Beliefs Underlying Teams Intention and Practice: 
An Application of the Theory of Planned Behavior

Carol Passos 1, Daniela S. Cruzes2,3, Manoel Mendonça4

1 Department of Computer Science (DCC), UFBA, Salvador-BA, Brazil
mcarolpassos@dcc.ufba.br
2 Department of Computer and Information Science (IDI), NTNU, Trondheim, Norway
dcruzes@idi.ntnu.no
3 SINTEF, NO-7465 Trondheim, Norway
4 Fraunhofer Project Center at UFBA, Salvador-BA, Brazil
manoel.mendonca@ufba.br

Abstract. Many theories aim to understand the beliefs underlying an intention or behavior. These theories are currently used to seek answers about how people progress from intention to practice in business environments. The Theory of Planned Behavior (TPB) is today one of the most popular socio-psychological models for the prediction of behavior. It is believed that people consider the implications of their actions and act based on a reasonable assessment of those implications. In this context, belief can be defined as the psychological state in which an individual holds a proposition or premise to be true. So, behavior is driven by what is believed, by what is culturally assumed to be true about the world. Our work aims to study and characterize a belief system applying TPB in project teams in terms of organizational and team levels factors associated to beliefs about the software development practices. A set of interviews on origins, sources and impacts of beliefs on software practices was conducted with professionals from different project teams and companies. The results point out to a strong influence of past experiences and show that it is possible to characterize belief systems in software project contexts within a behavioral perspective.

Keywords: belief system, attitude, behavior, planned behavior, perceived behavioral control, software development practice.

1 Introduction

A project team’s belief system is the foundation of software engineering practices adoption [31]. Most of the practitioners rely on beliefs to make their methodological choices. The definitive step towards the introduction of a new practice into an organizational culture is to capture and understand the beliefs and values of the project teams that can guide behavior and important decisions. This requires a deep grasp of the organization belief system and related factors. How do people come to believe that something is useful to them and reach the decision to use it in their
organization or particular project? This is a very interesting subject to be studied and understood by empirical software engineering researchers.

Beliefs can be defined as conceptions, personal ideologies, and perceptions of the world that shape practice and orient knowledge. The concept of belief implies the existence of a mental state with intentionality, interacting with goals and influencing ordinary actions [1]. These beliefs are built over a set of interactions, relationships, processes and activities of the group. Beliefs exist in the form of expectancy-rules and these rules are tested for a given situation. In this context, actions are driven by what is believed, by what is culturally assumed to be true about the world [13].

Declared and acknowledged project team's beliefs are often consistent with what is found in observations of software practices; and even the unannounced ones can influence specific aspects of the software process. Research has documented that practitioners' beliefs related to work processes have a significant impact on their behavior and that this influence on practices manifests itself in interesting ways [1][10][13][31]. However, most of the existing studies have not directly addressed or characterized the belief system in a software project context within a behavioral perspective.

There is a common understanding that beliefs and behavior are related. In order to determine what practices an individual is likely to perform at any given time, it is necessary to understand the set of beliefs about the individual's behavioral intention in a given circumstance. In the Software Engineering (SE) area, this lack of understanding may be due, in part, to a deficiency to investigate, understand and document the nature and effect of the belief system underlying current SE theory and practice.

Many theories have been developed to explain health-related intention and behavior and to seek answers for business administrators about how people progress from intention to behavior. The Theory of Planned Behavior (TPB) [2][3] is today one of the most popular social-psychological models for the prediction of behavior. The TPB provides a suitable framework for conceptualizing human behavior in organizational contexts, because it was developed specifically to account for behavior for which actual and perceived control may be low (spontaneous behavior).

