best. The participants had to choose between 5 options. Thus, the scale goes from 1 (agile) to 5 (traditional). A rating of 3 indicates that the practice is neutral and therefore more or less approach independent. The color-coding was added to easier assess whether a practice is good or not. A dark green background indicates a good rating and dark orange a bad one.

<table>
<thead>
<tr>
<th>Practices</th>
<th>Avg. Score</th>
<th>Applicability</th>
<th>Effectiveness</th>
<th>Efficiency</th>
<th>Methodology</th>
<th>Rating</th>
</tr>
</thead>
<tbody>
<tr>
<td>Practice 1</td>
<td>4.17</td>
<td>4.40</td>
<td>4.50</td>
<td>3.60</td>
<td>Rather traditional</td>
<td>3.90</td>
</tr>
<tr>
<td>Practice 2</td>
<td>4.23</td>
<td>5.80</td>
<td>4.50</td>
<td>4.40</td>
<td>Rather agile</td>
<td>2.60</td>
</tr>
<tr>
<td>Practice 3</td>
<td>4.70</td>
<td>5.00</td>
<td>4.50</td>
<td>4.60</td>
<td>Neutral</td>
<td>3.20</td>
</tr>
<tr>
<td>Practice 4</td>
<td>4.28</td>
<td>6.00</td>
<td>4.25</td>
<td>4.00</td>
<td>Neutral</td>
<td>3.70</td>
</tr>
<tr>
<td>Practice 5</td>
<td>3.32</td>
<td>4.20</td>
<td>2.75</td>
<td>3.00</td>
<td>Neutral</td>
<td>3.40</td>
</tr>
<tr>
<td>Practice 6</td>
<td>2.48</td>
<td>2.80</td>
<td>2.25</td>
<td>2.40</td>
<td>Rather traditional</td>
<td>3.90</td>
</tr>
<tr>
<td>Practice 7</td>
<td>2.96</td>
<td>3.20</td>
<td>2.50</td>
<td>3.00</td>
<td>Agile</td>
<td>1.20</td>
</tr>
<tr>
<td>Practice 8</td>
<td>3.15</td>
<td>3.40</td>
<td>3.25</td>
<td>2.80</td>
<td>Neutral</td>
<td>3.00</td>
</tr>
<tr>
<td>Practice 9</td>
<td>2.52</td>
<td>2.80</td>
<td>1.75</td>
<td>1.00</td>
<td>Neutral</td>
<td>3.00</td>
</tr>
<tr>
<td>Practice 10</td>
<td>3.76</td>
<td>4.00</td>
<td>3.75</td>
<td>3.60</td>
<td>Neutral</td>
<td>3.00</td>
</tr>
<tr>
<td>Practice 11</td>
<td>2.45</td>
<td>2.60</td>
<td>2.75</td>
<td>2.20</td>
<td>Neutral</td>
<td>2.80</td>
</tr>
<tr>
<td>Practice 12</td>
<td>2.25</td>
<td>2.20</td>
<td>2.75</td>
<td>1.80</td>
<td>Rather agile</td>
<td>2.00</td>
</tr>
<tr>
<td>Practice 13</td>
<td>4.70</td>
<td>5.00</td>
<td>4.50</td>
<td>4.60</td>
<td>Neutral</td>
<td>3.40</td>
</tr>
<tr>
<td>Practice 14</td>
<td>3.65</td>
<td>5.00</td>
<td>4.75</td>
<td>3.20</td>
<td>Neutral</td>
<td>3.00</td>
</tr>
<tr>
<td>Practice 15</td>
<td>3.92</td>
<td>4.20</td>
<td>4.00</td>
<td>3.60</td>
<td>Neutral</td>
<td>3.40</td>
</tr>
<tr>
<td>Practice 16</td>
<td>3.53</td>
<td>4.70</td>
<td>4.00</td>
<td>3.40</td>
<td>Agile</td>
<td>1.80</td>
</tr>
<tr>
<td>Practice 17</td>
<td>4.05</td>
<td>4.20</td>
<td>3.75</td>
<td>4.20</td>
<td>Neutral</td>
<td>2.80</td>
</tr>
<tr>
<td>Practice 18</td>
<td>3.45</td>
<td>3.20</td>
<td>3.75</td>
<td>3.00</td>
<td>Neutral</td>
<td>3.20</td>
</tr>
<tr>
<td>Practice 19</td>
<td>2.90</td>
<td>3.60</td>
<td>2.50</td>
<td>2.60</td>
<td>Neutral</td>
<td>3.40</td>
</tr>
</tbody>
</table>

Figure 3.9: Results of the verification survey

The results displayed in figure 3.9 as well as the comments provided by the different experts are analyzed in chapter 5.
3.6 Lessons learned

This section described the last step in the data gathering and verification process. While performing it, certain issues appeared and impressions and experiences were gathered. The most important ones are discussed in the next section before the final results are presented in chapter 4.

3.6 Lessons learned

The performed approach worked pretty good, but in retrospect, some improvement suggestions come to mind. First of all, a better and more structured organization of the interview results. By doing this, less effort would have been necessary to keep a good overview. Secondly, the early evaluation of results. In grounded theory, open coding and constant comparison (see 3.4.2) are applied constantly, thus core aspects can be identified early. During the interviews, the main focus was on the post processing of the recordings and the refactoring of the interview preparation material. Maybe a bigger emphasis on analysis of the collected results would have been beneficial. This conclusion is supported by another fact. Since there was a time gap of 2 weeks between interview 11 and 12, it was possible to analyze the aggregated data and to define the best practice candidates, which were then used in interview 12. As a consequence, the feedback based on the practice drafts was way more precise and valuable than in other interviews.

**Recommendation:** Use a more iterative approach!

Perform 3-5 interviews, then analyze the results and create possible best practice candidates. Ask the participants of the next 1 or 2 interviews for feedback on the created drafts. This approach can reduce the amount of redundant and unrelated information and helps people focusing on the specific practices. Thus, it will probably improve the quality of the overall results, because people can specifically suggest improvements, since the practices are already presented in a more abstract and compact form.

The only drawback which comes to mind when using this approach is that someone might miss other important aspects due to the early increase in focus. Although, based on the experiences from this thesis, the chances are rather low.

The learnings presented above conclude this chapter on the applied research approach. The next chapter focuses on the results and deliverables which were derived from the findings.
Chapter 4

Results

This chapter presents the two core deliverables of this thesis. Those are the practice catalog and the list of potential project risks (see practice catalog). Both of them are the outcome of the performed research approach described in chapter 3. This chapter explains the different deliverables, the relationships between them and it provides suggestions on how they should be used. Afterwards it focuses on the integration of the collected results into the BeST framework and how the artifacts can be maintained in the future.

4.1 Deliverables

4.1.1 Practice catalog

The practice catalog is basically a summary of the collected practices. The practices are grouped by the different project management areas. Within each area the practices are ordered based on their logical appearance within the project’s life-cycle.

To ease navigation within the catalog, it includes an overview page which is depicted in figure 4.1. The overview uses 2 criteria, which are the same as the ones used to structure the interview data (see figure 3.6). The first one is the project management area the practice belongs to. The second one refers to its purpose within risk management.

The resulting matrix indicates the intended purpose for each practice, whether it is preparation, identification, evaluation, management, controlling or treatment. Note: The term preparation is used instead of context because it seems to be more appropriate considering the collected practices. The term treatment was added to differentiate possible counter measures from pure management practices. The project management areas are colored differently to visually separate them.
4.1 Deliverables

The different background colors in the ID column refer to the rating a practice received in the verification process. See figure 3.9 and section 5.1 for additional information.

