The Journal of Systems and Software 125 (2017) 271–288

Contents lists available at ScienceDirect

The Journal of Systems and Software

journal homepage: www.elsevier.com/locate/jss

A study of value in agile software development organizations

Hiva Alahyari [a], Richard Berntsson Svensson [b,*], Tony Gorschek [b]

[a] Department of Computer Science and Engineering, Chalmers | University of Gothenburg, Gothenburg, Sweden
[b] Department of Software Engineering, Blekinge Institute of Technology, Karlskrona, Sweden

ARTICLE INFO

Article history:
Received 1 February 2016
Revised 26 October 2016
Accepted 6 December 2016
Available online 14 December 2016

Keywords:
Value
Agile software development
Empirical

ABSTRACT

The Agile manifesto focuses on the delivery of valuable software. In Lean, the principles emphasise value, where every activity that does not add value is seen as waste. Despite the strong focus on value, and that the primary critical success factor for software intensive product development lies in the value domain, no empirical study has investigated specifically what value is. This paper presents an empirical study that investigates how value is interpreted and prioritised, and how value is assured and measured. Data was collected through semi-structured interviews with 23 participants from 14 agile software development organisations. The contribution of this study is fourfold. First, it examines how value is perceived amongst agile software development organisations. Second, it compares the perceptions and priorities of the perceived values by domains and roles. Third, it includes an examination of what practices are used to achieve value in industry, and what hinders the achievement of value. Fourth, it characterises what measurements are used to assure, and evaluate value-creation activities.

1. Introduction

Agile and Lean Software Development have gained much popularity during the last decade. The very first principle of Agile Manifesto reflects on the highest priority to be satisfying customers through delivery of valuable software. Similarly Lean principles have a particular emphasis on Value and the first principle of Lean software development considers every activity or process to be waste unless it adds some Value to either the company or its customers (Poppendieck, 2011).

The focus on Value is in line with most studies looking at critical success factors for software intensive product development, distinguishing successful from failed software projects, showing that the primary critical success factors lie in the Value domain (Boehm, 2006b). The understating of Value as a concept is however somewhat limited (Dingsøyr et al., 2012; Dybå and Dingsøyr, 2008; Racheva et al., 2009). Value is traditionally seen as profit generation and adding Value is a pecuniary activity that needs to be taken into account from a software business perspective. Value is however a much more complex concept as described by Khurum et al. (2013); 2014) in their Software Value Map. There they elaborate not only on Customer Value, and Financial Value for the development company, but also on internal Value Aspects such as the Value (or Value degradation of assets) and legacy within the development company.

Independent how you define and use Value however, the basic aim for a company developing software intensive products and services (called company from now on) is to maximise Value creation for a given investment. For this to be possible it is necessary to understand what is considered Value and what are the strategies that drives Value and assures the Value creation (Aurum and Wohlin, 2007).

To the best of our knowledge no empirical study has investigated how different companies interpret the Value concept, to what extent Value and different Value Aspects are defined, what Value Aspects they consider important to achieve, and how Value Aspects are assured and/or measured. This paper presents the results of an empirical study that includes data collected through in-depth interviews with 23 participants from 14 different software development organisations in Sweden.

The remainder of this paper is organised as follows. In Section 2, the background and related work are presented. The research methodology is described in Section 3, and Section 4 presents the results and relates the findings to previous studies. Section 5 holds the main conclusions.

2. Background and related work

Agile methodologies with the promise of satisfied customers through early and continuous delivery of valuable software have

* Corresponding author.
E-mail addresses: hiva@chalmers.se (H. Alahyari), richard.berntsson.svensson@bth.se (R. Berntsson Svensson), tony.gorschek@bth.se (T. Gorschek).
URL: http://www.rbsv.eu (R. Berntsson Svensson), http://www.gorschek.com (T. Gorschek)

http://dx.doi.org/10.1016/j.jss.2016.12.007
0164-1212/

brought unprecedented changes to the software engineering field since the articulation of the agile manifesto in 2001 (Agile Manifesto). Poppendieck and Poppendieck (2003) state that the success of many of the practices of Agile Software Development (ASD) can be explained by understanding the principles of Lean software development. The main principle of Lean states that all activities and work products that do not contribute to the customer Value are considered waste (Poppendieck and Poppendieck, 2003).

While a substantial amount of papers (e.g. Conboy and Morgan, 2011; Korkala and Abrahamsson, 2007; Mishra and Mishra, 2011; Petersen and Wohlin, 2009; Wang et al., 2012) have been published in recent years on issues related to agile software development, contributions often have been around particular agile methods or comparing agile and other development processes. Conboy and Morgan (2011) looked into the applicability and implementation of open-innovation in agile environments and challenges when combining agile and open-innovation principles. In Korkala and Abrahamsson (2007), the authors conducted two case studies to investigate the communication in distributed agile development. In addition, Mishra and Mishra (2011) investigated how agile development methodologies and management approaches are used in development of complex software projects, while Petersen and Wohlin (2009) identified issues and advantages when implementing incremental and agile practices in large-scale organisations. Wang et al. (2012) looked into how lean software development approaches can be applied in agile software development. In a study by Dingsøyr et al. in 2012 they examined publications on Agile, during the decade after introduction of the Agile methods in 2001. The results show that the majority of research was related to the differences between process-oriented approaches such as CMM/CMMI and Agile methods such as XP (Dingsøyr et al., 2012). However, no study was found to have a dedicated focus on the concept of Value and Value assurance (Dingsøyr et al., 2012).

Although the majority of the published papers do not specifically look into Value and Value creation, some studies (e.g. de Azevedo Santos et al., 2011; Conboy, 2009; Hoda et al., 2011; Maruping et al., 2009; Petersen and Wohlin, 2010) have been published about Value creation through agile practices. However, they are limited to a few Value Aspects such as quality (de Azevedo Santos et al., 2011; Conboy, 2009), simplicity (Conboy, 2009; Maruping et al., 2009), frequent releases (Hoda et al., 2011; Petersen and Wohlin, 2010), and economy (Conboy, 2009). In addition, Racheva et al. (2009) conducted a systematic review on how business Value is created in agile projects. They found, with very few exceptions, that most published studies take the concept of business Value for granted and do not state what it means in general as well as in the specific study context. Racheva et al. (2009) could not find any study which clearly indicates how exactly individual agile practices, or groups of practices, create Value. The need for conducting empirical research into Value and Value creation in agile projects was mentioned as one of their implications.

Chase (2001) proposed a list of Value aspects that an individual task could contribute towards. However, a detailed account of value considerations relevant for different perspectives like customer and internal business Value are missing. Several other researchers (e.g. Conboy, 2009; Fogelstrom et al., 2010; Song et al., 2009) have presented Value constructs and corresponding valuation/measurement solutions needed for making decisions about software product development. However, the contributions are often isolated and with a limited focus on, for example, only cost, or only product characteristics such as simplicity and usability. Moreover, some researchers (e.g. Cleland-Huang, 2015; Golfarelli et al., 2013) have looked into how to use Value as input to prioritization and release planning. In Cleland-Huang (2015), the author describes an approach that takes value into account when prioritizing, while the Golfarelli et al. (2013) proposed an optimization model that creates a release plan that maximizes the business Value from a user perspective. However, none of these papers investigated what Value is, how it is defined and used in industry, or how is it measured and assured. Instead, in Cleland-Huang (2015), the author used return on investment as Value, while Golfarelli et al. (2013) used the value aspect of perceived value (from a customer perspective) as defined in Khurum et al. (2013).

A comprehensive description of existing software value aspects is provided by Khurum et al. (2013), who distinguish major four perspectives, *Customer, Internal Business, Financial*, and *Innovation and Learning*. The *Customer* perspective is concerned with the Value proposition that the company operates to satisfy customers, thus generate more sales to the most relevant (i.e. the most profitable) customer groups through the maximisation of Value aspects such as Perceived Value and Usability (Khurum et al., 2013). The *Internal Business* perspective focuses on Value aspects that are concerned with internal aspects that can be taken into consideration, such as architectural aspects, but also values tied to differentiation and maintaining quality of development base (Khurum et al., 2013). The *Financial* perspective includes the aspects and strategies that a company takes into account in order to contribute to the bottom-line improvement of the company. It embodies the long-term strategic goals of the organisation in traditional financial terms (Khurum et al., 2013). The *Innovation and Learning* perspective takes into account the intangible possessions of an organisation. It focuses mainly on skills and capabilities and internal practices that are required for supporting the Value creating processes (Khurum et al., 2013).

Despite the importance of Value, and that Value is considered critical in Agile software development, to what extent companies' utilise Value, how Value is defined, and used, is largely unexplored. Chase (2001) proposes a list of Value aspects; however, a detailed account of value considerations relevant for different perspectives missing. Although Khurum et al. (2013) provide a consolidated view on the concept of Value, they do not look into how different companies interpret the Value concept, nor what Value Aspects are considered important to achieve and how these Value Aspects are assured and/or measured, which is the purpose of this study.

3. Research methodology

The investigation presented in this paper was carried out using a qualitative approach, namely in-depth semi-structured interviews (Robson, 2002). The objective of qualitative research is to study and understand phenomena within their real-life context (Robson, 2002). A qualitative approach is useful when the purpose of the study is to explore an area of interest where the aim is to improve the understanding of the phenomena that has not yet been investigated fully (Robson, 2002). Although Khurum et al. (2013) provide four major perspectives of Value with associated Value Aspects (VA from now on), they do not look into how different companies interpret the Value concept, nor what VAs are considered important to achieve. Thus, a further in-depth understanding of Value is needed. Since the purpose of this study was to gain an in-depth understanding of Value, its definition (into different types, called, which of these VAs are considered the most important in industry, and how is Value used and measured/assured in agile software development organisations, a qualitative approach was chosen.

Predicting the probable diversity of definitions and set of VAs that could be collected, semi-structured interviews would best meet the objectives of this study. In addition, we choose to use interviews as the concept of Value could be very contextually dependent, and it could be defined and approached differently amongst companies. For this reason it was important to have a presence when eliciting the data making it possible to elaborate on what

Table 1
SDO characteristics.

SDO	Participant/Role	Type of interview	Identification of Value Aspects	Domain	Number of employees (SDO)	Number of used agile practices
A	1. Agile driver (PM)	Joint	Joint	Telecom	300	30
A	2. Product owner (PO)	Joint	Joint	Telecom	300	33
B	1. Product owner (PO)	Joint	Joint	Telecom	300	25
B	2. Project Manager (PM)	Joint	Joint	Telecom	300	29
C	1. Product owner (PO)	Joint	Separate	Telecom	300	11
C	2. Project Manager (PM)	Joint	Separate	Telecom	300	24
D	1. Program responsible (PM)	Joint	Separate	Telecom	55	46
D	2. Product owner & System architect (PO)	Joint	Separate	Telecom	55	42
E	1. Customer Project Manager (PM)	Single	Single	Telecom	28	20
F	1. Process Manager (PM)	Joint	Separate	Automotive	25 (only team)	15
F	2. Product owner/Technical expert (PO)	Joint	Separate	Automotive	25 (only team)	15
F	3. Architect/Requirements (PO)	Separate	Separate	Automotive	25 (only team)	15
G	1. Product owner (PO)	Joint	Separate	Automotive	15 (only team)	21
G	2. Scrum master (PM)	Joint	Separate	Automotive	15 (only team)	27
H	1. Scrum master (PM)	Separate	Separate	Automotive	100	9
H	2. Product owner (PO)	Separate	Separate	Automotive	100	26
I	1. Project manager/developer (PM)	Single	Single	Automotive	N/A	22
J	1. Scrum master (PM)	Joint	Separate	Defence industry	200	42
J	2. Product owner/System architect	Joint	Separate	Defence industry	200	31
K	1. Scrum master (PM)	Single	Single	IT-Consultancy	110	39
L	1. Scrum master/verification responsible (PM)	Single	Single	IT Management consultancy	180	39
M	1. Consultant/Agile coach (PM)	Single	Single	Consultancy	N/A	45
N	1. Business consultant (PO)	Single	Single	Consultancy	6 (only team)	21

we were looking for. Interviews also allow the possibility to ensure a deeper understanding of the Value perspectives. Interview based interactions between the interviewer and participant enabled us to uncover the ambiguities when necessary and compensate for different definitions and approaches. In addition, the interviewer had the chance to validate the questions with the interviewee lessening the chances of misunderstandings. That is, the interviewer went back to the interviewee to validate the interviewers interpretation of the results to minimise misinterpretations and validate the results.