We have run a case study [21][22] to characterize a belief system applying Theory of Reasoned Action (TRA) [4][7][11] in agile software project teams in terms of origins, sources and impacts of beliefs on self-management development practices [9][15][20]. The study addressed the influence factors associated to team beliefs, its attitudes toward behavior, the organizational culture and subjective norms to predict behavior intention and also document the inconsistencies between declared beliefs and real practice in agile software projects. With this study, we showed that it is possible and interesting to capture and represent a belief system in a project context and there are a strong influence of past experiences and organizational culture on self-management practices of agile teams in software industry.

The TPB is a generalization of TRA that is indicated to predict behaviors that are not entirely under an individual's volitional control. It fits well in the context of SE research within software industry and provides a suitable theoretical framework in mapping software project teams' behavior. After a literature review and evaluation of research objectives and questions, we decided to apply TPB to study team behavior in
software engineering and opted for an interview-based qualitative data collection approach.

The aim of this paper is to present a study to characterize a belief system applying TPB in project teams in terms of origins, sources and impacts of beliefs on software development practices. In order to reach this goal, we performed a set of interviews and collected information about the project contexts. A conceptual framework was built, based on the TPB model, to focus and bound the collection of data. The study addresses the influence factors associated to team beliefs, its attitudes toward behavior, the organizational culture and normative pressure to predict behavior intention and also document the inconsistencies between declared beliefs and real practice in software projects. We relate our findings to relevant behavioral literature [2][3][4][5][11][12][26] in order to contribute to an improved understanding on how to apply this kind of theory to better study SE practices.

The rest of the paper is organized as follows: Section 2 gives an overview of the background. Section 3 presents the research methodology and describes how we conducted the study. Section 4 presents the results of applying Theory of Planned Behavior in the context of software development practices. Section 5 discusses the implications and limitations. Conclusions and opportunities for further work are presented in Section 6.

2 Background

2.1 The Theory of Planned Behavior

Fishbein and Ajzen [11] gave us a robust definition of behavior, avoiding the confusion and ambiguity of some past theories. Their research was derived in a solid body of work for a more uniform study of these terms [14]. They sought out a way to not only predict behavior, but also to understand its relationship with beliefs and their strength.

In more recent publications, Ajzen [2][3] has extended the Theory of Reasoned Action (TRA) to the Theory of Planned Behavior (TPB) by including a measure of perceived behavioral control, which it is argued will increase the prediction of intention and behavior in those instances where the behaviors are not entirely under the control of the individual or group.

Although the TRA model can predict the probable behavior, it may not predict the actual behavior, because people do not always do what they intend to do and there may be other factors that will cause them to go against their initial intention.

When used to explain behavior that is not fully a conscious choice or decision, the TPB is expected to perform better than the TRA. According to TPB theory illustrated in Fig. 1, human behavior is guided by three kinds of considerations: (i) beliefs about the likely outcomes of the behavior and that the evaluations of these outcomes produce a favorable or unfavorable attitude toward the behavior; (ii) beliefs about the normative expectations of references, i.e. subjective norms; and (iii) beliefs about the presence of factors that may facilitate or inhibit behavior performance and that the perceived power of these factors gives rise to perceived behavioral control.
The evolution of the TPB and research is marked by a debate about the meaning of the third variable [32]. The current dual-aspect conceptualization of perceived behavioral control is determined by two important factors such as perceived autonomy and confidence related to how easy or difficult behaviors can be. Those factors can be both internal (knowledge, skill, willpower) and external (time, money, resources, cooperation of others). From this point of view, people will believe that they can carry out their intentions when they believe that they have the resources and opportunities to perform the behavior and when they believe that they can freely make the decision to use those resources and opportunities.

![Fig. 1. TPB Model.](image)

We believe that TPB can be helpful in generating rich and detailed accounts of software project teams, the interactions between their members, and, especially, the actions oriented toward certain software practices. The TPB model appears to provide a better conceptual framework for dealing with the complexities of organizational contexts and to understand the influence of a belief system on team's practices. However, the application of a theoretical approach to SE can be challenging. The problems and objects of study in SE require approaches suited to their dynamics and contexts.