<table>
<thead>
<tr>
<th>Category</th>
<th>ID</th>
<th>Practices</th>
<th>Risk related purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Risk management</td>
<td>1</td>
<td>Create a good proposal</td>
<td>X</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>Create an open and risk-friendly atmosphere</td>
<td>X</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>Create an initial risk list based on existing artifacts and know how</td>
<td>X X X</td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>Manage risks using a risk list</td>
<td>X</td>
</tr>
<tr>
<td></td>
<td>5</td>
<td>Visualize your known risks by using a risk map</td>
<td>X</td>
</tr>
<tr>
<td></td>
<td>6</td>
<td>Integrate risk-specific symbols into your risk board</td>
<td>X X</td>
</tr>
<tr>
<td></td>
<td>7</td>
<td>Track your risk management effectiveness</td>
<td>X</td>
</tr>
<tr>
<td></td>
<td>8</td>
<td>Perform scenario planning</td>
<td>X X X</td>
</tr>
<tr>
<td></td>
<td>9</td>
<td>Perform the hot chair game</td>
<td>X</td>
</tr>
<tr>
<td>Scope management</td>
<td>10</td>
<td>Create an architectural overview</td>
<td>X</td>
</tr>
<tr>
<td></td>
<td>11</td>
<td>Compare the company’s internal processes</td>
<td>X</td>
</tr>
<tr>
<td></td>
<td>12</td>
<td>Create a product vision box</td>
<td>X</td>
</tr>
<tr>
<td>Stakeholder</td>
<td>13</td>
<td>Perform a stakeholder analysis</td>
<td>X</td>
</tr>
<tr>
<td>management</td>
<td>14</td>
<td>Prepare important decisions</td>
<td>X</td>
</tr>
<tr>
<td></td>
<td>15</td>
<td>Actively manage changes and expectations associated with the project</td>
<td>X</td>
</tr>
<tr>
<td></td>
<td>16</td>
<td>Perform a daily standup meeting</td>
<td>X</td>
</tr>
<tr>
<td></td>
<td>17</td>
<td>Plan your communications</td>
<td>X</td>
</tr>
<tr>
<td>Monitoring &amp; control</td>
<td>18</td>
<td>Track the team’s progress</td>
<td>X</td>
</tr>
<tr>
<td></td>
<td>19</td>
<td>Perform post mortem analysis</td>
<td>X</td>
</tr>
</tbody>
</table>

Figure 4.1: Practice catalog
4.1 Deliverables

For additional information on the actual practices, please refer to the actual practice catalog. The catalog is separate document which explains how the different deliverables are related to each other and it provides suggestions on how to use them. Why the decision was made to create a separate document is explained in section 4.1.3.

4.1.2 Potential project risks

The second big deliverable is the list of potential project risks. In the very beginning, the potential project risks and suitable counter measures were collected together with the other responses. After reaching a certain number of risks, the risks were extracted and stored within a separate risk list. By now, the risk list contains a total of 64 potential project risks. To ease navigation, the identified risks are organized using a categorization system based on the one proposed by the PMBOK [IEE11]. Other categorizations like the one proposed by Knoll [KB14] appear to be less suited. Knolls version is simply to fine grained and not all categories are useful on the project level. Besides, the PMBOK is a very well known and established de facto standard.

The 5 categories as well as their subcategories are listed below:

- **Organizational**: Project dependencies, Resources, Funding, Prioritization
- **Project management**: Estimation, Planning, Controlling, Communication
- **Technical**: Requirements, Technology, Technical complexity, Quality, Performance & Reliability
- **External**: Subcontractors & Suppliers, Regulatory, Market, Customer
- **Business**: Processes, Business complexity

The PMBOK only suggests the first 4 major categories, namely Organizational, Project management, Technical and External. The 5th category, called Business was added because many interview participants emphasized the differences between technical and business aspects.

The risk list also contains probability and impact estimations for 43 risks. Of course, these estimations are not 100% accurate. First of all, they are based on personal experience, hence different people with different backgrounds may have different opinions. Secondly, the experts were "forced" to rate probability and impact without a specific project context. As a consequence, the risks may not even be an issue within a specific project context or the estimations can be way off. Lastly, the number of estimations per risk varied, because some risks were identified later than others, thus fewer experts had the opportunity to estimate them.
As mentioned in the beginning, the risk list contains a total of 64 risks, but only 43 were rated by the various experts. The other 21 risks are taken from various risk lists which can be found on the Internet [Mar13b][Mar13a][Lev08][MLS14][Raw14]. They were selected based on my subjective opinion, because they seemed to be valuable additions to the existing ones. The 21 risks which were added later were rated on a subjective basis and not by the experts. This is not ideal, but it was done to increase the risk lists uniformity and consistency.

The generic risk list contains a total of 11 columns with additional information. The next enumeration explains them from the left to the right.

1. **Category**: One of the 5 risk categories based on the PMBOK definition.
2. **Risk origin**: Describes where the risk usually originates from (context).
3. **ID**: The risk’s ID to enable references.
4. **Average probability**: The average value based on the expert estimations for the risk’s probability.
5. **Average impact**: The average value based on the expert estimations for the risk’s impact.
6. **Risk exposure**: This a simple metric to prioritize risks. It is calculated by multiplying probability and impact.
7. **Risk**: A description of the risk.
8. **Impact comments**: This provides some additional information on the potential impact.
9. **Counter measures**: Lists potential counter-measures and refers to suited best practices.
10. **Person responsible**: Indicates who is accountable for this risk and the initiated counter measures.
11. **Remarks**: Contains some comments on the risk, e.g. when the risk is related to another one.

An abridged part of the risk list is displayed in figure 4.2. A complete overview over the risk list can be found in the resulting Excel file (Risklist.xls) or in the practice catalog.

![Figure 4.2: The generic risk list](image-url)
4.1 Deliverables

To evaluate the distribution of the collected risks, the risk exposure was calculated by multiplying the probability and impact estimations for each risk. After calculating the risk exposure for each risk, they were sorted from low to high. The resulting graph (red line) is displayed in figure 4.3 and it indicates that the different risk scores are almost evenly distributed over the whole spectrum. The green line was added to indicate a perfect distribution. The y-axis displays the calculated risk exposures and the x-axis the different risk IDs. Thus someone can easily identify the most important project risks. At least with regard to their risk exposure.

![Graph showing risk distribution](image)

Figure 4.3: Distribution of risk scores

### 4.1.3 How to use the deliverables

In addition to different practices, the catalog provides introductions and overviews. Moreover, it explains how the different deliverables are related to each other. Even though this thesis is certainly interesting, most Accso employees probably prefer to get the results in a more concise version. As a consequence the results were removed from this thesis and transferred into a separate document, the practice catalog. This way, people interested in the results can print and use them more easily because they do not have to deal with the whole thesis. To ensure the catalogs consistency and usability, some of the content covered in this thesis is described there as well.

Since the previous sections introduced the main deliverables, figure 4.4 shows how they are related to each other. Note that the individual expert estimations included into the risk list are hidden by default. Only the averaged values are displayed.
Step 1: In the first step, project managers can take a look at the generic risk list to learn about potential project risks. (If someone is not responsible for the identification of potential risks, it is also possible to directly proceed with the practice catalog to learn about good risk management practices in general.) Since the generic risk list contains the probability and impact estimations for the different risks, a project manager can use them as an initial prioritization. When disagreeing with them, he or she can replace them with more appropriate ones.

