Author(s): Tuunanen, Tuure; Vartiainen, Tero; Kainulainen, Sanna; Ebrahim, Mehdi

Title: Development of an Agile Requirements Risk Prioritization Method : A Design Science
Research Study

Year: 2023

Version: Published version

Please cite the original version:

Tuunanen, T., Vartiainen, T., Kainulainen, S., & Ebrahim, M. (2023). Development of an Agile
Requirements Risk Prioritization Method : A Design Science Research Study. Communications of
the Association for Information Systems, 52, 609-637. https://doi.org/10.17705/1cais.05226

6-1-2023

Development of an Agile Requirements Risk Prioritization Method: A Design Science Research Study

Tuure Tuunanen
Faculty of Information Technology, University of Jyväskylä Department of Informatics and Media Uppsala University

Tero Vartiainen
School of Technology and Innovations, University of Vaasa

Sanna Kainulainen
Faculty of Information Technology, University of Jyväskylä

Mehdi Ebrahim
Advanced Management Systems, Ltd.

Follow this and additional works at: https://aisel.aisnet.org/cais

Development of an Agile Requirements Risk Prioritization Method: A Design Science Research Study

Tuure Tuunanen

Faculty of Information Technology,
University of Jyväskylä
Department of Informatics and Media
Uppsala University

Tero Vartiainen

School of Technology and Innovations,
University of Vaasa

Sanna Kainulainen

Faculty of Information Technology,
University of Jyväskylä

Mehdi Ebrahim

Advanced Management Systems, Ltd.

Abstract:

The practice of information systems development (ISD) has changed during the past two decades from very structured approaches to agile ISD methods. However, in the literature many methods available for managing requirements-related risks follow a structured approach to ISD. Few, if any, available methods offer solutions that prioritize requirements risks for agile ISD projects based on recognizing requirements-related risks and patterns in order to mitigate these. To fill this gap in the literature, we apply the design science research methodology to develop an agile requirements risk prioritization method together with industry experts (n=54) in Finland and New Zealand in a multi-year study. The method was developed by applying contingency theory, and our study makes an artifactual contribution to the literature. The method helps practitioners prioritize the overall requirements-related risks for ISD projects.

Keywords: Information Systems Development, Agile Requirements Risk Prioritization Method, Design Science Research.

1 Introduction

Today, information systems development (ISD) practices can be characterized as agile (Chen, 2015). However, the perceived view in the literature emphasizes the structured method for requirements risk management, see, for example, Hickey and Davis (2004) or Mathiassen et al. (2007). Our paper applies design science research (DSR) (Hevner et al., 2004; March & Smith, 1995; Walls et al., 1992; Walls et al., 2004) to develop an agile requirements risk prioritization method to address this gap in the literature. Below, we provide an argument for the need for this method, our research objective and question, and finally, a description of how we set out to accomplish this task.

During the 1970s and 1980s, plan-driven structured methods, such as Waterfall (Royce, 1970), were typical. Consequently, risk management has been studied mostly from the point of view of structured ISD methods. For example, in the early 1980s, the focus was on managing project portfolios and using different techniques to resolve requirements risks (Alter & Ginzberg, 1978; Davis, 1982; McFarlan, 1981). Later on, in the 1990s, researchers examined what risks ISD projects entail (Barki et al., 1993; Saarinen & Vepsäläinen, 1993) and which risks are more important at certain phases of the ISD project (Lyytinen et al., 1998). More recently, the focus has shifted to assisting practitioners in applying different requirements development techniques to mitigate requirement-related risks for a project (Hickey & Davis, 2004). For this purpose, Mathiassen et al. (2007) developed a contingency model to match developers and analysts with the requirements development techniques that fit the risk situations (Barki et al., 2001) they face in a project.

Later in the 2000s, there was a strong practitioner-led movement from plan-driven methods toward more agile ISD methods (Barlow et al., 2011). Plachkinova et al. (2018) suggested that the agile ISD has more seamless transitional phases for the development process than plan-driven structured approaches, like the Waterfall method. Short cycle times and quick releases are typical in these agile methods as there is a need for change due to rapid changes in the business environment (Truex et al., 1999). Consequently, ISD rarely follows a traditional life cycle that terminates the information systems (IS) anymore; instead, the IS are preserved and redeveloped to satisfy users' needs: consider, for example, Facebook, which is constantly changing its digital offerings. Thus, for our study, we only distinguish an overall life cycle iteration in the development process and that the main elements of such a life cycle are related to requirements, design, and implementation.

As the industry has adopted the agile ISD methods, interest in the agile way of doing requirements elicitation and analysis increased. Matook and Maruping (2014), for example, highlighted the increased need to manage the changes in agile ISD projects. Matook and Maruping (2014) argued that prioritizing requirements is one crucial way of managing requirements-related risks in agile ISD projects. Racheva et al. (2010) recognized a lack of understanding of how agile requirements prioritization happens in practice. This change to agile ISD in the industry also means that traditional ISD values, such as long IS life span, concise specifications, and complete systems analysis, are more outmoded. Instead, there should be an emphasis on continuous analysis, negotiated requirements, and maintenance activities (Truex et al., 1999). Therefore, we propose that requirements risk prioritization should also be done accordingly. However, the extant literature proposes that requirements risks should be assessed with appropriate intervals (e.g., Mathiassen et al., 2007; Lyytinen et al., 1998) and follow the traditional plan-driven structured ISD methods.

We argue that further emphasis should be given to the changing nature of ISD and what impact this changing nature has for requirements risk prioritization. Accordingly, our research question is: How should requirements risk prioritization be done in agile ISD? We propose that applying such a method in agile ISD would impact the overall efficiency of ISD and requirements risk prioritization efficacy. We recognize that general-purpose risk prioritization methods exist in the literature (e.g., Mathiassen et al. (2007) or Hickey and Davis (2004)). However, we see that recent changes in the industry practices warrant an investigation of 1) how industry practitioners see today's requirements risks and 2) whether we can develop a method for agile requirements risk prioritization. We worked with experienced industry practitioners (n=54) in Finland and New Zealand who manage ISD projects for their daily work.

The remainder of the paper is structured as follows. First, we review relevant literature on ISD risk management and agile requirements risk prioritization. Then, we depict the applied DSR approach, and different qualitative research methods, to develop our method. Then, we present the developed artifacts

and the demonstration and evaluation process we used with our industry participants. Finally, we discuss the implications of our work and conclude the paper.

2   Managing ISD Project Risks and Moving Towards Agile Requirements Prioritization

The following sections review the literature on managing ISD risks in general. Thereafter, we review the literature on agile in requirements risks management and, finally, we review how contingency theory offers a solution to prioritize requirements according to the identified risks.

2.1   Background

ISD risk management typically starts with identifying the project risks to control and mitigate them. There have been numerous studies that have put forward lists of risk items for this purpose. Barki et al. (1993), for example, proposed a risk measurement method that incorporates the uncertainty that projects are exposed to and the impact that project failure could have. Boehm (1991) introduced risk management as a two-step process involving assessment and management, which involves identification, prioritization, resolution monitoring, etc. Later, Keil et al. (1998) conducted a Delphi survey that identified risk items that affect ISD projects and proposed a framework to indicate the relative importance of each risk item. Schmidt et al. (2001) also conducted another Delphi study that developed a ranked list of risks affecting projects.

Barki et al. (1993) and Boehm (1991) proposed risk measurement tools for overall projects. However, their research did not specify how to interpret the resulting risk exposure. To respond to this, Tiwana and Keil (2004) proposed a one-minute measurement tool that would allow project managers to determine the current risk exposure of a project, varying from high risk to low risk. The measurement tool was based on risk drivers resulting from the examination of 720 projects. Like Keil et al. (1998), the risk drivers were classified based on the project managers' level of control over them. Wallace et al. (2004) later described how the different categories of risks varied for different types of projects, such as high-, medium-, and low-risk projects. Accordingly, the extant literature has approached risk measurement to identify the risks or main risk components that a project can be exposed to and their effect. For example, the approach adopted by Tiwana and Keil (2004) is consistent with how risk level perceptions are determined. Earlier, Florig et al. (2001) proposed a similar framework for ranking risks. Their framework called for categorizing the risks and utilizing a panel to rank the risk items. The framework proposed by Florig et al. (2001) also calls for the panel of participants to be provided with a summary/description of the risks.

The more recent studies based on control theory (Keil et al., 2013; Dreesen et al., 2020; Remus et al., 2020) suggest that the use of different elements of control (control modes and control styles) can benefit ISD projects. Keil et al. (2013) found that user and requirement risks reduce the positive influence of controls on process performance. According to the authors, this means that to ensure good process performance, solid controls need to be implemented. However, solid controls do not guarantee good process performance. Kremser and Blagoev (2021) found that when actors prioritized multiple routines (interdependent actions involving various actors), they mainly focused on a single person's patterns of actions. Kremser and Blagoev (2021) developed a process model of temporal coordination that explains how actors iteratively use three prioritizing mechanisms, sequence-, timing-, and role-based prioritizing when confronting temporal conflicts in their work. Geiger et al. (2021) studied how routines are coordinated among firefighters when there are high levels of temporal uncertainty, situations when the timing of critical events is unknown, and risks concerning misalignments. These findings are promising for risk management research, but these focus more on team member-related risks than others.

2.2   Agility in Managing Requirements Risks

Over the past two decades, ISD companies have adopted agile approaches to cope with change and mitigate business risks. Sebastian et al. (2017) argue that agile or development and operations (DevOps) methods are essential when developing digital service platforms. As service and customer requirements constantly change and new market opportunities emerge quickly, digital services need to be continuously updated. While IS research and practice have concentrated on agile ISD, requirement engineering research has also increased attention to agile requirements development (i.e., eliciting, analyzing, and specifying requirements). Furthermore, the literature shows how socio-technical approaches need to

replace the technical approaches in agile requirements development. The following studies show how researchers attempt to develop solutions for this.

Bergman et al. (2002) argued that existing requirements development techniques are based on the false assumption that political and business problems can be separated from technical solutions. To resolve this, they proposed that requirements should be developed as mappings between technical solutions and business problems, and developers need to iterate while developing requirements. Napier et al. (2006) studied bridging plan-driven and agile approaches for requirements development. They used the concepts of repeatability and response-ability. In repeatability, requirements are derived through specification, and changes in requirements are exceptions (plan-driven), whereas in response-ability, requirements are negotiated, and changes are expected (agile). They concluded that combining these two approaches is the most useful approach.