The following research questions provided a focus for the empirical investigation:

- **RQ1:** How is the concept of Value defined in Agile organisations developing software intensive products and systems?
- **RQ1.1:** What Value Aspects are considered most important?
- **RQ2:** How is the concept of Value used in Agile organisations developing software intensive products and systems?
- **RQ2.1:** How is Value measured/assured/evaluated?

3.1. Research design and data collection

The investigation can be divided into three phases:

Planning/Selection. The two target roles within each software development organisation were the process responsible, and the role/function responsible for decision of what is prioritised and selected for implementation (often product owner). That is, the role responsible for the overall development process, and the role responsible for what features, and what quality, is delivered.

A combination of maximum variation (Patton, 2002) and convenience sampling (Patton, 2002) were used for selecting the participants in this study. Within our industrial network, for each company, we conducted "gate-keepers" at each software development organisation. They helped us in identifying the two participants that were the most suitable to participate in this study. That is, the researchers neither influenced the selection of participants nor had any personal interest towards any of the participants. Depending on the size of the organisation and the definition of areas of responsibility, the two target roles could be handled either by two people or only one. The roles had different titles such as Project Manager/Leader, Agile expert, Scrum Master, Product Owner/Manager, and Technical expert. In total, 23 participants at 14 Software Development Organizations (SDO) from nine companies participated, resulting in 23 data points.

All nine companies and 14 SDO develop embedded systems and, according to the participants themselves use Agile methods. The SDO themselves vary in respect to size, type of products, and application domain; a characterisation (following the guidelines of Ivarsson and Gorschek, 2011) can be seen in Table 1 (more details are not revealed for confidentiality reasons) following the recommendations of Ivarsson and Gorschek (2009).

Data Collection. The data collection method used was semi-structured interviews with open-ended questions (Robson, 2002). In some of the interviews, one participant and one interviewer attended; while in others two participants and one interviewer were present, as shown in Table 1, column "Type of interview". In addition, the column "Identification of Value Aspects" in Table 1 shows if the identified VAs were identified jointly or separately. Although we preferred to conduct all of the interviews separately and having the interviewees to identify the VAs individually, some of the participants from a number of SDO insisted to have the interviews and/or the identification of VAs jointly. Moreover, for some SDO, only one participant was able to participate in the study to answer questions in relation to Value in ASD (see Table 1).

During the interviews, first the purpose of the study was presented to the participant., followed by demographic questions, e.g. the participants were asked to identify which agile practices that were utilised in their development processes based the Agile Alliance's list of 58 agile practices (Agile Alliance). Then, questions about how they define Value and what they consider as Value were discussed in detail. The participants answered the questions from the perspectives of their SDO and their role. In addition, the participants were asked to prioritise the identified VAs using the 100-dollar method (Leffingwell and Widrig, 2003). The prioritisation was conducted using an empty paper sheet with two separate columns, one for the identified VAs and one for the distribution of the 100 dollars. All interviews were conducted in English and varying in length from 60 to 120 min. All the interviews were conducted on-site and both recordings and written extensive notes were taken in order to facilitate and improve the analysis process.

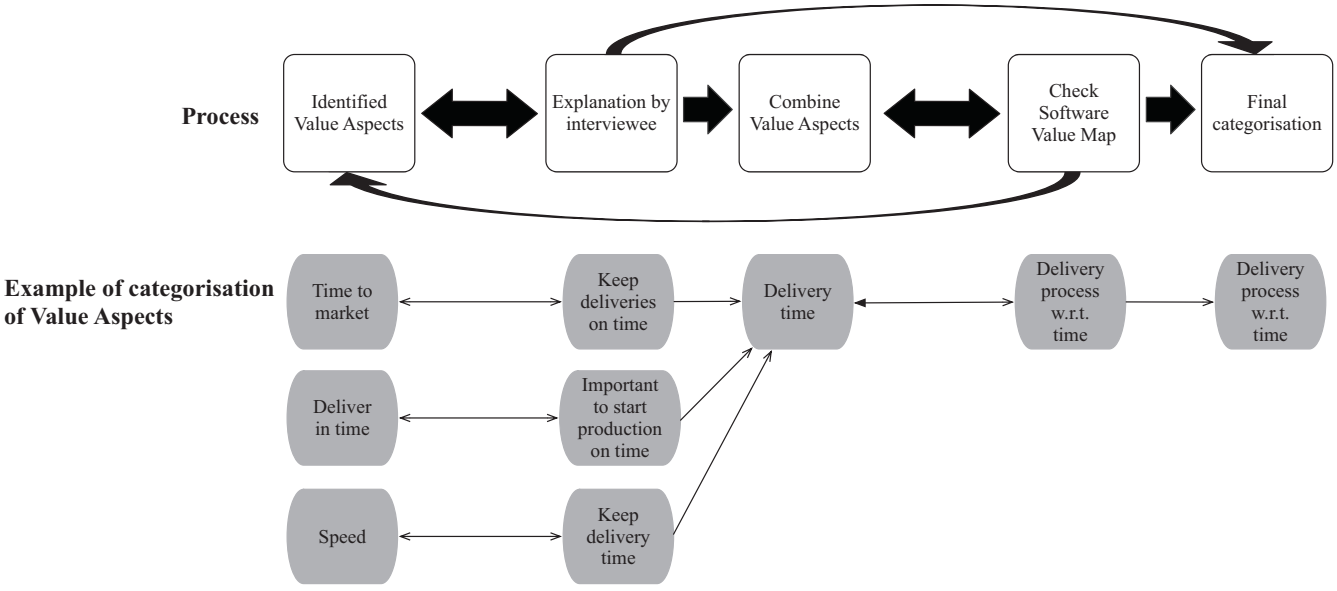

Fig. 1. The process of identifying the VA categories.

Analysis. All the written extensive notes from the interviews were imported into spreadsheets to enable analysis of qualitative data. Data was analysed in this study using content analysis (Robson, 2002), which includes marking and discussing interesting sections in the extensive notes.

In our study, once all VAs had been identified and categorised, we checked the explanation the interviewee gave for each VA to make sure that what the interviewee stated as a VA corresponded to their explanation, as shown in Fig. 1. For example, one interviewee stated that *time to market* is a VA and gave the explanation that it is about *keeping deliveries on time*, while another interviewee identified *speed* as a VA with the explanation that it is about *keeping delivery time*, as illustrated in Fig. 1. Then, we combined all VA in the same category into a higher abstract VA, e.g. the VAs *time to market, speed*, and *deliver in time* where combined into a new VA called *delivery time*, as illustrated in Fig. 1. Each of these combined categories was then discussed among all authors. The next step in the categorisation of the VA was to look into the SVM to see if there was any category that was similar to our combined categories. This step included to read the description of the VA in the SVM, match it, if possible without combined category and sometimes with the originally stated VA from the interviewee. This was done to make sure that we did not assign an identified VA to a wrong category. Then, if there was a matching category in the SVM (we compared the description in the SVM with the explanations given by the interviewees) we assigned our combined category to the one identified in the SVM. For example, the combined category of *delivery time* was assigned to the VA of *Delivery process w.r.t. time* from the SVM, as illustrated in Fig. 1. In all cases, we identified a similar category in the SVM.

During the data analysis, the authors examined the VAs and the categories from different perspectives. The analysis of data was an iterative and interpretative process. If a certain VA belonged to more than one category (see categories in Table 3), the assigned dollars by the participants were divided between the categories. The results from the analysis are found in Section 4.

3.2. Validity threats

For this study, as for any study, there are validity threats worth discussing. We consider the four perspectives of validity threats as presented in Runeson and Host (2009).

Construct validity. These threats refer to the relation between the research method and the observations from the study (Runeson and Host, 2009). There is a threat that the practitioners may have misunderstood the researchers and the questions asked due to the use of different terms and references. In order to minimise this threat, we used peer debriefing (Runeson and Host, 2009). Peer debriefing suggests that the research is carried out by a group of researchers instead of a single researcher, which may minimise the threat of being biased towards one researchers terms and references. Moreover, research colleagues were involved during the research design to review and give feedback. As a final step to minimise the threat of misunderstandings, the interview questions were designed based on the research questions and reviewed by the authors and research colleagues. Another threat is related to the presence of a researcher during the interviews. The practitioners may have felt threatened to express their opinions. Hence, the practitioners may have answered the questions in relation to what they thought were the researchers expectations. This threat was minimised by the guarantee of anonymity, and the answers from the interviews were only to be used by the researchers. A third threat to construct validity is the selection of the companies and the practitioners. The companies were selected within our industrial network, which provided the researchers with the needed trust from the companies. Since the practitioners were not fully randomly sampled, there is a threat to selection bias. That is, only practitioners with a positive attitude towards Value are selected. However, the practitioners were selected based on their roles by a "gate-keeper" at each company, as shown in Table 1.

Internal validity. Since this study is of empirical nature, the risk of identifying incorrect VA and factors is a validity threat. In order to mitigate this threat, we recorded the interviews to assure that we did not misunderstand the interviewee. In addition, the researchers had the chance to validate the questions and answers with the participants, which minimises the chances of misunderstandings. Furthermore, more than one researcher participated in each step of the analysis. This strategy also partly helps in minimise the threat of incorrect generalisations when abstracting VA, value creation and measurements, activities need for achieving the VAs, and barriers for achieving the VAs. This risk could be addressed through more case studies and systematic theory building based on empirical data (Eisenhardt and Graebner, 2007).

Table 2
The most and least used agile practices.

Agile Practice	Frequency	%
Backlog	23	100
Estimation	22	95.7
Agile Team	22	95.7
Backlog Grooming	21	91.3
Unit Testing	20	87.0
Acceptance Testing	19	82.6
Daily/Scrum Meeting	19	82.6
Integration	19	82.6
Iteration	18	78.3
Burn-down Chart	18	78.3
⟩	⟩	⟩
BDD (Behavior Driven Development)	4	17.4
Mock Objects	4	17.4
Rules of Simplicity	3	13.0
Story Mapping	3	13.0
Three C's (Card, Conversation, Confirmation)	3	13.0
Ubiquitous Language	2	8.7
INVEST	2	8.7
Given-When-Then	1	4.3
CRC (Class, Responsibilities, Collaborators) Cards	1	4.3
Project Chartering	0	0
Niko-Niko Calendar	0	0

Table 3
Identified value aspects.

ID	Category	Frequency (number of participants)	Amount of spent dollars
1	Delivery Process w.r.t. time	15 (65%)	372 (16%)
2	Perceived Quality	12 (52%)	257 (11%)
3	Cost (product, project)	11 (48%)	235 (10%)
4	Actual Quality (product, code, architecture, or stability)	10 (44%)	261 (11%)
5	Processes, Ways of Working (WoW), Tools	10 (44%)	175 (7.5%)
6	End-User Performance, Usability	8 (35%)	140 (6%)
7	Innovation, Knowledge of Organisation	7 (30%)	100 (4%)
8	Customer Relationship	6 (26%)	100 (4%)
9	Knowledge of feature Value for customer or product usage	6 (26%)	147 (6%)
10	Revenue, Business Value	5 (22%)	140 (6%)
11	Functionality	5 (22%)	135 (6%)
12	Non-Functional Requirements, Hedonic Value	4 (17%)	78 (3%)
13	Competitiveness	3 (13%)	70 (3%)
14	To keep positive attitude, Professionalism	2 (9%)	35 (1.5%)
15	Maintainability	2 (9%)	20 (1%)
16	Reliability	1 (4%)	35 (1.5%)

External validity. These threats are concerned with the ability to generalise the results beyond the case companies in this study (Runeson and Host, 2009). The results of this study are limited to the investigated case companies. However, the objective of qualitative studies is rarely to generalise beyond the actual setting. Instead, qualitative studies focus on explaining and understanding the investigated phenomena. In addition, the design of qualitative studies is difficult, if not impossible, to replicate since identical circumstances may not be possible to recreate. However, by understanding the investigated phenomena in one setting may help in understanding other situations. This means, for the findings to be generalisable, the context and characteristics of the case companies in this study needs to be compared with the context of interest. To help the reader to understand the relevance of the study, and to be able to compare with other situations, the characteristics of each company in this study are presented in Table 1.

Reliability. Threats to reliability refers to the dependency between the analysis and the specific researcher. To minimise the threats to the reliability of this study, an interview guideline was created to make sure that all relevant areas were covered. Moreover, in order to minimise the threat of wrong interpretations, all of the 23 interviews were recorded. Furthermore, member checking was used during the interviews for recording and prioritising the VAs. During the interviews, the participants identified VAs were added to an empty paper sheet with clear and separate columns, one for the identified VAs, and another column in which the participants prioritised each of the identified VAs by distributing the 100 dollars.

