

Building a Theory of Job Rotation in Software Engineering from an Instrumental Case Study

Ronnie E. S. Santos, Fabio Q. B. da Silva,
Cleyton V. C. de Magalhães

Centro de Informática
Universidade Federal de Pernambuco
Recife, PE, Brazil

ress@cin.ufpe.br, fabio@cin.ufpe.br, cvcn@cin.ufpe.br

Cleviton V. F. Monteiro

Departamento de Estatística e Informática
Universidade Federal Rural de Pernambuco
Recife, PE, Brazil

cleviton@gmail.com

ABSTRACT

Job Rotation is an organizational practice in which individuals are frequently moved from a job (or project) to another in the same organization. Studies in other areas have found that this practice has both negative and positive effects on individuals' work. However, there are only few studies addressing this issue in software engineering so far. The goal of our study is to investigate the effects of job rotation on work related factors in software engineering by performing a qualitative case study on a large software organization that uses job rotation as an organizational practice. We interviewed senior managers, project managers, and software engineers that had experienced this practice. Altogether, 48 participants were involved in all phases of this research. Collected data was analyzed using qualitative coding techniques and the results were checked and validated with participants through member checking. Our findings suggest that it is necessary to find balance between the positive effects on work variety and learning opportunities, and negative effects on cognitive workload and performance. Further, the lack of feedback resulting from constant movement among projects and teams may have a negative impact on performance feedback. We conclude that job rotation is an important organizational practice with important positive results. However, managers must be aware of potential negative effects and deploy tactics to balance them. We discuss such tactics in this article.

CCS Concepts

• Software and its engineering~Software development process management

Keywords

job rotation; software teams; performance; software engineering; case study.

1. INTRODUCTION

For several decades, the research on human resource management has investigated practices and approaches to improve performance of individuals at work [1]. These studies focus on individuals rather than teams or the organization as a whole, and regard the experience of workers and the individual work practices as one of the main elements to increase performance and the interest and attractiveness of the job.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than the author(s) must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from Permissions@acm.org.

ICSE '16, May 14 - 22, 2016, Austin, TX, USA

Copyright is held by the owner/author(s). Publication rights licensed to ACM.

ACM 978-1-4503-3900-1/16/05...\$15.00

DOI: <http://dx.doi.org/10.1145/2884781.2884837>

In this context, job rotation arises as one of these practices to be applied at the organizational environment to increase task variety, and to reduce the monotony, boredom and fatigue, resultant from job simplification, specialization, and repetition [2]. Woods defines job rotation as "the systematic movement of employees from job to job, or project to project, within an organization, as a way to achieve various different human resources objectives" [3].

Nowadays, the study of human factors have also become of great interest in software engineering research because software development depends on human-centered activities, in which individual characteristics and behavior have direct impact on the effectiveness of individuals and teams at work [4]. In this study, we investigated the practice of job rotation in a software organization guided by the following research question: *RQ: How does the practice of job rotation affect software engineers at work?*

We are interested in project-to-project rotations, as described by Woods [3], in which individuals are moved during the development of a software project, to work in another project, either in the same or in different roles. To find answers to our research question, we conducted a qualitative case study in a large software company in Brazil, in which senior managers and software engineers were interviewed with the objective of collecting data about their experience with the practice of job rotation in this company. This study provided empirical evidence about the impacts of job rotation on the work of software engineers and let to the proposition of a theory of job rotation in software engineering.

The remaining of this article is organized as follows. In Section 2, we present the background and related work. In Section 3, we describe the research method and the case study design. Section 4 presents the results and Section 5 discusses implications of these results together with the suggestions of future work. Finally, conclusions are presented in Section 6.

2. BACKGROUND

This section presents the background about job rotation that supports this study, as well as related works.

2.1 Definition of Job Rotation

The literature presents many definitions to describe the practice of job rotation, in which individuals are frequently moving from projects, departments, and jobs within the organization. In particular, Woods [3] broadly defines job rotation as "the systematic movement of employees from job to job or project to project, within an organization, as a way to achieve various different human resources objectives". We use this definition to guide our study because, in software development, this practice concerns more often the rotation of members among different software projects within the organization, and less often moving individuals among different jobs.

Therefore, we are interested in investigating the impact of project-to-project rotation in the work of the individuals at the software development environment. This concern arises from the analyses of the research literature on job rotation in several fields, in which different and, sometimes, conflicting results about the impacts of this practice have been observed, as discussed below.

2.2 Job Rotation in other Areas: Conflicting Evidence of its Effects

We analyzed the literature describing empirical studies about the effects of job rotation in other areas. This analysis revealed that the potential benefits might be overshadowed by negative effects, with an overall result that is complex to predict.

For example, Richardson et al. [7] identified a set of benefits obtained by the use of job rotation, e.g., improvement of knowledge and skills, promotion and development of opportunities, networking, ability to recruit and retain staff, improvement of interdepartmental relationships, and increase of motivation. On the other hand, Campion, Cheraskin, and Stevens [8] identified potential costs and negative impacts, such as the increase of workload and decrease of productivity, the increase of costs with learning process and the decreased of satisfaction and motivation, resulting in job burnout.

Several other studies in different areas also contributed to the understanding that the effects of job rotation on motivation, satisfaction, job burnout, performance, and many other factors are complex, systemic, and, often, conflicting [5][6][9][10]. This complexity has motivated and justified the development of empirical studies in several areas. However, despite the use of job rotation by several software companies to address diverse organizational and business needs, there is no consistent and strong body of evidence about the benefits and challenges of this practice in software engineering research, as discussed below.

2.3 Job Rotation in Software Engineering Research: Low Strength of Evidence

To understand the landscape of the research about job rotation in software engineering, we performed a systematic literature review covering studies published between 1997 and 2014. We performed automatic and manual searches, resulting in over 4,000 articles from which 18 relevant ones were selected. We analyzed these 18 articles, looking for empirical evidence about the use and effects of job rotation. Only one study addressed job rotation as its primary goal. We briefly discuss this study and then summarize the findings of the mapping study below.

Faegri, Dyba, and Dingsøyr [11] investigated the benefits and challenges to improve knowledge redundancy among developers using job-to-job rotation. This is the only study in the literature that specifically addressed job rotation in software engineering context (Faegri [12] published a summarized version of the same study). Different from project-to-project rotation, they investigated a practice in which software developers were rotated to different departments in the organization to perform a different job (other than software development, such as customer care). This study emphasized the lack of empirical evidence about the introduction and adoption of job rotation in software practice. They pointed out that job rotation could improve knowledge redundancy, support innovation based on integration of different knowledge domains, and improve organizational learning. However, the benefits obtained were not sufficient to justify the practice of rotation of developers, in this case, because professionals were not favorable

about being rotated to perform different tasks other than software development.