TPB has been widely used to predict and explain health-related intention and behaviour and attitude toward the behavior (whether the behavior is seen as good-bad or pleasant-unpleasant) and subjective norms (perceived social pressure from relevant others) were found to be significant direct predictors of intention and participation in many studies [8][19][23][24][25][29][30]. Facilitating and hampering conditions were also found to have significant effects in the prediction of participation [16][18][27][28][32].

### 2.2 Research Conceptual Framework

The conceptual framework idealized for this research was derived from an adaptation of the TPB model and is shown in Fig.2. In this framework, the values and beliefs of a project team, in addition to the attitude toward its behavior, represent the strength of beliefs. It motivates people toward a behavior intention given that attitude is a predisposition to act in a positive or negative way toward an object. Another factor,
represented by subjective norms added to organizational culture, can also impact the team's behavior intention and, consequently, the team's practices. Therefore, the behavioral intention is affected by what others think and the strength of their opinion on the organization in context [4]. Lastly, perceived behavioral control denotes people's perception of the degree to which they are capable of, or have control over, performing a given behavior [12]. Believing that they can perform a practice, because they have capacity and autonomy, motivates project teams to try to perform the respective behavior and increases the likelihood that they will expend effort and persevere in their attempts [32].

TPB fits in the context of our research, because it allows us to study the way beliefs, attitudes and sense of self-efficacy are formed and their relationship to behavior and practice with room to explore other relevant aspects. Moreover, TPB has been widely used to predict and explain health-related intention and behavior and these results were successfully replicated [18][28][32].

3 Methodology

We have being running a long-term case study, involving software development projects, which has gone through a main cycle of 18 months. This study is now undergoing another cycle, lasting approximately eight months, with the aim of characterizing a project team's belief system in three organizations in Brazil (see Table 1) and to investigate its origins, sources, and impacts on the team's software development practices. In particular, we have been focusing on the social behavior and relationships that arise as an intrinsic part of adopting new software practices. To do this, we have been applying an interview approach, asking insightful questions, drawing maps of the projects context, and collecting some artifacts.

All three companies studied provide software development and evolution services for customers of both the public and private sectors. The interviews was performed according to [17] in all companies. So far, the findings seem to confirm the influence of team's belief systems on software practices and methodological lessons learned
from the first cycle of this research [22] were used to define the next cycle. After results obtained in the first cycle and reported in [21], new research questions (RQ) arose to direct the next phase, as follows:

- **RQ1**: How do beliefs and attitude influence team practices in software organizations?
- **RQ2**: How do organizational culture and subjective norms influence software team behavior and practices?
- **RQ3**: How do team autonomy and confidence impact software practices?

A particular moment of the research is being explored in this paper. It involved a scope of nine interviews in three of the three companies during the second cycle of the study. This second cycle included the application of the behavior theories to map and analyze the relevant experiences of the software team members and uncover the beliefs which could hinder or benefit the adoption of new software practices. Using TPB as a guide, we could also focus on how the influence of organizational culture, subjective norms, team confidence and perceived autonomy, embedded in the software development context, might help to predict behavior intention.

### 3.1 Data Collection and Analysis

After a literature review, designed to address the key references to the behavior theories related, we prepared an initial version of the interview questions to identify some issues we intended to investigate. The purpose of this round was to cast light on the respondents’ past experiences, beliefs emerged or evolved from these experiences, impacts of new agile practices on projects and unexpected effects of known and new methods or techniques. Guided by TPP model, we asked them to retell and relive specific and directed stories that illustrated the beliefs and attitudes we were trying to capture.