4. Results and analysis

In order to provide a better understanding of the context and characteristics of the SDO in this study, their agile development processes, and their compliance with agile practices, the participants were asked to identify which agile practices that were utilised in their development processes. Across all 14 SDO, 56 out of the 58 agile practices identified in Agile Alliance were used. The average number of used agile practices amongst all of the 23 participants was 26. The most and least used agile practices can be seen in Table 2.

The most frequently used agile practice was *Backlog* (n = 23), followed by *Estimation* (n = 22), *Agile Team* (n = 22), and *Unit Testing* (n = 20). In contrast, the least frequently used agile practices were *Niko-Niko Calendar* (n = 0) and *Project Chartering* (n = 0), followed by *CRC Cards* (n = 1), and *Given-When-Then* (n = 1). The agile practice *Niko-Niko Calendar* was not a familiar practice among the participants. However, some participants had heard of its other name, namely *Mood Board*.

With regards to the domain and the size of the SDO, we could not find any statistical significant correlation between used agile practices and the domain, or the size of the SDO. The only difference between the domains and used agile practices was the number of used practices. The average number of used agile practices in the defence industry was 37, while the consultancy companies used 36 agile practices on average. For the telecom industry, 26 practices were used in average, while the average for the automotive industry was 19 used agile practices.

In general, the most used agile practices among the SDO in this study were similar to the most used practices reported in the literature (Diebold and Dahlem, 2014; Jalali and Wohlin, 2010; Version one). *Continuous Integration* has been reported as one of the most used agile practices in several studies, e.g. Diebold and Dahlem (2014); Jalali and Wohlin (2010);Version one, which is inline with our results (although *Continuous Integration* was not among the top ten used Agile Practices in Table 2, it was used by 17 participants, hence it was the 11th most used practice). Four of the top six agile practices in the VersionOne survey (Version one) (*Daily/Scrum meeting, Iteration, Unit Testing*, and *Estimation/Burn-down Chart*) also appear among our most used agile practices.

In a study by Jalali and Wohlin (2010), the two most frequently used agile practices were *Continuous Integration* and *Daily/Scrum meetings*, which also appear among the most frequently used agile practices in our study. However, there are differences between our results and the results in Jalali and Wohlin (2010). Among the top ten used practices in Jalali and Wohlin (2010), only three (*Continuous Integration, Daily/Scrum meetings*, and *Backlog*) were among our top ten. The difference between the results may be explained by the focus of the studies. In the study by Jalali and Wohlin (2010), the focus was most frequently used agile practices in the context of Global Software Engineering, while we focused on agile software development organisations.

Table 4
Actual stated value aspects from SDO C.

Participants from SDO C (Telecom domain)			
Participant one: PO		Participant two: PM	
Value Aspect	Dollars	Value Aspect	Dollars
User experience Value (in regards to both quality of the work and innovation aspect)	40	Perceived quality (external)	20
Value in reaction to cost	20	Innovations/patents	15
Differentiation Value	20	Customer relationships (cost)	15
Usefulness of features in general (pragmatic Value)	20	Customer relationships (speed, responsiveness)	15
		Human knowledge	10
		Cost of development	10
		Internal quality (e.g. good code)	5
		Customer specific	5
		Hygiene	5

Diebold and Dahlem (2014) conducted a systematic mapping study to identify empirical evidence on the current usage of agile practices in industry. The two most used agile practices in Diebold and Dahlem (2014) were *Time Boxing* and *Planning Meeting*, which were among the most used agile practices in our study. In our study, *Time Boxing* is called Iteration (see Table 2), while *Planning Meeting* in Diebold and Dahlem *[21]* refers to *Daily/Scrum Meetings* in our study (see Table 2). Moreover, The agile practice *Estimation* was the second most used agile practice in our study, which is inline with Diebold and Dahlem (2014) where *Team-Based Estimation* was assigned to be part of the *Planning Meeting* category of agile practices.

4.1. How value is defined and prioritised (RQ1 and RQ1.1)

We asked the participants how they define and use Value, and what Value means to them (RQ1). The participants could state anything that adds Value to their organisation, product, or customers. Our approach was not to impose preconceived definitions of Value but try to understand industrial practice and the participants' own interpretations of Value. In total, the 23 participants identified 134 Value Aspects (VA) with an average of six VAs per participant. The 134 VAs were categorised into 16 categories. The naming of the final 16 categories (see Table 3) that emerged from the 134 VAs were inspired by the VA in the Software Value Map (Khurum et al., 2013). In addition, we asked the participants to prioritise the importance of their identified VAs. The participants used the 100-dollar method (Leffingwell and Widrig, 2003) for the prioritisation.

Looking at Table 3, we see that *Delivery Process w.r.t. time* (15 of 23 participants, from all four domains) was the most frequently mentioned VA, followed by *Perceived Quality* (12 of 23, from telecom and automotive domain), *Cost* (11 of 23, from all four domains), *Actual Quality* (10 of 23, from all four domains), and *Processes, Ways of Working and Tools* (10 of 23, from all four domains).

In general, there is a relation between the frequency of the mentioned VAs and total amount of spent dollars, as illustrated in Table 3. However, there are a few exceptions, although the differences are minor. Looking at *Cost* and *Actual Quality*. *Cost* was mentioned by 11 participants while *Actual Quality* was mentioned by 10, but *Actual Quality* received 261 dollars in total while *Cost* received 235 dollars. Moreover, *Actual Quality* and *Processes, Ways of Working and Tools* (ID 5 in Table 4) were mentioned with the same frequency (10 participants), but *Actual Quality* was assigned more dollars. Although *Actual Quality* was not mentioned as frequently as *Cost* and *Processes, Actual Quality* seems to be very important for some of the participants.

Several participants explained that the importance of deadlines is the reason for stating *Delivery Process w.r.t. time* as a VA. One participant explained the importance of deadlines, *"delivery on time is important, project manager has deadlines based on customer and project needs"*. One reason for the frequency of *Perceived Quality* was explained by one participant from the telecom domain, *"quality is very important but sometimes you take it for granted"*. Another participant from the automotive domain explained that he would like to see their organisation to be known as "best in class" in "quality assurance". However, no details were offered as to if this constituted value from a (external) customer perspective, or if it was an internal business view.

One reason of why *Delivery Process w.r.t. time* was viewed as the most important VA could be that all SDO use Agile Software Development (ASD) processes. A key characteristic of any ASD process is an explicit focus on delivering valuable software to the customer (Agile Manifesto), hence the ASD process is a Value creation process where delivery time plays an important role, and is among the main principles of Agile Manifesto, Poppendieck and Poppendieck (2003). This was supported by one interviewee who reflected on this as an obvious fact by stating that *"deliver often and in time are Agile principles"*. *Perceived Quality* is partly related to Quality Requirements (QR), e.g. usability and performance. The importance of *Perceived Quality* is inline with Svensson et al. (2012) where QR were labeled as critical in software product development. In addition, not dealing with QR may lead to increased time-to-market, which is related to the VA of *Delivery Process w.r.t. time*. Therefore, it is not surprising that *Delivery Process w.r.t. time* and *Perceived Quality* are viewed among the most important VAs among the participants in this study. However, interesting enough is that no details in relation to if, or how *Perceived Quality* was evaluated.

The VA of *Cost* (ID 3 in Table 3) consists of three perspectives, (1) cost from a project perspective in terms of planning and adherence to plan to make sure that the company can handle the resources, e.g. the development team, (2) the relative perceived value by the customer given the cost of the product in relation to the value perceived, and (3) the company's return on the investment given the cost of the product. One reason why *Cost* (from a project perspective) was a frequently mentioned VA was explained by one participant: *"we have to stay on budget, project management wants that"*. Several other participants mentioned similar reasons (from a return on investment perspective), e.g. *"it is common sense"* that cost and to be cost efficient must be considered. Another participant explained that *Cost* is important because we need to consider the customer revenue by stating, *"based on the information from the feature owner and competitor analysis it is important to consider the customer revenue"*. McDougall and Levesque (2000), suggested that if companies are seeking to improve customer satisfaction through perceived Value, they need to consider the results or benefits customers receive in relation to the price paid and other costs associated with the purchase. This is supported by one of the participants (from the perceived value by the customer perspective), who explained that *"it is important for us that our customer can save cost and so the developed e.g. feature is profitable for them"*. According to Khurum et al. (2013), cost is not considered a VA in itself, cost is rather seen as a qualifier of VAs that needs/should be maximised. For example, including a feature that is aimed towards enhancing e.g. functionality, adds Functional Value, but has a cost. One explanation of why *Cost* in itself is seen as a VA among the participants in this study may be that it can be viewed as an internal perspective for the purposes of pricing, but also planning activities to utilise resources in the SDO (Aurum and Wohlin, 2007; Boehm, 2006b). However, this type of distinction was not made by any of the participants in this study when they identified cost as a VA.

The VA of *Processes, Ways of Working, and Tools* (ID 5 in Table 3) was the fourth most frequently mentioned category of VAs. Most

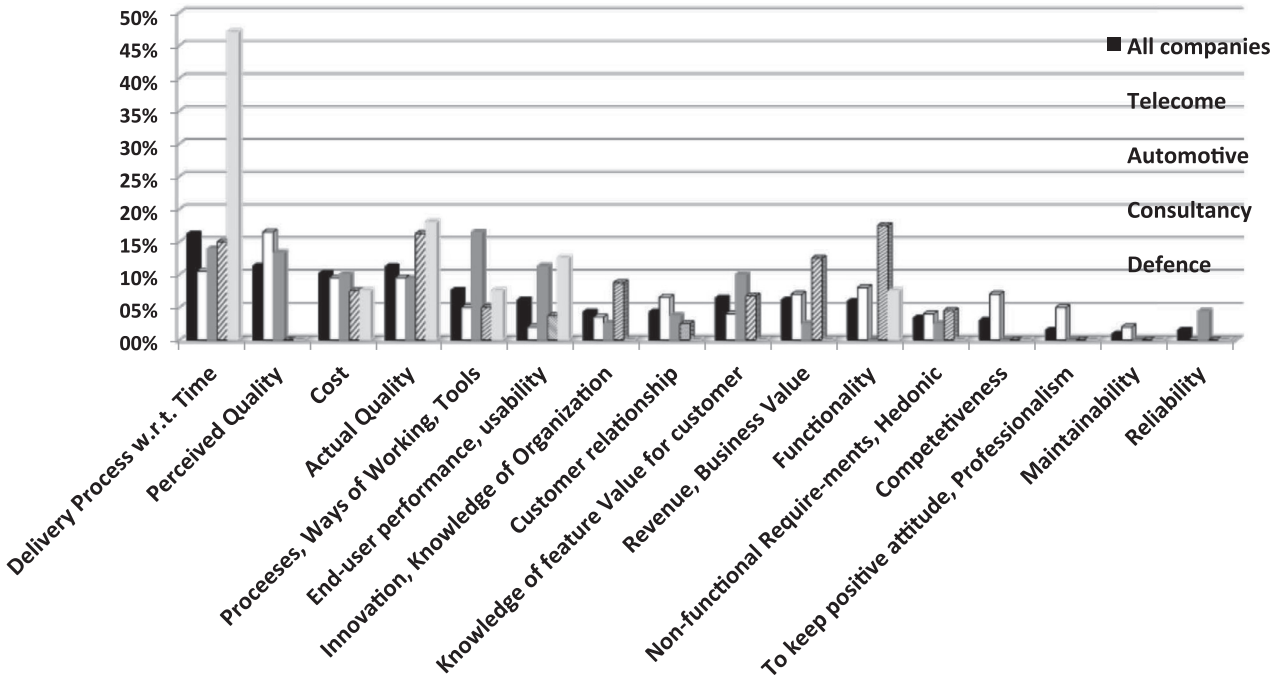

Fig. 2. Prioritised value aspects per domain.

of the participants mentioned the need to improve (at different levels) their agile ways of working as the reason. Other participants explained that the size of the organisation and following simplicity, in terms of processes, were other reasons. One participant further explained, *"simplicity in our processes is very important, specially that we are a big organisation"*. No connection to external (customer) VAs were mentioned, e.g. through simplicity we offer better external value. Rather this seems to be more of an internal VA as described by Khurum et al. (2013).

