

Aligning Requirements with Testing: Challenges and Practices

Poornima Joshi and Richard Berntsson Svensson

Department of Computer Science and Engineering, Chalmers, Gothenburg, Sweden,
Department of Computer Science and Engineering, Chalmers | University of
Gothenburg, Gothenburg, Sweden
poornim.joshi@gmail.com, richard@cse.gu.se

Abstract. [Context & motivation] A good alignment in requirements with testing helps in producing a good product that is really needed by the customer. [Question/problem] Knowing the importance of a good alignment between requirements and testing, people in the industry often tend to neglect it due to time and budget crisis. [Principal ideas/results] We have performed a multi-unit case study to explore the challenges that exist in aligning requirements and testing by interviewing eight practitioners from eight software developing companies. [Contribution] This study has exposed the motives, situational behavior and thinking styles of practitioners involved in aligning requirements with testing. Furthermore, the results have identified specific challenges to the automotive industry in aligning requirements with testing.

Keywords: Alignment; Requirements; Testing; Case study

1 Introduction

Requirements Engineering (RE) and Software Testing (ST) are one of the important steps in the development process of a software product. These processes are complex and lay foundation to the success of a project. Both RE and ST aim to support software development of products that will meet customers expectations regarding functionality and quality. However, to achieve this, both RE and ST need to be aligned, i.e. the "*adjustment of RE and ST efforts for coordinated functioning and optimized product development*" [18].

When RE and ST are aligned, they can effectively support the development activities from initial requirements and acceptance testing to the final product [3]. However, weak alignment of requirements with software development and testing activities may not only lead to inefficient development, but also to problems with the functionality and quality of the product [8]. Sabaliauskaite et al. [15] identified a number of problems, such as lower product quality, additional cost and effort required for removing defects, that are related to weak alignment between RE and ST. Moreover, weak alignment of RE and the remaining development activities within a company may lead to a product that does not satisfy business and/or client expectations [5].

Recent research shows that the study of the alignment between RE and ST are important [11], [15], [19]. Moreover, in a case study, Kukkanen et al. reported on lessons learnt in improving the requirements engineering and software testing processes [9], while Uusitalo et al. reported on a number of industry practices for linking requirements and testing [19]. A few studies have discuss the challenges in aligning requirements engineering and software testing. Sabaliauskaite et al. discusses the present challenges that exists in the software industry [15]. The identified challenges include, tools, managing requirements, and communication and cooperation among people. In a case study by Bjarnason et al., the results show that companies experience challenges in the achieving alignment including full traceability [2]. However, despite the importance of aligning RE and ST, few studies have specifically looked into the alignment between the two areas [1].

This paper presents the results of an empirical study that includes data collected through in-depth interviews with eight practitioners from eight different companies. The study focus on the challenges, motives, situational behavior and thinking styles of practitioners involved in aligning requirements engineering with software testing. This exploratory study can be seen as a study of state-of-practice in industry.

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

2 Related Work

There are a few research papers that discuss the challenges in aligning requirements engineering with verification and validation. Sabaliauskaite et al. discusses the present challenges that exist in the software industry in aligning the requirements engineering and verification & validation [3]. Their interview study report contains findings of the challenges in large software Development Company in Sweden. They also suggest that often the issues are with the tools used, managing requirements, test cases, traceability and communication and cooperation among people.

Bjarnason et al. conducted a multi-unit case study including of six software companies to gain insight into issues around requirements engineering and verification & validation. In their report [4], they discuss predominantly on the aligning the requirements with verification and validation process rather than the challenges that exist in both. Their result shows that the companies experience challenges in the achieving alignment including full traceability. However, their focus is mainly on the softer issues is a new turning. They have made excellent study and answered questions containing issues regarding the industrial alignment challenges, industrial practices for improving alignment and mapping between the challenges and practices. They conclude that human side is central and communication & coordination between people is vital, the quality and accuracy [7] of development organization play as a crucial stating point.

During our study, we have received similar responses despite the varied background of the companies under study. All the responses said that the people issues are a challenge that is currently of higher priority over the challenges related to tools and technology. These researches also discuss on the level of influence of the quality requirements, tools and cooperation/coordination and how they play a significant role to test or verify the product.

3 Research Methodology

This investigation aimed to understand and explain the challenges that exist in aligning requirements engineering and testing. The research questions (RQ) that provided the focus for this empirical investigation are:

- **RQ1:** What are the Challenges that exist in aligning requirements engineering and software testing?
- **RQ2:** What is state-of-practice in aligning requirements engineering and software testing?
- **RQ3:** How are requirements engineering and software testing alignment challenges addressed in industry?