Step 2: To get a better overview over the remaining risks, the Excel document also contains a risk map which is automatically generated from the different expert estimations. The risk map also includes 2 thresholds which help to identify the most important risks. The thresholds refer to the rating scale for the risk estimations, which is from 1.5 to 4.5. The first threshold (orange) is at 50% and the second one (red) at 75% of the overall scale.

Step 3: When the most important project risks are identified, the project manager can switch back to the generic risk list to replace the generic risks with specific ones. The definition
of specific project risks is necessary to make them manageable. A generic risk represents an arbitrary number of risks and management strategies, but when transforming it into a concrete risk it becomes "tangible" and therefore manageable. A good example is risk 68 ("Internal or external staff absences due to accidents, sickness, dismissal, ... "). It is a fact that people can be unavailable during the course of the project, but this is not a very precise description of a project risk and in some cases this may not be a risk at all. Therefore someone should look at the risk and analyze its consequences for the own project. Imagine that a team only has a single person who is an expert for the used database and if this expert is not available to perform the data migration, the whole team has to wait for his return. Thus, when thinking about risk 68, a possible derivation is: "The project's specialist for the used database is unavailable during the data migration phase". As a consequence someone should replace the generic risk 68 with the more specific version. If a generic risk results in multiple specific risks, someone should create a separate entry for each one of them. In case a generic risk is not relevant at all for the given project then someone can choose to keep or to simply delete it. When keeping it, the estimations should be modified to indicate the risk's little priority. Of course, someone could also use this generic risk list as a checklist and document the risks somewhere else. But in this scenario, people may simply ignore or forget about the check list, thus by "forcing" them to replace the generic risks, someone can make sure that they actually evaluate all the generic risks.

**Step 4:** After deriving a specific risk form a generic risk, someone should also look at the additional information which are provided for each generic risk. They may describe possible counter measures which can be used to mitigate the derived risk or contain other useful information. Where appropriate, the generic risk list refers to suited best practices. Additional information on the practices can be found in the practice catalog. Read the referenced practice, evaluate it and apply it if reasonable.

**Step 5:** The Excel file contains another spreadsheet with a risk burn down chart. This chart is created automatically and shows the project's risk exposure, based on the estimations in the risk list. To use it, someone has to copy the current risk exposure value from the top of the risk list into the provided table. When doing this periodically e.g. at the end of each sprint, this diagram provides a good visualization of how the project risk evolved over time.
4.2 Integration and maintenance of results

4.2.1 BeST integration

The integration into the BeST framework is rather implicit since it is mainly limited to updating existing power point slides. The updated slides simply indicate how the thesis outcome and the practice catalog fits into the framework. This is done by adding important artifacts and activities or by highlighting the relationships to other domains. Figure 4.5 shows a possible integration of the findings.

![Figure 4.5: BeST framework: Project management [Vos15]](image)

The figure also reveals why Accso offered a master’s thesis focusing on risk management and the related areas of scope- and stakeholder management. Compared to the other areas, those 3 are rather vague and not fully developed. All contributions made to the existing slide are highlighted by using a black font. The relationships of risk management to some of the other areas are indicated by the red arrows. Especially the planning area seems to be interesting form a risk management perspective, due to its huge potential impact on a project’s success. But since somebody else wrote a thesis on planning, this thesis mostly ignored it.
4.2.2 Maintenance

A lot of time and effort was invested to produce the described results. Unfortunately, things in computer science tend to change rapidly, thus these artifacts need to be maintained. Otherwise they will become obsolete in the near future.

To ensure the maintenance of the artifacts, management could distribute the risk list to interested project managers. While working with the risk list, the project managers may discover missing project risks, possible counter measures or other things which are currently missing. These could be documented in a separate section e.g. at the bottom of the risk list. When the project is completed, the post mortem analysis is performed. During this activity the project manager can re-estimate the different risks based on the insights he or she gathered while working on the project. Afterwards the list is emailed to a person who is responsible for merging the various results. The most up-to-date version of the list can be published on a network drive or re-distributed via email, e.g. every 3 to 6 months.

An even better possibility provides the intranet. The different artifacts can be transferred into a wiki or something similar. Since the intranet is accessible via the Internet, all employees can easily do so. Moreover, every project manager could integrate the changes directly, thus nobody is needed to merge the different results. Another positive side effect is that the intranet version is always up to date.

Lastly, wikis usually have an version control system, thus someone could use it to analyze the risks and how they changed/evolved over time. This might be useful to identify emerging trends.

To sum everything up, this chapter basically presented the different deliverables of this thesis and how to use them. Unfortunately, the question from the beginning has not been answered yet. In order to do this, the next chapter starts with an analysis of the collected practices followed by a more detailed evaluation of relationship between risk management and agile approaches.
Chapter 5

Analysis

This chapter starts by analyzing the collected practices. In the process each practice receives a rating to indicate whether it is better supported by traditional or by agile approaches. This will help to evaluate the relationship between the different practices and agile approaches. Afterwards the focus shifts towards Scrum and its indirect risk management capabilities. This is done by elaborating on the different activities and principles which are beneficial for risk management.

Based on the learnings from both parts, a conclusion on the relationship of risk management and agile methodologies is drawn.

5.1 Evaluation of best practices and their level of agility

This section examines the collected best practices in general and evaluates their relationship to agile approaches. Starting now, this thesis refers to this relationship as their level of agility. The examination of a practice is based on the ratings and the feedback it received during the verification process. Therefore each practice is supplemented by a little diagram which visualizes the ratings it received. To provide even more information, the confidence interval for each characteristic is shown as well. The chosen confidence level is 80% because there are only 5 estimations per characteristic. Higher values like 95% are likely to result in rather large confidence intervals due to the limited sample size.

As mentioned earlier, each practice is rated with regard to 4 categories. They are described in the next paragraphs.

- **Applicability**: Is it possible to apply this practice in general or only in certain cases?
- **Effectiveness**: How good does the practice support its goal?
- **Efficiency**: How is the cost vs. benefit ratio?
To ease the classification of the different practices, the scores from the 3 categories above are used to calculate an average score. Based on the average score, a practice receives one of the following classifications.

<table>
<thead>
<tr>
<th>Classification</th>
<th>Score Range</th>
</tr>
</thead>
<tbody>
<tr>
<td>Highly recommended</td>
<td>4.0 ≤ score ≤ 5.0</td>
</tr>
<tr>
<td>Recommended</td>
<td>3.0 ≤ score &lt; 4.0</td>
</tr>
<tr>
<td>Optional</td>
<td>2.0 ≤ score &lt; 3.0</td>
</tr>
<tr>
<td>Not recommended</td>
<td>1.0 ≤ score &lt; 2.0</td>
</tr>
</tbody>
</table>

Table 5.1: Classification scale for practices

The 4th category, the level of agility is a result of the 5 answer options the experts could select during the verification survey (see 3.5.1). Each option had a number associated with it starting at 1 (agile) to 5 (traditional). The averaged score is used to define the level of agility. Since the rating scale is from 1 to 5, each interval has a range of 4/5=0.8.