The remaining 16 studies (listed in Section 10) did not address job rotation as part of their initial goal. They presented conclusion about job rotation that were found during data analysis of the research, as secondary findings or after-thoughts. Therefore, the strength of the evidence presented is low because the investigation of job rotation was not planned in the design of the studies. Nevertheless, the findings are relevant because they provide pointer for future investigations. According to these studies, the use of job rotation in software organizations is frequently associated to the spread of knowledge and information, both related to the work itself and also about several aspects of the business (organizational understanding). Job rotations also contribute to create connections among the individuals and to improve the existing networking in the organization. Consequently, job rotation allowed individuals to build professional and social interactions with people from different teams, departments, or branches of the company. These interactions might be useful to support the establishment of multicultural teams, when needed. Job rotation is also a useful practice to improve team flexibility, helping individuals to be less vulnerable to changes and also preparing them to deal with a vast diversity of issues in software development. Further, the practice is often thought as a way to promote task variety, which is an important motivational factor [21].

On the negative side, about a third of papers (6/18) have also reported limitations and disadvantages of job rotation. According to these studies, this practice can be difficult to plan and its effects can negatively impact, directly or indirectly, the performance of individuals and teams. Further, rotations often require time for proper implementation or until the desired results are achieved.

2.4 The Motivations and Contributions of Our Research

The study of job rotation in software engineering is motivated by the nature of the software development tasks that have more task variety and task identity than traditional jobs. Therefore, classic job design strategies may not work as in other industries [5]. Further, project-to-project rotations among software development projects may increase job specialization and tap the benefits of modern software development task design characteristics. However, as far as we aware of, no other study addressed this issue in software engineering research.

This study extends and improves the current knowledge about job rotation in software engineering in two ways. First, this is the only study so far that was specifically designed to study project-to-project rotation in an industrial setting. Second, we seek for a deeper understanding of the complex and systemic relationships between job rotation and work related factors, such as performance. For this, we developed an explanatory theory that can help researchers and practitioners to better understand benefits and challenges of job rotation in software engineering practice.

3. METHOD

In this research, we are interested in understanding how individual software engineers interpret their experiences with job rotation in the workplace and how these interpretations shape their work due to the influences of this practice. Consistently with the nature of our research question and investigated phenomenon, we followed the method proposed by Eisenhardt [13] to build theories from case study research. The following sections describe the method.

3.1 Getting Started

We started with the definition of the research question and the case study design. We performed a broad literature review searching for: (i) definitions of job rotation to provide a better grounding of this construct (Section 2.1); (ii) studies regarding the impacts of job rotation in different research areas (Section 2.2); (iii) studies investigating job rotation in the context of software engineering (Section 2.3). This literature review was used to precisely define job rotation and the initial research question, to identify potentially important variables to be observed in the field, and to increase construct and external validity during data analysis and synthesis. We did not use any priori theory and, consequently, no pre-defined hypotheses to retain theoretical flexibility, as recommended by Eisenhardt [13].

3.2 Selecting the Case

Second step was the purposeful sampling of the case to be investigated. We selected a well-established, mature software company, in which:

- the job rotation practice was performed consistently with our chosen definition;
- job rotation was systematically applied as a managerial practice throughout the entire organization (all projects and teams);
- the application of this practice was known to all current employees and also to potentially new employees during recruiting and selection;
- there was a large number of projects running at the same time (over 50, in this case), of varying size, scope, duration, team size, etc.;
- and we could have full access to all data and individuals necessary for our investigation.

With these requirements, we selected a large software organization located in Recife, Brazil, in which members of software teams were frequently rotated amongst software projects, during the development process, depending on business needs. We provide a description of this company in Section 4.1.

3.3 Crafting Instruments

As recommended in the literature [13][14], we used multiple data collection methods: interviews, document analysis, and questionnaires. We did not perform observations because: the perceptions and feelings we were investigating are difficult (even impossible) to observe; it was difficult to pinpoint when and what to observe; thus, we would need access to places and events in the company that we did not have due to confidentiality issues.

Semi-structured interviews were performed with two groups of participants (using different interview scripts): a) the senior managers of the company, to collect data about the organizational context, and to characterize job rotation; b) project managers and software team members, to obtain information about their experience with job rotations. Both interview scripts followed the six types of questions described by Merriam [14] (*apud* Patton [16])[15]. See Technical Report for the scripts available on: <https://goo.gl/GAeg6a>.

We validated the interview scripts by conducting pilot interviews with a group of five professionals of different companies, who had prior experience with between-projects rotation. We made minor adjustments to the phrasing of some questions and also timed the pilot interviews to have an estimate of the duration of the actual interviews. This was important to give the participants an estimate

on the time they would spend while participating in the research. We estimated 40 minutes for the interviews, which was consistent with duration in the field. We applied questionnaires during member-checking of results, to verify and validate our findings (Section 3.7).

3.4 Entering the Field

Field research was performed in three major phases, with corresponding sub-phases as describe below. We first selected the target projects and the individual participants within each project. We then interviewed the senior managers and members of the project teams. Finally, we accessed documents to complement and triangulate data collect directly from the participants.

3.4.1 Project and Participant Selection

Consistently with our qualitative research approach, we purposively sampled two projects in the organization portfolio looking for maximum variation of information (Section 4.1.2). We selected one project with a great incidence of rotations and other one with a more stable team. This diversity allowed the understanding of distinct perceptions about the benefits and challenges of job rotation.

We then sampled participants from each of the two projects (Section 4.1.3). We selected different types of professionals, in different roles (developers, testers, team leaders, and project managers) to achieve a variation. We also chose variety of gender, time within the organization, education level, and age. Further, we selected individuals with different perspectives about the job rotations: participants who had been rotated at least once and individuals who had never been rotated, but experienced the indirect effect of this practice on the team.

3.4.2 Interviews

All interviews occurred in the organization's facilities and were performed by an interviewer and supported by a second researcher (that took notes to support the process of data analysis). All interviews were recorded. The interviews with senior managers lasted about 60 minutes each. We analyzed the results of these interviews to characterize the practice of job rotation as precisely as possible. These results were used to frame the data collection with software team members. The interviews with project managers and team members lasted an average of 40 minutes each. Altogether, they produced 9 hours of audio and over 190 pages of transcriptions.

3.4.3 Document Collection

We collected documents related to: the values and principles of human resource management in the company, managerial practices of the organization, the characteristics of the projects in which the participants of this study were allocated, and data about the personal and individual profile of participants, including performance evaluation information. These documents were provided by the human resources department of the company and used to triangulate the data collect directly from the participants in the interviews.

3.5 Data Analysis

The objective of qualitative analysis is to consolidate, reduce, and interpret data obtained from various sources, and make sense of them [14]. It involves labeling and coding all data in order to identify similarities and differences to describe the phenomenon under study. Data analysis was performed in parallel with data collection, in incremental and iterative steps, as recommended by Merriam [14] and Seaman [15].