Since the purpose of this study is to gain an in-depth understanding of the challenges an organization may face when aligning requirements engineering and testing, it is important to study software development organizations. A qualitative research approach was chosen because it allows the researcher to understand the studied phenomena and its context in more depth [20]. A qualitative approach is useful when the purpose is to explore an area of interest, and when the aim is to improve the understanding of the phenomena [14]. Due to the potential richness and diversity of data that could be collected, semi-structured interviews [14], [20] would best meet the objectives of this investigation. Semi-structured interviews help to ensure common information on pre-determined areas is collected, but allow the interviewer to probe deeper where required.

3.1 Research Design and Data Collection

The investigation can be divided into three phases:

Planning/Selection The sampling strategy used was a combination of maximum variation sampling [10] and convenience sampling [10] within our industrial collaboration network. The first author contacted a "gate-keeper" at each company who identified one subject that he/she thought was the most suitable and representative of the company to participate in this study. That is, the researchers did not influence the selection of subjects. Eight participants at eight software development companies participated. The research instrument (see Table 1) used in this study was designed with respect to the different areas of interest and inspiration from [15].

The companies themselves vary in respect to size, type of products, application domain, and geographical locations, which is illustrated in Table 2.

Table 1. The Interview instrument

Question
What are the tests based on?
What are the current challenges in aligning the requirements and the verification process?
How can you be sure that the test coverage is adequate?
How do you manage coordination between requirements and the verification process?
Do you have specific resources to verify if the requirements and verification process is aligned?
Do you have an existing solution for alignment?
Do you have tools or documents for alignment?
How are functional, business and quality requirements handled?
Does the testing involve customers?
What is the measure for alignment?
How is alignment verified in Product lines and outsourcing?
In what way will alignment benefit your organization?

Table 2. Company Characteristics

Company	Domain	Location	Role of interviewee	Type	Development Method
A	IT Services	Bangalore	Test Lead	Customer based	Waterfall
B	Automotive	Gothenburg	Usability Specialist	Internal support	Scrum
C	Automotive	Gothenburg	Software Architect	Internal support	Scrum
D	Automobile	Tokyo	Quality Assurance Manager	Internal support	Kanban
E	Construction Equipment	Chennai	Project Manager	Customer Based	Scrum
F	Product development	Boston	Developer	COTS	Kanban
G	Support and Services	Mountain View, CA	Delivery Manger	Customer Based	Scrum
H	Mechanical Spares	Gothenburg	Business Analyst	Internal integration	Iterative

Data collection The investigation used a semi-structured interview strategy [13]. One interviewee and one interviewer (the first author) attended all interviews. During the interviews, the purpose of the study and a general explanation of alignment were presented to the interviewees. Then, questions about the different areas of interests in relation to alignment between RE and ST were discussed in detail. In order to facilitate and improve the data analysis, for all interviews, which varied between 55 and 60 minutes, we took records in form of audio recordings and written extensive notes. Then, all interviews were transcribed.

Data analysis The content analysis [13] involved marking and discussing interesting parts in the recorded notes. The first author examined the transcribed data from different perspectives and carefully observed interesting results. The emerging patterns were then compared to the existing literature to understand and draw inductive inferences from the data. The data collection and analysis was an ongoing iterative process where the data was continuously collected and analyzed simultaneously. The entire data collection was analyzed for patterns in observations [16]. The results from the analysis is found in Section 4.

3.2 Validity Threats

For this study, as for any empirical study, there are validity threats worth discussing. We consider the three perspectives of validity and threats as presented in Robson [13].

Description validity: The two main threats to description validity in this investigation are the risk of participants not freely expressing their views during the interviews, and the risk of misinterpreting what is said during the interviews. The risk of participants not freely expressing their views was alleviated by the guarantee of anonymity as to all information divulged during the interviews. To mitigate the risk of misinterpretations, the interviews were recorded and written extensive notes were taken.

Interpretation validity: The main threat to providing a valid interpretation is that of imposing a framework or meaning on what is happening, rather than this emerging from what is learnt during the involvement with the setting. This does not preclude starting with a set of pre-defined categories, but these categories must be subjected to checking of their appropriateness, with possible modification. In this study, the threat of interpretation was managed by the researchers discussing the interviews and how the different researchers interpreted the interviewees answers. This was accomplished by having both researchers interpreted the interviewees answers, the coding of the statements, and the final results of the data.

Generalizability: Internal generalizability is concerned with conclusions drawn within the setting studied, i.e. the interviewees or observed situations should not be biased by the researcher. In addition, threat to selection bias is always present when subjects are not fully randomly sampled. However, given

that eight different companies from different industrial domains and geographical locations are included and interviewees were selected based on their roles by a "gate-keeper" at the company, these threats have limited effect.