#### Table 1. Companies under Study.

<table>
<thead>
<tr>
<th>Company</th>
<th>Age</th>
<th>Personnel</th>
<th>Software Process Certification</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>11 years</td>
<td>800</td>
<td>ISO-9001 and CMMI Level 2</td>
</tr>
<tr>
<td>2</td>
<td>03 years</td>
<td>15</td>
<td>---</td>
</tr>
<tr>
<td>3</td>
<td>19 years</td>
<td>42,000</td>
<td>ISO-9001 and CMMI Level 3</td>
</tr>
</tbody>
</table>

To define the interview questions, we opted to keep our interview-based qualitative approach based on the War Story technique [17]. War Story questionnaires usually have warm-up, past experience, lessons learned, and reaction questions. We used a few of each type in our interview questionnaire (see Appendix A and [21] for more details). Using the conceptual framework (Fig. 2) as a guide, we were able to capture and classify participants’ beliefs, related impacts and describe their attitude toward practices. We uncovered the living experiences affected by the organizational culture and subjective norms. We also identified evidence of the real impacts of participants’ beliefs on team practice and the influence of team confidence and level of autonomy on the adoption of new practices in software teams.
The data analysis was performed in a qualitative manner to cross reference beliefs and their impacts on team's behavior and practices. The transcription and coding were performed manually and validated with other researchers before analysis. All transcripts of the interviews, were categorized, tabulated, and also analyzed via cycles of pattern coding. The transcription of the nine interviews produced 44 pages in a text document and an average of seven codes per transcription. In total, we have built almost 50 patterns of code, focusing on the relevant actions, interactions and events in the past and current projects that might exert influence on the teams' behavior and practice.

Table 2. Participants Profile.

<table>
<thead>
<tr>
<th>Company</th>
<th>Project</th>
<th>Roles</th>
<th>Experience</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>STF</td>
<td>Development Center Manager</td>
<td>&gt; 10 years</td>
</tr>
<tr>
<td>1</td>
<td>STF</td>
<td>Coding Leader</td>
<td>&gt; 03 years</td>
</tr>
<tr>
<td>1</td>
<td>STF</td>
<td>Technical Leader</td>
<td>&gt; 03 years</td>
</tr>
<tr>
<td>2</td>
<td>FD</td>
<td>Scrum Master</td>
<td>&gt; 03 years</td>
</tr>
<tr>
<td>2</td>
<td>FD</td>
<td>Developer</td>
<td>&gt; 02 years</td>
</tr>
<tr>
<td>2</td>
<td>FD</td>
<td>Developer</td>
<td>&gt; 01 year</td>
</tr>
<tr>
<td>3</td>
<td>SIG</td>
<td>Project Manager</td>
<td>&gt; 03 years</td>
</tr>
<tr>
<td>3</td>
<td>DO</td>
<td>Quality Manager</td>
<td>&gt; 01 year</td>
</tr>
</tbody>
</table>

After reducing the data to a limited number of belief classes, we characterized them in terms of frequency, sources, context and associated impact. We have identified a total of 10 beliefs to 32 related impacts. Each impact was grouped by similarity and type (negative or positive). We also recorded information about the profile of each participant of the study (see Table 2). This information served as a context to better understand the points of view of each participant connected to the beliefs found and also to uncover the aspects of organizational culture and subjective norms as well as the influence of team confidence and autonomy.

4 Results

In Table 3, we list the beliefs most frequently mentioned by the participants; their respective class of beliefs, according to topics related to the software team practice; their attitude in past or current experiences; the influence of the organizational culture; the evidence of the perceived behavioral control; the impact occurred of each belief; and finally, their impact type, positive (+) or negative (-). The interviewees seem to have a common concern and interest in new approaches for project management. Another point is related to the Knowledge Management Belief. It is a common belief for three of nine participants. The impacts of these beliefs on SE practices are positive, contributing to improve the implementation of key software team practices.
Table 3. Salient Beliefs.