In general, Time (*Delivery Process w.r.t. time* in Table 3), Quality (*Perceived Quality* and *Actual Quality* in Table 3), and *Cost* (ID 3 in Table 3) were viewed as the most important VAs. This is inline with the golden triangle, i.e. schedule (time), budget (cost), and quality (Westerveld, 2003) of how Project Management success is defined. According to Bloch et al. (2012), large IT projects tend to run 45% over budget, 7% over time, and deliver 56% less Value than predicted, thus it is no surprise that the participants in this study prioritised time, quality, and cost as the most important VAs. Although the VAs of *Delivery Process w.r.t. time* and *Cost* were viewed among the most important VAs, and are inline with the golden triangle, even if the project is delivered on time and within budget (*Cost*) it does not mean that the product delivers high value for the customers.

Few of the participants motivates their important VA with regards to product Value, instead the VAs are motivated in relation to internal Value. Hence, there seems to be a project focus among the participants in this study, despite that we interviewed Product Owners who should focus on transforming the product vision into a successful product (Agile Alliance). This is not inline with Gorschek and Davis (2008), who looked into sub-optimisation, e.g. that project managers' focus on the project and not the product. Thus, *Functionality*, and *Perceived Quality* (e.g. usability, performance) were expected to have a higher priority. Hence, the short-term values of the project get a higher priority than the long-term values of the product (which is inline with the results of RQ2, see Section 4.2). Not prioritising, e.g. usability and performance leads to a lower overall quality of the entire system, and thus the value of the system decreases (Svensson et al., 2012).

Processes, Ways of Working, and Tools (ID 5 in Table 3) was viewed as the fifth most important VA among the participants in this study. This may be explained by the common assumption that the quality of the software development process is directly related to the quality of the software product (Kitchenham and Pfleeger, 1996). Hence, *Perceived* and *Actual Quality* are important VAs, which is inline with the results in this study. However, it is noticeable that no real distinction was made between *Actual* and *Perceived Quality* among the participants in this study. *Actual Quality* is related to identifying the important quality characteristics that are important for the product, and then find factors in development and testing, which affect these quality characteristics, to ensure that the product works as intended. On the other hand, *Perceived Quality* is the customer's judgment about the excellence or superiority of the product (Zeithaml, 1988). It is important to deliver a software product with high *Actual Quality*; however, if the customer's perception of the quality is low, the product may not be successful. In addition, an excellent software development process is more likely to lead to successful projects, and at the same time the quality of the product increases, while time-to-market improves (Kitchenham and Pfleeger, 1996).

4.1.1. Difference between domains

In order to improve the understanding of the important Value Aspects (VA) in industry, we analysed the identified and prioritised VAs based on the four different domains. The participants in this study were from 14 SDO from nine different companies in four domains. Five SDO were from the Telecom domain, four from the automotive domain, four from consultancy, and one SDO from the defence domain.

Looking at Fig. 2, we see that there are differences between the domains in terms of the importance of different VAs, but also in terms of how many VAs that are prioritised per domain.

Looking at the VA of *Delivery Process w.r.t. time*, overall, all SDO spent 16% of the total amount of dollars on this aspects. However, the defence domain assigned 47% of their total amount of dollars to this VA, while the consultancy domain spent 15%, the automotive domain spent 14%, and the telecom domain 10%. Furthermore,

the defence domain prioritised six of 16 VAs (prioritised *Delivery Process w.r.t. time, Cost, Actual Quality, Processes, WoW and Tools, End-User Performance-Usability*, and *Functionality*), while the SDO in the consultancy domain prioritised 11 VAs (no priority given to *Perceived Quality, Competitiveness, Professionalism, Maintainability*, and *Reliability*), the automotive domain prioritized 12 (no priority given to *Functionality, Competitiveness, Professionalism*, and *Maintainability*), and the telecom domain prioritised 15 of 16 VAs (no priority given to *Reliability*), as illustrated in Fig. 2.

The defence domain in this study viewed *Delivery Process w.r.t. time* as their most important VA. The telecom and automotive domains ranked *Delivery Process w.r.t. time* as the second most important VA, while it was viewed as the third most important aspect for the consultancy domain.

One reason why the defence domain ranked *Delivery Process w.r.t. time* as more important than the other domains may be due to their large systems and existence of many subsystems within the system. One system architect from the defence domain stated that they have informal integration deadlines with associated subsystems, which makes delivery time very important. Defence has large necessity to be compliant to specifications and standards, and thus often focus on what can be measured instead of Value creation. This is supported by the prioritisation of the most important VA in the defence domain, which were delivery (e.g. deliver on time and integration deadlines), performance, and code quality (measured by statistical code analysis).

Another possible explanation may be related to the use of ASD processes. The SDO from the defence domain had adopted most agile practices among the SDO in this study. It was also stated by a Scrum Master that they are trying to have incremental deliveries to internal customers, hence delivery in time is important for internal deadlines and for the continuous integration purpose. In addition, it was also mentioned by one of the managers that late delivery to customers can end up in penalties and financial loss. Similar to the discussion about project focus instead of product focus (see Section 4.1), the defence domain increased the overall priority of the *Delivery Process w.r.t. time* VA. The main reason for this is due to hard contractual obligations with regards to time. That is, being able to deliver in time was more important than delivering all functionality, or even the correct functionality, while for other domains, e.g. the consultancy domain, deliver correct functionality was more important than deliver in time (see Fig. 2).

For the telecom domain, *Delivery Process w.r.t. time* was considered important with regards to market pull (Regnell and Brinkkemper, 2005). One participant stated that, *"delivery in time is importantÉwith tremendous market push, we need to get out new functionality in time"*. Another participant further explained that *"market push is willing to pay for functionality than anything else"*, thus it is important to *"get out the new functionality in time"* as several participants emphasised. Although the participants talked about market push, their explanations during the interviews showed that they meant market pull, and that the customers (the market) are willing to pay for the latest functionality. This is inline with Svensson et al. (2012) who reported that, although high quality is important, having an extra function (new functionality) is considered more important. However, little in terms of research as to what maximises customer value has been conducted in relation to this.

For the SDO in the consultancy domain, they are concerned with receiving often and early feedback in order to have a better understanding of their customers' project, or the particular product that they are involved in. According to our participants, in order to benefit from the feedback from their customers and *"get close tie with customers"* they need to have short delivery times and frequent deliveries. This works well if the release overhead is reasonable (Bjarnason et al., 2011).

When it comes to the automotive domain, where release overhead is substantial and it is harder to release actual new products cars regularly as means to gather customer input, one participant emphasises the importance of timely deliveries rather as, *"delivery in time is very important in order to start the production of a new car"*. Since teams work on different parts of the bigger system, there are many dependencies between teams and deliveries, which may explain the importance of delivery time. Another reason may be related to the distributed development and working with suppliers, which was further explained by one participant, *"in order to make parallel working possible, it is important to keep deliveries in time"*. When it comes to testing and the system test, delivery time is of particular importance according to some of our participants. It was stated *"there are not many winter or summer time in a year to test the climate control"*. This is because the whole system does not exist in real engine or with real hardware. The simulated environments are not sufficient or good enough. Especially since not all of the vehicles will be used in similar climate/countries, as our participants stated.

Interestingly, *Perceived Quality* (second most frequently mentioned VA, third in priority) was seen as very central for the telecom and automotive domains, but not for the defence and consultancy domain. One telecom participant explained, *"it is very important that our company is perceived as one with a high quality since we have premium prices"*. The importance of keeping a balance between quality and price to increase customer satisfaction is confirmed in the study by Wang et al. (2004). Similar to the telecom domain, the automotive domain needs to maintain a certain level of quality, especially in terms of following standards and quality assurance criteria.

None of the participants from the defence or consultancy domains prioritised *Perceived Quality*; however, both these domains focused more on *Actual Quality* than the telecom and automotive domains. One reason why defence did not prioritize *Perceived Quality* may be related to their type of products. This is supported by the two managers from the defence domain that explained that the importance of quality for their products is related to the stability of the software and having high quality code and being able to measure this. One of the managers further explained, *"it is important for our company to achieve a reliable base-product with high quality code in which we can separate customer adaption and base functionality"*. In addition, the defence domain develop safety critical systems that are inherently to be used in applications that are not prone to market-pressure, thus, *Actual Quality* such as robustness and reliability are of more importance than *Perceived Quality*.

The focus on *Actual Quality* rather than Perceived Quality is not inline with research in business and economy, which argue that consumers expect and desire 'experiences' rather than functionality and services (Pine and Gilmore, 1998). Hence, to be profitable, the development organisation should focus on 'experiences' (Pine and Gilmore, 1998). Similarly, SDOs need to deliver a unique experience to users in order to survive in a competitive market (Chapman and Plewes, 2014). Although the defence domain is not prone to as much market-pressure as other domains, e.g. telecom and automotive, there is a change in the market context in the defence domain. The market is changing from a bespoke development to a more market-driven industry (Börjesson and Elmquist, 2008). Hence, both hedonic (e.g. emotions and feelings) and user experience Values may affect the defence domain since this market changes radically change the future Value creation.

The consultancy domain did not consider *Perceived Quality* as an important VA. A possible explanation may be related to the type of development that is related to the consultancy domain. That is, they are mainly developing software for larger organisations and in most cases; they do not develop the whole product, but rather a part of a product or project, e.g. manning a project. Hence, devot-

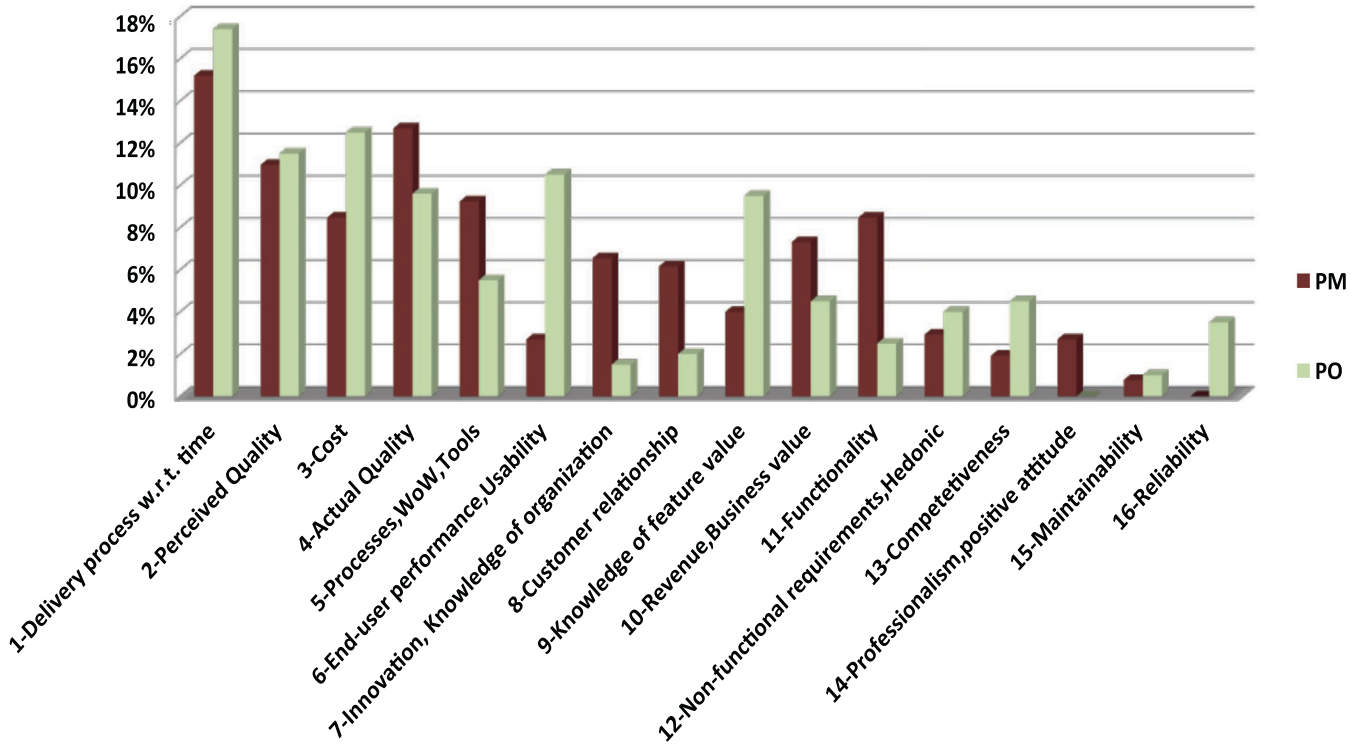

Fig. 3. Prioritised value aspects per role.

ing focus to *Perceived Quality* is not seen as important. Instead, *Actual Quality* was of higher importance for the consultancy domain (the second most important VA for this domain (see Fig. 2)). One reason for prioritising *Actual Quality* is related to having a well-designed software architecture. One participant explained, it is important *"to reduce risk and decrease technical debts"* of the project, denoting a project perspective. This may be related to that consultants are often hired as technical experts and may not have an overview of the entire product. Instead, the main focus for consultants is to deliver in the project.