- **Agile [1.0 ≤ score ≤ 1.8]**: This category contains practices which strongly benefit from agile project management approaches.
- **Rather agile [1.8 < score ≤ 2.6]**: This category contains practices which benefit a little more from agile than from traditional approaches.
- **Neutral [2.6 < score ≤ 3.4]**: This category contains practices which are independent of a specific methodology.
- **Rather traditional [3.4 < score ≤ 4.2]**: This category contains practices which benefit a little more from traditional than from agile approaches.
- **Traditional [4.2 < score ≤ 5.0]**: This category contains practices which strongly benefit from traditional project management approaches.

As a consequence the level of agility indicates, whether a practice benefits from an agile approach, a traditional approach or if it is approach independent. Someone should keep in mind that the ratings are based on the rather subjective opinions of the participants, thus other people might disagree.

The practices are presented in the same order as they are in the practice catalog. Since the different practices are not included in this thesis, it may be a good idea to have the practice catalog at hand while reading the next sections.
5.1 Evaluation of best practices and their level of agility

5.1.1 Risk management

Practice 1: *Create a good proposal*

This practice is highly recommended by the different experts. It has an averaged score of 4.17. Since almost every project contains a proposal, this practice should be applied as well, because it addresses common sources of problems.

As someone can see, the practice received a methodology rating of 3.80 which translates into “rather traditional”.

This indicates that this practice probably benefits from careful planning and documentation. When reading the practice, this rating is not surprising, because it mainly deals with removing uncertainty and documenting agreements to avoid interpretation problems. An unambiguous proposal with correct effort estimations is especially important in fixed price projects, where it has a huge influence on whether you make or lose money.

Since the methodology rating is “only” 3.80, people should also remember the agile manifesto, which states "Customer collaboration over contract negotiation". Thus, even if documenting contract issues is important, it should not replace face to face communication.

Practice 2: *Create an open and risk friendly atmosphere*

This practice is also highly recommended with an average score of 4.23. Even though some experts stated that the main challenge is to maintain a good atmosphere during the course of the project.

It received a methodology rating of 2.40 which translates to "rather agile". This may indicate that risk management benefits from the agile value system. The agile culture and the different values probably ease the implementation of a good project atmosphere. Additional influences which are likely to support
5.1 Evaluation of best practices and their level of agility

it are self organization and responsibility which are promoted in agile teams. **Expert 1** also states that a certain risk awareness usually increases the team’s feeling of responsibility.

**Practice 3: Create an initial risk list based on existing artifacts and know-how**

Practice 3 also received the “highly recommended” status with a rating of 4.70. The verificators seem to agree, that it is a good idea to reuse existing existing risk lists or other risk related information. The fact that this practice received such a high rating is especially good, because it confirms that it was a good idea to collect the risks of past projects to create a reusable list of potential project risks.

**Expert 2** added that “A risk list should only contain specific risks, not generic ones. [translated]”. This supports the idea to replace the generic risks with more specific ones in order to make them manageable.

The practice received a methodology rating of 3.40, thus it is still classified as “neutral”. The reason why it did not receive a perfect 3.00 rating may be due to the fact that explicit risk management is more common in traditional projects. The same is true for extensive documentation. Even though risks are something worth documenting, even in agile projects.

**Practice 4: Manage risks using a risk list**

This is another “highly recommended” practice. Its avg. rating is 4.28. This is probably the core of risk management, because it is the central artifact which contains all the important information.

This practice is also classified “neutral” with a score of 3.20. Again, it is no perfect 3.00, but this is probably due to the same reasons already stated in practice 3.
5.1 Evaluation of best practices and their level of agility

**Practice 5: Visualize your known risks by using a risk map**

This is the first practice in the "recommended" category due to an average score of 3.32. The experts agree, that it is a good idea to use this practice for reporting purposes e.g. in presentations or reports when a certain of abstraction is required. Beyond reporting, they concur that it has little use and people who are really interested in the current risk situation should use the risk list instead.

The low efficiency score seems surprising at first, but **Expert 4** stated that he rated the practice’s efficiency with 3 because he is not aware of any automatic Excel templates who create this visualization automatically. Maybe the others had the same reason. Good news is, the Excel template containing the generic risk list etc. provides this functionality.

This practice is also classified “neutral” with a score of 3.40. Again, it is no perfect 3.00, but this is probably due to the same reasons already stated in practice 3.

**Practice 6: Track your risk management effectiveness**

This practice only received a score of 2.48 and is therefore classified “optional”. The efficiency score of 2.4 is also surprisingly low, considering that it is possible to generate the "diagram" automatically.

The methodology rating of 3.60 which labels the practice as “rather traditional” is also surprising. Unfortunately, the comments do not provide enough information to explain these ratings. Agile approaches in particular seem to be suited because of their constant and clearly defined iterations. The end of an iteration should be a good moment to update the project’s risks status. Also the suggested diagram is basically an agile burn-down chart.
5.1 Evaluation of best practices and their level of agility

**Practice 7: Integrate risk specific symbols into your task board**

This is another activity which did not fulfill the expectations. It is also classified as "optional" with an overall rating of 2.90. Interesting is the big deviation between the different expert ratings. Some are very good while others are rather bad. The same is true for the comments.

**Expert 1** stated: "These flags show directly what's going on and should wake people up."

**Expert 4** stated that "the team should concentrate on the tasks at hand and should not do risk management as well [...] [translated]". Maybe this reflects their personal experiences and whether they mostly do agile or traditional projects. This would also explain the rather low applicability rating of 3.20, because typically only agile projects make use of Task Boards. Another indicator is the methodology rating of 1.20 which clearly categorizes this practice as "agile".

The effectiveness of 2.50 seems reasonable since it only adds some additional informal to remind people about certain things. The efficiency of 3.00 on the other hand is a surprise because it should not be a lot of work to add an additional symbol or a reference to a certain risk on a story card.

But as implied by the "optional" rating, people reading the practice can decide for oneself whether they want to use it or not.

**Practice 8: Perform scenario planning**

This practice describes a good method to identify additional risks or to further evaluate existing ones. This is reflected by its "recommended" rating (3.15). Especially when comparing it to practice 9, the experts like this one more, even though there are some deviations in the ratings and comments.

**Expert 1:** "This is necessary to gain alternative plans and bring out options in scenarios."

**Expert 2:** "Very good to get specific risks [translated]."
5.1 Evaluation of best practices and their level of agility

Expert 5: [T]oo much effort for too little result. Also shifts the mindset too far towards possible failure. Never seen this done.

The "neutral" methodology rating (3.00) is also no surprise since discussing scenarios is independent of a specific approach.

Practice 9: *Perform the hot chair game*

This practice has the same purpose as the one described in practice 8, but the experts clearly voted that they like practice 8 better than this one. Especially the effectiveness received with 1.75 the lowest rating within the whole survey. Expert 4 provided an explanation for this. To sum it up, he stated that this practice probably requires rather experienced experts who know each other and have established a certain level of trust between them.

As a consequence this practice is only categorized "optional" due to an average rating of 2.52. Like in practice 8, the "neutral" methodology rating (3.00) is no surprise as well.

5.1.2 Scope management

Practice 10: *Create an architectural overview*

With a score of 3.78 this practice is classified as "recommended", but it is very close to "highly recommended". In their comments, some of the experts stated that this practice may not uncover many additional risks, but that it is important to understand the major dependencies between the involved systems. A better understanding of the overall picture may help them to better assess the consequences of certain decisions, thus improving their quality. Being knowledgeable about the overall picture and how the own system fits into this landscape is important in almost all software development projects. All experts
5.1 Evaluation of best practices and their level of agility

seem to agree that this practice is crucial and that the chosen development methodology does not really matter here. The fact that all experts selected the “neutral” option for the methodology rating confirms this as well.

