

Requirements Engineering Process Improvement

Analyzing the Organizational Culture Impact and Implementing an Empirical Study to Evaluate the Benefits of Improvement

Jiyoung Jung, Sanghee Lee, Sangho Choi

LG Electronics.

Seoul, Republic of Korea

{jeeyoung.jung, sanghee3.lee, sangho109.choi}@lge.com

Seok-Won Lee

Dept. of Software Convergence Technology

Ajou University, Republic of Korea

leesw@ajou.ac.kr

Abstract— This paper describes an industrial experience of requirements engineering for the business process improvement in one of the biggest electronics companies in Korea. To improve the definition of the TV department business process requirements, we have applied a few methods to facilitate communications amongst stakeholders, based on their requirements. Our methods include iterative review processes, shared templates, and professional technical writing training. After applying our methods in a pilot project, the project stakeholders have confirmed that the approach provides a better requirements understanding and has improved requirements elicitation for the given examples. While implementing the project, we were also able to learn about both technical and nontechnical obstacles. Nontechnical obstacles were created by the organizational culture, including issues such as reduced empowerment, low levels of communication with other stakeholders, and a non-uniformly defined and not clearly understood mission statement. Most of the developers are very good at accomplishing their goals. They are very quick to respond to their management's requests, without excuses. However, often times, stakeholders have usually emphasized the importance of the results, rather than focusing on developing a strong quality process. The quality of work was highly dependent on the product development process. Therefore, in this study, the authors have analyzed the impact on the requirements gathering created by the stakeholders having a manufacturing process background and evaluating most decisions from a manufacturing perspective. The impact of the national cultural work style on the requirements engineering processes was also examined. In the future, we will continue to apply and expand the mentioned findings to further improve the requirements business process management.

Index Terms— Requirements engineering process improvement, organizational culture, product development process, national work patterns.

I. INTRODUCTION

Many studies have shown the benefits of requirements engineering process improvement [1]. It is difficult to change and improve the internal processes only considering the technical points of view, because communication patterns are strongly related to organizational characteristics [2]. Some studies have reported that cultural change is one of the success factors for Requirements Engineering (RE) process improvement [3].

Because of aforementioned studies, both technical and culture sensitive strategies should be used when approaching these kind of problems. So when trying to apply a certain strategy to an organization, the cultural characteristics of that organization should be considered [4].

One of the main goals of this study is to analyze the effects of cultural aspects on requirements engineering process improvement and understand the complex interplay between the technical and cultural perspectives. This paper describes the lessons learned from our case study, while trying to apply our findings for further requirements engineering process improvements.

II. THE CASE STUDY

A. Background

The analyzed company develops TV sets. It has a few hundred employees who work on developing only the software. It is a very successful company, and has developed successful products for many years. Because of that, some parts of the management do not think that further requirements specifications and refinement are required. Further, the SW has not played a big role in the TV development in the past. So the stakeholders are used to work on hardware centric product development processes.

While evolving the product line to align it to the latest trends in the industry, the company got involved in building smart TVs. The market in that segment is very dynamic. To quickly adapt to the trend, some technical expertise was brought in house by both insourcing and acquisitions. So the stakeholder numbers had grown, which had increased communication costs. The company has two groups for each role including those such as product planner, UX designer, GUI designer, for both developers and tester, both having similar roles. But they have different cultures for each group. Both want to clarify/analyze from their point of view. As an immediate impact, it was felt that the stakeholders' number growth had decreased the overall signal/noise ratio regarding the internal requirements alignment. It takes lots of time to develop new smart TVs. The company needed to improve, control and communicate changes more efficiently, in order to further improve time to market.

B. Approach to RE Process Improvement

For the first step, a sequential RE process improvement procedure was used - see “Fig. 1. [1].

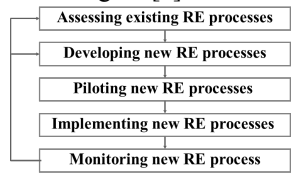


Fig. 1. Process improvement procedure of the study

The real procedure is more iterative than the “Fig. 1. ”. There are so many factors that affect RE process improvement that cannot be controlled and changed at once. Therefore, all five steps in “Fig. 1. ” should be activated, while process improvement is carried out continuously.

C. Findings

Our target of RE process improvement is the internal TV department in Korea. We need to establish our requirements before communicating with other organization.

We collected data through informal interviews, official meetings, documents and system inspections, and observations.