Furthermore, Matook and Maruping (2014) determined that providing, clarifying, and prioritizing requirements are important in agile projects. Customer representatives need to collaborate closely (e.g., in face-to-face meetings) with the development team to manage the changes. Alsaqaf et al. (2019) studied the challenges of requirements development in large-scale distributed agile ISD projects. Their study revealed several major risks in agile requirements development and identified existing practices to mitigate them. For example, for the "conceptual challenges of quality requirements," they recognized the mitigating practice of "maintain an assumption wiki-page." Tuunanen and Kuo (2015) studied how culture affects requirements prioritization. They proposed a value-based requirements prioritization approach to better understand features perceived as more important in different cultural settings. According to the authors, this approach contributes to the more efficient use of agile methods such as SCRUM. Tuunanen and Peffers (2018) developed a design theory, population-targeted requirements acquisition, for the effective use of requirements acquisition techniques. They designed five principles for action for adapting and using existing techniques to enable effective requirements acquisition activities with populations.

Thus, while the literature recognizes the importance of agile risk management for ISD projects (see, e.g., Dorofee et al., 1996; Persson et al., 2009) or software process improvement (see, e.g., Iversen et al., 2004), there seem to be few specific methods available for agile requirements risk management. One such method was developed by Racheva et al. (2010). It is a conceptual model for understanding the inter-iteration prioritization process regarding inputs and outcomes. Their model focused on understanding the value offered by different requirements and using this knowledge to prioritize the requirements in an ISD project. However, their model abridged risks to a single category of implementation-related problems for the requirements.

2.3 Contingency Model as a Solution Approach

The contingency model for requirements development by Mathiassen et al. (2007) does focus on requirements-related risks by categorizing them into identity (the availability of requirements), volatility (the stability of requirements), and complexity of requirements. Their model also specifies that the project's risk level (high, medium, low) is based on its type of requirement risk. They provide risk resolution techniques to resolve risks in different project settings. This approach follows the general contingency model introduced earlier by Barki et al. (2001), which looks at the risk exposure of an ISD project and risk management methods' fit to improve project performance.

Mathiassen et al. (2007) characterized risks related to requirements identity for situations where there is a communication gap between developers and would-be users, such as developing generic applications or mass-market software (Regnell et al., 2001). Furthermore, requirements identity risks relate to the availability of requirements; high risk means that requirements are unknown or indistinguishable. Second, Mathiassen et al. (2007) emphasized requirements volatility as another key risk in software development (Davis, 1982; Lyytinen, 1987). Requirements volatility relates to the stability of requirements (Davis, 1982; Mathiassen et al., 1995); high risk indicates that requirements change easily. Finally, Mathiassen et al. (2007) point out that requirements complexity is a key risk in requirements and software development (e.g., Brooks, 1987). Requirements complexity measures how easy requirements are to understand (Mathiassen et al., 1995), with high risk indicating that requirements are difficult to understand, specify, and communicate. Consequently, Mathiassen et al. (2007) categorize requirements risks into identity, volatility, and complexity groups. We follow this grouping and use the term requirements risk item to describe specific requirements-related risks.

Earlier, we argued that the ISD practice has moved to a more agile approach. Furthermore, the literature does not offer many specific agile requirements risk prioritization methods, but it does provide foundations for developing such. The contingency model proposed by Mathiassen et al. (2007) provides a theoretically firm ground for developing such a method. We were also inspired by Keil et al.'s (1998) study, which applied a Delphi survey to identify and rank the most important software development risks. In the following sections, we present the chosen research methodology and then explain how we developed the method in cooperation with the industry.

3   Research Methodology

In a multi-year study, we employed DSR (Hevner et al., 2004; Peffers et al., 2018) as the research approach to develop the method. We applied the design science research methodology (DSRM) (Peffers et al., 2007) to conduct our research. The DSRM starts with identifying the research problem(s) and the motivation for the research. Based on the evidence, reasoning, and inference, the process continues toward defining the objectives of a solution to solve the research problem. This process should be based on prior knowledge in the given field of research. This knowledge is then used to design and develop an artifact evaluated for effectiveness or efficiency. This approach eventually leads to disciplinary knowledge. To develop the artifact in three DSR cycles, we applied different qualitative research methods (depicted in the next section and summarized in Table 1) in each DSR cycle in cooperation with industry experts (n=54).

3.1   Study Design

First, we applied a focus group interviewing method to identify risk themes and the underlying risk items. Organizations commonly use focus groups (e.g., to conduct marketing research to collect opinions from customers about a given company's products and services) (Morgan, 1997; Stewart et al., 2007). The strength of the focus group method is that it enables the focus group moderator (researcher) to gather a homogeneous group of participants who can easily relate to each other due to similar backgrounds and expertise in a particular field and discuss a topic interesting to the organizational unit (Morgan, 1997). We invited participants by email from, but not limited to, the following titles: project manager, IT manager, application systems developer, product manager, business analyst, and consultant. We included participants from three major New Zealand cities: Auckland, Wellington, and Christchurch (n=17). To recruit qualified participants more effectively and promptly, we approached two professional membership associations and requested to circulate the research invitation to their members. Software New Zealand and Project Management Institute (PMI) are these two associations. Finally, two specific subgroups within the PMI network were utilized to narrow our search scope for qualified participants. They were Project Management Institute IS Special Interest Group (PMI-ISSIG) and Project Management Institute Risk Management Special Interest Group (Risk-SIG).

Second, we did a Delphi survey to identify and assess further the risk items. The Delphi method has been previously used in IS research by Akkermans et al. (2003), Schmidt et al. (2001), and Keil et al. (1998), for example. A Delphi study involves gathering opinions from a panel of experts in a particular field and trying to achieve a level of consensus from the panel (Linstone & Turoff, 2002). Prior Delphi survey studies reported in the IS literature have been conducted with panel sizes of 23, 10, and 11 (Akkermans et al., 2003; Keil et al., 1998). For this study, fifteen (n=15) was an ideal number of panel members based on their work experience. The members had a minimum of five years of experience and had performed one of the following roles: project manager, business analyst, or technical architect. On average, the work experience on the panel was 16 years. Panel members had also undertaken various roles in their industry, including programming, development manager, information technology (IT) manager, delivery manager, integration consultant, and team leader. Some of the panel members had started as programmers and changed roles to become business analysts and/or project managers over the years. Panel members also included professionals who had developed and delivered projects using various ISD methods, including agile. Lastly, a free-marginal kappa was used to determine the level of agreement among panel members. A free-marginal kappa was selected because the panel members were to choose a particular proportion response, be it risk impact level or the phase affected by a particular risk item (Brennan & Prediger, 1981; Randolph, 2005). A free-marginal kappa was calculated using the online calculator provided by Randolph (2008).

Third, we conducted an interview study with three companies in Finland during the year 2017, and in this study the method was evaluated. The study organizations varied from a company that operates as a

business-to-business (B2B) conference organizer to a large company operating in the transportation industry to a medium-sized software developer company. Interviewees were industry and project professionals working on agile ISD projects. We used semi-structured and theme-centered interviews (n=22) to map the views and experiences of the project professionals. Next, we describe the applied research methods.

Table 1. Method Development, Demonstration, and Evaluation Process

Cycle	Phase	Objective	Method	Developed Artifact	Evaluation Outcome
DSR Cycle #1	Method Design	Identify requirements risk themes and their underlying risk items and the level of impact for each requirement risk item while adding possible new items by applying focus group interviews and analysis	Focus group method	Weighted requirements risk items	
	Demonstrate and Evaluate	Assess and find consensus on the impact level of requirements risk on ISD projects with the PMI	Delphi method		Evaluation and improvement of risk items and assessment of impact levels
DSR Cycle #2	Method Design	Identify and assess ISD phases most likely affected by each requirements risk item	Delphi method	Requirements Risk profiling	
	Demonstrate and Evaluate	Find consensus on ISD phases most likely affected by the PMI	Delphi method		Evaluation and improvement of the risk profiling tool
DSR Cycle #3	Method Design	Generation of checklists to be used with the method	Delphi method	Requirements risk checklists	
	Demonstrate and Evaluate	Evaluate the developed method with an interview study with three companies for its applicability and utility	Thematic analysis		Proof-of-value for the utility of the method and its applicability to industry projects

3.2 Focus Group Data Collection and Analysis

3.2.1 Participant Recruitment

An email invitation that described the research objectives was sent to members through the software New Zealand (NZ) and Project Management Institute (PMI) networks to obtain participants for this research. The same invitation message was also included in the regular e-newsletter published through internal networks. There were about 2,228 recipients in total. One week after the mailing of the first invitation, a second email invitation was also re-sent to the same recipients. The purpose of this repeated step was to promote a higher participation rate.

Once we had received emails with expressions of interest, we immediately contacted those senders by phone and email and asked questions to see if they would qualify for our research. This was to ensure higher contribution values of the participants to the research outcomes. We asked questions about their work experience in software/IS projects, more importantly, what they had done or were doing in relation to requirements development. To qualify as a participant, they would need to tell us about the projects they had been involved with in the past, explain the key challenges faced with requirements development, and articulate them in detail. We evaluated each participant based on the quality of their responses. Figure 1 summarizes the participant recruitment procedure.