We used coding techniques [15] to code, categorize, and synthesize data, towards the construction of a central story that explains the effects of job rotation in this organization. Initially, all audio from the interviews was verbatim transcribed. We used QSR NVivo 10 to support the data analysis and synthesis.

Data analysis began with open coding of the transcripts. Post-formed codes were constructed as the coding progressed and were attached to particular pieces of the text (Figure 1). Then, the codes arising from each interview were constantly compared to codes in the same interview and from other interviews. From the constant comparisons of the codes, we grouped them into categories that represent factors affected by job rotation (Figure 2). As the process of data analysis progressed, relationships among categories were built (Figure 3).

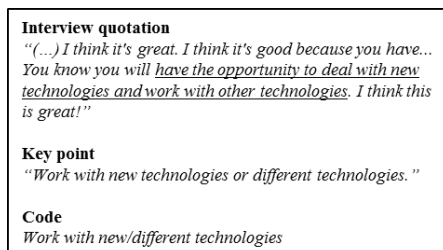


Figure 1. Open Coding: Building Codes

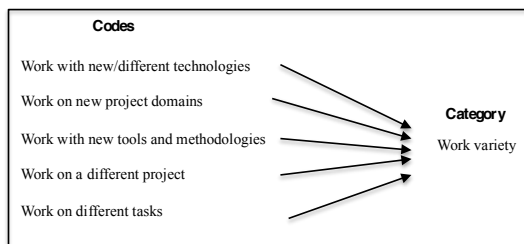


Figure 2. Open Coding: Building Categories

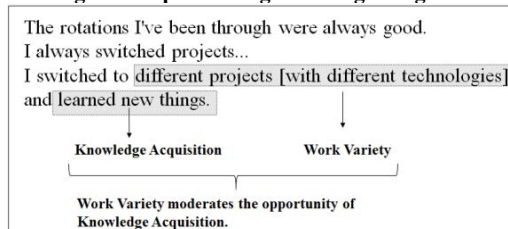


Figure 3. Axial coding: Building Relationships

Finally, core categories were chosen according to their general explanatory power, propositions emerged, and a narrative was created to describe the central story of the case. From this narrative, we defined two central hypotheses towards the definition of a theory of job rotation.

3.6 Enfolding Literature

Following the guidelines of Eisenhardt [13], after completing the data analysis and shaping initial hypothesis, we looked at the literature to sharpen construct definitions and generalizability, and raise theoretical level. We contrasted and compared our results with the findings from the literature review from other areas and our mapping study, discussed in Section 2.3.

3.7 Reaching Closure

Throughout the iterations of data collection and analysis, we checked for theoretical saturation [13]. We stopped data collection

when the results from new interviews and document analysis were not adding significantly new information. At this point, we consolidate the results and used member checking techniques to validate these findings, improving accuracy, credibility, and internal validity of our interpretations [18][19]. We performed a 15 minutes session with each participant following a simple protocol: (1) we presented the results of the study; (2) we applied a questionnaire to evaluate the level of agreement of the participants with our interpretation of the data. Below, we present a sample of the questionnaire:

Questionnaire 1 – Item 1: Job rotation allows software engineers to work with different projects and technologies: () I completely agree () I partially agree () I partially disagree () I completely disagree.

Most participants completely agreed with our interpretations and those that partially agreed were probed for suggestions for improvement in each specific item. These improvements were analyzed with other participants and added to the results. We mention these improvements when presenting the results. No participant disagreed with the findings.

As the final activity to verify consistency (raising internal validity) and theoretical saturation, we presented the consolidated results to a group of 32 project managers and 2 senior managers of the organization, who validated the results and did not add new information to our findings. At this point, we concluded that our results provided an accurate interpretation of the investigated phenomenon in this organization and finished the case study.

3.8 Ethics

We followed the norms of Resolution 466/12 – CNS-MS of the Brazilian National Health Council that regulates research with human subjects. The company signed a Term of Authorization and the researchers signed a Non-disclosure Agreement (covering access to sensitive information). Both documents granted the researchers access to facilities, to the participants, and to necessary documentation. They also authorized the participants to use work hours for the interviews. We believe that this formalization reduced the possibility of participants concealing information that they would consider sensitive. Before the interviews, each participant signed an Informed Consent Form that explained the overall objective and relevance of the research, guaranteed data confidentiality, the anonymity of the participation, the non-obligatory nature of the participation, and the right to withdraw from the research at any moment. All invited individuals freely agreed to participate and no participant withdrew from the research.

4. RESULTS

We start with a description of the company, the projects, and participants of the interviews. We then show the characterization of job rotation in the company obtained from the interviews with senior managers. Finally, we describe the construction of our theory of job rotation.

4.1 Context Description

This section describes the context in which we performed the research: the software company, selected projects and participants.

4.1.1 The Software Company

The selected case is a Brazilian software company, founded in 1996, specialized in on demand system development for external clients. By the time of this case study, the company had just over 500 employees of which about 70% worked directly on software development teams. The company executed an average of 50 projects concurrently in various business areas, such as finance,

telecommunication, government, manufacturing industry, services, and utilities. The organization had a typical hierarchical organizational structure, with a CEO controlling six major departments: Administration & Finance, Human Resources, Business & Marketing, IT Infrastructure, Advanced Engineering and Innovation. Advanced Engineering was responsible for all software development projects. It also had a typical structure headed by a Chief Operation Officer (COO) and supported by a Project Management Officer (PMO). Both COO and PMO were in charge of resource allocation for all projects and also job rotation, with support from the Human Resources department.

4.1.2 Selected Projects

Project A – this team was developing a web-based system for a multinational logistics company. At the time of data collection, the project was running for 2.5 years and the team was working on the development of new features, integrating the system with other systems of the client, fixing bugs, and working on the improvement of system's performance and scalability. This team was composed of 13 professionals. One project manager, four software engineers, one test engineer, and one technical team leader were interviewed in this research. Project A used an Agile process based on SCRUM.

Project B – this team was developing a 3D visualizer for a multinational printer company. This was a research project with the objective of creating and implementing new products using the technologies developed by the client's research lab. Because of this particularity, there was no long-term roadmap; product backlog and user stories were incrementally defined as new ideas, emerging from the research lab and the team itself. One project manager, four software engineers, one test engineer, and a technical team leader were interviewed in this research. Project B also used an Agile process based on SCRUM.

4.1.3 The Participants

As discussed in Section 3.4.1, we selected participants from each project to achieve large variation in data collection. We only present aggregated profile of participants in Table 1 to comply with ethical norms regarding anonymity of the participation.

