# Stakeholder Management Methodologies for Software Development Projects Having Complex Stakeholder Environment with Different Sense of Urgencies

Selin Sengur
(4209370)
Faculty of Technology, Policy and Management
Delft University of Technology
Jaffalaan 5, 2628 BX Delft, The Netherlands
S.Sengur@student.tudelft.nl
August, 2015

**Abstract**

The needs of companies are evolving and becoming more demanding each year, which resulted in more complex stakeholder relations. Stakeholders do not always have the same sense of urgencies and their expectations from the same project might differ. Although current literature suggests successful methodologies for software development projects, it remains inadequate when different sense of urgencies take place among the stakeholders. This creates the risk of developing negotiated nonsense software. There is a need of a more broaden approach where human types of issues are involved together with the traditional project management methods. This research aims to propose a set of stakeholder management methodologies that could be implemented in a software development project environment where different sense of urgencies exist. Suggested methodologies are validated by expert opinion.

## 1. Introduction

Stakeholder relations are getting more and more complex because of the evolving needs of companies and different sense of urgencies among stakeholders (Ruuska & Vartiainen, 2003) (Strang, 2003).

The software development projects studied in the literature usually have one client with the same sense of urgency. Therefore, current project management methodologies mostly focus on the development of the end-product and involving the stakeholders into the project to get their feedbacks and increase their understanding on the end delivery. They do not focus on solving the conflicts among stakeholders.

For instance, agile approach is a contemporary and effective methodology that increases the success rates of the software development projects by reducing the planning complexity, emphasizing the customer value and encouraging the stakeholder involvement and collaboration. (Highsmith, 2002). However, it remains inadequate on addressing different sense of urgencies among stakeholders. Therefore, agile methodology should be supported by more human kind of approaches.

The key point of this research is that the lessons learned from the literature is elaborated in the way that can be implemented in a project management environment. Stakeholder management techniques derived from the literature are mostly designed for the political environment, where there is a lot of negotiation takes place but no end-product is developed. However, unlike politicians, project managers have the responsibility of delivering a successful end product with several restrictions such as cost, time and

resource. This fact urges project managers to keep their focus on the end delivery rather than ongoing and endless negotiations. Still, negotiation is an inevitable and crucial part of a project with multi-stakeholder environment especially when different sense of urgencies takes place. Therefore, it is project manager's responsibility to find the balance on putting the right amount of effort between the product development and negotiation to avoid either negotiated or design nonsense software.

Apart from the methodologies designed for political environment, this chapter also includes literature that is already being used in project management environment but not particularly in software development projects such as stakeholder mapping.

The research starts with explaining the dilemma between developing negotiated nonsense vs design nonsense product, which the project manager should take into account. The first methodology that is suggested is mediator to ease the conversation between the users and software developer. Second one is stakeholder mapping technique to understand the power relations between the stakeholders. Third is multi issue decision making technique to involve the reluctant parties into the decision making loop. The question of whether the proposed methodologies could be applied and succeed in software development projects is validated by expert view.

## 2. Agile Approach and a dilemma: design vs negotiated nonsense end-product

If the project manager puts excessive amount of effort and time on negotiation, all stakeholders are satisfied with the tradeoffs and negotiation results but there is not enough time to improve the quality of the software or end product itself. On the other hand, if the project manager gives little attention to the negotiation, there is a risk of building a robust and high quality software but not accepted by stakeholders (design nonsense), which is non-implementable solution (De Bruijn, Ten Heuvelhof, & In't Veld, 2010). Therefore, project manager should make a good balance between putting enough effort both to the product design and the negotiation. Agile methodology, which is described thoroughly in the literature is an effective approach to make this balance in software development projects. Some of the studies in the literature on Agile approach is as follows:

Dingsøyr et. al., (Dingsøyr, Dybå, & Moe, 2010) compares the studies related to agile approach between the years 2003 and 2011. Strode et. al., (Strode, Huff, Hope, & Link, 2012) concentrate on improving the coordination and Drurya et. al., (Drurya, Conboy, & Power, 2012) discuss decision-making in agile software development projects. The applicability of agile approach on post-adaptation stages has been investigated as a case study by Senapathi et. al. (Senapathi & Srinivasan, 2012). Furthermore, Adolph et. al. (Adolph, Hall, & Kruchten, 2011) has showed that improvements on agile approach is achievable based on considering social factors in software development. Although these studies how agile approach is successful on managing software development projects, none of them give a focus on complex stakeholder environment with different sense of urgencies. Conflicting sense of urgencies require more effort on negotiation that may risk the balance between the design vs negotiated nonsense dilemma. Therefore, agile methodology should be supported with soft systems methodologies. The next section provides a set of methodologies that can be

implemented in the planning phase of the software development projects.