We received 47 emails with interest and inquiries about this research two weeks after mailing the initial invitation. We assessed the profiles and background information given by each participant individually. This ensured they were qualified participants who would contribute quality input to this research. However, after screening all potential participants' profiles, we discovered that only 35 of them matched the

background and experience requirements listed in the project description and were also willing to participate in our research. We then sent another email to those 35 potential participants asking about their current residence and schedule for the next few weeks. Some of the responses, however, needed to be disregarded. After scanning through all feedback emails, we discovered that three senders were based in the United States, and another was based in the United Kingdom. Furthermore, four respondents were considered qualified but could not attend a session in one of the selected cities because they were based in other geographical areas in NZ. Another two eligible respondents could not participate within the chosen time because they had other commitments.
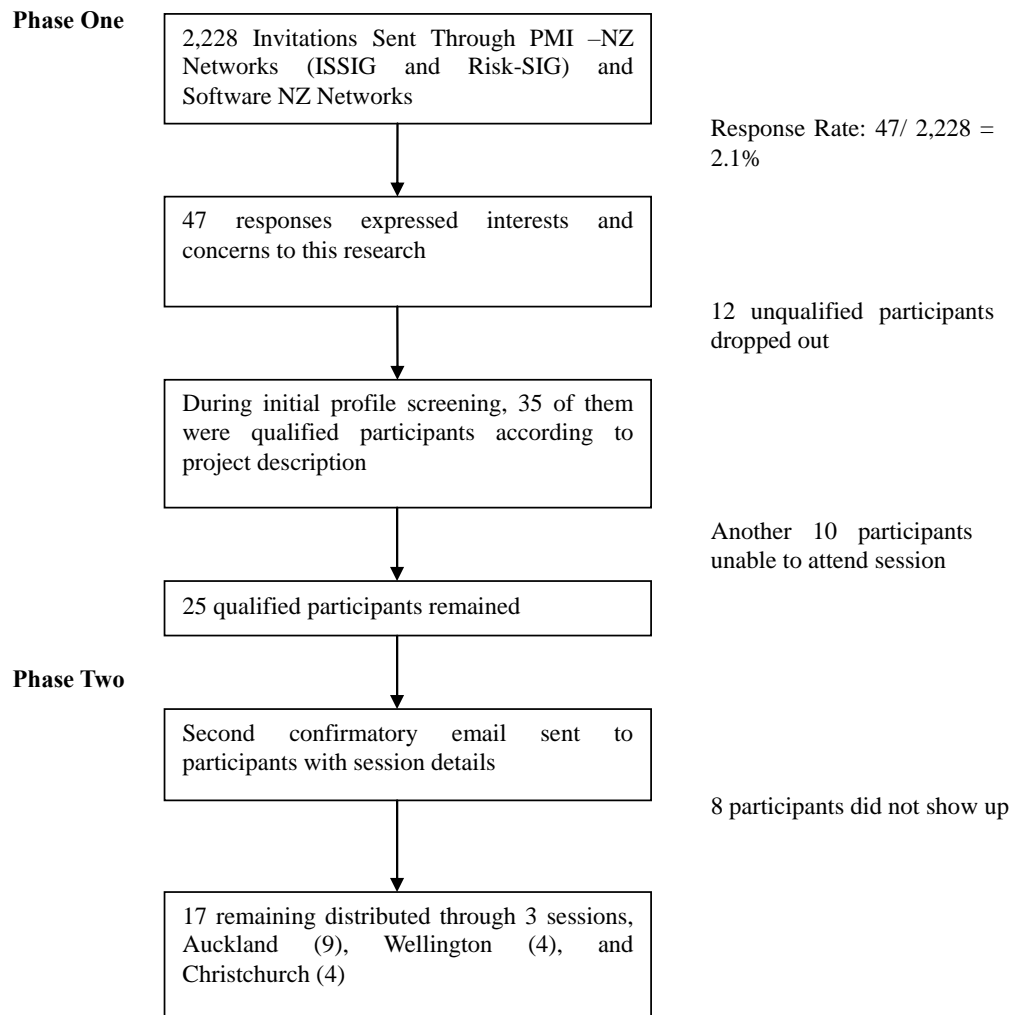
Phase One

2,228 Invitations Sent Through PMI –NZ Networks (ISSIG and Risk-SIG) and Software NZ Networks

Response Rate: 47/ 2,228 = 2.1%

47 responses expressed interests and concerns to this research

12 unqualified participants dropped out

During initial profile screening, 35 of them were qualified participants according to project description

Another 10 participants unable to attend session

25 qualified participants remained

Phase Two

Second confirmatory email sent to participants with session details

8 participants did not show up

17 remaining distributed through 3 sessions, Auckland (9), Wellington (4), and Christchurch (4)

Figure 1. Overview of the Recruitment Process for the Focus Group Study

The whole recruitment process took three weeks to complete. The time spent included screening profiles of participants, asking, and answering questions, and organizing focus group sessions. At the end of the process, we had 25 valid responses, and participants were symmetrically distributed across three selected cities. Nine (9) participants were based in Auckland, eight (8) participants were from Wellington, and eight (8) were from Christchurch. Another email was sent to remind all participants one week before the sessions. Following that email, we made an additional phone call two days before the session to ensure they were well-prepared for the session. Each onsite session took about two hours, longer than the normal 90-minute time frame. The focus group in Auckland was successfully conducted. All nine participants showed up, so the attendance rate was 100%. However, the attendance rates for the Wellington and Christchurch sessions were only 50%. We had eighteen (n=17) participants for our focus groups. Table 2 summarizes the focus group participant details.

Table 2. Focus Group Participants (n=17)

Title	Organization Type
Senior Systems Engineer #1	International
Project Manager #1	Local small and medium-sized enterprise (SME)
Senior Systems Engineer #2	International
Senior Business Analyst	National
Senior Consultant	Local SME
Managing Director/Developer	Local SME
Software Developer	National
Business Analyst/Project Manager	Local SME
Project Manager #2	International
Project Manager #3	National
Project Manager #4	National
Project Manager #5	International
Program Support office Test manager	National
Operations Manager	International
ERP Consultant	International
Software Sales and Marketing Manager	National
Project Manager #6	International

3.2.2 Conducting Focus Group Data Gathering

All focus group sessions were tape-recorded, and the entire dialogues were manually transcribed into verbatim copies. To bring participants into discussion mode, the underlying objective of this research was clearly outlined to participants at the beginning of the session: What sources of risks are involved in the requirements development of a system development project?

3.2.3 Thematic Data Analysis

Thematic analysis was employed as the data analysis method (Braun & Clarke, 2006). Qualitative researchers use thematic analysis to identify and analyze themes across qualitative data (Braun & Clarke, 2006). The thematic analysis focuses on producing and applying codes to data and is a special approach to theme development. A core step of thematic analysis is to perform a thematic coding exercise. In this study, open coding was used to assist the coding exercise. A code is a category that manifests the meanings of a particular subset of data; by grouping similar data under relevant codes, the researcher can further aggregate these codes into the main theme. In other words, the theme represents the patterned meanings of the data, and it is strongly related to the research question (Braun & Clarke, 2006). Table 3 shows the six steps of thematic analysis and the application of the steps in this study.

Encoding software NVIVO 8 was used to generate the initial codes. Similar codes were grouped together to find candidate themes. Four candidate themes emerged from the analysis, and each candidate theme also contained many subthemes aggregated from the texts. However, we discovered that some codes did not seem to fit into candidate themes or subthemes. Therefore, we temporarily placed them into a different theme called "miscellaneous" to review them later (Braun & Clarke, 2006). Reviewing candidate themes ensured that each candidate theme matched the meanings of its embedded codes and that all candidate themes could accurately depict a meaningful story from the data set (Braun & Clarke, 2006). We scanned through the details of all candidate themes and concluded that they were "real themes" and of significant value, as our data had shown they could well represent the types of requirements risk in ISD projects. We also assessed each theme to ensure there was no overlap so that each theme could be clearly distinguished from the others. A common guideline in revising themes is that by looking at the heading of each theme, the researcher can clearly describe its meaning in less than two sentences (Braun & Clarke, 2006). Consequently, the headings of four themes were refined to Requirements Identity, Requirements Volatility, Requirements Complexity, and Requirements Integrity.

Table 3. Thematic Analysis Process and Implementation in this Study Based on Braun and Clarke (2006)

Name of Step	Implementation of the step in this study
Becoming familiar with the data set	Transcribe focus group discussions. Thoroughly read through all focus group data and create initial analytical interests and thoughts about the data.
Generating initial codes	Identify interesting features of the focus group data as they occur to the researcher and encapsulate meanings with codes. Assimilate similar data into relevant codes.
Searching for themes	Group similar codes together according to their similarity. Use themes to represent the shared meanings of these codes.
Reviewing themes	The key task in this phase is to produce a thematic map of the analysis, which involves checking the representativeness of the generated themes for the given codes. It also checks the consistency of all themes to see if they are portraying the "full story."
Defining and naming themes	Revise and rename themes to avoid ambiguity and misunderstanding so that readers can clearly understand the phenomena under investigation.
Producing an analysis report	The task for this phase is to produce comprehensive reports showing the final analysis results, along with justifications on how these results relate back to the research objective and literature.

3.3   Delphi Data Collection and Analysis

The development of requirements risk profiles was based on a Delphi study. A Delphi study involves gathering opinions from a panel of experts in a particular field and trying to achieve a level of consensus from the panel (Linstone & Turoff, 2002). Information supplied by the panel members is shared among the panel members. However, the individual panel members are not identified when sharing the information. This anonymity allowed panel members to voice their opinions and reasoning safely without being labeled or the target of peer pressure. While Delphi involves a panel of experts to gather the necessary information for addressing the research problem, it is not until a reasonable consensus has been reached with responses from the panel that the Delphi process is complete. To achieve this consensus, it is necessary to gather the information supplied by each panel member, analyze the information, summarize it, or return it with initial results to the panel members for additional feedback. This process is repeated until the responses are similar and an agreement or consensus among the panel members is reached. It is crucial that while the process is repeated, none of the feedback received is identified as being supplied by a particular panel member. This anonymity is important in Delphi studies as it allows all members to provide their opinions without pressure.

The Delphi study was carried out in two main rounds. Each round included three iterations. The focus of Round 1 was to understand the level of impact each risk item can have on an ISD project. The focus of Round 2 was to understand which phase was most likely to be impacted by the risk item and any other phases that were also likely to be affected. Most Delphi studies start with an explicit brainstorming round. In this study, the risk item used to gain agreement from the panel members was sourced from the focus group interviews. The identified risk items were developed based on the focus group study described in the previous section. Generally, Delphi studies aim to get a level of agreement among panel members on the topic or issue being discussed or explored. In this study, a free-marginal kappa was used to determine the level of agreement among panel members. A free-marginal kappa was selected because the panel members were to choose a particular proportion response, be it risk impact level or the phase affected by a particular risk item (Brennan & Prediger, 1981; Randolph, 2005). A free-marginal kappa was calculated using the online calculator provided by Randolph (2008). We report the details of the data analysis, including the kappa values, in the demonstration and evaluation section.

3.4   Interview Study and Analysis

Researchers conducted the interview study in 2017. The data was collected with semi-structured theme-centered interviews and analyzed with thematic analysis. Analyzing data provided a comprehensive description of industry professionals' opinions, views, and experiences testing the method. A total of 22 professionals working with agile ISD were interviewed. The primary focus of the studies was to collect knowledge from the professionals on how this method could be used and developed and how they see the ways in which risk lists can be extended.