Table 1. Profile of Participants

Projects	PA	PB
Team Roles	4: Software engineers 1: Test engineer 1: Team Leader 1: Project Manager	4: Software engineer 1: Test engineer 1: Team Leader 1: Project Manager
Genders	6: Males 1: Female	6: Males 1: Female
Ages	3: < 25 3: 26 – 35 1: > 36	1: < 25 4: 26 – 35 2: > 36
Education	3: Undergraduates 1: Technician 2: B.Sc. 1: M.Sc.	1: Undergraduates 3: B.Sc. 3: M.Sc.
Time in the job	1: < 3 years 4: 3 – 5 years 2: > 6 years	2: < 3 years 3: 3 – 5 years 2: > 6 years
Experience with rotation	3: Never rotated 2: Rotated 1 or 2 times 2: Rotated > 3 times	1: Never rotated 4: Rotated 1 or 2 times 2: Rotated > 3 times

4.2 Characterizing Job Rotation

We coded the interviews with senior managers and synthesized the characterization of job rotation in the organization. In summary, job rotation was an explicit organizational practice (Definitions and Triggers); involved a large percentage of the work force (Frequency

of Rotations); and was clearly associated with tactical and strategic goals of the company (Motivations). We therefore concluded that the use of job rotation in this company was consistent with our conceptual definition. Figure 4 illustrates this characterization. During member checking sessions, all participants validated this characterization.

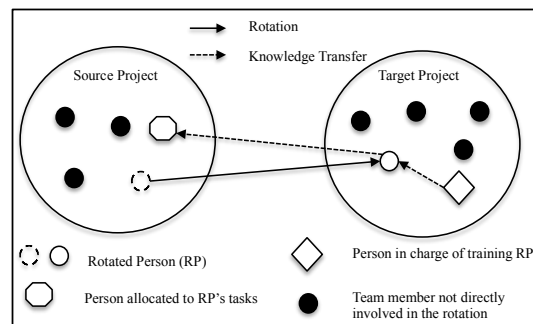


Figure 4. Job Rotation

Definition – job rotation (called “movement” by the interviewees) was defined as the practice of moving one professional from one software project (which we named “source project”) to another project (named “target project”), within the organization. In most situations, the role (software engineer, test engineer, team leader, software architect, etc.) performed by the rotated person (RP) would remain the same, but the role could change under various circumstances, e.g., when a test engineer changed to the role of software engineer to fulfill a resource need in the target project.

Triggers – a rotation was initiated by four reasons. First, when the source project finished (although this could be seen as a reallocation of resources, the COO and PMO treated it as a trigger). Second, target project needs: when the target project required more manpower (quantity) or different set of skills (diversity). Third, source project needs: when the performance of the rotated person was not compatible with the requirements of the source project. In this case, source project became target project. Fourth, individual request: an individual manifested the desire to change projects. Project needs (source and target) were the most frequent trigger and had priority over individual request. Further, individual requests were fulfilled only if the individual was not essential for the source project at that moment and there was a target project with suitable vacancy.

Process – no formalized process description existed for job rotation. The COO with the support of the PMO identified the trigger for a rotation and performed various actions: identify individuals in other projects with the skills as close as possible to the ones needed in the target project; check if the potential source project could release that individual; rotate the individual. The COO and PMO solved conflicts of priority between source and target projects in an *ad hoc* way.

Frequency of rotation – At the aggregated level, Human Resources department accounted an average of 130 rotations per year, over the three years prior to our study (2012-2014), which represented a rotation of nearly 40% of the entire workforce per year. Over the same period, over 65% of the workforce have been rotated at least once.

Moment of rotation – a rotation could occur at any moment during the life cycle of a project. It would depend mainly on the needs and level of priority on the target project, but also on the need to change a person that was not performing as required in the source project.

Motivations for using job rotation – the main motivation for using this practice was to fulfill the needs of projects for human resources. Nevertheless, the Human Resources Chief Manager also pointed out that rotation was important to increase the flexibility of the teams and to provide to employees the opportunity to participate of different projects, increasing the variety of tasks and the individual learning. COO and PMO did not mention this motivation in their interviews, but agreed with them during the validation of these results. Finally, rotations were motivated also to allow exchange of tacit knowledge about practices among different projects in the company.

4.3 Effects of Job Rotation

We present our findings regarding the effects of job rotation on the work of the software engineers in this company. As shown in Figure 4, a given rotation would directly affect the rotated person (RP) and, potentially, two other individuals (or groups of individuals in a more general way): the person in the source project allocated to the tasks left by the RP (AP for short) and the person in the target project in charge of training RP in her new tasks (TP for short). Interview questions asked participants to assume all three points of view, whenever possible.

4.3.1 Direct Effects Attributed to Job Rotation

Participants identified work related factors that were directly affected by job rotation according to their perceptions. All factors were identified during open coding and their precise definition evolved throughout data analysis, member checking, and literature enfolding processes. We analysed the potential effects of job rotation on each factor and how participants perceived the importance or impact of each factor on their work (Table 2). When presenting the results, we use excerpts from interviews as supporting evidence to build internal validity.

Table 2. Factors Affected by Job Rotation

Factor	Effect of job rotation on factor	Impact of factor on the work
Work Variety	+	+
Knowledge Acquisition	+	+
Knowledge Transfer	+	/
Well-defined Work	-	+
Feedback	-	+
Performance	-	+

Job rotation provided software engineers the opportunity to work with a wide diversity of projects and technologies. Therefore, job rotation helps to create a working environment that is rich in **Work Variety**, which is positively valued by software engineers:

"(...) It is very good because one is going to work with several different stuff."

"(...) to work with new technologies or different technologies. I think it is great!"

Participants also perceived the lack of variety at work as being negative, reinforcing the positive value of Work Variety:

"Man, there is no other project for me? I am doing this for a long time."

"... I can't stay in the same project for long."

Job rotation also created opportunity for the RP to acquire new knowledge (**Knowledge acquisition**), both technical and business related, increasing professional experience and skills regarding technologies, process, tools, business domains, etc.:

"(...) I think it is good because you know you will have the opportunity to learn."

Knowledge acquisition was also perceived as positive for the job, emphasizing the constantly changing nature of the technologies and practices in software engineering. During member checking, participants noted that this effect was moderated by the nature of the target project (amount of Work Variety) and also by the skills and competences of the RP (**Individual Skills**). Therefore, if the RP already had the needed skills and knowledge about business domain to work in the target project, the Knowledge Acquisition would be reduced.

As described in Section 4.2, job rotation could happen at any point in the life cycle of a project. Therefore, individuals could be transferred to a new project before completing their assignments in the source project. Two effects of this characteristic of the job rotation practice were identified.

First, participants emphasized that when they were rotated and left unfinished assignment, they still had to perform some activities in their previous project, mostly to transfer knowledge about their prior tasks. In this case, the RP spent time and effort with no new knowledge gain for herself, as can be seen from the following quotes:

"You will keep supporting the previous project. So you will have to help that person on what you were doing"

Job rotation, thus, creates the need for **Knowledge Transfer** and the RP perceived no positive effect on her work. Nevertheless, no participant emphasized this as having a negative effect. Therefore, we marked the perceived effect on the job with the "/" sign. During member checking, the participants noted that the moment in the life cycle of the source project when the rotation happened (**Moment of Rotation**) moderated the need for Knowledge Transfer: if the person was rotated after completing her part of the project or at the end of the project, less knowledge would need to be transferred or none at all. The skills of the AP that assumed the tasks left by the rotated person also moderate the need for knowledge transfer.