<table>
<thead>
<tr>
<th>Belief Class</th>
<th>Belief</th>
<th>Attitude</th>
<th>Organizational Culture</th>
<th>Perceived Behavioral Control</th>
<th>Impact</th>
<th>Impact Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>Process</td>
<td>Project management using SCRUM Methodology.</td>
<td>Toward agile software process</td>
<td>IT market and competition</td>
<td>High confidence</td>
<td>better productivity better quality high team-level effectiveness</td>
<td>+</td>
</tr>
<tr>
<td>Task Estimation</td>
<td>Project management using SE metrics supported by tools.</td>
<td>Toward precise task estimation practice</td>
<td>Motivated by fixed and early deadlines</td>
<td>High autonomy</td>
<td>better response time better project monitoring</td>
<td>+</td>
</tr>
<tr>
<td>Project Management</td>
<td>Bad project management increases the chance of failure in software projects.</td>
<td>Toward task delivery</td>
<td>High deadline pressure</td>
<td>Low autonomy Low confidence</td>
<td>bad quality high rework bad scope and cost management</td>
<td>-</td>
</tr>
<tr>
<td>Knowledge Management</td>
<td>Knowledge sharing practice through software documentation and planning increases the chance of success in software projects.</td>
<td>Toward project information sharing</td>
<td>CMMI certification program</td>
<td>High confidence</td>
<td>better quality better productivity better response time</td>
<td>-</td>
</tr>
</tbody>
</table>

4.1 Case 1

RQ1. In case 1, the company is motivated by the IT market demands to adopt an agile methodology and the technical leaders are declaring explicit empathy and claim to be friendly to this kind of methodology. At the beginning of the project, the STF project team showed some resistance to this adoption, but during the project, the team accepted the new practices and started to work very well as a team. At first, they seemed to be afraid of taking a bigger responsibility as a self-management team, since this approach presumes that the team have significant authority and responsibility for many aspects of their work, such as planning, scheduling, delegating, and making decisions.

Also, the company's development center manager has an attitude toward the use of an individual productivity rate as a metric for project monitoring. To him, a more precise task estimation has improved team autonomy in his projects, which led to better negotiations with customers and, consequently, higher team-level effectiveness.

Regarding the requirement traceability control, for Company 1 it is mandatory to use a bad format and an apparently useless requirements traceability artifact. The technical leaders believe that this practice reduces configuration management and change management effectiveness. However, it is part of the CMMI certification program, so it is required for every project in the company.

RQ2. Our study highlights the influence factors associated with organizational culture and subjective norms to predict behavior. The values and beliefs of a project team, when in agreement with the organizational culture, reinforce the strength of their belief that the behavior will lead to positive consequences, which will exert influence on the team's behavior intention.

In company 1, the CMMI certification culture influences the adoption of software engineering metrics for project monitoring. The company's development center manager believes that a better task estimation using individual productivity rate can lead to project success.
The STF project team believes that the adoption of CMMI practices increases the company's competitive edge and can stimulate company's evolution process bringing more profitable and demanding projects. Nevertheless, some members of the project team reported that the CMMI organizational culture seems to frame software practices to a more traditional development process in a cascade model, which is not so compatible with agile methodologies.

**RQ3.** The STF project team seems to believe that the compliance with CMMI practices increases the chance of success in software projects. The knowledge sharing through software documentation and planning, driven by the CMMI certification program, also increases the chance of success in software projects. For this team, a defined development process and the existence of adequate planning can contribute to project goals achievement. So, the team confidence level related to this subject is substantially high.

The same applies to the adoption of agile methodologies for project management. The team has a high level of confidence and also autonomy to use the SCRUM\(^1\) model and its practices to conduct the STF project.

### 4.2 Case 2

**RQ1.** Our findings indicate that the attitude toward an object is based on how favorable the total set is, because one considers each belief about that object and its evaluation according to the project context. For example, most members of the FD project team agree that achieving knowledge sharing through project tracking blog is a good practice. For them, this practice improves team productivity and integration, which, in turn, leads to team-level effectiveness.