The studied organizations vary from small and medium companies to large corporations. The study contexts are summarized in Table 4. The small company #P operates in more than ten European countries in the B2B conference business. They organize more than 100 major events annually and over 100.000 meetings with several thousand solution providers (company #P). The medium-size software company (company #M) is based in Finland and employs several hundred IT consultants in major Finnish cities. They primarily work with industrial manufacturing companies locally. The third company (company #K) is a large Finnish corporation operating in the transportation industry. The company manages large infrastructure and construction projects and other development projects. The company had 8600 employees, and the turnover was 1.2B€. During the study, the companies did different types and sizes of agile ISD projects in several industry areas. Interviewees were industry and project professionals working on agile ISD projects. Two companies followed the agile approach in their software development and used Scrum practices (companies #P and #M). The third company described its ISD process as agile and incremental: the new features are specified, implemented, and installed in releases, each containing a predefined set of features (company #K). Due to the confidentiality agreements with the participating companies, we are not permitted to provide further details about the companies or the demographics of the interviewees.

Table 4. Summary of study contexts and the collected data.

Study organization	Project context	Data collection
Small conference organizer (company #P)	An agile software project (Scrum) on customers' self-service channel	Eight semi-structured interviews
The medium-sized software developer (company #M)	Several agile (Scrum) software projects	Nine semi-structured interviews
Large transport corporation (company #K)	Agile, large infrastructure and construction projects, and other development projects	Project specification workshop observations, five semi-structured interviews

The researchers analyzed the collected qualitative data with thematic analysis - with an inductive approach - to analyze the research data and provide a comprehensive description of professionals' views about the method. The data sets (i.e., the interview transcripts) were analyzed using criteria for artifact evaluation (Prat et al., 2015). The evaluation taxonomy comprises five meta-level criteria: goal, environment, structure, activity, and evolution. We applied two of the meta-level criteria for our study: goal and environment. These two criteria are summarized in Table 5. The goal describes how the artifact achieves its goal in a real situation, works correctly, and measures the value of achieving it. It also describes feasibility from the technical, operational, and economic points of view and defines the generality of the artifact. The goal criterion is divided into four categories: goal attainment, utility, feasibility, and generality. The goal attainment, in turn, is divided into efficacy, effectiveness, and validity subcategories. In addition, the feasibility criterion is divided into technical, operational, and economic feasibility subcategories. The environment criterion focuses on the artifact's coherence with people, organization, and technology. The taxonomy divides the environment criterion into three subcategories: people, organization, and technology. People subcategory has four subcategories: usefulness, ease of use, ethicality, and absence of side effects. Organization has two subcategories: alignment with business and absence of side effects. Technology has two subcategories: fit into technical IS architecture and alignment with IT innovation, and absence of side effects.

Table 5. The applied evaluation taxonomy (adapted from Prat et al. (2015)).

Meta level criteria	Detail level criteria
Goal	1) Goal attainment, 2) utility, 3) feasibility, and 4) generality
Environment	People: 1) usefulness, 2) ease of use, 3) ethicality, and 4) absence of side effects Organization: 1) Alignment with business, and 2) absence of side effects Technology: 1) Fit into technical IS architecture and alignment with IT innovation and 2) absence of side effects

4   Developed Artifacts

4.1   Risk Items

The focus group analysis results are presented in Table 6. The risk items were extracted from the focus groups for each theme: "identity," "volatility," and "complexity". The themes and the risk items are elaborated on below.

Table 6. Requirements risk items identified in the focus group interviews.

Requirements Identity Risks	Volatility Risks	Requirements Complexity Risks	Requirements Integrity Risks
Conflicting requirements	Emerging requirements dependency	Access to clients	Fixed budget and deadline
Compliance with external regulations	Change in external regulation	Communication medium	Lack of requirements review
Misunderstood business needs	Underestimate the magnitude of changes	Constrained by users' knowledge	Original source of requirements
Incorrect stakeholders	Technology changes	Knowledge gap between coworkers	
Absence of project sponsor	Change in business strategy and direction	Lack of collaboration	
Ambiguous requirements	Emerging requirements dependency	Negative perception	
Client commitment	Unrated requirements	Translation of conceptual requirements to detailed specification	
Hostile users	Project team member turnover	Verbal precision	
Delivering what the client needs			
Omitted requirements			

For Requirements Identity, having clearly defined requirements would lay a solid foundation for subsequent development efforts and ensure that the proposed system matched clients' expectations to achieve higher customer satisfaction. In following are examples of 1) ambiguous requirements, 2) delivering what the client needs, 3) incorrect stakeholders, and 4) hostile users risk items:

1)   "One of the risks in requirements is they're vague. They are not very well defined. They can be interpreted in multiple manners." – Senior Project Manager

2)   "That's why you need to understand who the real user is in a broader sense so that you are delivering exactly what is needed as opposed to what was asked for." – Project Manager

"On time, to budget, and scope, that's the definition of what the customer said they wanted." – Project Manager

3)   "Identify the right stakeholders, and all the stakeholders as well, not just the ones that you think might be important." – Business Analyst/ Project Manager

4)   "I've been experienced with hostility, but it's more operational level people busy with day-to-day activities... because we are taking too much of their time during the day." – Business Analyst/ Project Manager

For Requirements Volatility, requirements can easily change for a range of reasons. One typical example concerning requirement dynamics is the change of business strategy and direction (changing business strategy and direction risk item):

"*A change in business strategy and direction represents a radical change to the project. It can significantly impact the requirements that were previously defined.*" –Business Analyst/Project Manager

Concerning Requirements Complexity, requirements are essentially complex because they must be effectively communicated and understood across different entities. For example, end-users have to communicate what they want as requirements to the development team. Those requirements also need to be shared and understood by different development team members to produce artifacts that satisfy end users. As more people are working collaboratively to develop the system, the communication process can easily go wrong and become unmanageable (communication medium risk item):

> "*Soon, they use rational rules, visit the management agency… they say rational rules and UML are the requirements for modeling language, now, two months into the project, customers thought it would satisfy the communication needs, and it didn't. It is useless....*" –Senior System Engineer #1

However, our focus group results also emerged a new risk dimension, Requirements Integrity. This dimension extends Mathiassen et al. (2007) model's risk dimensions. The found requirements risk items did not match the dimensions described above. We define this emerging new risk dimension as follows.

Integrity risk measures the completeness and accuracy of the requirements elicited from end-users. These issues were related to the lack of requirements review and not understanding where the requirements originated. Furthermore, the results show that setting a fixed budget and deadline for a given project can limit the time spent on requirements gathering and thereby restrain requirements from being fully captured. In the following are examples of the fixed budget and deadline risk item:

> "...*they decide on the due date before even defining the requirements. And then everything else somehow cascades in trying to meet the due date.* " – Project Manager #1

> "*To squeeze upfront, to short notice, what happened is you miss requirements, badly written, all matters of things.*" – Senior System Engineer #2

In addition, here are examples of the lack of requirement review risk item:

> "*But in a big project where you have a million lines of code, having some sort of tools that can trace original ideas back to particular test procedures is very, very important. And checks all changes if you update the requirements, you can trace all the way through, who says what, why, when, and how much it cost.*" – Senior System Engineer #2

Finally, here is an example of the original source of the requirements risk item:

> "*This comes back to your point of requirement. Requirements are such a diverse range of things. Range from huge requirements documents written by, who knows, with what level of qualification, you got no idea, to the other end you actually sit down with customers to define the requirements.*" – Project Manager #3

Next, a Delphi study was carried out. It started with 29 risk items identified in the focus group interviews. We also asked each panel member to indicate risk levels. The impact levels of interest were high, medium, and low. Panel members were also provided the opportunity to add additional risk items if they felt that certain requirements' risks were being left out of the study. However, they were only allowed to do so if they considered the risk requirements related. There were 14 risk items that the majority of panel members agreed on with regard to the impact these risk items can have on ISD projects. The responders also added five risk items. These are summarized in Table 7.

Table 7. Requirement Risk Items Added by Panel Members

Requirement Risk Items	Description Provided by Panel Members
Key team members are not compensated adequately	Key team members can be pressured to work additional hours to meet timeframe expectations
Team conflict	Conflict develops amongst team members who need to work together
Quality of testing	Time or other pressures create gaps between the testing time needed or planned and the time spent testing
Gap between actual project status & status reported	Normally done by those who report a different status from the actual one due to their belief that time will be made up and the gap will go unnoticed
Gap between skills needed and resourced	Poor fit between skills needed and those recruited

4.2   Risk Profiles

Next, we asked the panelists to connect risk items to general ISD phases and identify the activity most likely affected by a risk item. These results are summarized in Table 8. For this study, three main activities of an ISD project were considered: requirements, design, and implementation. Only the risk items for which the panel reached an agreement regarding the impact on an ISD project were included.

Table 8. Risk Profiling

Requirements Specific Risks	Impact	Design Specific Risks	Impact	Implementation Specific Risks	Impact	Risks Affecting All Phases	Impact
Absence of project Sponsor	High	Missing requirements	High	Hostile users	Medium	Ambiguous requirements	High
Access to clients (proximity to source)	High	Delivering what the client requires	High	Project team member turnover	Medium	Unrated requirements	High
Incorrect Stakeholder	High	Compliance with external regulations	Medium			Client commitment	High
Misunderstood business needs	High	Conflicting requirements	Medium			Change in external regulations	Medium
Change in business strategy and direction	Medium	Emerging requirements dependency	Medium			Underestimation of change magnitude	Medium
Constrained by users' knowledge	Low	Knowledge gap between coworkers	Medium			Fixed Budget and Timelines	Medium
		Lack of collaboration	Medium				
		Technology changes	Low				

4.3   Activity-specific Checklists

Checklists for the requirements, design, and implementation activities were constructed (Tables 9, 10, and 11, respectively) next. The checklists can be used in projects to determine risk exposure. As there may be other risk items that have not been captured in this study, the risk lists can be extended by the person using them based on their experience and knowledge. The 23 risk items provided here give a baseline for the project risk profiling exercise.