The VA of *Processes, WoW and Tools* was prioritised by all four domains; however, it was viewed as the most important VA for the automotive domain. In addition, the automotive domain assigned the most dollars to this VA, as illustrated in Fig. 2. There are several main reasons of why the SDO in the automotive domain viewed this VA as the most important one. First, the SDO are in an adoption phase of agile development. The organisations have not yet fully adopted agile software development processes. This was supported by participants from the SDO who explained that the internal customers *"are not working very agile"*. Hence, it is important to get the processes in place to be able to deliver software on time, both to internal and external customers (*Delivery Process w.r.t. time* is viewed as the second most important VA). Second, in some of the SDO in the automotive domain, in-house software development was not yet in place, thus outsourcing and distributed development introduces challenges to the agile processes and to deliver software on time. This is inline with several studies, e.g. Korkala and Abrahamsson (2007); Ramesh et al. (2006), that report on challenges such as difficult communication and delayed deliveries, which supports the importance of having well defined and efficient processes and ways of working to at least offer control. Third, several participants from the automotive domain explained that the current agile development processes need improvement. One participant stated, *"we need good working foundations, e.g. better guidelines"*.

Traditionally, the automotive domain is seen as system engineers. Hence, the automotive domain has a tradition of integrated system development where processes and tools, such as autosar, are seen as a coordination process in itself.

The VA *Competitiveness* (ID 13 in Table 3) was only prioritised by the telecom SDO. This is inline with Wang et al. (2004) that show the competition between the telecommunication companies are intense, and companies expecting to make and maintain competitive advantages in their market, need to turn the customers behaviour intentions into a purchasing behaviour. It is surprising that *Competitiveness* was not prioritised by the automotive SDO. One reason may be related their large system integrations and long cycles from idea to release, thus *Competitiveness* was not considered due to a higher focus on the project and internal delivery perspectives.

4.1.2. Difference between roles

We analysed the identified and prioritised Value Aspects (VA) based on the different roles of our participants. The roles were divided into two main categories. First, Project Manager (PM) includes the roles that deal with processes, scrum master, agile driver, or project manager. In total, 13 participants are classified as PM. Second, Product Owners (PO), which includes the roles of product owners and roles dealing with backlogs, technical experts, and architects. In total, 10 participants are classified as PO. Fig. 3 shows the prioritised VAs (percentage of total spent dollars) per role (the numbering and the order of VAs is the same as in Table 3).

Delivery Process w.r.t. time was viewed as the most important VA for both PM and PO, as illustrated in Fig. 3. This is inline with the general view of the participants in this study (see Table 3), and with the view of the most important VA among the four domains (see Fig. 2).

Both *Perceived Quality* and *Cost* were viewed as one of the most important VAs for PO and PM. This is not surprising since PM use the simple golden triangle (Westerveld, 2003) that besides schedule (time), deals with budget (i.e. *Cost*) and quality (which includes *Perceived Quality*). In addition, in general PM refers to planning, monitoring, and controlling the projects, which is inline with

Interview	List of value aspects	Interviewee	Value aspect	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Joint	Joint	A1		20	40	10								10	20				
Joint	Joint	A2		20	40	10								10	20				
joint	Joint	B1			10	25				5			25			25		10	
Joint	Joint	B2			10	25				5			25			25		10	
Joint	Separate	C1							20			40	20			20			
Joint	Separate	C2			20	25	10			25	20								
Joint	Separate	D1		15	15			25			25						20		
Joint	Separate	D2		20	30		25	15			10								
Single	Single	E		15			30	5			5			30			15		
Joint	Separate	F1		20		20	30			30									
Joint	Separate	F2		25	25	25		25											
Separate	Separate	F3		25			30		15			30							
Joint	Separate	G1		30			35												35
Joint	Separate	G2			25			25			30		20						
Separate	Separate	H1		10	10	30			10			40							
Separate	Separate	H2		10	10	20			40						20				
Single	Single	I		7.5	22.5			45	15			10							
Joint	Separate	J1		50			30	10	10										
Joint	Separate	J2		44		15	6	5	15					15					
Single	Single	K		60			20			20									
Single	Single	L						10				2		70	18				
Single	Single	M					45			5			50						
Single	Single	N				30		10	15	10	10	25							

Fig. 4. Distribution of spent dollars per value on individual basis.

the view of the most important VA for the PM role. In a similar way, POs are responsible for the projects success in Scrum teams/projects, which explains why *Perceived Quality* and *Cost* are among the most important VAs.

Looking at the VA of *Functionality*, it was among the least important aspects for POs, while *Functionality* was the fifth most important VA for PMs. Looking at Table 3, five participants prioritised *Functionality*, of which three were from the telecom domain, and one from the consultancy domain. The participant from the consultancy domain that prioritised *Functionality* mainly works with customers in the telecom domain. Thus, only one out of five (from the defence domain) that prioritised *Functionality* was not from/worked in the telecom domain. Hence, *Functionality* may be of high importance in the telecom domain. This is supported by, both (Ramesh et al., 2006) who found that functionality is important, and the view of important VAs per domain in Fig. 2. When combining the telecom and consultancy domain (since the only consultancy participant works with customers from the telecom domain), 89% of the total amount of dollars spent on *Functionality* comes from the telecom domain, see Fig. 4.

PMs prioritised *Customer Relationship* (ID 8 in Table 3) higher than the PO. The PMs viewed *Customer Relationship* as an important VA as they need to sustain a good relationship in order to benefit customer feedback and customer collaboration. Maintaining a good relationship with customers is central in agile projects (Chow and Cao, 2008) In addition to *Customer Relationship*, participants with the PM role had a higher focus on *Knowledge and Innovation* aspect too. Conboy and Morgan (2011), in their study, emphasise the importance of customer collaboration and the presence of customer in order to share ideas/knowledge outside the team and increase innovation. This may explain why PMs prioritised *Customer Relationship* and *Knowledge and Innovation* (ID 7 in Table 3) higher than the POs.

Besides looking into the identification of VAs (see Table 3) and the distribution of assigned dollars by domain and role, we analysed the data on an individual basis, as illustrated in Fig. 4. In Fig. 4, participants who share the same letter ID and cell color (e.g. "A" in A1 and A2) were from the same organisation (see Table 1). The first two columns in Fig. 4 show if the participants from the same organisation participated in a separate or joint interview session (column "Interview") and if they listed and prioritised the VAs separately or jointly (column "List of VA"). For example, B1 and B2 participated in a joint interview session and identified and prioritised the VAs together, while C1 and C2 had a joint interview session, but the participants identified and prioritised the VAs individually.

In general, looking at Fig. 4 it seems like the participants from the same organisation that identified and prioritised VAs separately were in agreement. Only participants C1 and C2, and G1 and G2 were completely off in terms of agreement of what Value is and which aspects are deemed most important, while participants from SDO F were partly in disagreement. SDO C and D were the only SDO from the telecom domain that identified the VAs separately. While participants from SDO C had different views of what Value is, it looks like the participants from SDO D were in agreement. However, the VAs in Fig. 4 (columns 1–16) are high-level categories of VAs that were used during the analysis process (same categories as in Table 3). To have a better understanding of the agreement level of participants in SDO D, and to understand the difference between the two SDO from the telecom domain, Table 4 shows the original identified (i.e. the actual stated VAs by the participants) and prioritised VAs from SDO C, while Table 5 shows the actual VAs from SDO D.

As shown in Table 4, the identified VAs from the two participants in SDO C were very different, and have different priorities. This could be an indication that there was no standard way of defining and prioritising Value within the SDO. One participant said, *"it is complicated to be able to pinpoint the values"*. Another participant explained that *"it is not clear what value is and how to work with value"*. This is supported by, both the participants from SDO C, but also from several participants in several SDO in this study, when asked what sources they used, or had access to when defining Value. Some participants referred to standards and regulatory requirements, while others referred to project managers, product owners, guidelines or processes of agile development methods, but none were specific in terms of exact definitions, rather were

Table 5
Actual stated value aspects from SDO D.

Participants from SDO D (Telecom domain)			
Participant one: PM		Participant two: PO	
Value Aspect	Dollars	Value Aspect	Dollars
To keep our promises	15	Quality	30
Speed (of delivery)	15	Keep delivery time	20
Simplicity	15	Keep it simple	15
We deliver quality	15	Software architecture (life cycle/backward compatible)	15
We deliver customer Value, and what they really want	10	Architecture rule	10
Open and transparent communication	10	Customer first	10
To keep the positive attitude	10		
We are proud of what we do	10		

this information can probably be found. Other VAs were emerging from test results, reviews, previous project's issues, or dependencies between development teams. One reason for this spread of different sources for identifying Value may be that all of the sources were valid references to Value; however, there is no general agreement of when, how, or to what extent the participants should refer to a particular source or definition of Value. One participant reflected on this, *"there are different prioritisation between deferent departments"*. In addition, it is important to consider what VAs and sources of references are inline with the short-term organisational Value, and what VAs are focused on the long-term Value perspective of the company and product, short- vs. long-term was not elaborated upon by the participants other than that they realised that different VAs were aimed at different perspectives.

On the other hand, the participants from SDO D were more in agreement of what Value is, as shown in Table 5. For example, the participants identified quality, delivery time, and simplicity as highly prioritised VAs. There are several reasons that may explain the differences between the two SDO from the telecom domain. First, the size (number of employees) of organisation differs between the two SDO, where SDO D is smaller than SDO C (55 vs. 300 employees as shown in Table 1). To have a shared understanding of common goals and VAs is a key issue in terms of having a clear focus on what type of value to deliver, e.g. customer value (Aurum et al., 2006), and it may be easier to share these goals and priorities of VAs in smaller teams. Another explanation may be related to the number of used agile practices, the average number of agile practices used in SDO C was 18, while the average in SDO D was 44. This could indicate that SDO D has a more rigorous development process and a mature utilisation of agile development - where a common view on value has had time to mature. On the contrary SDO C might be focused on adopting agile and new practices. The maturity of SDO D is supported by both of the participants from SDO D who stated that several of the VAs are part of organisation's overall Value/culture vision.

A third possible explanation may be related to the type of products developed. SDO C develops software for a hardware-intensive system (embedded), while SDO D's product is pure software. Since software has been a central part in telecom for a long period of time, it may be easier for the participants to identify individual VAs in pure software products compared to scenarios where the software itself is only a part of the offering. In the embedded case VAs might be easier to identify and qualify for the end-product which is not the focus on SDO C.

Table 6
Actual stated value aspects from SDO G.

Participants from SDO G (Automotive domain)			
Participant one: PO		Participant two: PM	
Value Aspect	Dollars	Value Aspect	Dollars
Robustness (you know how the software will behave)	35	To have no customer complaints in our development division	30
Reliability	35	"Best in class" quality assurance	25
Deliver in time	30	"Best in class" tools and processes for software development	25
		To be able to work with "business development" for future	20

Table 7
Actual stated value aspects from SDO H.

Participants from SDO H (Automotive domain)			
Participant one: PO		Participant two: PM	
Value Aspect	Dollars	Value Aspect	Dollars
Knowing the feature and function Value for the customer	40	More usable product - higher productivity for our customers	30
Quality (perceived and intrinsic)	20	Customer revenue	20
Project cost	15	Hedonic Value	20
Product cost	15	Fuel economy	10
Time to market (keep deliveries on time)	10	Deliver in time	10
		High quality on our product	10

Looking into the two SDOs G and H, from the automotive domain, the participants listed and prioritised the VAs separately. We see a similar pattern as in the telecom domain, as illustrated in Tables 6 and 7. In SDO H, the participants had in general an agreement of what Value is, while in SDO G the participants were in disagreement. For example, both of the participants in SDO H identified quality and delivery time as VAs. Moreover, although named differently (participant one said "knowing the feature and function Value for the customer", while participant two said "more usable product", as shown in Table 7), both participants prioritised product usability for their customers as the most important VA.

Although the pattern in the automotive domain is similar to the telecom domain, the situation is the opposite. That is, SDO H, where the participants were in agreement, is larger than SDO G in terms of number of employees, use less agile practices, and the product is more hardware-oriented than the product for SDO G.