**Practice 11: Comprehend the company’s internal processes**

Based on the 2.45 avg. rating this practice should be classified as “Optional”, but in the comment section, many experts stated that this practice has little value form the risk management perspective. Therefore the “Not recommended” classification seems to be more appropriate. As a consequence it does not really matter that the methodology rating is “neutral” (2.80).

**Practice 12: Create a product vision box**

This practice is another example which is not backed by the verification group. They rated it on average with 2.25 which puts it into the “optional” category, but again the rating is changed to “not recommended”. The reason for this change are again the comments. The experts agree altogether that this practice may be useful for a customer/sponsor who wants to clarify the project’s goal, but that it has little to no benefit from the risk management perspective.
5.1 Evaluation of best practices and their level of agility

5.1.3 Stakeholder management

Practice 13: *Perform a stakeholder analysis*

This practice is "highly recommended" with an average rating of 4.70. The very good avg. and the perfect applicability rating make this practice a must have for basically every project. Many of the experts also emphasized it in their comments.

**Expert 1:** "A good tool to get the worst cases from everybody who should have an opinion."  
**Expert 4:** "[translated] In my opinion, this is absolutely mandatory [...]"

The fact, that the stakeholder analysis received such a good rating, indicates the huge impact stakeholders can have on a project. Considering the fact that this practice is viewed from the risk management perspective, it suggests that stakeholders are a huge source of potential project risks some should manage actively. Since it has a "neutral" (3.40) methodology rating there is no reason not to use it in agile projects as well.

Practice 14: *Prepare important decisions*

The 3.00 applicability rating may indicate that it is not possible or not reasonable to prepare all important decisions in all projects. This may have various reasons. One of them could be that big steering committee meetings where all the big decision makers come together, are probably more common in rather traditional projects or in bigger ones.

**Expert 2** provides another one. He stated that steering committee members do usually not have enough time to discuss the content before the meeting.

But the very high effectiveness rating of 4.75 suggests that is usually pay’s off when it can be applied. Overall, it is a solid practice with an average of 3.65 and is "recommended" by the experts. The practice received a "neutral" (3.00) methodology rating, which indicates that it is independent of a specific approach. Personally, I would not be surprised to learn
that it is easier to do this in agile projects, where customer and contractor are supposed to have a closer relationship. But maybe this is not necessarily the case in reality.

**Practice 15: Actively manage changes and expectations associated with the project**

This practice may be a little blurry but its average rating of 3.93 indicates that it is addressing an important issue. Even though the experts "recommended" this practice, they made heavy use of the comment function. Some emphasized that it is especially important to talk to people when the project has a big front-end part and therefore a high "visibility". In primarily back-end focused projects this practice may be less important. Some experts also mentioned that someone should only ask for feedback when having the willingness to accept and consider it. Moreover, one of them mentioned that this practice should not be limited to end-users and operations.

The methodology rating may still be neutral (3.40), but the reasons for the tendency towards "rather traditional" would be interesting.

### 5.1.4 Communication management

**Practice 16: Perform a daily stand-up meeting**

Someone may not associate this practice with risk management, because its focus is clearly on managing the daily work. Even if it's contribution to risk management is not the biggest, the experts still "recommend" it. They probably do this because of its various indirect benefits like improving the atmosphere or because it helps to identify emerging problems early. Even though this practice became popular with the appearance of agile approaches, it's
5.1 Evaluation of best practices and their level of agility

methodology rating of 1.80 (agile) tends stronger towards "agile" then expected. I personally think that this practice can be valuable in most projects.

Practice 17: Plan your communications

This practice is a typical example of traditional project management. It is probably closer related to stakeholder management than it is to risk management, but it still helps to mitigate certain risks.

Expert 1 summarized it perfectly in his comment, by concluding with "communication is everything in an IT-Project". This is probably the reason why the experts "highly recommend" it with an average rating of 4.05. The methodology rating of 2.80 is still "neutral" but it has a small tendency towards agile approaches. This is unexpected, because it is a rather sophisticated artifact which requires a lot of information and frequent maintenance.

5.1.5 Monitoring and Control

Practice 18: Track the teams productivity

The confidence intervals for this practice are rather big, but most of the experts seem to agree that this practice is useful to identify emerging problems caused by scope creep or wrong estimations. The average rating of 3.45 classifies this as a "recommended" practice. In the end, this is a rather simple method which helps to realize when the project gets on the wrong track.

The "neutral" methodology rating of 3.20 indicates that the fact which is most important, is that someone tracks the progress at all. Especially small projects do not require extensive techniques for this. A simple burn-up chart can be sufficient to compare planned and actual progress.
5.1 Evaluation of best practices and their level of agility

Practice 19: ***Perform a post mortem analysis***

This practice only has an average rating of 2.90 which makes it "optional". This is less then expected. The reason for this rather low rating can be found in the comment section. Many experts stated that this practice may be a good thing for the individuals performing it, but that it is not very effective / efficient with regard to continuous improvement (see comments below).

![Figure 5.19: Practice 19 ratings](image)

**Expert 2:** "This is usually done (in a small group) during project touch-down. Reality shows: The same problems are repeated over and over again - individuals learn from mistakes, projects or companies do not (companies repeat the same mistakes over and over again) [translated]"

**Expert 4:** "[[... For me, the real challenge is transferring the learnings back into the company. [translated]]"

**Expert 5:** "sometimes difficult to do, because people fly of to other projects. also difficult to do right, because of trust issues."

The past paragraphs analyzed the different practices individually. To get a better impression of the overall results, figure 5.20 depicts all 4 ratings for all the collected practices. As expected, not all practices are backed by all experts, but most of them are. Only 2 practices received the "not recommended" status. This leaves a total of 17 practices, from which 6 are "highly recommended", 7 are "recommended" and 4 may depend on the situation and are therefore "optional".

As mentioned earlier, these ratings are rather subjective, but the fact that the experts rated most of the presented practices "neutral" is another strong indicator that risk management is rather independent of the applied project management approach.

To increase the soundness of these results a series of of hypotheses tests is performed on the gathered statistical data. MS Excel is used to perform the required calculations. In the first step, all the methodology ratings of the different practices were combined into a single data set. This new dataset is used to determine the relationship of the collected practices and risk management.
5.1 Evaluation of best practices and their level of agility

Figure 5.20: Comparison of best practice candidates

A t-test is used to analyze the data set which contains a total of 95 (19 x 5) methodology ratings. The next table summarizes the most important facts.

<table>
<thead>
<tr>
<th>Test type</th>
<th>t-test (two sided)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Degree of freedom</td>
<td>94</td>
</tr>
<tr>
<td>Alpha value</td>
<td>0.05 (⇒ 1-0.05 = 95% )</td>
</tr>
<tr>
<td>t-test quantiles (95%)</td>
<td>[-1.960 ; 1.960]</td>
</tr>
<tr>
<td>Mean (Methodology ratings)</td>
<td>2.9474</td>
</tr>
<tr>
<td>Standard deviation</td>
<td>0.855</td>
</tr>
<tr>
<td>Sample size</td>
<td>95</td>
</tr>
</tbody>
</table>

Table 5.2: t-test data

On this basis, 5 hypothesis tests are performed. To calculate the test result, the following formula is used:

\[
\frac{(Mean - H0)}{Standard\ deviation} \times \sqrt{Sample\ size}
\]

(5.1)

The value of H0 refers to the different levels of agility (see 5.1). Since this classification is based on intervals, the interval mean is used instead. [H0 values: Agile=1.4, Rather agile=2.2, Neutral=3.0, Rather traditional=3.8, Traditional=4.6]
In order to perform the hypothesis test, a null-hypothesis (H0) and an alternative hypothesis (H1) are required.