Table 9. Requirements Activity-Specific Checklist

Requirement Risk	Risk Type	Project is/can be exposed?
Absence of Project Sponsor	*Identity*	
Access to Clients (Proximity to Source)	*Complexity*	
Ambiguous Requirements	*Identity*	
Change in Business Strategy and Direction	*Volatility*	
Change in External Regulations	*Volatility*	
Client Commitment	*Identity*	
Constrained Users' Knowledge	*Complexity*	
Fixed Budget and Timelines	*Integrity*	
Incorrect Stakeholder	*Identity*	
Misunderstood Business Needs	*Identity*	

Underestimation of Change Magnitude	*Volatility*	
Unrated Requirements	*Volatility*	
Any other risks that could affect the design and implementation		

Table 20. Design Activity-Specific Checklist

Requirement Risk	Risk Type	Project is/can be exposed?
Ambiguous Requirements	*Identity*	
Change in External Regulations	*Volatility*	
Client Commitment	*Identity*	
Compliance with External Regulations	*Identity*	
Conflicting Requirements	*Integrity*	
Missing Requirements	*Identity*	
Delivering What the Client Requires	*Identity*	
Emerging Requirements Dependency	*Volatility*	
Fixed Budget and Timelines	*Integrity*	
Knowledge Gap between Coworkers	*Complexity*	
Lack of Collaboration	*Complexity*	
Technology Changes	*Volatility*	
Underestimation of Change Magnitude	*Volatility*	
Unrated Requirements	*Volatility*	
Any unresolved risks from requirements and risks that could affect the implementation		

Table 13. Implementation Activity-Specific Checklist

Requirement Risk	Risk Type	Project is/can be exposed?
Ambiguous Requirements	*Identity*	
Change in External Regulations	*Volatility*	
Client Commitment	*Identity*	
Fixed Budget and Timelines	*Integrity*	
Hostile Users	*Identity*	
Project Team Member Turnover	*Volatility*	
Unrated Requirements	*Volatility*	
Underestimation of Change Magnitude	*Volatility*	
Any unresolved risk items from design and requirements		

4.4   Agile Requirements Risk Prioritization Method

The developed agile requirements risk prioritization method is summarized in Figure 2. The method provides three elements to accomplish risk requirements prioritization. Our method assumes iterations between requirements, design, and implementation activities. The requirements risk analysis and resolution layer is similarly based on iterations between three key elements: identifying risks, assessing the risk profile, and intervening. Below we elaborate on these elements and link our findings to the previous literature.

Like Mathiassen et al. (2007), we propose that the number of risk items faced by the ISD project and the impact each risk item causes on the project should play a significant role in the project's risk profile. Our method assumes that the profiling effort begins with identifying risks with the developed checklists (Tables 9-11), following what Persson et al. (2009) presented with their literature synthesis. The analyst can recognize the requirements-related risks that affect or impact the project by applying the checklists. The project manager or analyst attempting to profile the project risk should also identify the risks the project faces or is likely to face when the project profiling exercise is being carried out. The person can utilize one of the three checklists for this purpose. Each of the checklists contains risk items that are most likely to affect the requirements, design, and implementation activities of an ISD project.
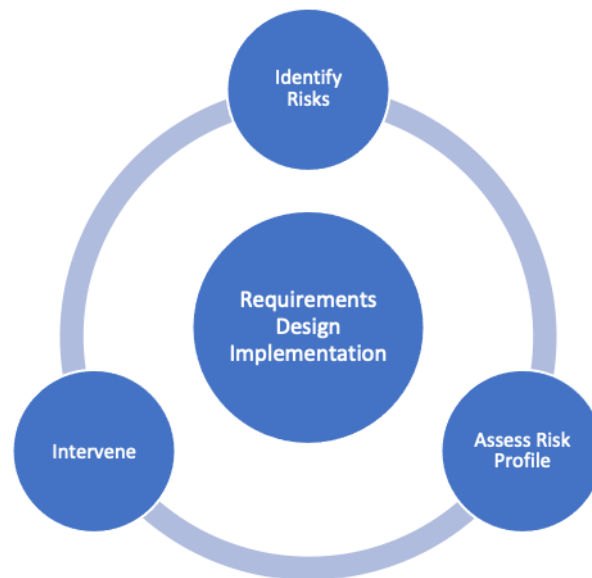

Figure 2. Agile Requirements Risk Prioritization Method.

Table 12. Requirements Risk Resolution Patterns

1.	If identity risks are high, put a high emphasis on discovery techniques
2.	If integrity risks are high, put a high emphasis on prioritization techniques.
3.	If volatility risks are high, put a high emphasis on experimentation techniques.
4.	If complexity risks are high, put a high emphasis on specification techniques
5.	If three or more risk items are high, follow the above sequence of applying techniques (1 to 4).

As the ISD project progresses, the evaluator of the project risk profile should consider the number of risk items occurring in the project. In Table 7, we have depicted the perceived risk impact for different phases in projects. Similarly, as with the original contingency-theory-based risk management model (Mathiassen et al., 2007), we assume that there can be several requirements risk categories marked "high" simultaneously. In this situation, we follow the interaction proposition: adopting a requirements development technique can require compensating actions to reduce the adverse effect on risks other than the ones targeted by the technique. Thus, when several requirements risk categories are marked "high," we suggest first inspecting which of the requirements risk resolution patterns is in a recommended priority proposition. The technique categories (discovery, prioritization, experimentation, and specification) are considered the same, as Mathiassen et al. (2007) argued. Mathiassen et al. (2007) listed 85 different requirements risk resolution techniques categorized into discovery, prioritization, experimentation, and specification groups. This list is not considered exhaustive, but it still provides an extensive review of a range of techniques available for requirements risk resolution (see Appendix A for details). The priority proposition has been amended to include integrity risks linked to prioritization techniques.

We also see that it is helpful to consider portfolios of such patterns to apply the risk resolution patterns. For this purpose, we need to define different risk profiles. Table 13 summarizes these sixteen project

profiles that can be used with the developed method and the five requirements risk resolution patterns. Those responsible for projects need to understand how they can use the risk profiles to guide the transition between high, medium, and low-risk resolution portfolios. Figure 3 below depicts this process. The process starts by determining which risks the project is exposed to. Based on the risks exposed, a risk profile for the project in the current phase is determined. While the project is in that phase, more active monitoring of the risks and resolution techniques to facilitate risk profile transitions will occur. The project should also be actively monitored for any new risks. If it is determined that the project is exposed to new risks, the risk profile is re-assessed. Accordingly, the risks the project is exposed to are re-assessed when the project moves on. The project continues with the same risk profile if there are no new risks. Finally, if there are new risks or some risks have been mitigated, then the project's risk profile is re-assessed.

Table 43. Project Risk Profiles

Profile	Description	Risk resolution
#1	**Profile 1** has all requirement risk types as high.	Focus on discovery techniques to reduce some of the identity risks faced in the project. You should also utilize parallel experimentation and specification techniques to avoid adverse effects on the other risk categories.
#2	**Profile 2** is like profile 1, except the project is not expecting, or has not faced, any integrity risk issues with requirements.	Focus on identity risks. You should consider using discovery techniques and specification techniques to reduce the overall risk profile of the project.
#3	**Profile 3** is like profiles 1 and 2, except projects in this profile do not face or have not yet faced requirements complexity risk issues.	Focus on identity, volatility, and integrity risks. You should consider using discovery techniques along with experimentation and specification techniques to reduce the overall risk profile of the project.
#4	**Profile 4** is like the above profiles, except the requirements are not seen as unstable.	Focus on discovery and specification techniques, which will help to reduce the overall risk profile of the project.
#5	**Profile 5** has a reduced impact on the project due to identity risks.	Focus the other three requirement risk categories that are causing or potentially have a high impact on the project. Use prioritization and experimentation techniques to reduce the overall risk profile of the project.
#6	**Profile 6** represents projects that are not anticipating requirements complexity and have not yet encountered integrity issues.	The focus is on discovery techniques and experimentation to reduce the overall risk profile of the project.
#7	**Profile 7** represents projects facing, or having the potential to face, identity and complexity risks associated with the requirements.	Focus on discovery techniques to develop client relations and engage in experimentation to reduce the overall risk profile of the project.
#8	**Profile 8** represents projects that are facing requirement identity and integrity risk issues	Focus on discovery techniques and engage in specification techniques to remove inconsistencies in the requirements and to reduce the overall risk profile of the project.
#9	**Profile 9** represents projects that face volatility and integrity requirement risks.	Focus on prioritization and experimentation techniques to resolve the risks and to reduce the overall risk profile of the project
#10	**Profile 10** represents projects that should determine requirements for successful project delivery.	Use prioritization techniques to reduce the overall risk profile of the project.
#11	**Profile 11** represents projects that face unstable requirements.	Focus on prioritization and experimentation techniques. In addition, specification activities also should be carried out to reduce the overall risk profile of the project.
#12	**Profile 12** represents projects facing a high degree of ambiguity issues with requirements.	Focus on discovery techniques to reduce the overall risk profile of the project.
#13	**Profile 13** represents projects still facing issues with the stability of requirements.	Focus on experimentation techniques. You should also focus on specification techniques to keep integrity issues manageable and to reduce the overall risk profile of the project.
#14	**Profile 14** represents projects with complexity issues.	Focus on prioritization techniques to reduce the overall risk profile of the project.

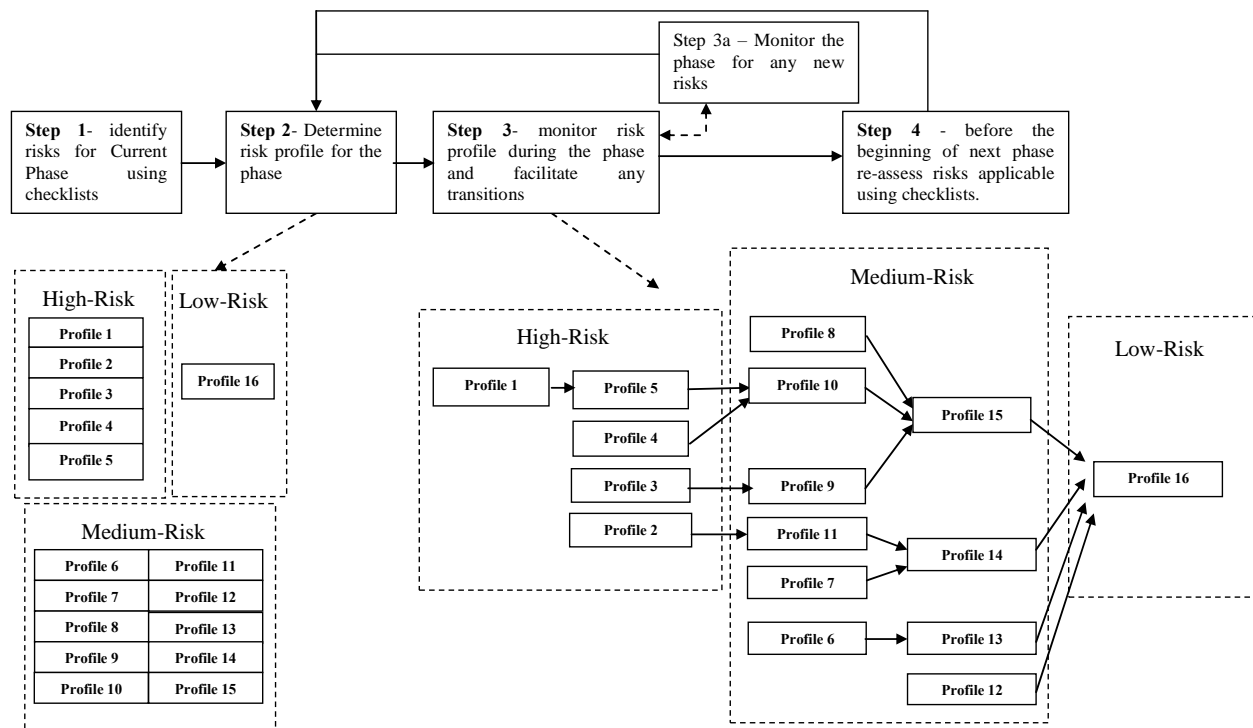| #15	**Profile 15** represents projects that have inconsistencies in the requirements.	Focus on discovery and specification techniques to reduce the overall risk profile of the project.
#16	**Profile 16** represents projects that have successfully utilized various risk resolution techniques that do not adversely affect requirement risks in all four categories.	