A second effect of rotating a person before the end of the project or the person's assignments was the perception that they were not developing a well-defined work. Participants emphasized that they liked to perform complete tasks, i.e., from start to end, before moving to another one. That is, participants positively valued **Well-defined Work**. As can be seen below, job rotation decreases the perception of well-defined work:

"I would not say it [job rotation] would be good. Because it is good when you do something and finish it".

"During the on-going project, if I'm moved, then I would be mad, because I'll get that feeling that my work remains unfinished."

"I don't like this situation, leaving something unfinished".

This effect is moderated by the moment in the development of the source project that the rotation is performed. If the individual is rotated either at the end of the project or at the end of her assignments on a still on-going project, the perception of not completing well-defined work decreased. If the RP left unfinished tasks, this perception increased.

Participants felt that job rotations could negatively impact the **Feedback** provided by managers and team leaders about their performance, from the point of view of the RP. They felt that managers and team leaders did not have enough information about their performance or capabilities at work because they were frequently moving between projects, as the quotes below emphasize:

"... There are some guys who had to wait too much time to get a good feedback because they were always switching between projects".

"... Another manager that knows nothing about you will have to start an evaluation from scratch".

Therefore, job rotation impacted the production of consistent performance feedback because managers and leaders did not have sufficient time (and perhaps resources) to obtain performance information of individuals rotated to their projects. Participants also related this impact on feedback as being potentially negative on their overall performance evaluation, although we did not find corroborating evidence from the performance evaluation data we obtained from human resources.

Finally, participants pointed out that job rotation had a potential negative impact on the **Performance** of all three individuals directly involved in the rotation (Figure 4). Therefore, job rotation had a potential negative effect on performance and, as expected, the participants understood that high performance was important for their work, as seen in the following quotes:

It is always traumatic [job rotation]. You end up having loss of performance" [point of view of the RP].

"Then you have to work more, more focused, to fill the gap of that person [who left]" [point of view of the AP].

"We will have to prepare the new person who just joined us. And it takes some time" [point of view of the TP].

Participants found that the effect job rotation on performance was direct, but we understood during data analysis that performance loss was better explained by an indirect effect due to the interactions between Knowledge Transfer and Knowledge Acquisition. We kept the performance factor here to be consistent with the results of the open coding and show these interacting effects in the next section.

4.3.2 Interacting Effects of Job Rotation

After finding and interpreting the work factors directly affected by job rotation, we interrogated our data again looking for explanations for those effects in terms of potential relationships among factors and the existence of other factors revealed by these relationships. The relationships were found during axial coding and were refined during selective coding, member checking, and the literature enfolding process, resulting in four key relationships.

Relationship 1: Work Variety and Individual Skills moderate the opportunity of Knowledge Acquisition.

The movement of software engineers to a different type of project that uses different or new technologies, new tools and methods, or is in a different business domain created opportunities for the RP to acquire new knowledge:

"The rotations I've been through were always good. I always switched projects... I switched to different projects [with different technologies] and learned new things".

The bigger the Work Variety (difference between previous and new projects) the more Knowledge Acquisition could be required, potentially representing a better learning opportunity for the RP. Therefore, participants perceived that they had better opportunities to learn when moving between projects than if they stayed working always in the same project. The skills of the RP also moderated the effect of job rotation on Knowledge Acquisition, as noted by the participants during member checking.

Relationship 2: The need for Knowledge Acquisition and for Knowledge Transfer increases Cognitive Workload.

The rotation to a different type of project required the acquisition of technical knowledge and the assimilation of new business domains. In addition, the RP could need to transfer knowledge about her former assignments to the AP in the source project. Due to time constraints of source and target projects, knowledge transfer and knowledge acquisition often happened concurrently. The need to perform several concurrent tasks in short period of time, potentially with extra work hours, increased the **Cognitive Workload** on individuals. Participants reported that this situation often lead to cognitive overload, i.e., the excessive or unbalanced mental or cognitive workload due to very distinct and, sometimes, excessive work tasks, with negative impact on their work:

"Sometimes you enter in a project that is going to be harder, that will require working extra hours or during weekends".

"Look, it is not good. At the moment I joined [the target] project, I knew just a little about the project, so I had to learn quickly".

Relationship 3: The need for Knowledge Acquisition and Knowledge Transfer negatively impacts Performance, and this impact is increased due to increased **Cognitive Workload**.

Participants identified that a drop in performance after a rotation was caused by the activities related to Knowledge Acquisition that were not immediately productive. There was also drop in performance if the RP was required to spend effort transferring knowledge to the AP in the source project (Knowledge Transfer). Both effects on performance were increased if the RP was experiencing cognitive overload due to the very diverse nature of the activities and the excessive amount of work:

During the development of my tasks [target project], I also have to learn all about this new project, and... I'd have to look after the previous project, to the tasks that I was responsible for, and now is designated for another person to finish it. You have to help that person with what you were doing [in the source project]. And this is bad.

Relationship 4: Performing Well-defined Work is important for receiving consistent individual Feedback.

Participants identified that rotating from a project before it was finished impaired the possibility of receiving consistent feedback from managers and leaders, because they left their assignments unfinished. Therefore, receiving consistent Feedback was also related with the possibility of performing a work assignment from start to end, and job rotation could impair this possibility.

There are some cases where people that I know waited too long for a good feedback, because they were always switching projects, with a different manager, and they were always restarting their tasks, they rarely finish their tasks, and so they rarely get good feedbacks.

4.3.3 The Central Story of this Case: A Theory of Job Rotation

After relationships among constructs were identified and refined during axial coding, selective coding was performed to identify the central categories that better explained the studied phenomenon. We then created a narrative that describes this explanation and forms our theory of job rotation.

The central story that explains the effects of job rotation on this case study is based on the balance among aggregated categories, that we labelled *Lasting Gains*, *Temporary Losses* and *Permanent Losses*, with respect to their potential impact on Performance and Feedback. This central story is presented below and is illustrated in Figure 5.

Lasting gains were associated with the increase of Work Variety and the opportunities for new Knowledge Acquisition that were

created by job rotations. The greater the Work Variety in a given rotation, the bigger was the opportunity for Knowledge Acquisition both in terms of technical and business aspects of the work (Relationship 1). These are considered *Lasting Gains* in the sense that the acquired knowledge was used in the target project and potentially would be useful in other situations in the individual's career. These gains were moderated by the skills that the RP already had, because if she possesses the necessary knowledge for the new assignments, less or no new knowledge would be acquired.