**H0**: Risk management benefits from approach X.  
**H1**: Risk management does not benefit from approach X.

<table>
<thead>
<tr>
<th>Methodology</th>
<th>Test result</th>
<th>Consequence</th>
</tr>
</thead>
<tbody>
<tr>
<td>Test1 (Agile)</td>
<td>17,636</td>
<td>=&gt; H0 is rejected</td>
</tr>
<tr>
<td>Test2 (Rather agile)</td>
<td>8,518</td>
<td>=&gt; H0 is rejected</td>
</tr>
<tr>
<td>Test3 (Neutral)</td>
<td>-0.5998</td>
<td>=&gt; H0 is accepted</td>
</tr>
<tr>
<td>Test4 (Rather traditional)</td>
<td>-9,7183</td>
<td>=&gt; H0 is rejected</td>
</tr>
<tr>
<td>Test5 (Traditional)</td>
<td>-18,836</td>
<td>=&gt; H0 is rejected</td>
</tr>
</tbody>
</table>

Table 5.3: Hypothesis tests

**Conclusion:**
The t-quantile values for a confidence level of 95% define the boundaries of the test value. H0 is accepted when the test value is within the acceptance range of [-1.9860 ; 1.9860], otherwise H0 is rejected. Test1, Test2, Test4 and Test5 reject the null hypothesis because the test value is outside of the defined range. The only test which supports the null hypothesis is Test3. As a consequence, someone can conclude that risk management is rather independent of the selected approach. It is important to note that the significance of this statistical analysis is limited to the practices, collected during the interviews with experienced Acesso employees.

This concludes the first part of this chapter which focused on the analysis of the collected practices. The second part of this chapter will focus on Scrum and its inherent risk management capabilities.

### 5.2 Risk management in agile environments

This section further elaborates the relationship of risk management and agile project management approaches, based on the literature research (see chapter 2) and the performed expert interviews (see chapter 3).

As mentioned in section 2.1, the main purpose of risk management is to manage potential project risks. Since Scrum was developed to address at least some of these issues, it is no surprise that the Scrum process described in section 2.2.2 contains activities and principles which support risk management. Some of them are rather implicit while others are more explicit. The next paragraphs analyze the most important ones.
5.2 Risk management in agile environments

**Scrum core values**: Scrum’s core values are a very good example of indirect risk management support. By emphasizing *Focus, Courage, Openness, Commitment* and *Respect* they help to create a good project atmosphere. When people respect each other and can discuss anything, then they are probably more willing to address potential risks as well. Due to the trusting atmosphere they do not have to fear the "shoot the messenger" scenario. Other supporting factors can be commitment and self responsibility. Since everybody commits to the defined work package during *Sprint Planning*, everybody is somehow responsible for the resulting outcome. Naturally, the motivation to address a risk is bigger, when oneself has to live with its consequences.

**The Product Backlog**: While traditional risk management mainly uses risk lists to manage and to track known risks, *Scrum* offers an additional possibility. The *Product Backlog* is basically a prioritized list of tasks, which are still awaiting completion. These tasks are not necessarily development specific, hence someone could also integrate and manage identified risks within the *Product Backlog* [Mor14]. To ensure a better overview, it may be a good idea to create a separate risk Backlog. Besides the storage and management of known risks, the Backlog has another advantage which is a result of the *Backlog Grooming*. The *Backlog Grooming* activity focuses on re-prioritization of the various items contained within the *Product Backlog*. When performing this activity the *Product Owner* can also re-evaluate the known project risks and change their priority if necessary. Thus, risks which are integrated into the *Product Backlog* are less likely to be ignored or forgotten.

In the end it does not make a lot of difference whether the risks are stored in a backlog or e.g. within an Excel sheet. Both ways are very similar, whoever is responsible for the risk management activities should do as he or she prefers. In the end, someone only has to make sure that the risks are not forgotten and re-evaluated regularly.

**The iteration**: The concept of iterations or *Sprints* in *Scrum* works actually very well with risk management, because the risk management process itself is also a rather iterative process. Once risks are identified, they are analyzed and evaluated. Based on the results, management steps are initiated to deal with the risks. Somewhat later, the whole process starts over again. The major reason for this is the fact that it is unlikely that someone can identify all potential project risks at the beginning of a project. Possible reasons are missing knowledge, a lack of information or that certain events simply have not occurred yet. While the project progresses, more information become available, e.g. from other
5.2 Risk management in agile environments

project management areas. At the same time new risks can arise as well. By repeating the
different steps frequently, the risk management process uses the most recent information,
thus ensuring that emerging issues and information are considered.
The iteration, one of the agile core principles allows the incremental delivery of product
parts which can be evaluated by the customer at an earlier stage. As a consequence the
received feedback can be used to influence and to improve the final product. The constant
evaluation and adaption implicitly reduces the risk that the customer does not like the final
product. This process also indirectly shares the risks with other stakeholders, because all of
them were given the chance to provide feedback on the different increments.

Besides increasing feedback, the end of an iteration is also a very good point to evaluate
the project’s current status. Based on the outcome of this evaluation, decisions about the
project’s future can be made. In the case of insufficient progress or results, the project can
easily be canceled before more money is wasted. This course of action is very similar to the
one recommended by Barry Boehm [BLKT14], who is an advocate of phase based projects.
The whole idea about phase based projects is that you split a project into multiple phases
and after each phase, e.g. after someone finished the feasibility study or created a prototype,
you evaluate whether it makes sense to fund the next phase or to abort the project. Barry
Boehm concludes that this approach can help so safe money, which may be burnt otherwise.
Moreover, this is probably the fairest approach for contractor and customer, because either
of them could abort the project after completing a phase. Unfortunately, this approach
makes budgeting less predictable and this is probably one of the reasons why many sponsors
prefer fixed price projects instead.

The Daily Scrum: The Daily Scrum can be considered to be another risk management
activity. An Accso internal study [Vos13] indicated that the team’s internal communication
is one of the most important factors for a project’s success. Without good communication,
a lot of problems can arise.
The risk of bad communication can be effectively reduced by the Daily Scrum, because it
ensures that team members meet and talk on a daily basis.
Every day, all team members gather in a room to talk about their completed work, their
plans for the next day and about possible problems. As a consequence, everyone of them has
a pretty good idea of what the others are doing and what problems they are facing. As a re-
result, they may be able to help each other more easily. This frequent exchange of information
also improves the projects transparency / openness, which is an important Scrum core value.
The Task Board: The Task Board also contributes to transparency/openness, just like the Daily Scrum. The standard version is about visualizing the projects status, but with some modifications it can also become be a valuable asset for risk management.
When thinking about it, multiple modifications come to mind, e.g. enriching it with risk management information [Mor14]. In order to do so, someone could print and pin the projects agility profile, the risk map, the risk burn-up chart or other rather visual and abstract artifacts next to the Task Board. That way, all people can quickly asses the current project or Sprint status and learn about the known risks as well. This is especially effective, since the Task Board is something most team members look at on a daily basis.
Another possibility is the extension of user stories. Someone could add additional symbols to those story cards in order to indicate associated risks, dependencies or other problems [Mor14]. This increases the awareness of people who are working on this specific user story. These things are also reflected in the best practice "Integrate risk specific symbols into your Task board".