One explanation may be related to the relatively (compared to the telecom domain) late introduction of software as a central part of their products. That is, the participants from SDO H may have used hardware related VAs as an input when defining the value of software as well.

4.2. Use of value in agile organizations (RQ2)

We asked the participants how the concept of Value is used, i.e. what do the participants do to achieve the identified Value Aspects (see Table 3) in their organisations. In addition, we asked the participants what barriers to achieve Value may exist. Table 8 shows which activities are used to achieve (column "activities used to achieve Value") a certain VA (column "VA ID", which refers to the identified VA in Table 3), and which activities that first need to

Table 8
Activities for achieving value.

VA ID	Activities used to achieve Value		Activities needed to be in-place or improved
1	Test Driven Development	Agile methodology	Setting up a new process
	Optimising the processes, minimising the handovers between teams	Program/project planning	
	Development loops, sprints	Team planning	
	Iterative way of working, many deliveries between main deliveries	Prioritising the deadlines	
	Automating parts of the process and the build environment	Backlog meetings	
	Design, test and requirement teams sitting in the same room	Fulfilling the requirements as the priority	
	Monitoring the progress in backlog	Rejecting unneeded changes	
	Scrum of Scrum	Estimation by experience	
2	Verify with feature owners and internal teams	System testing	N/A
	DoD on requirements/user-stories	Use of experience and judgment	
	Verification of the latest software version	Use of Trouble Reports	
	Transformation towards Agile and Lean WoW		
3	Transparency amongst employees in the whole organisation	Cost engineers	In-house software development
	Improve and automate some of the processes	Cost measurements	
	Super modelling that supports several projects	Monitoring the budget usage	
	Project related meeting with the developers	Adjusting the backlog content	
4	Pair programming	Code reviews	Making better test strategies
	Static code analysis	Peer reviews	Automating regression test
	Unit and system testing		Re-design
			Separation between customer adaptation and base functionality
5	Scoping	Simplicity	Creating better guidelines, missioning
	Automating parts of processes	Toolbox	To keep delivery time short/the culture to keep delivery times short
	Internal communications with team members/with other teams	Working in sprints	Improving WoW (many manual steps in different systems)
	More of the Lean way of working and PDCA (Plan, Do, Check, Act)	Sprint demoes	Group (cross functional teams) discussion to identify the areas that needs improvement
	Informal collaborations within the company to support the system level test, informal requirements	Team retrospectives	More open and transparent communication (through reporting, presentations, meetings)
6	Making our processes concrete	Testing	N/A
	Putting myself in the customer's situation	Happy Employee	
	Participate in system level testing	Meetings with the end user	
	User groups workshops, User Experience (UX) reviews, and UX experts (within the team)	Use case tests with end users	
	Give feedback/problems to system/function owner		
7	Focus and also sharing between teams/individual	Incremental approach	Understanding how the SW is done (referring to in-house software development) so to be able to take smarter decisions as a company
	Agile development projects		A need to have a focus area for innovation and company's knowledge
			To create SW drop tests with innovation, and test them in the products
8	Stay with the customer, never walk out on the customer	Close work with customer units	N/A
	Continuous follow up, being interested in product and people	Always treat the customer with respect	
9	A lot of meetings with the customer		
10	Using cost-estimation	Quality (of products)	A need to have a focus area for this and to know what are values
11	Three C's (Card Conversation Confirmation)	Scoping	N/A
	Participation in system level testing	Backlog meetings	
12	N/A		N/A
13	N/A		A need to have a focus area for this, so to be able to formalise it
14	Skilled personnel	Training	N/A
	Take responsibility and interest	Make it as a culture	
15	N/A		N/A
16	To make sure team(s) understand each other		N/A

be in-place or improved before a certain VA can be achieved (column "activities need to be in-place or improved"). Table 9 shows what barriers exists for achieving a certain VA (column "barriers for achieving Value"). In Tables 8 and 9, a "N/A" (Not Applicable) means that no activity or barrier was mentioned by any participant.

Looking at Table 8, there are several activities that are already in place that are used to achieve Value, and some activities that need to be in place, or to be implemented before Value can be achieved.

In general, the most common mentioned activities to achieve Value were related to agile practice, optimising and automating development processes, in particular the testing processes and strategies. For example, to achieve the Value Aspect (VA) of *Delivery Process w.r.t. time*, the majority of the answers were about optimising the processes and different agile practices such as iterations, back-

Table 9
Barriers for achieving value.

VA ID	Barriers for achieving Value	
1	Late scope changes	Unclear definition of delivery
	Hardware based issues	Feedback not given in time
	Dependencies amongst different development teams	
2	Perceived quality taken as granted	
3	Too much focus on re-planning	Keeping promises for fast deliveries
	Not enough detailed requirement specification	Looking at near future only
	Short time projects and not enough money allocated to the project	
4	Market push	Late feedbacks
	Deadlines/delivery time to customer/Insufficient time and focus	Vague guidelines
	Lack of enough testers	Cost resources
	Lack of access to the target environment for the developed software/test	
5	Large organisation	Mindset of people
	Receiver is not always to receive the "truth" about plans	Skeptical people
	New development teams	Communication
	Customer requirements	Autosar
	PDCA feels like waste when we are already using retrospectives	
6	Somewhat vague requirements	Time
	End users with very different experience level	Cost resources
	Delivery capacity and prioritisation (change requests)	Waste of double testing, late feedbacks
	Different level of knowledge within the customers	
7	Access to customer	Time and focus
	Different prioritisation between different departments	Financial margins
8	"Tricky" customers	Unforeseen risks
	Customers don't always know what they want	
9	A more complete solution was needed	
10	It is complicated to pinpoint the values	IT project costs
	Governance models	Architecture
11	Technical issues in the verification environment	Somewhat vague requirements
	Time constraints, sometimes too costly	
12	N/A	
13	N/A	
14	Unrealistic expectations from sponsors	Stress and high work loads
15	N/A	
16	N/A	

log meetings, development loops, and scrum of scrum. One of the participants reflected on this and said, *"this is what we want to do as agile drivers"*. Optimising the processes and different agile practices were the most frequent mentioned activities to achieve Value. A common assumption is that the quality of the software processes is directly related to the quality (e.g. *Perceived* and *Actual Quality*) of the developed software, and to the time-to-market (i.e. delivery time) (Kitchenham and Pfleeger, 1996), which is discussed in more detail in Section 4.1 (RQ1).

To achieve the VA of *Cost*, some activities were directly related to cost, such as cost measurement, use of cost engineers or monitoring the budget usage, as illustrated in Table 8. In addition, some participants mentioned improving the development processes and automating part of the processes as activities for achieving cost related Values. An interesting reflection of activities for reducing cost was made by one participant who said that they were working towards achieving a full in-house software development, which was considered as an important cost reduction activity. This is not inline with Herbsleb and Mockus (2003) who found that companies could reduce cost by introducing Global Software Development.

Looking into the VA of *Processes, Ways of Working*, and Tools (ID 5 in Table 8), besides agile practices and process improvement activities, transparency was mentioned by several participants as an important activity. Transparency and communicating the information through meetings or presentations amongst different levels of the organisational hierarchy, or between different parts of the organisation was seen as an important strategy towards the general improvement of the software development process, and thus increasing the VA. Most of the VAs in Table 8 are utilised through different process improvement activities, thus process improvement can be seen as the main key activity to achieve Value as it enables the ability to achieve Value through Delivery on time, and *Actual Quality*. However, in general, *Process, Ways of Working, and Tools* was only viewed as the fifth most important VA. To achieve both *Perceived* and *Actual Quality* related VAs, answers were mostly related to testing activities and improvement of testing strategies.

One reason why the participants mainly stated testing activities and improvement of testing strategies for Value creation may be related to that several of the participants reported using the agile practice of *Test-Driven Development* (9 of 23), and that 17 of 23 participants stated that several of the used agile practices are related to the agile "tribe" called Testing (Agile Alliance). While there are several techniques and testing strategies to predict how many faults, bugs, and defects that remains in the software product (Mockus and Li, 2005), customer satisfaction (including *Perceived Quality*) is a far more complex than minimising the number of defects in a system (Chulani et al., 2001).

Despite the importance of the VA of *Perceived Quality* (viewed as the second most important VA, see Table 3), no real activities for achieving *Perceived Quality* were mentioned. *Perceived Quality* cannot be achieved by testing activities, e.g. *Test-Driven Development*, since the decision of what to include in the product, or in coming releases, has already been made, i.e. the functionality and level of quality have already been prioritised, planned, decided and added to the backlog. Thus, improved testing activities and strategies can only be used for realisation of the chosen functionality and quality.

Moreover, other project related activities such as improved development process, or improved cost estimation may not help in achieving the VA of *Perceived Quality*. One reason is that these activities are used for tracking and monitoring project activities (Boehm, 2006a), and not for considering/realisation the business Value. Hence, a project can be successful in terms of cost, i.e. the project was completed within its budget, but may fail to add any business Value (Barney et al., 2008). Instead, business Value and

the VA of *Perceived Quality*, but also other VA such as *Actual Quality* and *Functionality*, can be achieved through good Requirements Engineering (RE) (Favaro, 2003). Favaro (2003) argues that the purpose of RE is to add business value to the software product. In addition, value-based RE aims to maximise the value of a release of a software product through the selection of requirements (Aurum and Wohlin, 2007). Hence, requirements engineers are in a position of managing requirements (pre-project) to take strategic opportunities into account for value realisation.

Previous studies, e.g. Gordjin and Akkerman (2003) have shown that despite a significant effort from companies to increase customers' perceived value in the development process, determining how value should be added has been a challenge. The results from this study are inline with Gordjin and Akkerman (2003), the SDOs seem to have a project focus where monitoring the project (e.g. through activities related to *Cost*) and identifying faults and defects (e.g. improved testing strategies) are more important than realisation of the identified VA (see Table 3). These results indicate that agile development processes are rather project focused and may not consider the product perspective, thus making the realisation of value a reactive rather than proactive effort.

Looking into what barriers exist towards achieving the VA of *Delivery Process w.r.t. time*, a frequently mentioned barrier was unclear definition of delivery, as illustrated in Table 9. Other barriers include late scope changes, late feedback, and dependencies to other development teams, or to hardware.

Only one barrier was mentioned to achieve *Perceived Quality*, which is related to the mindset of the people. One participant explained, *"sometimes we take the perceived quality as granted"*. *Perceived Quality* is partly related to Quality Requirements (QR), e.g. performance and reliability. Therefore, that *Perceived Quality* is assumed and taken for granted is not surprising, which is inline with the findings in Svensson et al. (2012) who reported that QR were often assumed.

Reflecting on barriers towards the *Cost* related VA, one participant recognised that *"looking at near future only"* was one of the important existing obstacles. The participant believed that many activities are considered and judged by their short-term effect while the long-term cause and effects are overlooked. Several other participants pointed at similar concerns where one explained that one barrier was *"the short-term projects with not enough money"*. One reason, as explained by several participants, when looking from a long-term perspective, these short-term projects may cost much more, take more effort and resources compared to the connected income.

Barriers for achieving *Actual Quality* were related to time, e.g. deadlines and market push, late feedback, and lack of test resources. Barriers in regards to *Processes, WoW and Tools*, include tools such as Autosar that is widely used in automation industry. Autosar was seen as a barrier rather than an aid towards improving the development process. The reason, according to our participants, is because of the complexity of the tool, i.e. it has a difficult and time consuming learning process. Another interesting barrier that was mentioned by several participants, was the *"mindset of the people"*.

RQ2.1: How is Value measured/assured/evaluated? In general, few of the identified VAs are measured, assured, and/or evaluated. The most evident measurements for most of the SDO are delivery time and dead-lines for measuring the VA of *Delivery Process w.r.t. time*. The main reason for this was because they must deliver to other teams or departments, which are dependent on these deliverables. In addition, test results and number of discovered bugs are common measurements to assure high *Perceived* and *Actual Quality* among all SDO. Some of the participants mentioned measurements in relation to the *Cost* VA. However, the participants mainly referred to staying within the assigned project budget for measuring *Cost*, and not in relation to the long-term measurement of cost or revenue for the organisation.

For the majority of the identified VAs (see Table 3), the SDO did not have any specific measurement or metrics to assure high Value. Reflecting on this, one participant stated that *"we need a baseline to measure, even hard to say if we are developing faster, because we do more test"*. Another participant further explained while relating to the difficulties of identifying what is Value by stating, *"if we don't even know what is the value, then we can't communicate it to everyone in the organisation and we can end up doing the features incorrectly"*.