# 3. Methodologies

Methodologies given in this section should be used in addition to the agile approach, in the planning phase of the software development projects to set up effective negotiation settings. The methodologies suggested in this research is as follows:

1. Mediator
2. Stakeholder Mapping Technique
3. Two Poles Approach
4. Multi-issue Decision Making

## 3.1 Mediator:

Mediator is decided by the contractor after the client reaches out for help. He becomes the part of the team. In the literature this person (mediator) is called as scrum master. One of the unique properties of the agile methodology is the existence of the scrum master. The scrum master is the main link between the development team and the product owner. The scrum master also controls the progress of the development team in the progress (Schwaber & Beedle, 2002). Assigning a scrum master (mediator) is important to have an effective communication between the development team and the client. As he acts like a glue between these two stakeholders, he can direct the scope of the negotiation sessions, which will ease to manage the complex stakeholder environment.

## 3.2. Stakeholder Analysis and Management Methodologies

The decisions are made in a process of interaction between the interdependent players in a network. In a software development project with complex stakeholder environment, the initiator of the project should inquiry how to influence the process of interaction. He should know which other actors may contribute on serving his interests and how could the actors be committed to those interests. Therefore, the actor should give the primary focus on the actors, whose support is needed, rather than the content of the problem (de Bruijn & ten Heuvelhof, 2008).

### 3.2.1. Stakeholder Mapping Technique

In order to apply the right strategy on actors involved, manager should make a stakeholder analysis to identify the interest of each actors and the power relations in between. For this, stakeholder mapping technique is recommended. Murray et. al., (Murray-Webster & Simon, 2006) recommends the following analysis to have understanding about the stakeholders:

**Power vs Interest:** It helps the analyzer to understand the power of the stakeholder but also whether the stakeholder will be active (interested) or passive in the project. However, it is useless without knowing whether the stakeholder is “against” or “for” to the project.

**Interest vs Attribute:** It is important to understand whether the stakeholder is active backer or active blocker. However, before this analysis it is important to know the power of the stakeholder, so it will be clear whether the stakeholder is influential or not.

Newcombe (Newcombe, 2010) argues in his book that stakeholder-mapping technique is an effective methodology for project managers on conducting the analysis of power, predictability and interest of key project stakeholders. He supports his arguments by exemplifying a large construction project. Furthermore, Bryson et. al., (Bryson, 2007) give insight in his book on how and why managers should use stakeholder identification and analysis techniques to help their organizations to fulfill their targets and creating a public figure. His research argues that wise use of

stakeholder analyses can help framing the solvable issues that are technically feasible and politically acceptable. Moreover, Crosby et. al., (Crosby, 1991) explains the purpose of stakeholder analysis as having an indication of whose interests should be taken into account when making a decision. He argues that stakeholder analysis is a vital tool for strategic managers on the recognition of the key roles of the stakeholders on determining the policies, their implementations and outcomes.

### 3.2.2. Which Stakeholders to Involve: Two-poles approach:

In a software development project with complex stakeholder environment and divergent requirements, project manager should make a selection among the stakeholders to give a prioritization. Prioritization is important on a software development project to decide whose expectations are met first and on which stakeholders' interests should be invested most.

The purpose of the decision makers/project managers in such projects should be involving the key stakeholders into the design and development process of the software as actively as possible. For this purpose they should design the most optimal strategy to involve those actors into the process. Sengur (Sengur, 2015) concludes in her study that two-poles approach is the most efficient approach to develop software in a complex stakeholder environment and changing requirements. When there are diversified requirements among the stakeholder, it is difficult to set up the requirements in the first iteration of the project. Therefore, it is recommended to select two powerful stakeholders, which have the most diversified requirements. As there will be very small overlap between the requirements of two selected actors, the requirement set up will continuously evolve in each iteration and the requirements will be prioritized and then implemented as a result of the negotiation sessions of the project.

Two-poles approach stands for selecting two giant pilot stakeholders that has the most diversified requirement and systems requires designing the software according to the requirements of the most diversified stakeholders and adapting others after the tool is being implemented in pilot actors (Sengur, 2015).

*Benefits of the two-pole strategy:*

If the biggest fighters are managed to compromise, it is easier to align compromise with the less powerful stakeholders because of the following reasons (Sengur, 2015).:

- If the proposed solution has already implemented by the biggest and the most diversified players and worked, it will be easier to convince the other parties to involve because the benefits of the new system will be visible. Their perception about the proposed solution will be like: "If this solution works with the biggest players, it will work with rather smaller parties too!"