The Sprint Planning: The beginning of an iteration or Sprint is very suited for risk management activities. Because knowing about potential risks is the most important benefit from risk management [Car13]. Since Sprint Planning uses the latest available information to create and to refine the plan, it is a good moment to identify new risks for the upcoming sprint. Since a Sprint is rather short, someone should focus on the risks which are associated with the planned set of features. Long term risk management is probably more effective when performed on the Product Backlog or release planning level.

The Sprint Review and the Sprint Retrospective: The Sprint Review and the Sprint Retrospective are great ways to reflect on the completed Sprint. Both are a kind of post mortem analysis, which is usually performed at the end of the project. By identifying strengths and weaknesses at the end of a Sprint, future Sprints can benefit from the gained insights. The insights can also help to improve Scrum process itself and reduce the likeliness of repeating the same mistakes in the future.

5.3 Summary of analysis results

The introduction to Scrum and the second part of this chapter showed that Scrum does not define any explicit risk management specific measures. Instead, the topic of risk management is addressed more indirectly by the structure of the framework. Even though Scrum has these
indirect risk management "activities", which mitigate certain risks "automatically", this is not enough to replace "real" risk management.

As indicated by the second part of this chapter, a total of the 13 practices received a neutral rating. Only 2 practices were rated agile, 2 others are rather agile and the remaining 2 practices belong to the rather traditional category. This supports the initial conclusion that within typical Accso projects, risk management is rather independent of a specific approach. As a consequence most practices can be used in Scrum as well to improve its existing risk management capabilities. This also includes the "proven" traditional methods.

This procedure may create some sort of gray area between agile and traditional approaches, but as in section 2.5 indicated this is already the case in most Accso projects.

It is certainly not reasonable to claim that this conclusion has general validity. First of all, as mentioned several times before, all results are subjective. Another reason why risk management and Scrum seem to fit together so well may be due to the fact that typical Accso projects have between 5 and 10 team members and a duration of 6 to 12 months. Especially the team size is basically identical to what Scrum recommends. As a consequence, many of the extensive and heavy weight methods described in certain risk management books were out of scope.

Even though the practices turned out to be rather independent of the chosen approach, timing may still be an issue. While traditional projects are rather sequential, Scrum has multiple nested cycles with different durations. Therefore timing deserves a more closer look. Some common cycles and typical durations in agile approaches are:

- A single workday
- The sprint (1-4 weeks)
- The release cycle (4-12 weeks)
- The projects duration (6 to 12 months)

When thinking about the collected practices, it is obvious that not all of them are performed in the same cycle or in the same phase. Some of them are performed multiple times, while others are only performed once. An activity like creating the proposal is performed only once. It is more intense and usually has a much higher impact on the project than activities which are performed every day. Thus, someone can conclude that the intensity and the impact of most activities is also related to their cycle time [Mor14].

Keeping this in mind, figure 5.21 shows how the different practices can be integrated into a typical project life cycle.
Figure 5.21: Practices in the project life cycle

The figure refines the life cycle from the top to the bottom. The black lines indicate to which part the next level of abstraction refers too. The phases are represented by the various boxes and the numbers refer to the different practices.

The additional arrow on some practices indicates that they should be performed frequently within their phase. This was done to avoid duplications and clutter.

This summary section concludes the analysis chapter. The only thing left is an overall conclusion on this thesis.

Chapter 6

Conclusion

This thesis evaluated existing literature on risk management and agile methodologies. This was done to learn about best practices, potential project risks and the relationship of risk
management and agile methodologies. To identify best practices and potential project risks, a series of interviews with experienced project managers was performed. The gathered data was analyzed by using some concepts from grounded theory. This approach resulted in the best practice candidates and the generic risk list, in which each risk is rated with regard to likeliness and impact.

The defined practices were verified by a second group of experts. The experts rated the practices with regard to applicability, effectiveness, and efficiency. Moreover, they specified whether a presented practice benefits more from an agile or a traditional project management approach. The analysis of the data from the verification survey and the performed hypothesis tests confirmed the initial theory that risk management is rather independent of the selected project management approach, at least within typical Accso projects. Therefore, it makes sense to use "proven" practices, including traditional ones, in agile projects as well. They can help to improve existing risks risk management capabilities.

The verification data was also used to classify the different practices from "highly recommended" to "not recommended". To ease the usability, all practices as well as some additional information were moved into a separate document, the practice catalog. This catalog and the list of potential project risks can be used by every Accso employee who is interested in risk management.

The previous paragraph summarized the different chapters, deliverables and conclusions of this thesis. Besides this main conclusion, there is another key learning.

A lot of risk management seems to happen rather indirectly. Meaning that experienced project managers simply perform a lot of tasks which help to mitigate risks, even though they do not explicitly think about risk management when doing them. While talking to them I realized that experience as well as empathy and some knowledge of human nature are very influential factors in risk management. Probably, because they help to quicker assess people, the project's specific context and the circumstances.

Since a project manager usually does not start off with a lot of experience, the generic risk list and the proposed practice catalog should be valuable resources to get started with risk management.
A.1 Expert identification survey

Accso expert survey

This survey is performed by Alexander Eiermann. It is a part of my masters thesis about "Best practices for risk-, scope- and stakeholder management, in traditional and especially agile projects."

Notes:
- This survey takes ~5 min to complete
- The results of this survey are not public and can only be accessed by myself.
- The purpose of this survey is to identify potential interview candidates

*a Erforderlich

General questions

Please enter a name/pseudonym of your choosing.*
Please remember your chosen pseudonym for further inquiries.

For approximately how many years do you work in software development? *

How many years of (traditional) project management experience do you have? *
(In the area of software development)

How many years of (agile) project management experience do you have? *
(In the area of software development)

What was your ratio of traditional and agile projects in the past 5 years? *
(If you work for less than 5 years in the software industry, that's fine too.)

☐ Only agile
☐ Mostly agile
☐ Balanced
☐ Mostly traditional
☐ Only traditional
A.1 Expert identification survey

In which roles did you work in past projects?
- Requirements Engineer
- Developer
- Project Manager
- Product Owner / Sponsor
- Technical Architect
- Tester
- Others: __________________________

How familiar are you with the following methodologies? *

<table>
<thead>
<tr>
<th>Methodology</th>
<th>Not at all</th>
<th>A little</th>
<th>Moderately</th>
<th>Very familiar</th>
</tr>
</thead>
<tbody>
<tr>
<td>PMI PRINCE2</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>XP</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Scrum</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Kanban</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>DSDM (Dynamic</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Systems Development</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Method)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>RUP (Rational</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Unified Process)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SAFe (Scaled</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Agile Framework)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>DAD (Disciplined</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Agile Delivery</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Risk management specific questions

How would you rate your overall risk management knowledge/experience? *

None  ☒  ☒  ☒  ☒  ☒  Very good

How would you rate your capability to perform the following risk management tasks? *

<table>
<thead>
<tr>
<th></th>
<th>None</th>
<th>Low</th>
<th>Moderately</th>
<th>Good</th>
<th>Very good</th>
<th>Don't know</th>
</tr>
</thead>
<tbody>
<tr>
<td>Risk analysis</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Risk assessment</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Risk mitigation / reduction</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