Job rotations could also create the need for extra activities related to Knowledge Transfer and Knowledge Acquisition that occurred concurrently with the work activities in the target project. Performance on the project activities would be temporarily impaired, until Knowledge Acquisition was finished (which depended on the learning curves of the new technologies and business domain, and the previous skills of the RP) and the need for Knowledge Transfer to the source project decreases (which also depended on learning curves, the moment in the life cycle of the source project that the rotation was performed and the skills of the AP). A rotation could also temporarily increase Cognitive Workload due to the potentially diverse types of mental efforts needed to cope with Knowledge Acquisition and Knowledge Transfer, concurrently with the work on the target project, and this workload increased the impact on Performance. These negative effects were considered *Temporary Losses* because they would eventually disappear considering that the RP stayed for enough time in the target project.

Job rotations decreased the perception of the RP about performing a Well-defined Work due to the possibility of leaving a project before the end of her assignments, thus giving the impression of an unfinished or incomplete work. This effect was moderated by the

moment when rotation happens in the life cycle of the source project. Well-defined work was found to be important for receiving consistent Feedback on Performance. The negative impact on Well-defined Work was, therefore, associated with a negative impact on Feedback. Different from the temporary losses related to decrease in performance, the impact on Well-defined Work and the consequent impact on Feedback was considered as *Permanent Losses* in the sense that, once incurred they would not be (easily) recovered.

The overall effect of job rotations on individuals, therefore, depended on balancing the effects of the *Lasting Gains*, the *Temporary Losses*, and the *Permanent Losses* on Performance and Feedback, and the three components are linked. From this central narrative, we can shape two hypotheses to finish the proposition of a Theory of Job Rotation in software engineering:

Hypothesis 1 – Increasing *Lasting Gains* increases *Temporary Losses* moderated by Individual Skills and Moment of Rotation: The larger is the difference in terms of technology and business domain between source and target projects, the larger is the opportunity to learn and also the larger is the size and duration of the negative effects on performance, moderated by the Individual Skills and the Moment of Rotation.

Hypothesis 2 – Increasing the frequency of rotations increase *Lasting Gains* and *Permanent Losses* moderated by the Moment of Rotation: The higher the frequency of job rotations the more opportunities to acquire new knowledge will be created and also the larger will be the negative impact on the perception of performing well-defined work, moderated by the Moment of Rotation.

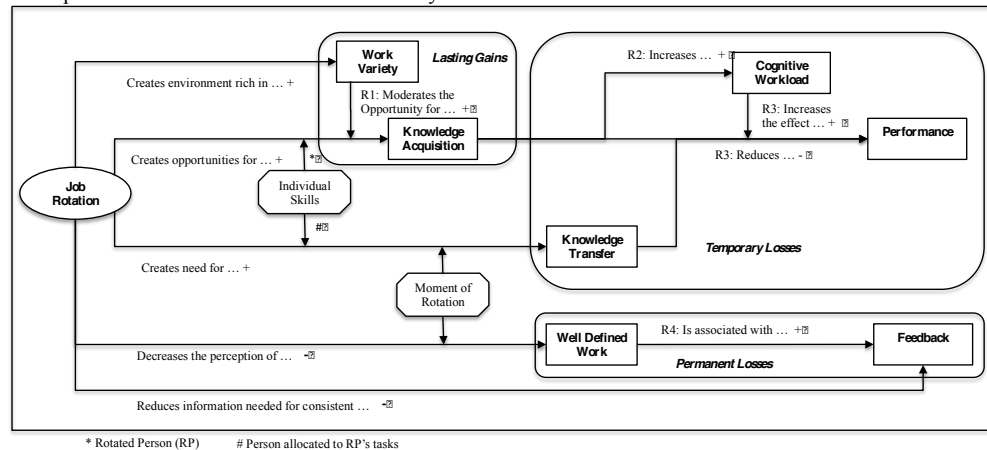


Figure 5. Central Story of Job Rotation in the Organization

5. LITERATURE ENFOLDING

According to Eisenhart [13], a central element of theory building is the comparison with evidences, theories, or propositions from existing literature. We interrogated the studies from software engineering and other fields reviewed in Section 2, looking for similarities and contradictions with our findings to enhance generalizability and to frame the scope of our results.

Similar to the observations of Ollo-Lopez, Bayo-Moriones, and Laraza-Kintana [1] in the area of Business Administration, in our study job rotation increased task variety. However, complementing their findings, we found that job rotation could decrease and impair

task identity (Well-defined Work). Our findings also showed that software engineers place great value on performing well-defined work and this is consistent with recent studies on work motivation and job satisfaction in software engineering [20][21].

We also confirmed the negative impacts observed by Campion, Cheraskin, and Stevens [8]. In our study, job rotation could increase the workload (in particular cognitive workload) and decrease productivity. We found that these negative impacts tended to be temporary and were moderated by individual skills and the moment of the rotation, as discussed above.

The benefits reported by Richardson et al. [7], such as improvement of knowledge and skills, promotion and development of opportunities and networking, were partially found in our study. We identified the improvement of knowledge and skills related to Work Variety and Knowledge Acquisition. Development of networking was one of the motivations to perform job rotation in our case study, as pointed out by senior managers, because they wished to enhance knowledge exchange among projects. However, the participants did not mention the enhancement of opportunities and networking in our study. Two explanations are possible. First, the participants were not stimulated to produce this information because it is not directly related to individual work in software development. Second, development of opportunities and networking requires other management practices, perhaps in combination with job rotation. Both explanations can be tested in future studies.

Other benefits of job rotation identified in our mapping study and literature from other areas, such as the improvement of team flexibility, ability to recruit and retain staff, and the improvement of interdepartmental relationships, were not addressed in our study, because we focused on understanding the impact on the work of the individuals. Future studies to address these issues should focus on other organizational aspects not necessarily related to software development, but that are important to enhance our understanding of job rotation. Some directions for these future works are provided in Section 6.3

6. DISCUSSIONS

In this section, we discuss limitations and validity of our results, the implications of the results for the practitioners, and directions for future research.

6.1 Addressing Limitations, Validity and Reliability

Validity and reliability assessments used in positivist experimental studies do not apply directly in interpretive qualitative research. We discuss validity and reliability of our results from the perspective proposed by Merriam [14].

Construct validity in qualitative research is related to the precise and clear-cut definition of constructs that is consistent with the meanings assigned by the research participants. We employed member checking to ensure our interpretations were consistent with those of the participants. We also compared and contrasted our construct definitions with the literature. Whenever the meaning assigned by the participants differed from the literature, we double-checked with the participants until a consistent definition was reached.

Internal validity, or *credibility*, is related to the extent that the results match reality and that the researchers were able to capture reality as close as possible. To increase credibility, we tried to achieve maximum variation on the sources of data. We collected direct data from participants with different roles in the company, in the teams, and with different perspectives, regarding rotations (already rotated and never rotated). We then contrasted and compared directly collected data with document information. We then used member checking to verify the completeness of our findings. Finally, we compared the findings with the literature on software engineering and other fields to sharpen construct definitions and increase internal validity.