- The smaller players will want to follow the pattern that bigger players adopt. People have a tendency to follow the successful and powerful examples (bandwagon effect)

- If the biggest players are already committed to the new system, small players will feel the pressure to adapt it as well or big players can use forceful power on the smaller once. Smaller or less powerful players will not resist so much to accept the new system because they will not want to harm their relations with the bigger players. It

is likely for them to need the support of big players on other issues in the future.

- Earlier the bigger players are involved into the project, less costly to give up the project. If the biggest fighter cannot compromise, it will be realized at the first step before other countries are involved. The regret cost will be lower. No matter how many stakeholders are convinced to adapt the new solutions, if the biggest players do not accept to involve, the project will fail and will not be implemented. The regret cost will be much higher if the biggest players are involved to the project at the end and do not accept the solution.

- Earlier the biggest players are involved into the project, quicker the return on investment from the project is gained. The impact of the biggest players on the outcome is usually higher than the smaller players. Therefore, if the biggest stakeholders start implementing the solution at an earlier phase, the impact of the project will be felt earlier.

- If the biggest players join the project at first, the project will be designed based on the inputs and the requirements given by the biggest players. That means, it is higher chance to convince the biggest players to implement the solution. Once the biggest players implements the solution, it is easier to involve smaller actors because of the reasons mentioned above.

### 3.2.3. Multi-issue decision making

Stakeholder involvement in every part of the project cycle is crucially important in software projects because the software is implemented based on the continuous input provided by the client/stakeholders. However, in most cases involving the client to the software development process becomes a major issue. The reasons vary (de Bruijn & ten Heuvelhof, 2008).:

- Software projects are very technical for many clients, their understanding is shadow on the final delivery and it's benefits. That causes stakeholders showing reluctance on involving and contributing the process.
- They resist on change because it is difficult. They have a tendency to keep their own system, which is already in use, even though the new system offers more efficiency or less cost, etc.
- They have conflicting interests with some of other stakeholders, so lack of belief on achieving their goals on the project. Therefore the project is not always interesting to them.
- They already have their backup system, they want to keep their old system, which they are used to or committed to.

Given the reasons above, involving the stakeholders to the development process of the software becomes a major difficulty. In order to increase the speed of their involvement to the process, multi-issue decision making game that is introduced by de Bruijn et. all., (de Bruijn & ten Heuvelhof, 2008) can be a useful tool. The initiator of the project can make the project attractive to the stakeholders they want to involve by using the multi-issue gaming.

Multi-issue game might be a useful tool to draw the attention of each stakeholder and involve them into the development of the software. If an incentive is created for each stakeholder, they will be willing to provide more input and feedback to speed up the process of the implementation (de Bruijn & ten Heuvelhof, 2008).

*How to Apply* (de Bruijn & ten Heuvelhof, 2008)*:*

- The initiator of the software development project asks each stakeholder he wants to involve into the decision making/software development process to write down their expectations/wishes from the initiated project.
- Initiator puts all the wishes/requirements written by the stakeholders into the agenda to make the project appealing for each actor and prevent them to make excuse not to join the process.
- Initiator invites the stakeholders into the decision making process for the software development.
- Once they are in the decision making process, they will not be eager to leave the process even if their initial wishes that are put into the agenda at the planning phase of the project are not totally realized. It is because they will jeopardize not only themselves but also other stakeholders' prospect of gain in the project once they decide to leave.

The literature is quite limited on how to apply the multi-decision-making technique but the importance of multi-issue decision-making is being discussed and several studies exist on how to improve the efficiency of the multi-issue decision-making can be improved. One of these studies on multi-actor decision-making is done by Coehorn et. al., (Robert M. Coehoorn & Jennings, 2004). They highlight the importance of learning opponent's preferences to make effective multi-issue decision-making. For this, they suggest kernel density estimation to make negotiations more efficient regarding the utility and the time of agreement. Moreover, Raymond Y.K. Lau (Lau, 2005) emphasizes the characteristic of real-world negotiations such as complex negotiation spaces, tough negotiation deadline, limited information about the opponents and volatile negotiator preferences. They propose GA-based adaptive negotiation mechanism because it supports multi-party multi-issue negotiations based on a distributive decision-making model. According to the study of Lau, his mechanism guarantees theoretically optimal negotiation model. Lastly, Luo et. al., (Luo , Jennings, Shadbolt, Leung, & Lee, 2003) develops a fuzzy constraint based model for bilateral multi-issue negotiation in trading environments. They seek for a fair deal for both parties while maximizing their own pay-offs.