Qualitative studies rarely attempt to generalise beyond the actual setting since they are more concerned with characterising, explaining and understanding the phenomena under study. The nature of qualitative designs also makes them impossible to replicate since identical circumstances cannot be recreated. However, the development of a theory can help in understanding other cases and situations. The fact that more than one participant acknowledged several of the discovered results and challenges increase the possibility of transferring the results to other situations.

4 Results and Analysis

This section presents the results and analysis of this study. The three sub-sections each corresponds to one of the three research questions presented in Section 3. In each sub-section, the findings of the interviews are presented in a table and followed by a discussion. In Tables 3 - 5, a '✓' indicates that there exists a challenge in that field, a 'X' indicates that there is no challenge, while an empty square indicates that challenges in the specified are is not applicable.

4.1 Current challenges in requirements engineering and software testing (RQ1)

In analysing Research Question 1 (RQ1), this sub-section discusses the current challenges and problems that exist in having a well aligned requirements engineering and software testing with respect to the type and size of company. Table 3 shows that there are some important issues to be considered with respect to requirements and testing.

Requirements related challenges As shown in Table 3, all the automotive based companies (B, C, D and F) say that requirements change continuously and most other companies also agree that prioritizing the existing requirements is one of the biggest challenges that exist today. All the companies agreed that well conformed requirements ease the process of testing as everyone knows what the end result should look like, similar to the discussion in Elizabeth et al. who states, it was very easy to report when we found defects, and there were not a lot of discussions between testers and developers, because everyone knew what was expected [2].

In the requirement process the customers are unaware of the exact need themselves. They have a vague idea of what they are expecting; however, they do not have a clear picture of what they want delivered. This slows down the process of development and many times creates confusion and uncertainty for the business analysts and developers.

Table 3. Challenges in RE and ST

Challenge	A	B	C	D	E	F	G	H
Requirements								
Changing requirements	✓	✓	✓	✓	✓	X		✓
Ability to express customer needs	✓	✓	X		X			✓
Complete requirement specification	✓	✓	X			X		✓
Prioritizing the requirements			✓	✓		✓	✓	✓
Understanding important features		✓	✓	✓	X	✓	✓	X
Testing								
Early tester involvement		✓	X	X		X	✓	X
Customer involvement	X	✓		X	✓	✓	✓	X
Adequate time	✓	✓	✓		✓	✓	✓	✓
Adequate budget	X	✓	✓	✓		✓	✓	✓
Edge cases	✓		✓		✓	✓	✓	✓

Its hard to reach the end users, states the UX analyst. From this statement, it is quite evident that it is not possible to always develop any product so accurately to the user needs.

Requirement engineers do not think about the fact that, all of the requirements elicited needs to be implemented and tested at a later stage. Hence, the requirements must be quantifiable/ testable. They tend to forget or ignore this aspect most times.

Another major issue that was discussed during the interviews is that, the requirements engineers must make sure that the requirements are aligned to the goals and visions of the company. They say that missing on these kinds of details may end up in unimaginable situations for the project.

Software testing related challenges As shown in Table 3, all the companies have a generic problem of insufficient time, especially towards the end of the project. It is not possible to fully test the product before release. Time issues can be easily solved if there is high budget or more resources. However, they are both inter-related to each other. A tradeoff has to be made in the TCQ time, cost, quality model in order to achieve the desired goals. Another challenge that most companies agree is that, there is a need for edge cases just like from Elizabeth et al. results, having full test coverage with unit tests gives a better security check that I have interpreted things correctly with acceptance tests. But building full test coverage has high difficulty level because of the unclear / non-verifiable requirements and incomplete requirements that are elicited in the requirements phase [2].

It is observed that, it is hard for a tester to fully understand the current situation and need for present market situation. Hence making a huge change in

the testing schedules at the last minute leaves dissatisfaction and imperfection. One of the interviewees said that time and changing requirements are the biggest challenges, that one can face in verification and validation. If verification and validation are not done in the initial phase, it may lead to lot of changes and rework. It also indirectly affects the traceability as well.

4.2 Aligning requirements with test (RQ2)

There is need for coherency in the requirements that was elicited and the final product that is being tested. In this section, traceability of this coherence in agile methods of development is discussed. Below, Table 4 shows the issues that exist in aligning requirements with test.