Reliability refers to the extent that the results can be replicated. We do not expect that results from qualitative research to be replicable because, in short, human behavior, feelings, and perceptions

change. The more important question is whether the results follow consistently from the data and that the researchers did not make any inference that cannot be supported by the data, which Merriam refers to as *consistency* [14]. To increase consistency, we performed data analysis in pairs. Coding was performed by one researcher and reviewed by two other researchers. Member checking with participants was used also to check the consistency of our interpretations. Finally, project and senior manager of the company fully evaluated and validated the results.

Our result is a mid-range theory about job rotation, which is transferable (instead of generalizable). In this sense, although we do not expect that all our findings will be applicable to other contexts, it is possible to learn from the case description and decide to what extent the findings can be applied or transferred to other situations. Two strategies were employed to enhance *transferability* of the results. First, we tried to provide rich description of the research method, context in which the research was performed, and the results themselves, although we believe this is one of the limitations of this article, since space restriction impacts the possibility of rich and thick descriptions. Second, we sampled the participants to achieve maximum variation because this would provide richer data and, consequently, a richer theory more widely applicable.

Finally, a common challenge and potential limitation in qualitative studies is to reach *theoretical saturation*. The quality, rather than the size, of the sample of participants is essential to increase our confidence on theoretical saturation. In this study, we interviewed 14 participants with different points of view and perceptions about the studied phenomenon. We checked for theoretical saturation during data analysis and stopped sampling further participants when the increments of new information were very small. We member checked the results with interviewees and a larger group of 34 project managers. We then compared and contrasted our findings with the literature also to check if we were not overlooking important facts. We believe that these procedures produced a consistent and comprehensive account of the phenomenon.

6.2 Implications for practice

In software projects, the quantitative (amount) and qualitative (roles and skills) requirements for manpower change very dynamically throughout the life cycle of a project. Our results confirmed that job rotation is a suitable task to address the dynamic human resource needs of software projects. However, rotations can have negative effects over several factors of the work of the individuals. We believe our theory can be instrumental to provide awareness about the potential conflicting outcomes that can result from using job rotation.

One key factor that must be evaluated when performing a rotation is the amount of work variety that will be involved. Little variety (moving a person to the same role in a project with similar technical requirements and on the same business domain) creates less opportunity to learn and, therefore, would bring less direct benefits to the rotate person. On the other hand, too much variety may create a long and steep learning curve for the rotate person and other individuals involved in the rotation, which could cause loss of performance and increase cognitive workload. It is desirable to plan a rotation taking the work variety into consideration whenever possible in practice.

Similarly, it is also important to take into consideration if the rotated person will need to still support the source project after the rotation. This creates the need of knowledge transfer that can increase cognitive workload and decrease performance. It is possible that creating practices to increase knowledge redundancy

in the software teams will make rotations more effective by reducing the need of knowledge transfer.

Another important factor is the moment to perform a rotation. It is important to have a balance on the frequency of rotations of a given person. Rotations performed too often and in the middle of the project development decreases the perception of well-defined work and might have a negative impact on feedback. On the other hand, leaving a person for too long in the same project is not desirable because software engineers value work variety, as found in our study and in the literature about motivation [20][21].

In summary, certain desirable goals with the use of job rotation can be conflicting and some of these conflicts may not be reconciled due to practical reasons. Our theory can be useful to raise the awareness of the potential conflicts and prepare managers and team members to deal with negative impacts.

6.3 Future work

This research proposed a preliminary theory of job rotation that is the first we are aware of in software engineering. Although we believe that it can be instrumental to practitioners as it is, there are several directions for future research that can complement the explanatory power and extend the generalizability of the theory.

One direction would be to investigate the effects of job rotation on team processes, such as cohesion [22] and conflict [23]. A similar methodological approach, using qualitative case studies, would be adequate for such exploratory inquires.

Most factors affected by job rotation identified in our research can be related to individual motivation and job satisfaction. Another direction for future research would be to investigate the effects of job rotation under the lens of a theory of motivation and job satisfaction, such as the TMS-SE proposed by França [21]. We plan to perform a reanalysis of our results comparing and contrasting our theory with TMS-SE, aiming at increasing the explanatory power of our theory of job rotation.

Finally, the need for balance that our theory proposes has an important temporal component. Temporary losses tend, as the name implies, to disappear over time. Permanent losses and lasting gains should be more persistent. Longitudinal studies would offer a complementary perspective on our findings. We also plan to conduct such study by investigating the trajectory of rotate individuals over a longer period of time, using an ethnographically based approach.

7. CONCLUSIONS

We presented the results of an instrumental case study performed with the objective of building a theory of job rotation in software engineering. Our case was a large software company that had over 500 employees and executed about 50 projects concurrently at the time of the study. The company employed job rotation to fulfil the dynamic needs for manpower in the projects, and also to increase work variety and create more flexible teams. Our results indicates that the practice of job rotation in the company indeed increased work variety and this, in turn, created opportunities to acquire new technical and business knowledge. We did not investigate whether the goal of having more flexible teams were achieved.

We employed qualitative research techniques to collect, analyse and synthesize data via interviews and documentation provided by the company. Qualitative coding techniques were employed to identify the effects of job rotation on work factor and also to build a theory that explains these effects. We then used member checking techniques and literature enfoldng to check the consistency of our interpretations and to raise theoretical level.

In summary, our results showed that the effects of job rotation on the work environment and the individuals were complex and interrelated. The lasting gains obtained with more variety at work and the opportunity to learn created an adverse effect on performance. At the same time that an increase in work variety was considered positive, job rotation also negatively impacted the perception of performing a well-defined work. Job rotation could also have a temporary negative effect on performance and a more lasting negative effect on individual performance feedback.

We have three findings that are more specific to software engineering. First, cognitive workload plays a more central role in job rotation in software engineering than in other activities that are less cognitive intensive [21]. Second, in software engineering job rotation decreases “task identity” and this seems to be due to the nature of software development in which projects are different from each other more than in other types of businesses. Third, “individual skills” and “the moment of rotation” are important moderators of the increase on workload and decrease on productivity in software engineering. The former because the tasks in software development require more specialization and the latter because individual pieces of software and the tasks related to their development, usually take longer to be completed than in other types of job.

Our main conclusion is that job rotation is an important managerial practice to account for the dynamic changes in manpower requirements in software projects, but that it is necessary to be aware of the potential negative effects of this practice. Balancing the positive effect on work variety and the negative effects on well-defined work and, ultimately, on performance may not be always feasible in practice, but managers and team members must understand and prepare for mitigating possible problems. Our theory of job rotation is the first that provides a comprehensively understanding of the impact of this practice in software engineering. As a mid-range theory, we do not claim that it is universally generalizable, but it can be modified and extended with data and results from new contexts. Therefore, our theory provides a well-founded starting point for future research. Further, researchers and practitioners can use our theory to learn the complexity of the studied phenomenon and to assess how to transfer its findings to other contexts.