In addition, the Scrum Master of the FD project is motivated to use an agile methodology to speed up software process and delivery the software product. He believes that a good task estimation practice should be supported by appropriate tools, even though some team members are still resistant to this practice.

**RQ2.** The evidence suggests that a good fit between the organizational culture and subjective norms embedded in the software development context leads to an easier assimilation of a new practice or behavior.

Company 2 seems to be motivated by competition to adopt an agile methodology. Now it is part of its organizational culture and exerts huge influence on the behavior of the employees and how they act toward an agile practice.

---

\(^1\) SCRUM approach. [http://www.scrum.org/](http://www.scrum.org/)
RQ3. The Scrum Master of the FD project is quite confident that the adoption of a task estimation practice supported by tools can help in the implementation of agile software methodology. He believes that it is a big challenge to predict team productivity without a systematic process for scheduling and task estimation. Some team members declare explicit resistance to this practice, but they perform it anyway, which denotes a low level of team autonomy.

4.3 Case 3

RQ1. In Company 3, the project managers overcame the challenge of dealing with geographically distributed teams improving team communication practices, not only adopting new communication tools, but mainly using past experiences and lessons learned to improve the communication process.

Also, the project managers are motivated to use individual productivity control as part of a task estimation practice for a new change management process during project monitoring. They care about proposing new work processes which can become a good practice for the whole company.

RQ2. Evidence also shows that there is a significant connection between a team belief system and organizational culture. According to the participants, the company's senior managers foster an attitude toward the ISO-9001 processes and procedures because of the company's certification program. With respect to CMMI certification program, it seems to fit very well in the context of a development center operational model, but not so well in an agile project scenario.

RQ3. In the middle of the CMMI certification program, the project teams of Company 3 demonstrated low autonomy to adapt software practices to their needs. The evidence shows that there are cases of mandatory production of complex, time consuming and useless reports, only to be in compliance to the CMMI model.

In addition, the project managers are not motivated to build a suitable project plan because, in most cases, they don't have full autonomy to conduct the project in compliance with the necessary practices for their project context.

5 Discussion

In this section, we summarize our results and answer the research questions. We discuss our findings in contrast to related work and present the limitations of this study.

Considering research question 1, the study indicates that the team members are concerned about the lack of productivity metric for project monitoring, which is influencing their attitude toward task estimation practices in some ways. The team members' beliefs appear to come from a personal hands-on experience that did not work well on previous projects. The related experiences were described in specific and directed stories reported by the participants during the interviews. So, it was possible to point out that all the participating companies are involved with agile
methodologies and introducing new agile practices in software projects routine. When applying the TPB model, we analyzed the relevant actions, interactions and events in the past of the team members, trying to uncover common or conflicting beliefs between them, which could benefit or hinder the adoption of new software practices.

With respect to question 2, the evidence suggested that a good fit between the CMMI culture, embedded in the software project context, that already involves task estimation practices, leads to an easier acceptance of this practice. We confirmed that there is a significant connection between organizational culture and subjective norms around the project team and its behavior intention as the TPB model indicates. Evidence to support this was found in the statements declared by the participants about how much organizational support and culture are essential to achieve team effectiveness. Our findings also indicated that when the values and beliefs of a project team are in agreement with the organizational culture they will exert greater influence on the team's behavior intention, because the organizational culture can reinforce the strength of these beliefs and the confidence that the particular behavior will lead to positive consequences.

The main findings that relate to research question 3 are associated with the low autonomy of the project teams of all three companies to adapt software practices to their project requirements and contexts. In some cases, it is because of the CMMI certification demands and, in others cases, because of unviable deadlines set by customers. This type of situation breaks the project team confidence and reduces its effectiveness. In accordance with TPB theory, the perceived power of influence factors that may inhibit behavior performance will cause the project teams to go against their initial intention. Thus, TPB proved to fit better then TRA for the cases in which the behavior is not entirely under volitional control.