How familiar are you with the following terms and their resulting implications for risk management? *

<table>
<thead>
<tr>
<th></th>
<th>Not at all</th>
<th>A little bit</th>
<th>Moderately</th>
<th>Familiar</th>
</tr>
</thead>
<tbody>
<tr>
<td>Technical dept</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Cost of delay</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

How would you rate your capability to create the following risk management artifacts? *

<table>
<thead>
<tr>
<th></th>
<th>None</th>
<th>Low</th>
<th>Moderately</th>
<th>Good</th>
<th>Very good</th>
<th>Don't know</th>
</tr>
</thead>
<tbody>
<tr>
<td>Risk list (including probability)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Risk management</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Risk / Uncertainty diagrams</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>(probability distributions)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Decision</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Risk / Uncertainty diagrams</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Which risk management artifacts are you missing in the list above?
## Scope management specific questions

### How would you rate your overall scope management knowledge/experience? 

<table>
<thead>
<tr>
<th></th>
<th>None</th>
<th>Low</th>
<th>Moderately</th>
<th>Good</th>
<th>Very good</th>
<th>Don't know</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

None ☐ ☐ ☐ ☐ ☐ Very good

### How would you rate your capability to perform the following scope management tasks? * 
(Assume that the requirements are already known and that you are familiar with the domain.)

How to define a scope definition. (What is in/out?)

<table>
<thead>
<tr>
<th></th>
<th>None</th>
<th>Low</th>
<th>Moderately</th>
<th>Good</th>
<th>Very good</th>
<th>Don't know</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Tracking the project's progress.

<table>
<thead>
<tr>
<th></th>
<th>None</th>
<th>Low</th>
<th>Moderately</th>
<th>Good</th>
<th>Very good</th>
<th>Don't know</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Managing scope.

<table>
<thead>
<tr>
<th></th>
<th>None</th>
<th>Low</th>
<th>Moderately</th>
<th>Good</th>
<th>Very good</th>
<th>Don't know</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Scope control: (adjusting the scope if necessary)

### How would you rate your capability to create the following scope management artifacts? * 
(Assuming you have the required domain knowledge.)

How to create a work breakdown structure.

<table>
<thead>
<tr>
<th></th>
<th>None</th>
<th>Low</th>
<th>Moderately</th>
<th>Good</th>
<th>Very good</th>
<th>Don't know</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Domain models.

<table>
<thead>
<tr>
<th></th>
<th>None</th>
<th>Low</th>
<th>Moderately</th>
<th>Good</th>
<th>Very good</th>
<th>Don't know</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Product backlog.

<table>
<thead>
<tr>
<th></th>
<th>None</th>
<th>Low</th>
<th>Moderately</th>
<th>Good</th>
<th>Very good</th>
<th>Don't know</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Burn down chart.

<table>
<thead>
<tr>
<th></th>
<th>None</th>
<th>Low</th>
<th>Moderately</th>
<th>Good</th>
<th>Very good</th>
<th>Don't know</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Product vision.

<table>
<thead>
<tr>
<th></th>
<th>None</th>
<th>Low</th>
<th>Moderately</th>
<th>Good</th>
<th>Very good</th>
<th>Don't know</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### Which scope management artifacts are you missing in the list above? 
(please separate your answers with line breaks.)

Please separate your answers with line breaks.
A.1 Expert identification survey

Stakeholder management specific questions

How would you rate your overall stakeholder management knowledge/experience *

1 2 3 4 5

None ☐ ☐ ☐ ☐ ☐ Very good

How would you rate your capability to perform the following stakeholder management tasks? *

<table>
<thead>
<tr>
<th>Stakeholder management tasks</th>
<th>None</th>
<th>Low</th>
<th>Moderately</th>
<th>Good</th>
<th>Very good</th>
<th>Don't know</th>
</tr>
</thead>
<tbody>
<tr>
<td>Stakeholder identification</td>
<td>☐</td>
<td>☐</td>
<td>☐</td>
<td>☐</td>
<td>☐</td>
<td>☐</td>
</tr>
<tr>
<td>Stakeholder prioritization</td>
<td>☐</td>
<td>☐</td>
<td>☐</td>
<td>☐</td>
<td>☐</td>
<td>☐</td>
</tr>
<tr>
<td>Negotiation</td>
<td>☐</td>
<td>☐</td>
<td>☐</td>
<td>☐</td>
<td>☐</td>
<td>☐</td>
</tr>
<tr>
<td>Conflict resolution (escalation)</td>
<td>☐</td>
<td>☐</td>
<td>☐</td>
<td>☐</td>
<td>☐</td>
<td>☐</td>
</tr>
</tbody>
</table>

How would you rate your capability to create the following stakeholder management artifacts? *

<table>
<thead>
<tr>
<th>Stakeholder management artifacts</th>
<th>None</th>
<th>Low</th>
<th>Moderately</th>
<th>Good</th>
<th>Very good</th>
<th>Don't know</th>
</tr>
</thead>
<tbody>
<tr>
<td>Stakeholder register</td>
<td>☐</td>
<td>☐</td>
<td>☐</td>
<td>☐</td>
<td>☐</td>
<td>☐</td>
</tr>
<tr>
<td>Personas</td>
<td>☐</td>
<td>☐</td>
<td>☐</td>
<td>☐</td>
<td>☐</td>
<td>☐</td>
</tr>
<tr>
<td>Communication plan</td>
<td>☐</td>
<td>☐</td>
<td>☐</td>
<td>☐</td>
<td>☐</td>
<td>☐</td>
</tr>
</tbody>
</table>

Which stakeholder management artifacts are you missing in the list above?

(Please separate your answers with line breaks.)

.
# List of abbreviations

<table>
<thead>
<tr>
<th>Abbreviation</th>
<th>Full Form</th>
</tr>
</thead>
<tbody>
<tr>
<td>Accso</td>
<td>Accelerated Solutions GmbH</td>
</tr>
<tr>
<td>BABOK</td>
<td>Business Analysis Body of Knowledge</td>
</tr>
<tr>
<td>DAD</td>
<td>Disciplined Agile Delivery</td>
</tr>
<tr>
<td>DSDM</td>
<td>Dynamic Systems Development Method</td>
</tr>
<tr>
<td>PM</td>
<td>Project manager</td>
</tr>
<tr>
<td>PMBOK</td>
<td>A Guide to the Project Management Body of Knowledge</td>
</tr>
<tr>
<td>PMI</td>
<td>Project Management Institute</td>
</tr>
<tr>
<td>PRINCE2</td>
<td>PRojects IN a Controlled Environment, version 2</td>
</tr>
<tr>
<td>RUP</td>
<td>Rational Unified Process</td>
</tr>
<tr>
<td>SAFe</td>
<td>Scaled Agile Framework</td>
</tr>
<tr>
<td>SLA</td>
<td>Service level agreement</td>
</tr>
<tr>
<td>TDD</td>
<td>Test Driven Development</td>
</tr>
<tr>
<td>WBS</td>
<td>Work Breakdown Structure</td>
</tr>
</tbody>
</table>
Bibliography


[MS14] Meyer-Stabley, Bertrand: Agile! The good, the hype and the ugly. Springer International Publishing Switzerland, 2014


77
BIBLIOGRAPHY


[Vos15] Voss, Markus: BoST framework. 2015