“Figure. 2” shows the current state of RE processes in the company.

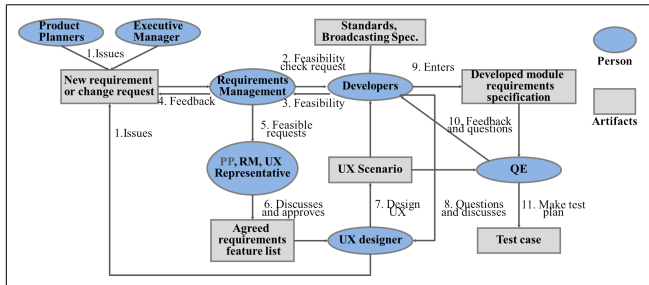


Fig. 2. Current state of RE process

1. Product planners (PP) and executive managers usually initiate their ideas as a new requirements or change request.
2. The requirements management (RM) team checks that the requirements are properly defined, but the checking only involves a developer doing a basic feasibility check.
3. The developer's check is usually correct. However, sometimes, the developer cannot be sure if the requirement is feasible or not, especially when requirements are too ambiguous. If that is the case, the approach proceeds without validation.
4. RM sends the developer's feedback to the PP.
5. The specific moment that they have a workshop making a baseline. PP, UX and RM give priority for each feature. Based on the results of the vote, PP chooses what features are included in the product scope.
6. An agreed upon requirements feature list is produced by representatives, but mostly by product planners. The requirements agreed upon don't always have a clear

meaning. There is also a continuous change requirements flow coming from planners and managers without rearranging schedule or priority. Once they get their ideas/suggestions/requirements in, product planners typically get working on other things and are not always easily accessible. So the developers develop given ambiguous requirements in their limited schedule. Some developers end up working very hard but sometimes just guessing, instead of working with a properly defined set of requirements.

7. The UX team then produces the UX design for the accepted requirements. However, the UX team needs to rephrase the requirements first because the original ones are too ambiguous.
8. Developers use an UX scenario as a requirements document, which is not enough to develop the final components. Therefore, the developers ask UX designers to specify quality attributes and exceptional case handling for the current scenario.
9. Developers specify software requirements specifications (SRS) after development. The specifications are used for making test cases.
10. QE asks developers to specify the test cases.
11. QE usually produces the test plans from the SRS.

The team business process flow then becomes very complex and difficult to trace both by the participants and by the management.

While working on the requirements improvement process, a few key issues had been identified, both from technical and communications points of view.

- Un-prioritized requirements – There was only one decision for specs that were “in” or “out” by a feasibility check, but not a prioritization and a more gradual evaluation. There was also no prioritization in the more detailed level of requirements.
- No impact analysis done for requirements changes -- If there were any schedule changes, the scope of the requirements scope needed to be reevaluated. There was no evaluation process for each requirement and it was not a big factor for prioritization.
- Incomplete requirements – Sometimes only the best -case scenario was described, without full exceptional case analyses and alternative use cases, including usage flow and nonfunctional requirements.
- Ambiguous requirements – There was no common standard sentence format. The lack of standardization and translation dilution made it easy to misunderstand requirements. There was no rationale for each requirement. It was difficult to understand and change requirements when necessary.

Our goals of RE process improvement were to reduce developer reworking by improving communication. To achieve our goal, we have selected some problems to target from our identified issues.

- Product planner's involvement – Product planners are only involved in high-level business requirements. We want to help them get more involved in daily

communications. Because of PPs have decision-making roles, developers and UX designers have to communicate without decisions if they can't reach a PP. No one can make decisions specifically enough, so developers implement software by trying to complete the missing parts as they find it suitable.

- Tester's involvement – the QE and QA are not involved in before features are complete. They review requirements and make test cases only after the implementation is complete. Making test cases after development causes lots of changes in requirements.
- Writing good requirements – There were no well written requirements for the developers. We made a shared template to improve stakeholders' understanding. We were focused on specifying the rationale of requirements. The template included a guide describing how to write and improve the quality of requirements. Properties like "complete", "unambiguous", "consistent" and "verifiable" have been added to the requirements.

We applied the pilot project with a few stakeholders, interested in only one feature. We wrote requirements for the feature they were interested about, using both the old approach and the new template. Stakeholders confirm that they can better understand the requirements than before while using the template, in the given example. They also agree that they need to clarify poorly defined requirements. It was difficult to follow the template at first, but they already have adapted so that all contents could fill the template. So after practice, they could easily write good requirements as compared to the previous one.