Table 4. Traceability and Agile Challenges

Challenge	A	B	C	D	E	F	G	H
Traceability								
Tracing changes	✓		X	✓	X	✓		✓
Prioritising over implement	X		X	X	✓	X	✓	✓
Adequate test coverage	✓		X		✓	✓	X	✓
Agile methods								
Works very well	X	X	✓	✓	✓	✓		✓
Design thinking	✓	X			✓	✓		X

Traceability related issues The IEEE Standard Glossary of Software Engineering Terminology [7] defines traceability as the degree to which a relationship can be established between two or more products of the development process, especially products having a predecessor-successor or master subordinate relationship to one another [. . .]. Research into traceability indicates that good traceability supports impact analysis [4], [6], [12], [19] and lowers test and maintenance costs [18]. In this section, traceability of this coherence is discussed. Table 4 shows the issues that exist in being able to trace the semantics of the project being developed.

The most basic kind of traceability is having a line of thought (not necessarily documented) from the requirements through to the defining and assessing of the test cases [2]. Company G states that, the most challenging issue is to make people understand the need of a good requirements analysis and its effect in traceability. People in the company usually want to directly implement and test without understanding if it is really needed or not.

However, in large companies people change projects and companies leaving behind no documentation. One interviewee said that due to reorganization is our

company, the responsibilities and tasks are transferred with limited knowledge transfer. Tracing back to the root of an issue will consume a lot of time and effort. So many a times there is a need to down prioritize traceability. This clearly indicates a loop hole so complex to solve.

Agile method related issues s seen in Table 4, most companies that we have interviewed, following agile methodology is the current trend and scrum is the most popular of them all.. However, to have people over process and working software over documentation lead to discussions and varied opinions. As the companies grow larger and have diverse projects in them, the companies feel the need to imply certain processes on its people, without which it feels impossible to operate and the same goes with having documentation. The software architect discussed that, documentation is a double edged sword, it could consume a lot of time or benefit in knowledge transfer. They also mentioned that, creating exhaustive documentation is definitely a cost and time overhead, but without which it feels impossible.

One of the interviewees states that it is not possible to fully follow some of the agile methods or traditional methods. There is a need for tradeoffs between them. One has to find the middle path that suits the situation.

4.3 Organizational Issues (RQ3)

Table 5 presents the softer issues that exists in the industry. It aims at people and their involvement with the product development and interaction with the product.

Table 5. Organizational Challenges

Challenge	A	B	C	D	E	F	G	H
People								
Persuading people	✓	✓		✓	✓	✓	✓	✓
Understanding the domain	X	X	✓	X	✓	X	X	
Managing the expectations	✓	✓	✓	X	✓	X	X	✓
Getting the right people to approach	✓			X		✓	X	✓
Market Strategy								
Entry to market	✓	✓		✓	✓	✓	✓	✓
Tool support usage			X	✓	✓	X	X	
UX/Customer Satisfaction								
Align UX with business analysts	✓	✓				✓		X
Basic feature implementation	X	X	✓		✓	✓	X	X
End users involvement	✓	✓	✓		✓	✓		X

People related issues As seen in Table 5, all the companies we interviewed said that convincing people about a new process or method is beneficial and is also the hardest task. And again it is evident from the findings, that the automotive industry has a major issue with managing the expectations of the customer. But on contrary to Gorschek et al. who says Professionals do not always have good technical knowledge and understanding about the work of other units, the software architect said that, Generally the competence of the expert is not what is the problem but the attitude or the state of mind of the expert to accept new things is a major issue. This could be for the reasons that:

- Lately, the companies recruit highly specialised people. They have started to master the art of picking the most competent people.
- People at the higher levels of management are filled with lots of experience but the same fact that they know it all objects them from exploring and learning new things.

At the end of the interview study, we realized that people problems are the biggest issues that the industry is currently facing. First is to build and understand a good working team. Next is for everyone in the project to have the same direction in terms of goals, vision and mission. Goal alignment is another common practice of basing launch decisions made by management on test reports produced by testers. Furthermore, tracing between artefacts provides a technical basis for supporting efficient communication of requirements [2].. Another major issue that all of our interviews were concerned about is to work in distributed teams. Company A, B, C, E, F and G were all global companies based almost everywhere in the world. They all mentioned that, time zones, cultures, ways of thinking and perception of end result always differs from person to person and bridging these issues is a major task. Specially in the automotive industry, the end product is tightly couple with human factors. It takes time, resources and many compromises. Since the vision of the requirements engineer, business analyst, developers and testers are not the same, the project is pulled in all possible directions giving the project an awkward shape.