While expressing their concerns of lack of measurements, some participants uttered their future plans of introducing measurement plans for Value. A general view among the participants in this study was the importance of having a unified understanding of the VAs, evaluation of attaining high Value, and reflecting on the organisation's revenue and return of investment. One participant explained, *"I am trying to get acceptance for this to get things as written [documents] we need to know what is value for customers"*. Another participant further explained, *"I am working on this in fact, we need to formalise it, [to know] what kind of information is valuable, who documents it, we are not really there yet sometimes I must say that the entire organisation does not know the entire value of a feature and it is not their fault, we have to start"*. Another participant referred to the need to comparing the cost to return of investment, *"we always have focus areas each year, I think we do not have that at all the [development] levels at the moment, but we are getting better in communicating the value aspect and also in comparing this to return of investment"*.

However, despite the expressed awareness of the importance of Value realisation, communication in the organisation, and the need for a balance between organisation's VAs, revenue and long-term success, we could not identify any activity that had been dedicated to Value evaluation in particular, or a focused practice to examine, e.g. if the existing processes are serving towards Value creation in a long-term perspective. This is surprising since most of the mentioned activities for achieving Value were related to software processes and software process improvement. Although there are various metrics to measure software systems and processes, measurements programs mainly focus on the internals of the measurement program (Niessink and Vliet, 1999), which may explain the lack of metrics for Value realisation among the SDO in this study.

4.3. Mapping value aspects to the software value map

The purpose of this study is to gain an in-depth understanding of the concept of Value and how it is used and defined in industry. Although Khurum et al. provides a consolidated view on the concept of Value in the Software Value Map (SVM) (Khurum et al., 2013), they do not look into how different companies interpret the Value concept, what Value Aspects are considered important to achieve, nor which of the Value perspectives in the SVM are used, or not used in industry. Therefore, to be able to see which of the identified VAs in the Software Value Map are important in industry, and which of the VAs are not used in industrial practice, we applied a second analysis of the identified VAs in this study (in Section 4.1) based on the Software Value Map. The 134 VAs that the 23 participants identified in this study were categorised into the different Value perspectives, Value Aspects (VAs), and sub-Value Aspects (SVAs) of the Software Value Map by the first author. The mapping between the 134 identified VAs and the categories of the SVM (see Figs. 5 and 6) where then discussed among all authors. The SVM provides a broad view of software Value through four major perspectives, *Customer, Internal Business, Financial Value*, and *Innovation and Learning*. A short explanation of the four per-

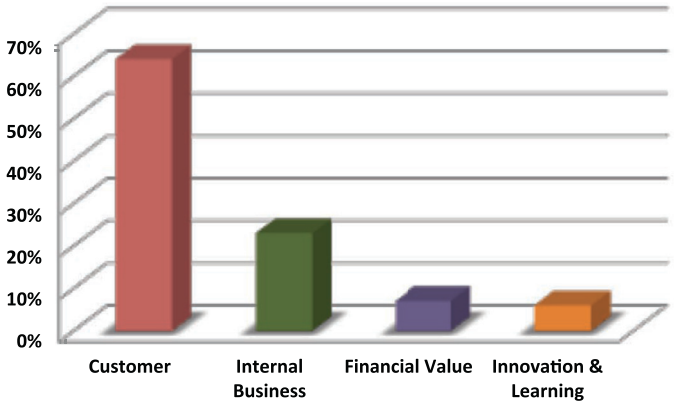

Fig. 5. Value aspects mapped to the Software Value Map's perspectives.

spectives can be found in Section 2, while a more detailed description of the SVM can be found in Khurum et al. (2013).

Fig. 5 illustrates the distribution of the total amount of dollars spent on VAs for the four perspectives of the SVM. Before mapping the VAs to the perspectives of the SVM, the data was normalised per each SDO.

Looking at Fig. 5, we see that the *Customer* perspective (with 896 dollars out of 1400, i.e. 64%) is the highest prioritised Value perspective among our SDO. In terms of importance, *Customer* is followed by *Internal Business* (321 dollars, i.e. 23%), *Financial Value* (95 dollars, i.e. 6.8%), and *Innovation and Learning* (87 dollars, i.e. 6.3%).

Comparing the different domains, we can see that there were no major differences, in terms of order of the prioritised Value perspectives, between the four domains, as illustrated in Table 10.

Table 10
Software Value Map perspectives per domain.

Value Perspective	Consultancy	Telecom	Automotive	Defence
Customer	57.5%	63.0%	69.2%	74.5%
Internal Business	21.3%	21.5%	25.8%	25.5%
Financial Value	12.5%	7.0%	2.5%	0%
Innovation and Learning	8.8%	8.5%	2.5%	0%

Apart from prioritising *Customer* (ranging from 57 to almost 75% of the total dollars spent) and *Internal Business* (ranging from 21 to 25% of the total dollars spent) perspectives as the two most important VAs, there was a small difference in order of importance for *Financial Value* and *Innovation and Learning*. *Financial Value* and *Innovation and Learning* received almost equal attention from the four domains with the exception from the consultancy domain, which prioritised *Financial Value* perspective slightly higher. However, it is worth mentioning that one manager assigned all of the dollars (50 dollars) for the *Financial Value* perspective from the consultancy domain. The manger explained that this *"is a part of our policies"*. No further elaboration was made.

Innovation and Learning was the least prioritised Value perspective for three of the domains with the exception of Telecom domain. However, the difference between *Innovation and Learning* and *Financial Value* for the telecom domain was small. Zero dollars were assigned to the *Innovation and Learning* perspective by the defence domain, and 2.5% by the automotive domain. Innovation is widely seen as the basis of a competitive economy (Porter and Ketels, 2003), hence product innovation is an important challenge for companies that want to be competitive in the market and to survive in the unprecedented economic (Patterson et al., 2009). Considering that most of a products' innovation lies in its software components (Briand, 2012), the Value perspective of innovation re-

Value Perspective	%	Value Aspect (VA)	%	Sub-Value Aspect (SVA)	%	Sub-Value Aspect (SVA)	%
Customer	64	Perceived Value	51	Intrinsic value	16,3	Functionality	8,4
						Maintainability	0,7
						Reliability	1,3
						Other	5,9
				Delivery process value	25,1	w.r.t Time	15,4
						w.r.t Quality	9,7
				User Experience Value	9,6	Pragmatic Value	6,2
						Hedonic value	2,0
						Other	1,4
		Customer lifetime value	13	Revenue and Cost	8,9	N/A	
				Retention rate	4,1	N/A	
Internal Business	23	Production value	20,5	Physical value w.r.t Quality	20,5	process	8,5
						Product architectural value	10,6
						other	1,1
		Differentiation Value	2,5	N/A			
Financial Value	6,8	N/A					
Innovation and Learning	6,3	Intellectual Capital Value	1,8	N/A			
		Other	4,5	N/A			

Fig. 6. Detailed mapping of value aspects to the Software Value Map.

ceived relative little attention from the participants in this study in general, and in particular from the defence domain and the automotive domain. This is surprising since it is estimated that 90% of new innovations in the automotive domain are in the field of electronics, of which 80% is software based (Swedsoft).

One reason why the defence domain did not view innovation as important was related to their long tradition of working with customers on project-based contracts with no base-products. One manager explained, if they achieve a base-product in which they can separate the base functionality from customer specific requests, as a result, the need for innovational aspects rises. One of the managers reflected on this and said, *"we have discussed and decided to dedicate more attention to innovational aspects and it is a part of our future plans also"*. This is an indication that much of the software product development is reactive rather than proactive. Thus, product management may not be able to plan and rely on delivering high Value to achieve competitive advantages, which is inline with the findings in Gorschek et al. (2010), and Khurum et al. (2015), that among other things highlight that the relative importance and value generating properties of software in products such as cars is hard to gauge.

Looking at the result that Customer perspective was ranked as the most important perspective, and that the Innovation and Learning perspective received a very low priority, according (Treacy and Wiersema, 1997) it is more likely that the ASDOs in this study mainly focus on *operational excellence* (i.e. providing customers with reliable products at competitive prices) rather than *product leadership* (i.e. providing products that continually redefine the state of the art). If companies would like to change their focus e.g. from operational excellence towards product leadership or perhaps move towards a more equal focus on both aspects, they need to reconsider the organisations strategy map (Treacy and Wiersema, 1997).

In the SVM (Khurum et al., 2013), each Value perspective is further divided into VAs, and each VA is further broken down to Sub-Value Aspects (SVA). We mapped the VAs mentioned by the participants in this study (see Table 3) to the VAs and SVAs in the SVM according to the guide provided by the SVM (see Fig. 6).

The first two columns in Fig. 6 show the four major Value perspectives and their percentage of the total amount of spent dollars, which is the same information as in Fig. 5. The remaining columns illustrate the distribution of VAs and SVAs and their total percentage of the spent dollars for each major Value perspective. An "empty" white cell in Fig. 6 indicates that there are SVAs in the SVM; however, we could not map the mentioned VAs among our participants to these SVAs. There are two reasons for this. First, the focus of this study was not to map all VAs to the SVM. Second, in some cases the participants did not provide enough details about the mentioned VA to allow for a deeper mapping.

Looking at the highly prioritised Customer Value perspective in Fig. 6, we can see that the greater share of the spent dollars is given to the VA *Perceived Value* (51% of the total amount of spent dollars). *Perceived Value* is related to the benefits derived from the product/feature. That is, *"it is a trade-off between perceived benefits and the cost of ownership"* (Khurum et al., 2013). This is inline with McDougall and Levesque (2000) who emphasised the importance of *Perceived Value*. In addition, McDougall and Levesque (2000) argue that customers may be satisfied with the core quality (i.e. what is delivered) and/or the relational quality (i.e. how it is delivered), but may not be happy in general because they are conscious about the cost, i.e. if they got Value for the money. Looking at the *Customer* perspective, its related VAs of *Perceived Value* and *Customer Lifetime Value*, and their related SVAs in Fig. 6, the participants in this study seem to share the view of McDougall and Levesque (2000).

Looking further in the SVA of *Perceived Value, Delivery Process Value* (25.1% of the total amount of dollars spent) was seen as the most important SVA, followed by *Intrinsic Value* (16.3%). According to Khurum et al. (2013), *Delivery Process Value* is defined as the "quality of process in in-stalling / upgrading / receiving the product", while *Intrinsic Value* is defined as being "embedded into the software as functionality and attributes, for example, usability, security". The importance of *Intrinsic Value*, i.e. what the company promise that this product will deliver in terms of functionality and quality, which is partly related to quality requirements (e.g. usability), is not surprising since Quality Requirements (QR) are seen as critical for software product development (Svensson et al., 2012), and increases the likelihood of market success; thus QR can be seen as a key competitive advantage. However, the most important SVA of *Intrinsic Value* was *Functionality* (i.e. what functions the company promise to deliver in the product), which is inline with both the result in RQ1 (see Section 4.1.1) and Svensson et al. (2012) who found that new functionality was considered more important than higher quality.

When looking into the SVAs of *Delivery Process Value* we can see that the *Time* aspect (15.4%) is the most important one, followed by *Quality* (9.7%). In the SVM, there is a third SVA of *Delivery Process Value*, namely *Cost*, which did not receive any dollars among our participants. In Table 6 we can see that the most important SVA of *Intrinsic Value* is *Functionality* (8.4%), followed by *Other* (5.9%), and *Reliability* (1.3%). The importance of *Functionality* was explained by several participants as *"market pull is willing to pay for functionality than anything else"*, thus it is important to *"get out the new functionality in time"*, which is discussed in Section 4.1.1. The importance of reliability in relation to Value is inline with Svensson et al. (2012) who found that reliability was seen as one of the most important QR because of the importance to provide long-term Value for the customers.

5. Conclusions

In conclusion, this paper presents the results of an empirical study that examines how the concept of Value is perceived in 14 agile software development organisations. Data are collected from 23 participants using in-depth semi-structure interviews.

In relation to RQ1, how Value is defined and which Value Aspects (VA) are considered most important, the findings reveal that (1) in general, *Delivery Process w.r.t. time* is deemed the most important VA among the participants in this study, (2) for the defence domain, *Delivery Process w.r.t. time* was identified as the most important VA and more important than for any other of the three domains in this study, (3) *Perceived Quality* was only prioritised by participants from the telecom and automotive domains in this study, while *Actual Quality* was viewed more important among the participants from the defence and consultancy domains in this study.

Looking into the differences between roles, the results show that (1) PO and PM viewed *Perceived Quality* as one of the most important VA, and (2) *Functionality* was viewed among the least important VA for PO and PM.