5.1 Limitations

To our knowledge, this study is one of the first initiatives of applying behavioral theories in the context of SE practices to guide research in software organizations. We should say that we do not have a complete list of implications and answers for the research questions, thus, further studies should be performed to point to other possibilities of applying behavioral theories in SE contexts.

Construct validity is concerned with design of a study and whether the studied scenario represent the real world. In our study we tried to generalize the findings from empirical statements to theoretical statements. It involved generalizing data from interviews and perceptions by discussing them in accordance with the behavioral literature. In this respect, we related our findings to relevant examples of application of TRA and TPB and compared them with related theories used in information systems research.

Another possible limitation is that we were working with the findings of software projects within only three participating organizations. It may not be possible to generalize the results in this context. However, the participants were professionals of three completely different companies using typical development technologies in a typical working environment, e.g., the natural setting demanded by case study approach. This makes the results easier to be validated.
We had to do a trade-off between the number of participants, the duration and the cost of this study. We understand that nine interview subjects is not the ideal number of participants for the interview approach, but we had to balance that with our need for a case study of a software team's belief systems. However, it seemed to be enough to show that it is possible and interesting to capture and represent a belief system in a software project context. Our intention is to increase the relevance of the obtained results to software industry, and contribute to an improved understanding on how to apply this kind of theory to study software practices.

Lastly, there is also a risk that our findings could be influenced by factors that escaped our attention. To mitigate that, we chose to discuss and validate findings with other researchers to seek the completeness of the conclusions.

6 Conclusion and Further Work

This paper has presented a study that characterizes a belief system applying the Theory of Planned Behavior (TPB) in software project teams with professionals from three different companies. Using a conceptual framework as a guide, we captured participants' beliefs and described their main attributes and the impacts on software development practices.

Scientific research has tried to explain attitude and behavior and TPB has received considerable attention within fields related to social behavior. It was most often tested in environments where individuals have a low perceived behavior control. TPB fits well in the context of software development practices and serves as a good theoretical framework in mapping software project teams' behavior.

In order to contribute to an improved understanding on how to apply a behavioral theory to study software practices, we cast light on relevant knowledge and experience on the characterization of belief systems and their impact on software industry practices. Overall, we have the aim of identifying and mapping organizational and team factors of influence on the adoption of new practices through an approach that has proven to lead to practical and useful recommendations for companies.

Our next step is to conduct new case studies with a focus on organizational culture factor. Through the synthesis of all the evidence, we intend to contribute to provide rich narrative accounts for this type of SE research activity, and elucidate more questions and issues that arise from the practice of software development.

Acknowledgment

This work was partially supported by the scholarship from the CAPES Foundation, process number 5744-11-3 and by the Fraunhofer Institute for Experimental Software Engineering. The authors are grateful to all involved in this study, specially the interviewees for their insights and cooperation and to the SOLUTIS organization for supporting this work.
References


Appendix A: Interview Questionnaire

Warm-up questions:
1. In your opinion, what is the main challenge of your project? (productivity, quality, deadline, cost or other)

Past experience questions:
2. Could you cite a past experience where the project was conducted without planning, defined schedule, without following a software development methodology and without risk analysis, and how this impacted (positively or negatively) the project results?

Lessons learned questions:
3. Is there any practice in your current project that is new to you? Do you think this new practice is beneficial to your project? Please, explain in what sense. And after using this new practice for a while, have you changed your opinion regarding its usefulness and importance?

4. Is there any practice in your current project (or past projects) that you adopted just because it was required to use it in the organization that you work, but you don't see its usefulness? Why do you not believe in this practice?

5. Is there any practice that you have introduced to your current project and that was not used in the organization in which you work? why did you think it was important to introduce this practice? And was there any problem with acceptance? What are the results of this practice in your current project?

Reaction questions:
6. Is there any new methodology that the organization in which you work is adopting? Do you know the reason for that change? Do you believe in this new methodology? Why? In which aspects is this company's new methodology affecting your current project?