UX/ Customer satisfaction related issues Agile development assumes a close connection to users and user-centered design assumes rapidly iterating designs with users [17]. A design is what the user can really see and interact with. The interviewee stated that, the purpose of any soft-ware is only fulfilled when the user can use it. A general question that is commonly discussed by the other companies is; how should the UX be like? At what stage should the design be implemented? Should you follow design upfront method or inte-grate the design in the last minute? If the design is made upfront there is a lot of re-work upon feedback. If the design is made in the end it limits the possible options. The product manager expresses his concern that, design decisions are the hardest as they the most tangible of all the features and the developers often neglect this factor. Due to which a lot of times, the project is delayed. The other issues discussed by company A are that the people in the top level

management assume that user experience/usability of a software is often not an important thing and it is down prioritized in case of low budget or time. Another similar consequence was discussed by company B and they suggested a solution to this problem. The solution is to have the prototypes merged in the requirements documents so that the designs are ready from the very first phase. This ensures that there is a good UX and also simplifies the work of a developer as the requirements feel more clear to implement.

5 Conclusions

References

1. Barmi ZA, Ebrahimi AH, Feldt R (2011) Alignment of requirements specification and testing: A systematic mapping study. In: Proc 4th Int. Conf. On Software Testing, Verification and Validation Workshops (ICSTW) pp. 476485
2. E. Bjarnason, P. Runeson, M. Borg, M. Unterkalmsteiner, E. Engström, B. Regnell, G. Sabaliauskaite, A. Loconsole, T. Gorschek and R. Feldt, 'Challenges and practices in aligning requirements with verification and validation: a case study of six companies', Empirical Software Engineering, pp. 1–47, 2013.
3. Damian D, Chisan J (2006) An empirical study of the complex relationship between requirements engineering processes and other processes that lead to payoffs in productivity, quality, and risk management. IEEE Trans Softw Eng 32(7):33453.
4. Daniela Damian, James Chisan, Lakshminarayanan Vaidyanathasamy, and Yogen-dra Pal. 2005. Requirements engineering and downstream software development: Findings from a case study. Empirical Softw. Eng. 10, 3, 255283.
5. Gorschek T, Davis AM (2007) Requirements engineering: in search of the dependent variables. Inf Softw Technol 50(12):6775
6. O. C. Z. Gotel and C. W Finkelstein. 1994. An analysis of the requirements traceability problem. In Proceedings of the 1st International Conference on Requirements Engineering (RE). IEEE, 94101.
7. IEEE. 1990. IEEE standard glossary of software engineering terminology. IEEE Std 610.12-1990, 184.
8. Kraut RE, Streeter L (1995) Coordination in software development. Commun ACM 38(3):6981
9. Kukkanen J, Vakevainen K, Kauppinen M, Uusitalo E (2009) Applying a systematic approach to link requirements and testing: A case study, In: Proc of Asia-Pacific Software Engineering Conference (APSEC 09):482488
10. M.Q. Patton, Qualitative Research and Evaluation Methods, Sage Publications, 2002.
11. POST, H., SINZ, C., MERZ, F., GORGES, T., AND KROPF, T. 2009. Linking functional requirements and software verification. In Proceedings 17th International Conference on Requirements Engineering Conference (RE). IEEE, Atlanta, USA, 295302.
12. Ramesh and M. Jarke. 2001. Toward reference models for requirements traceability. IEEE Trans. Softw. Eng. 27, 1, 5893.
13. C. Robson, Real World Research, Blackwell, Oxford, 2002.
14. P. Runeson, M. Höst, A. Rainer, and B. Regnell, Case study research in software engineering - guidelines and examples. Wiley, 2012.

15. G. Sabaliauskaite, A. Loconsole, E. Engström, M. Unterkalmsteiner, B. Regnell, P. Runeson, T. Gorshek and R. Feldt, Challenges in aligning requirements engineering and verification in a large-scale industrial context, Springer, pp. 128–142, 2010.
16. Savenye, Wilhelmina C., and Rhonda S. Robinson. "Using Qualitative Research Methods in Higher Education." *Journal of Computing in Higher Education* 16.2 (2005): 65-95.
17. Da Silva T, Martin A, Maurer F, Silveira M (2011) User-Centered Design and Agile Methods: A Systematic Review. 77–86.
18. M. Unterkalmsteiner, R. Feldt, T. Gorshek, 'A taxonomy for requirements engineering and software test alignment', *ACM Transactions on Software Engineering and Methodology*, 2013.
19. UUSITALO, E. J., KOMSSI, M., KAUPPINEN, M., AND DAVIS, A. M. 2008. Linking requirements and testing in practice. In *Proceedings 16th International Conference on Requirements Engineering (RE)*. IEEE, Catalunya, Spain, 265270.
20. R.K. Yin, *Case Study Research: Design and Methods*, Sage, London, 2003