Figure 3. Transition Process Between High, Medium, and Low-Risk Profiles

5   Demonstration and Evaluation

5.1   Risk Items

The risk items were collated and supplied to the Deplhi study panel to evaluate the results. The panel members were informed that they could change the impact level they assigned to a particular risk item if they agreed with the majority of the panel members (if they had not done so) or wanted to change the impact level based on the feedback received. Panel members were also encouraged to provide additional feedback. To include these additional risk items in the study period, it was necessary for panel members to agree upon whether the risk items could be considered as requirements risks and, if so, the type of requirements risks.

Next, a summary survey was initiated to study whether the panel members agreed with the view perceived as the panel's message. The results showed that by treating both "strongly agree" and "agree" as separate responses, there was fair agreement (free-marginal kappa = 0.343915) among the panel members. But by treating the responses from panel members who indicated that they agreed and panel members who indicated that they strongly agreed to mean that the member agreed with the perception, there was near-perfect agreement among the panel members (free-marginal kappa = 0.86665).

The responses also showed that some of the panel members had chosen to change the impact perceptions of some risk items, resulting in six additional risk items now showing that a majority of panel members agreed on their impact perceptions (one risk item perceived as causing a high impact, five risk items perceived as causing a medium impact). Panel members were asked whether they agreed with the impact perceptions, similar to how they were solicited in the summary round in Round 1. The responses

indicated that the panel agreed upon the impact perception (free-marginal kappa = 0.714285). The risk items identified in Table 14 above provide the list of risk items and their impact perceptions upon which panel members agreed.

Table 54. Impact Perception Agreed Upon by the Panel

Impact to ISD Project	Requirement Risk Items
High	1. Absence of project sponsor 2. Access to clients 3. Ambiguous requirements 4. Client commitment 5. Delivering what the client needs 6. Incorrect stakeholder 7. Missing requirements 8. Misunderstood business needs 9. Unrated requirements
Medium	10. Change in business strategy and direction 11. Change in external regulations 12. Compliance with external regulations 13. Conflicting requirements 14. Emerging requirements dependency 15. Fixed budget and deadline 16. Hostile users 17. Knowledge gap between coworkers 18. Lack of collaboration 19. Project team member turnover 20. Underestimation of change magnitude
Low	22. Constrained by users' knowledge 23. Technology changes

5.2   Risk Profiles

A Delphi survey was initiated to evaluate if panel members agreed with the earlier results of constructing the risk profiling guideline. The combined responses for both "strongly agree" and "agree" regarding the choice requirements risks that affected the requirements, design, and implementation activities of an ISD project were near-perfect agreement among the panel members (free-marginal kappa = 0.999999). The results from the survey also suggest that even though certain risk items have been perceived and agreed upon by panel members as most likely to affect a particular phase of ISD projects, the risks can affect any activity of an ISD project if they are not appropriately managed. The panel members' views on this, when considering both "strongly agree" and "agree," showed near-perfect agreement (free-marginal kappa = 0.999999). The panel could not agree on the activities most likely affected by the remaining six risk items. Therefore, for these six risk items, based on the panel's agreement with the statement that "a risk item that has not been identified, or that has not been properly addressed in a timely fashion, can affect any activity of the project," we concluded that these six risk items could potentially affect any activity.

5.3   Method

Finally, in the interview study with the three companies we evaluated the method's applicability and utility. This step examined how requirement risks are managed in IS projects and how the developed method is applicable for prioritizing requirement risks. Our study focused inductively on two meta-level criteria: goal and environment. These two criteria were then used to evaluate the developed method. The entities describing the goal and environment criteria were mapped hierarchically, after which the suitability for the subcategory criteria was mirrored. Table 15 summarizes the analysis, indicating strong support for the utility of the developed method (cf. goal criteria) and its applicability to ISD projects (cf. environment criteria). Below, we depict, in more detail, how the interviewees perceived the method to support the risk identification process and help to identify and manage risks that might otherwise be overlooked.

Table 65. Method Evaluation Results

Evaluation Criteria	Total		Count
Goal	41		
		Generality	20
		Goal attainment, Validity	17
Environment	56		
		People	20
		Organization	26

The goal/generality criterion was used most in examining the study results. The studies examined how extensively and generally the method can be used in various development projects. Responses varied widely for and against. Interviews were widely influenced by the interviewees' experience and work history. Some respondents saw the method as a good part of risk management and found it useful in rapidly changing environments. The method's checklists were seen as a complete and accurate overview of possible requirements-related risks. Interviewees from company #M felt that the method was a suitable tool for a beginner, while some felt that it was very well suited as a tool for risk analysis that helps to identify risks that are not usually addressed enough in projects. The interviewees commented on the findings of the case studies:

> "*The case study suggests that the requirement prioritization method was applicable in the case project arrangement. The requirements risk analysis and prioritization based on the risk tables was found meaningful for evaluating the risks associated with requirements.*" – Interviewee (company #K)

> "*Results suggest that the method indeed would help its user to focus on topics that might have otherwise been ignored or forgotten*..." – Interviewee (company #P)

Environment/people provided a view that the method was seen more as a communication tool that could help to share knowledge between different stakeholders. However, it was also pointed out that it needs to cover risk relations so that people in different roles understand concepts similarly (interviewees from company #P) and that experience is the key value when defining risk items (interviewees from company #M).

> "*If I would have to explain to a decision-maker what is wrong, this could be an eye-opener that which issues are related... This is not really recognizing risks, but more communicating them through a known model to someone who is not familiar with the whole development process.*" – Interviewee (company #P)

> "*When you read these* [risk items] *from a checklist, you probably recognize things in a different way as you do not know yet how to make it based on experience.*" – Interviewee (company #M)

Environment/organization criteria presented information that the method can support identifying and prioritizing the risks on a high level. Still, the interviewees from company #M told us that the more precise prioritization and finding resolutions to the risks were seen as important, also practical experience and consideration of the project-specific characteristics. The method was also seen as a useable tool that can be implemented as part of the organization's project toolbox:

> "*Method's framework could be added to project toolbox for project risk management or requirements management purposes.*" – Interviewee (company #K)

The goal/goal attainment/validity criteria fit the goal of the research questions to determine whether the method is a suitable tool for requirement management and how well it achieves defined goals. The interviewees from company #K agreed that the method was useful in bringing structure to analyzing the risks related to project requirements, which was considered valuable by all interviewees.

> "*Maybe it does* [bring up new risks associated with requirements]. *Because it is good that there are prepared risks to consider. Our approach has been when making a risk table, that we start to think about the risks from the scratch. And then the risks will always be the sufficiency of resources and so forward.*" – Interviewee (company #K)

Finally, interviewees from company #M stated that the method was also valuable for detecting and prioritizing several requirements management-related risks. It can help collect and maintain the necessary knowledge to produce quality project work.

> "*I would say that this all is very useful when thinking about our projects. … We do have the knowledge, but it would be good to also concretize and maintain it.*" – Interviewee (company #M)

6 Implications for Research and Practice

The study contributes to the literature in several ways. In the literature, general-purpose requirements risk prioritization methods, Mathiassen et al. (2007) or Hickey and Davis (2004), offer ways to integrate these methods into the plan-driven structured ISD. However, while researchers have highlighted the need to prioritize requirements (Matook & Maruping, 2014) and thus consider the related risks, the literature does not offer many agile requirements prioritization methods. The available methods are often tailored to a specific agile ISD method, like SCRUM (Racheva et al., 2010) or automotive software-based tools to assist in categorizing and managing the requirements-related risks (cf., e.g., Achimugu et al., 2016). Furthermore, the agile requirements prioritization literature typically treats risks as highly general. For example, Racheva et al. (2010) define risks as something caused by a requirement's implementation, but they do not define what this might be. The literature considers the requirements risk as a more versatile concept. Moreover, the literature often focuses on what value agile requirements prioritization offers for the customer to guide the prioritization efforts. The value of the requirements (for a business, for instance) is estimated during agile ISD, and requirements are prioritized accordingly. Our study offers a risk management-based approach for agile requirements prioritization, complementary to the business value approach.

Our study also extends and tests Mathiassen et al.'s (2007) conceptualization of the requirements risk resolution model. Mathiassen et al.'s (2007) model consists of eight permutations of project profile states to describe the use of the model and four risk resolution heuristics. Our model consists of sixteen project profiles as we add a requirements risk category and a corresponding risk resolution heuristic: we argue for the need to recognize integrity-related requirements risks and apply prioritization techniques to mitigate these. The original model by Mathiassen et al.'s (2007) was developed based on a systematic literature review. Our study offers an evaluation of the original study, which led to the further development of the model, but also to the development of an agile requirements risk prioritization method.