## 4. Validation

The methodologies suggested in section 3 are validated by two software development managers by setting up interviews. First expert has extensive management experience with software development projects having complex stakeholder environment. As he uses these techniques regularly in his projects within the consultancy agency he is working, he confidently approved and validated all the approaches mentioned in this research. Second expert is a software development manager, who has seven years of experience in this field. She approved and validated all the methodologies suggested in the management guideline.

## 5. Conclusion

Software development projects, having stakeholders with different sense of urgencies require significant amount of discussion and negotiation as all stakeholders need to align on a consent of what to implement and what not to. While putting effort on negotiation, project manager should not forget that he needs to deliver a successful product as an output of the project. Therefore, he needs to find the right balance in between. By this balance, it

will be possible to develop a robust software that the stakeholders like and willing to implement.

To be able to manage different sense of urgencies while guaranteeing developing both negotiated and design sensible product, stakeholder management techniques should be used in software development projects together with the agile methodology. This paper provided a set of stakeholder management methodologies that can be applied in software development projects, where different sense of urgencies exist among the stakeholders: Mediator, stakeholder mapping technique, two poles approach and multi-issue decision making. The methodologies are validated by software development experts, who have managed software development projects for years. Experts confirmed that suggested methodologies are effective to deal with different sense of urgencies in a complex stakeholder environment with different sense of urgencies and can be implemented in software development projects.

## 6. Bibliography


Adolph, S., Hall, W., & Kruchten, P. (2011). Using grounded theory to study the experience of software development. *Empirical Software Engineering, 16*(4), 487-513.

Bryson, J. M. (2007). What to do when stakeholders matters. In *Public Management Review* (pp. 21-53).

Cowie, G. (2003). The importance of people skills for project managers. *Indust Commer Train, 35*(6), 256-258.

Crosby, B. L. (1991). Stakeholder Analysis: A Vital Tool for Strategic Managers. In *Technical Notes.*

de Bruijn, H., & ten Heuvelhof, E. (2008). *Management in Networks On multi-actor decision making.* Routledge.

De Bruijn, H., Ten Heuvelhof, E., & In't Veld, R. (2010). *Process Management: Why Project Management Fails in COmplex Decision Making Processes* (2nd Edition ed.). Springer.

Dingsøyr, T., Dybå, T., & Moe, N. B. (2010). *Agile software development: current research and future directions.* Springer Science & Business Media.

Drurya, M., Conboy, K., & Power, K. (2012). Obstacles to decision making in Agile software development teams. *Journal of Systems and Software, 85*(6), 1239-1254.

Highsmith, J. A. (2002). *Agile software development ecosystems* (Vol. 13). Addison-Wesley Professional.

Kliem, R., & Ludin, I. (1997). The people side of project management.

Lau, R. Y. (2005). Towards Genetically Optimised Multi-Agent Multi-Issue Negotiations. *Proceedings of the 38th Hawaii International Conference on System Sciences* . IEEE.

Luo , X., Jennings, N. R., Shadbolt, N., Leung, H.-f., & Lee, J. H.-m. (2003). A fuzzy constraint based model for bilateral, multi-issue negotiations in semi-competitive environments. *Artificial Intelligence, 148*(1-2), 53-102.

Murray-Webster, R., & Simon, P. (2006, November). *Making Sense of Stakeholder Mapping.* Retrieved December 18, 2014, from PM World Today: http://www.pmforum.org

Newcombe, R. (2010). From client to project stakeholders: a stakeholder mapping approach. In R. Newcombe, *Construction Management and Economics* (pp. 841-848).

Robert M. Coehoorn, R. M., & Jennings, N. R. (2004). Learning an Opponent's Preferences to Make Effective

Multi-Issue Negotiation Trade-Offs. (pp. 59-68). ICEC.

Ruuska, I., & Vartiainen, I. (2003). Critical project competences – a case study. *Journal of Workplace Learning, 15*(7/8), 307-312.

Schwaber, K., & Beedle, M. (2002). *Agile Software Development with Scrum.*

Senapathi, M., & Srinivasan, A. (2012). Understanding post-adoptive agile usage: An exploratory cross-case analysis. *Journal of Systems and Software, 85*(6), 1255-1268.

Sengur, S. (2015). *Management Guideline for Software Development Projects.* Delft.

Strang, K. D. (2003). Achieving organizational learning across projects. *PMI North America Global Congress.* Baltimore, USA.

Strode, D., Huff, S. L., Hope, B., & Link, S. (2012). Coordination in co-located agile software development projects. *Journal of Systems and Software, 85*(6), 1222-1238.