The findings for RQ2, activities used for realising Value and what barriers exist in the realisation of Value, reveal that: (1) most used activities are related to agile practices, i.e. software processes (2) the most common barriers for *Delivery Process w.r.t. time* are unclear definition of delivery, late scope changes, and dependencies to other development teams.

Looking into what measurements and metrics are used to assure Value, the results show the most evident measurements were delivery time and deadlines to achieve the VA of *Delivery Process w.r.t. time*. In addition, testing related metrics, such as number of discovered bugs were mentioned by the participants. However, for

the majority of the identified VAs, no specific measurements or metrics was used.

In general, the results indicate that there might be differences between domains in relation to the importance of VAs. The type of products may explain the differences. That is, the large complex systems with both software and hardware from the defence and automotive domains have a more traditional systems engineering perspective, thus, coordination (i.e. time) is the most important VA.

References

Agile Alliance, (accessed December 2014). [link]. URL http://www.agilealliance.org/.

Agile Manifesto, (accessed December 2014). [link]. URL http://agilemanifesto.org.

Aurum, A., Wohlin, C., 2007. A value-based approach in requirements engineering: explaining some of the fundamental concepts. In: Proceedings of the 13th International Working Conference on Requirements Engineering: Foundation for Software Quality, pp. 109–115.

Aurum, A., Wohlin, C., Porter, A., 2006. Aligning software engineering decisions. Int. J. Softw. Eng.Knowl. Eng. 16, 795–818.

de Azevedo Santos, M., de Souza Bermejo, P.H., de Oliveira, M.S., Tonelli, A.O., 2011. Agile practices: an assessment of perception of value of professionals on the quality criteria in performance of projects. J. Softw. Eng. Appl. 4, 700–709.

Barney, S., Aurum, A., Wohlin, C., 2008. A product management challenge: creating software product value through requirements selection. J. Syst. Archit. 54, 576–593.

Bjarnason, E., Wnuk, K., Regnell, B., 2011. Requirements are slipping through the gaps - a case study on causes and effects of communication gaps in large-scale software development. In: Proceedings of the 19th IEEE International Requirements Engineering Conference, pp. 37–46.

Bloch, M., Blumberg, S., Laartz, J., 2012. Delivering large-scale it projects on time, on budget, and on value. McKinsey Q. October.

Boehm, B., 2006a. Value-Based Software Engineering. In: Ch. Value-based SE: seven key elements and ethical considerations. Springer, pp. 109–132.

Boehm, B.W., 2006b. Value-Based Software Engineering. In: Ch. Value-based Software Engineering: Overview and Agenda. Springer, pp. 3–14.

Börjesson, S., Elmquist, M., 2008. Aiming at innovation in the swedish defence industry - the difficulties of responding to disruptive market change in a technology driven context. In: Organizational Learning, Knowledge and Capabilities Conference.

Briand, L., 2012. Embracing the engineering side of software engineering. IEEE Softw. 29, 93–95.

Chapman, L., Plewes, S., 2014. A ux maturity model: effective introduction of ux into organizations. In: Design, User Experience, and Usability. User Experience DesignPractice, pp. 12–22.

Chase, J. P., 2001. Value creation in the product development process.

Chow, T., Cao, D., 2008. A survey study of critical success factors in agile software projects. J. Syst. Softw. 81, 961–971.

Chulani, S., Santhanam, P., Moore, D., Davidson, G., 2001. Deriving a software quality view from customer satisfaction and service data. In: European Conference on Metrics and Measurement.

Cleland-Huang, J., 2015. Injecting value-thinking into prioritization decisions. IEEE Softw. 32 (2), 14–18.

Conboy, K., 2009. Agility from first principles: reconstructing the concept of agility in information systems development. Inf. Syst. Res. 20, 329–354.

Conboy, K., Morgan, L., 2011. Beyond the customer: opening the agile systems development process. Inf. Softw. Technol. 53, 535–542.

Diebold, P., Dahlem, M., 2014. Agile practices in practice: a mapping study. In: Proceedings of the 18th International Conference on Evaluation and Assessment in Software Engineering, pp. 30:1–30:10.

Dingsøyr, T., Nerur, S., Balijepally, V., Moe, N.B., 2012. A decade of agile methodologies: towards explaining agile software development. J. Syst. Softw. 85, 1213–1221.

Dybå, T., Dingsøyr, T., 2008. Empirical studies of agile software development: a systematic review. Inf. Softw. Technol. 50, 833–859.

Eisenhardt, K., Graebner, M., 2007. Theory building from cases: opportunities and challenges. Acad.Manage.J. 50, 25–32.

Favaro, J., 2003. Value-based management and agile methods. In: Proceedings of the 4th International Conference on XP and Agile Methods.

Fogelstrom, N.D., Gorschek, T., Svahnberg, M., Olsson, P., 2010. The impact of agile principles on market-driven software product development. J. Softw. Maint. Evol. 22, 53–80.

Golfarelli, M., Rizzi, S., Turricchia, E., 2013. Multi-sprint planning and smooth replanning: an optimization model. J. Syst. Softw. 86, 2357–2370.

Gordjin, J., Akkerman, J., 2003. Value-based requirements engineering: exploring innovative e-commerce ideas. Requirements Eng. 8, 114–134.

Gorschek, T., Davis, A., 2008. Requirements engineering: in search of the dependent variables. Inf. Softw. Technol. 50, 67–75.

Gorschek, T., Fricker, S., Palm, K., 2010. A lightweight innovation process for software-intensive product development. IEEE Softw. 27, 37–45.

Herbsleb, J., Mockus, A., 2003. An empirical study of speed and communication in globally distributed software development. IEEE Trans. Softw. Eng. 29, 481–494.

Hoda, R., Noble, J., Marshall, S., 2011. The impact of inadequate customer collaboration on self-organizing agile teams. Inf. Softw. Technol. 53, 521–534.

Ivarsson, M., Gorschek, T., 2009. Technology transfer decision support in requirements engineering research: a systematic review of rej. Requirements Eng. J. 14, 155–175.

Ivarsson, M., Gorschek, T., 2011. A method for evaluating rigor and industrial relevance of technology evaluations. Empirical Softw. Eng. 16, 365–395.

Jalali, S., Wohlin, C., 2010. Agile practices in global software engineering - a systematic map. In: Proceedings of the 5th IEEE International Conference on Global Software Engineering, pp. 45–54.

Khurum, M., Fricker, S., Gorschek, T., 2015. The contextual nature of innovation - an empirical investigation of three software intensive products. Inf. Softw. Technol. 57, 595–613.

Khurum, M., Gorschek, T., Wilson, M., 2013. Software value map - an exhaustive collection of value aspects for the development of software intensive products. J. Softw. 25, 711–741.

Khurum, M., Petersen, K., Gorschek, T., 2014. Extending value stream mapping through waste definition beyond customer perspective. J. Softw. 26, 1074–1105.

Kitchenham, B., Pfleeger, S., 1996. Software quality: the elusive target. IEEE Softw. 13, 12–21.

Korkala, M., Abrahamsson, P., 2007. Communication in distributed agile development: acase study. In: Proceedings of the 33rd EUROMICRO Conference on Software Engineering and Advanced Applications, pp. 203–210.

Leffingwell, D., Widrig, D., 2003. Managing Software Requirements: A Use Case Approach. Pearson Education.

Maruping, L.M., Venkatesh, V., Agarwal, R., 2009. A control theory perspective on agile methodology use and changing user requirements. Inf. Syst. Res. 20, 377–399.

McDougall, G., Levesque, T., 2000. Customer satisfaction with services: putting perceived value into the equation. J. Serv. Mark. 14, 392–410.

Mishra, D., Mishra, A., 2011. Complex software project development: agile methods adoption. J. Softw. Maint. Evol. 23, 549–564.

Mockus, P.Z.A., Li, P., 2005. Predictors of customer perceived software quality. In: Proceedings of the 27th International Conference on Software Engineering, pp. 225–233.

Niessink, F., Vliet, H.V., 1999. Measurements should generate value, rather than data [software metrics]. In: Proceedings of the Sixth International Software Metrics Symposium, pp. 31–38.

Patterson, F., Kerrin, M., Gatto-Roissard, G., Coan, P., 2009. Everyday innovation: how to enhance innovative working in employees and organizations. NESTA Res. Rep. December, 1–54.

Patton, M.Q., 2002. Qualitative Research & Evaluation Methods. SAGE Publications.

Petersen, K., Wohlin, C., 2009. A comparison of issues and advantages in agile and incremental development between state of the art and an industrial case. J. Syst. Softw. 82, 1479–1490.

Petersen, K., Wohlin, C., 2010. The effect of moving from a plan-driven to an incremental software development approach with agile practices. Empirical Softw. Eng. 15, 654–693.

Pine, B., Gilmore, J., 1998. Welcome to the experience economy. Harv. Bus. Rev. 76, 97–105.

Poppendieck, M., 2011. Principles of lean thinking. IT Manage. Select 18.

Poppendieck, M., Poppendieck, T., 2003. Lean Software Development: An Agile Toolkit. Addison-Wesley Professional.

Porter, M., Ketels, C., 2003. Uk competitiveness: moving to the next stage. DTI Econ. Pap. 3.

Racheva, Z., Daneva, M., Sikkel, K., 2009. Value creation by agile projects: methodology or mystery? In: Product-Focused Software Process Improvement, pp. 141–155.

Ramesh, B., Cao, L., Mohan, K., Xu, P., 2006. Can distributed software development be agile? Commun. ACM 49, 41–46.

Regnell, B., Brinkkemper, S., 2005. Engineering and Managing Software Requirements. In: Ch. Market-Driven Requirements Engineering for Software Products. Springer, pp. 287–308.

Robson, C., 2002. Real World Research. Blackwell.

Runeson, P., Host, M., 2009. Guidelines for conducting and reporting case study research in software engineering. Empirical Softw. Eng. 14, 131–164.

Song, M., Parry, M., Kawakami, T., 2009. Incorporating network externalities into the technology acceptance model. J. Prod. Innov. Manage. 26, 291–307.

Svensson, R.B., Gorschek, T., Regnell, B., Torkar, R., Shahronki, A., Feldt, R., 2012. Quality requirements in industrial practice - an extended interview study at eleven companies. IEEE Trans. Softw. Eng. 38, 923–935.

Swedsoft, A strategic research agenda for the swedish software industry. (last accessed January 18, 2015) URL http://www.swedsoft.se/wpcontent/uploads/2014/05/SwedsoftSRA2010.pdf.

Treacy, M., Wiersema, F., 1997. The Discipline of Market Leaders: Choose Your Customers, Narrow Your Focus, Dominate Your Market. Basic Books.

Version one, 7th annual state of agile development survey. (last accessed December 2014) URL http://www.versionone.com/pdf/2013-state-of-agile-survey.pdf.

Wang, X., Conboy, K., Cawley, O., 2012. 'leagile' software development: an experience report analysis of the application of lean approaches in agile software development. J. Syst. Softw. 85, 1287–1299.

Wang, Y., Lo, H.-P., Yang, Y., 2004. An integrated framework for service quality, customer value, satisfaction: evidence from China's telecommunication industry. Inf. Syst. Front. 6, 325–340.

Westerveld, E., 2003. The project excellence model: linking success criteria and critical success factors. Int. J. Project Manage. 21, 411–418.

Zeithaml, V.A., 1988. Consumer perception of price, quality and value: a means-end model and synthesis of evidence. J. Mark. 52.

Hiva Alahyari is a Ph.D. candidate at Chalmers University of Technology. Her research interests include process assessments and improvements, organizational and human aspects of software engineering, requirements engineering and innovation. Hiva received her Licentiate degree in software engineering from Chalmers University of Technology, Sweden.

Richard Berntsson Svensson is an assistant professor in Software Engineering at Blekinge Institute of Technology, Sweden. His research interests include quality requirements, requirements engineering, software product management, Agile and Lean software development, human aspects of software engineering, creativity and innovation. Dr. Berntsson Svensson is the chairman of the Swedish Requirements Engineering Research Network (SiREN). He received his Ph.D. in Software Engineering from Lund University, Sweden, 2011.

Tony Gorschek is a Professor of Software Engineering at Blekinge Institute of Technology. Dr. Gorschek has over fifteen years industrial experience as a CTO, senior executive consultant and engineer, but also as chief architect and product manager. In addition he is a serial entrepreneur – in areas ranging from logistics to Internet based services and algorithmic stock trading. At present he works as a research leader and in several research projects developing scalable, efficient and effective engineering solutions, Product Management, Value based product development, and Real Agile™ and Lean product development.