Our method is also relatively unaffected by the underlying ISD methods used in specific agile projects. We see that our method can be used in projects that use traditionally structured ISD methods, agile methods, or combinations of both (e.g., Niederman et al., 2018). Our method recognizes that the contemporary ISD can often be characterized as continuous development (Gall & Pigni, 2022). For this purpose, our method proposes two key aspects important for practice regarding requirements risk management. First, the risk profiling should be continuous, and we should be able to identify current requirements risks, assess the overall situation, update the risk profile for the project, and finally intervene by using suitable risk resolution techniques. Second, our method also proposes that in continuous development, we can recognize three important emphasis areas of ISD: requirements, design, and implementation. When applying agile ISD methods, we realize that the distinction between the phases is blurred. However, we still argue that, based on the previous literature, ISD projects have a different emphasis on activity during the project life cycle, regardless of the ISD method or methodology applied.

Our study also makes a methodological contribution to the DSR literature. We offer an example of how to apply the DSRM for conducting a DSR project in several cycles. The applied cyclic DSRM can be used in other method development projects or any DSR projects that require several iterations of the DSRM. Tuunanen and Peffers (2018) offer a similar approach for design principle and design theory development purposes. Our approach also follows Ågerfalk and Karlsson (2020), and we seek to make an artifactual contribution. Our method is developed based on a contingency theory to manage risks (Barki et al., 2001; Mathiassen et al., 2007), which is instantiated in the risk resolution patterns. Thus, while our method is based on the contingency theory, we do not seek to develop a design theory but instead a method artifact. Furthermore, we also present a way other researchers can follow to evaluate an artifact. We apply Prat et al.'s (2015) taxonomy of five meta-level evaluation criteria to evaluate our method: goal, environment, structure, activity, and evolution. The evaluation was inductive. We started with the meta-level criteria, and then with the result of three studies with IS developer companies, we found that the industry practitioners offered strong support for the utility of the developed method (cf. goal criteria) and its applicability to ISD

projects (cf. environment criteria). This contributes to the DSR methodology. The perceived view in the DSR literature (e.g., Peffers et al., 2007) is that the evaluation criteria should be set before evaluating the artifact. In contrast to this view, our study inductively analyzed the interview data using a meta-level and found more specific evaluation criteria during the analysis process. In addition, our study depicts how to use the Delphi method (Keil et al., 1998; Linstone & Turoff, 2002) in a DSR project for data collection and analysis.

Our work contributes to practice in several ways. Our findings show that the method helps to focus on previously ignored topics, but it requires a deep understanding of IS projects, system development, and the organizational environment. The method also is most beneficial at the beginning of the project or development iteration, and our study participants suggested to use as a communication tool in agile development. Furthermore, the checklists were valuable for highlighting key issues, but it was seen as good that the method suggests assessing risks outside the checklists. The method helped detect and prioritize several requirements management-related risks, and the checklists contained a complete and accurate overview of possible requirement-related risks. Our participants thought the method fit well with agile projects' rapidly changing environments and workflows and is easy to learn and use. In addition, the method was perceived as especially useful for less experienced project managers. The requirement risk analysis approach was also found to be novel by the study participants. We have also included a way to follow the risk profile of the project (Figure 3), which can help practitioners further understand how risks dynamically evolve during an ISD project. Finally, our work allows practitioners to prioritize the overall requirements-related risks for agile ISD projects. For this purpose, the method is grounded in the literature yet was developed together with industry professionals. Our method proposes four different requirements risk categories (identity, integrity, volatility, and complexity). Our method also provides practitioners with easy ways to apply guidelines in practice.

7   Concluding Remarks and Limitations

This paper proposes an agile requirements risk prioritization method for ISD based on the literature on contingency-based risk management and identifying and mitigating requirements risks. Our paper contributes to the literature by developing a method that 1) provides a requirements risk item list, 2) a guideline for project risk profiling, 3) how to prioritize the use of requirements risks resolution techniques, and 4) the requirements risk prioritization method for agile ISD that the practitioners can use. We developed the method together with industry professionals in Finland and New Zealand. The study applied the DSRM (Peffers et al., 2007) to structure and it guided the research process.

Although we strived to conduct a rigorous and well-planned DSR study, we recognize some limitations. For example, the number of participants in the Delphi study could have been larger to improve the generalizability of the results. However, we were satisfied that the number of participants was in line with the previous literature that has applied the Delphi method in IS research (Akkermans et al., 2003; Keil et al., 1998). Thus, we do not consider this a significant limitation of the study. Similarly, it would be interesting to conduct several case studies in parallel in a particular industry context in future research to see how this would enhance or change the results. This would, however, require substantial resources to run the study, as running Delphi studies is laborious.

We also note that we have not focused on risk resolution techniques with this study. Moreover, we built on the previous research on this (Mathiassen et al., 2007). However, during this study, we did not see any compelling reasons to extend the scope of the study in this direction. The industry participants did not report any significant need for developing new risk resolution techniques. Still, they were more interested in the requirements risk prioritization aspects and the impact of requirements risks on a project. Mathiassen et al. (2007) report many different requirements risk resolution techniques (cf. Appendix A). The list provides ample resources for the practitioners to choose techniques. Our method provides guidelines for how to do this in practice.

We also recognize that our method has not yet been fully tested. However, we have evaluated the method's proof-of-value (Nunamaker et al., 2015). However, we are currently working on a research project with a major multinational company to apply the developed method in real-world ISD projects to gain proof of use (Nunamaker et al., 2015) and further develop the method. We also call for further application of the method in different industry contexts to give more evidence of its utility and applicability. We look forward to doing this soon and investigating how the method applies to different domains of ISD, like DevOps, but also to see how different agile ISD methods impact the method's efficacy for

requirements risk prioritization and whether it needs to be altered to accommodate different contexts where the method is applied.

References

Achimugu, P., Selamat, A., & Ibrahim, R. (2016). Reprotizer: A fully implemented software requirements prioritization tool. In N. T. Nguyen & R. Kowalczyk (Eds.), *Transactions on computational collective intelligence XXII* (pp. 80-105). Springer.

Ågerfalk, P. J., & Karlsson, F. (2020). Artefactual and empirical contributions in information systems research. *European Journal of Information Systems*, *29*(2), 109-113.

Akkermans, H. A., Bogerd, P., Yücesan E., & van Wassenhove, L.N. (2003). The impact of ERP on supply chain management: Exploratory findings from a European Delphi study. *European Journal of Operational Research, 146*(2), 284-301.

Alsaqaf, W., Daneva, M., & Wieringa, R. (2019). Quality requirements challenges in the context of large-scale distributed agile: An empirical study. *Information and Software Technology*, *110*, 39-55.

Alter, S., & Ginzberg, M. (1978). Managing uncertainty in MIS implementation. *Sloan Management Review, 20*(1), 23-31.

Barki, H., Rivard, S., & Talbot, J. (1993). Toward an assessment of software development risk. *Journal of Management Information Systems, 10*(2), 203-225.

Barki, H., Rivard, S., & Talbot, J. (2001). An integrative contingency model of software project risk management. *Journal of Management Information Systems*, *17*(4), 37-69.

Barlow, J. B, Giboney, J. S., Keith, M. J., Wilson, D. W., Schuetzler, R. M., Lowry, P. B., & Vance, A. (2011). Overview and guidance on agile development in large organizations. *Communications of the Association for Information Systems, 29*, Article 2.

Bergman, M., King, J. L., & Lyytinen, K. (2002). Large scale requirements analysis as heterogeneous engineering, *Scandinavian Journal of Information Systems*, *14*(1), Article 2.

Boehm, B. W. (1991). Software risk management: Principles and practises. *IEEE Software. 8*(1), 32-41.

Braun, V. & Clarke, V. (2006). Using thematic analysis in psychology, *Qualitative Research in Psychology*, *3*(2), 77-101.

Brennan, R. L. & Prediger, D. J. (1981). Coefficient kappa: Some uses, misuses, and alternatives. *Educational and Psychological Measurement*, *41*(3), 687-699.

Brooks, F. P. (1987). No silver bullet - Essence and accidents of software engineering. *IEEE Computer, 20*(4), 10-19.

Chen, L. (2015). Continuous delivery: Huge benefits, but challenges too, *IEEE Software, 32*(2), 50-54.

Davis, G. B. (1982). Strategies for information requirements determination. *IBM Systems Journal, 21*(1), 4-30.

Dorofee, A. J., Walker, J. A., Alberts, C. J., Higuera, R. P., & Murphy, R. L. (1996). *Continuous risk management guidebook*. Carnegie-Mellon University.

Dreesen, T., Diegmann, P., & Rosenkranz, C. (2020). The impact of modes, styles, and congruence of control on agile teams: Insights from a multiple case study. In *Proceedings of the 53rd Hawaii international conference on system sciences*.

Florig, H. K., Morgan M. G., Morgan K. M., Jenni, K. E., Fischhoff, B., Fischbeck, P. S., & DeKay M. L. (2001). A deliberative method for ranking risks (I): Overview and test bed development. *Risk Analysis*, *21*(5), 913-921.

Gall, M., & Pigni, F. (2022). Taking devops mainstream: A critical review and conceptual framework. *European Journal of Information Systems, 31*(5), 548-567.

Geiger, D., Danner-Schröder, A., & Kremser, W. (2021). Getting ahead of time–performing temporal boundaries to coordinate routines under temporal uncertainty. *Administrative Science Quarterly*, *66*(1), 220-264.

Hevner, A. R., March, S. T., Park, J., & Ram, S. (2004). Design science in information systems research. *MIS Quarterly*, *28*(1), 75-105.

Hickey, A. M., & Davis, A. (2004). A unified model of requirements elicitation. *Journal of Management Information Systems*, *20*(4), 65-84.

Iversen, J. H., Mathiassen, L., & Nielsen, P. A. (2004). Managing risk in software process improvement: An action research approach. *MIS Quarterly, 28*(3), 395-433.

Keil, M., Arun, R. & Shan, L. (2013). How user risk and requirements risk moderate the effects of formal and informal control on the process performance of it projects. *European Journal of Information Systems*, *22*(6), 650-672.

Keil, M., Cule, P. E., Lyytinen, K., & Schmidt, R. C. (1998) A framework for identifying software project risks. *Communications of the ACM. 41*(11), 76-83.

Kremser, W., & Blagoev, B. (2021). The dynamics of prioritizing: How actors temporally pattern complex role–routine ecologies. *Administrative Science Quarterly*, *66*(2), 339-379.

Linstone, H. A., & Turoff, M. (2002). The Delphi method: Techniques and application. *New Jersey Institute of Technology.* Retrieved from http://www.is.njit.edu/pubs/delphibook/

Lyytinen, K. (1987). Different perspectives on information-systems - Problems and solutions. *ACM Computing Surveys, 19*(1), 5-46.