We also defined a new process such that the review process was reinforced to involve product planners who make decisions, and testers. We also suggested reviewing the step when the developer checks the feasibility, to have the developer confirm if the requirements are clear enough, so the developer can use them to develop the required components.

After applying the pilot project, we introduced our pilot result, which was a new process and plan to apply to high-level managers, including product planners and development organizations. They have not accepted our methods yet, because the product planner's manager didn't think what was described before was a problem.

Currently, we are trying to apply the methods bottom-up. We persuaded the requirement management team to work with our methods. We are writing user requirements and acceptance criteria, and getting reviews from every related organization for each requirement. The developers and testers are mostly happy about our work. But we feel it is difficult to proceed quickly, mostly because of the mentioned cultural problem in the organization.

These problems have been analyzed by applying the framework developed by Denison and Mishra [5]. The framework has been typically used to diagnose cultural problems in organizations. It is based on four cultural traits of effective organizations, which are involvement, consistency,

adaptability, and mission. We could find many problems from cultural point of view, are:

- Weak empowerment – No one can make decisions in many cases, especially to eliminate specific requirements. While the time spent in meetings increased, few issues were resolved.
- Hierarchical system - If a high level manager suggested an idea, it was implemented as the highest priority without proper impact analysis. The developers just implemented it without questions. This has roots in the Korean Confucian culture, in which there is a high level of respect for the elders (or higher ranked persons). That is a great idea in general, but in cases where the elders may not be familiar with the problem domain, some decisions can be detrimental to the stakeholders. A more balanced decision process, split between the management and problem domain experts, is highly recommended in this case.
- Developers created software even if the requirements were not sufficient. They also applied changes quickly. This was in principle very efficient, but sometimes generated something different than the expected results. The products were made fast and well, but without a proper requirements definition, it was difficult to show the product users the product value, even though the product was functional.
- There was no shared vision for the specific product. It is difficult to share information directly across even small organizations. So the developers should ask questions as specifically as possible, and share information after every work component is done already.
- We can elaborate on the listed problems during the workshop.

We are trying to overcome these problems while applying our methods, by showing the benefits of our work. But we have yet to solve all of those problems. It takes too much time to proceed for each step. We need to follow all of the procedures and persuade every single person to do so as well. We are also currently working without the product planners because we are not convinced them to participate as of now. We will introduce our method again with persuasive and impactful results to much higher levels of management after finishing our current project. We feel that if the higher management understands the issues within the process, it will immediately make sense as to why we need to change the business process requirements definition.

III. CONCLUSIONS

Considering that RE research is done in countries with different cultural backgrounds (i.e. USA and Europe), we found there is a significant mismatch when translating that research to Korean culture or when sharing good Korean practices with other international stakeholders. Most of the experienced electronics companies have changed their focus from hardware to software. We have tried to improve the challenging cross-cultural technical and nontechnical issues,

while adapting the above-mentioned steps for business process improvement. We will continue to further improve the cross-stakeholders communication to help the company benefit from good Business Process RE practices.

REFERENCES

- [1] D. Damian, D. Zowghi, L. Vaidyanathasamy, and Y. Pal, "An industrial case study of immediate benefits of requirements engineering process improvement at the Australian center for Unisys software", *Empirical Software Engineering Journal* 9(1–2): 45–75, Mar 2004.
- [2] T. Hall, S. Beecham and A. Rainer, "Requirements problems in twelve software companies: an empirical analysis", *IEE Proceedings:Softw.* Vol. 149, No.5, October 2002
- [3] M. Kauppinen, S. Kujala, T. Aaltio, L. Lehtola, "Introducing requirements engineering: how to make a cultural change happen in practice", *IEEE Joint International Conference on Requirements Engineering*, 2002.
- [4] Gert Jan Hofstede, *Cultures and Organizations: Software for the Mind*, McGraw-Hill Companies, 2004.
- [5] Daniel R. Dension and Aneil K. Mishra, "Toward a theory of organizational culture and effectiveness", *Organization Science*. Vol.6, No.2, March-April 1995.
- [6] Carl F. Fey and Daniel R. Denison, "Organizational culture and effectiveness: can American theory be applied in Russia?", *William Davidson Institute Working Paper Number 598*, July 2003. .