Lyytinen, K., Mathiassen, L., & Ropponen, J. (1998). Attention shaping and software risk - A categorical analysis of four classical risk management approaches. *Information Systems Research, 9*(3), 233-255.

March, S., & Smith, G. (1995). Design and natural science research on information technology. *Decision Support Systems, 15*(4), 251-266.

Mathiassen, L., Saarinen, T., Tuunanen, T., & Rossi, M. (2007). A contingency model for requirements development. *Journal of Association of Information Systems, 8*(11), 569-597.

Mathiassen, L., Seewaldt, T., & Stage, J. (1995). Prototyping and specifying: Principles and practices of a mixed approach. *Scandinavian Journal of Information Systems, 7*(1), 55-72.

Matook, S., & Maruping, L. M. (2014). A competency model for customer representatives in agile software development projects. *MIS Quarterly Executive, 13*(2), 77–95.

McFarlan, F. W. (1981). Portfolio approach to information systems. *Harvard Business Review*, *59*(5), 142-150.

Morgan, D.L. (1997). *Focus groups as qualitative research* (2nd). Sage Publications.

Napier, N., Mathiassen, L., & Johnson, R. (2006). Negotiating response-ability and repeat-ability in requirements engineering. In *Proceedings of the international conference on information systems (ICIS).*

Niederman, F., Lechler, T., & Petit, Y. (2018). A research agenda for extending agile practices in software development and additional task domains. *Project Management Journal, 49*(6), 3-17.

Nunamaker Jr, J. F., Briggs, R. O., Derrick, D. C., & Schwabe, G. (2015). The last research mile: Achieving both rigor and relevance in information systems research. *Journal of Management Information Systems, 32*(3), 10-47.

Peffers, K., Tuunanen, T., & Niehaves, B. (2018). Design science research genres: Introduction to the special issue on exemplars and criteria for applicable design science research. *European Journal of Information Systems*, *27*(2), 129-139.

Peffers, K., Tuunanen, T., Rothenberger, M. A., & Chatterjee, S. (2007). A design science research methodology for information systems research. *Journal of management information systems, 24*(3), 45-77.

Persson, J. S., Mathiassen, L., Boeg, J., Madsen, T. S., & Steinson, F. (2009). Managing risks in distributed software projects: An integrative framework. *IEEE Transactions on Engineering Management, 56*(3), 508-532.

Plachkinova, M., Mood,y, G., & Peffers, K. (2018). Selecting communication artifacts for requirements engineering. *Journal of Information Technology Theory and Application (JITTA), 19*(4), Article 6.

Prat, N., Comyn-Wattiau, I., & Akoka, J. (2015). A taxonomy of evaluation methods for information systems artifacts. *Journal of Management Information Systems, 32*(3), 229-267.

Racheva, Z., Daneva, M., Herrmann, A., & Wieringa, R. J. (2010). A conceptual model and process for client-driven agile requirements prioritization. In P. Loucopoulos & J. L. Cavarero (Eds.) *Proceedings of the 2010 fourth international conference on research challenges in information science (RCIS)* (pp. 287-298). IEEE.

Randolph, J. J. (2005). Free-marginal multirater kappa (multirater κfree): An alternative to fleiss marginal multirater kappa. In *Proceedings of the Fifth Joensuu Symposium on Learning and Instruction.*

Randolph, J. J. (2008). *Online kappa calculator*. Retrieved from http://justus.randolph.name/kappa

Regnell, B., Höst, M., Natt och Dag, J., Beremark, P., & Hjelm, T. (2001). An industrial case study on distributed prioritisation in market-driven requirements engineering for packaged software. *Requirements Engineering Journal, 6*(1), 51-62.

Remus, U., Wiener, M., Saunders, C., & Mähring, M. (2020). The impact of control styles and control modes on individual-level outcomes: A first test of the integrated is project control theory. *European Journal of Information Systems*, *29*(2), 134-152.

Royce, W. W. (1970). Managing the development of large software systems: Concepts and techniques. In *Proceedings of the 9th International Conference on Software Engineering.*

Saarinen, T., & Vepsäläinen, A. (1993). Managing the risks of information systems implementation. *European Journal of Information Systems*, *2*(4), 283-295.

Schmidt, R., Lyytinen, K., Keil, M., & Cule, P. (2001). Identifying software project risks: An international Delphi study. *Journal of Management Information Systems, 17*(4), 5-36.

Sebastian, I. M., Ross, J. W., Beath, C., Mocker, M., Moloney, K. G., & Fonstad, N. O. (2017). How big old companies navigate digital transformation. *MIS Quarterly Executive*, *16*(3), 197-213.

Stewart, D., Shamdasani, P., & Rook, D. (2007). *Focus groups theory and practice*. Sage Publications.

Tuunanen, T., & Kuo, I. T. (2015). The effect of culture on requirements: A value-based view of prioritization. *European Journal of Information Systems*, *24*(3), 295-313.

Tuunanen, T., & Peffers, K. (2018). Population targeted requirements acquisition. *European Journal of Information Systems*, *27*(6), 686-711.

Tiwana, A., & Keil, M. (2004). The one minute risk assessment tool. *Communications of the ACM*, *47*(11), 73-77.

Truex, D. P., Baskerville, R., & Klein, H. (1999) Growing systems in emergent organizations. *Communications of the ACM*, *42*(8), 117-123.

Wallace, L., Keil, M., & Arun, R. (2004). Understanding software project risk: A cluster analysis. *Information & Management*, *42*(1), 115-125.

Walls, J. G., Widmeyer, G. R., & El Sawy, O. A. (1992). Building an information system design theory for vigilant EIS. *Information Systems Research*, *3*(1), 36-59.

Walls, J. G., Widermeyer, G. R., & El Sawy, O. A. (2004). Assessing information system design theory in perspective: How useful was our 1992 initial rendition? *Journal of Information Technology Theory and Application (JITTA), 6*(2), 43-58.

Appendix A: Requirements Risk Resolution

Requirements risk resolution techniques and categorization, adapted from (Mathiassen et al. 2007)

Name	Specification	Experimentation	Discovery	Prioritization
Affinity technique			*	
Aspect mining in requirements specification			*	
Attributed goal-oriented analysis	*		*	
Behavior analysis	*		*	
Box structure specification and design	*			
Brainstorming			*	
Business information analysis and integration technique	*		*	
Business process planning (BSP)	*		*	*
Card sorting			*	*
Cognitive mapping			*	
Contextual design	*		*	*
Cooperative prototyping		*		
CREV	*		*	
CREWS	*		*	
Critical success factors			*	*
Data flow diagram	*			
Decision analysis			*	
Delphi method			*	*
Deriving requirements from the existing system			*	
Domain-specific modeling	*			
EasyWinWin			*	*
Email/bulletin board			*	
Ends/means analysis			*	
Entity-relationship modeling	*			
Facilitated team			*	
Focus group			*	
Future analysis			*	
Goal modeling-oriented requirements elicitation	*		*	
Goal-oriented approach	*		*	
Group support systems and strategic business objectives			*	*
Guided brainstorming			*	
Human, social, and organizational requirements elicitation			*	
Inquiry cycle model – structure and describe requirements discussions	*		*	
Joint application design		*	*	
KAOS	*			

Laddering			*	
Lyee	*			
Machine rule induction	*			
Marketing and sales			*	
MIS intermediary	*		*	
Multidimensional data models	*			
Multidimensional scaling	*			
Nominal group technique			*	*
Normative analysis			*	
Object-oriented Z	*			
Open interview			*	
Open systems task analysis			*	
Participatory design		*	*	
Petri nets	*			
Petri nets combined with use cases	*		*	
Precision model			*	
Prime-CREWS	*		*	
Process analysis			*	
Protocol analysis			*	
Prototyping		*		
Quality function deployment	*			*
Repertoire grids			*	
Requirements generation model			*	
Requirements prototyping		*		
Requirements workshops			*	
Rich pictures	*		*	
Scenario-based requirements elicitation	*		*	
Semantic maps			*	
Socio-technical analysis			*	
Statecharts	*			
Strategic business objectives			*	*
Strategy set analysis			*	
Structured group elicitation method			*	
Structured interview			*	
Structured walkthroughs			*	
Support line			*	
Surveys			*	
Teach-back interview			*	
Testing		*	*	
Text analysis			*	
Trade show		*	*	
Usability lab		*		
Use cases			*	

Use of video in requirements elicitation				*	
User group				*	
User-interface prototyping			*	*	
Variance analysis				*	
VDM ++,VDM-SL		*			
Warnier-Orr diagrams		*			
Z		*			

About the Authors

Tuure Tuunanen is Chair Professor of Information Systems at the Department of Informatics and Media at the Uppsala University and Professor of Information Systems and Vice Dean of Research in the Faculty of Information Technology at the University of Jyväskylä. He leads the Value Creation for Cyber-Physical Systems and Services Research Group and Finnish Hub for Digitalization. He is also a global faculty fellow of the Center for Service Leadership at Arizona State University. He holds M.Sc. and D.Sc. (Econ.) degrees from the Aalto University School of Business.

Tero Vartiainen is professor in the Department of Computing Sciences at the University of Vaasa, Finland. He is also an adjunct professor at University of Jyväskylä. He holds a Ph.D. from University of Jyväskylä. His research interests lie in the areas of cybersecurity, IT services, energy informatics, IT project management, and computer ethics. Dr. Vartiainen's research has been published, e.g., at *Information Systems Journal*, *European Journal of Information Systems*, *Communications for Association of Information Systems*, and *IT and People*.

Sanna Kainulainen is a doctoral researcher at the University of Jyväskylä, Finland, and a member of the Value Creation for Cyber-Physical Systems and Services research group. Sanna is an experienced Business Information System Manager with a demonstrated history of working in the automotive market. She is a strong program and project management professional with a Master of Science (M.Sc.) focused on Information Technology from the University of Jyväskylä. Her doctoral research focuses on requirement risk management in continuous development.

Mehdi Ebrahim is currently working in NZ Health IT Sector. During the course of his 20+ year career in the IT industry, he has been part of many successful deliveries to Airports, Government Departments, and Telco Sector. Mr Ebrahim has also completed his Master Thesis in Information Systems from the University of Auckland with Honours. His topic was "Profiling ISD Risk: A Measurement Tool", which received the PMI NZ Research Achievement Award in 2009.