8. ACKNOWLEDGMENTS

Fabio Q. B. da Silva holds a research grant from CNPq #314523/2009-0. Cleyton V. C. de Magalhães and Ronnie E. S. Santos are PhD students and receive a scholarship from CNPq. FACEPE also supported this research under the MSc scholarship IBPG-0651-1.03/12. Last but not least, we are very grateful to all participants for dedicating their time and attention to our study.

9. REFERENCES

- [1] OLLO-LOPEZ, A.; BAYO-MORIONES, A.; LARRAZA-KINTANA, M. The relationship between new work practices and employee effort. *Journal of industrial relations*. Vol. 52, no. 2, 2010.
- [2] COYNE, P. F. An evidence-based model of job rotation (PhD Dissertation). Middlesex University. London, UK, 2011.
- [3] WOODS, R. *Human Resources Management*. AHMA. Michigan, 1995.
- [4] PIRZADEH, L. *Human Factors in Software Development: A Systematic Literature Review (Master Thesis)*. Chalmers University of Technology. Göteborg, Sweden, 2010.
- [5] HSIEH, A.; CHAO, H. A Reassessment of the Relationship between Job Specialization, Job Rotation and Job Burnout: Example of Taiwan's High-technology Industry'. *International Journal of Human Resource Management*, 2004.

- [6] KUIJER, P. P.; VRIES, W. H.; VAN DER BEEK, A. J.; VAN DIEËN, J. H.; VISSER, B. FRINGS-DRESEN, M. H. Effect of job rotation on work demands, workload, and recovery of refuse truck drivers and collectors. *The journal of human factors and ergonomics society*, 2004.
- [7] RICHARDSON, A.; DOUGLAS, M.; SHUTTLE, R.; HAGLAND, M. R. Critical care staff rotation: outcomes of a survey and pilot study. *Nursing in Critical Care*, 2003.
- [8] CAMPION, M.; CHERASKIN, L.; STEVENS, M. Career-related antecedents and outcomes of job rotation. *Academy of management journal*, 1994.
- [9] KAYMAZA, K. The Effects of Job Rotation Practices on Motivation: A Research on Managers in the Automotive Organizations. *Business and Economics Research Journal*, 2010.
- [10] HO, W. H.; CHANG, C. S.; SHIH, Y. L.; LIANG, R. D. Effects of job rotation and role stress among nurses on job satisfaction and organizational commitment. *BMC Health Services Research*, 2009.
- [11] FÆGRI, T. E.; DYBÅ, T.; DINGSØYR, T. Introducing knowledge redundancy practice in software development: Experiences with job rotation in support work. *Information and Software Technology*, v. 52, 2010.
- [12] FÆGRI, T. E. Improving General Knowledge in Agile Software Organizations Agile Conference. Chicago, 2009.
- [13] EISENHARDT, K. M. Building Theories From Case Study Research. *The Academy of Management Review*, 1989.
- [14] MERRIAM, S. B. *Qualitative Research: a Guide to Design and Implementation*. San Francisco, CA: Jossey-Bass, 2009.
- [15] SEAMAN, C. Qualitative Methods in Empirical Studies of Software Engineering. *IEEE Transactions on Software Engineering*. Volume 25, Issue 4, 1999.
- [16] PATTON, M. Q. *How to use qualitative methods in evaluation*. Newbury Park, CA: Sage, 1989.
- [17] STRAUSS, A.; CORBIN, J. *Pesquisa qualitativa: técnicas e procedimentos para o desenvolvimento de teoria fundamentada*. Tradução Luciane de Oliveira da Rocha. 2. ed. Porto Alegre: Artmed, 2008.
- [18] HARPER, M.; COLE, P. Member Checking: Can Benefits Be Gained Similar to Group Therapy? *The Qualitative Report*, 2012.
- [19] KREFTING, L. Rigor in qualitative research: The assessment of trustworthiness. *The American Journal of Occupational Therapy*, 1991.
- [20] FRANÇA, A. C. C.; SHARP, H.; DA SILVA, F. Q. B. Motivated software engineers are engaged and focused, while satisfied ones are happy. *Proceedings of the 8th ACM/IEEE International Symposium on Empirical Software Engineering and Measurement (ESEM)*. Torino, Italy, 2014.
- [21] FRANÇA, A. C. C. *A Theory of Motivation and Satisfaction of Software Engineers* (PhD Dissertation). Center for Informatics, Federal University of Pernambuco, Recife, 2014.
- [22] A. V. Carron, L. R. Brawley. Cohesion: Conceptual and Measurement Issues, *Small Group Research*, 31, 1, 89-106, 2000.
- [23] K.A. Jehn. A multimethod examination of the benefits and detriments of intragroup conflict, *Adm. Sci. Q* 40, 256-282, 1995.

10. UNCITED STUDIES SELECTED IN THE SYSTEMATIC REVIEW

- [24] PASSOS, C.; CRUZES, D. S.; MENDONÇA, M. Applying Theory of Reasoned Action in the Context of International Conference on Evaluation and Assessment in Software Engineering. Porto de Galinhas, 2013.
- [25] COHEN, C. F.; BIRKIN, S. J.; GARFIELD, M. J.; WEBB, H. W. Managing Conflict in Software Testing. *Communications of the ACM*, 2004.
- [26] ANG, K. T.; THONG, J. Y. L.; YAP, C. S. It Implementation through The Lens of Organizational Learning: A Case Study of Insuror. *International Conference on Information Systems*, 1997.
- [27] SIY, H. P.; HERBSLEB, J. D.; MOCKUS, A.; KRISHNAN, M.; TUCKER, G. T. Making the Software Factory Work: Lessons from a Decade of Experience. *International Software Metrics Symposium*, 2001.
- [28] BIRK, A.; HELLER, G.; JOHN, I.; SCHMID, K.; MABEN, T. MÜLLER, K. *Product Line Engineering: The State of the Practice*. IEEE Software, 2003.
- [29] PATNAYAKUNI, R.; RAI, A.; TIWANA, A. Systems Development Process Improvement: A Knowledge Integration Perspective. *IEEE Transactions on Engineering Management*, 2007.
- [30] KUKKO, M.; HELANDER, N.; VIRTANEN, P. Knowledge management in renewing software development processes. *Hawaii International Conference on System Sciences*, 2008.
- [31] MOE, N. B.; DINGSØYR, T.; DYBÅ, T. Overcoming barriers to self-management in software teams. *IEEE Software*, 2009.
- [32] XU, Y.; ZHANG, C.; ZHANG, C. Information Seeking in an Information Systems Project Team. *IEEE Transactions on Professional Communication*, 2010.
- [33] JAAKKOLA, H.; HENNO J.; LINNA, P. Software Development in a Multicultural Context: Adaptive and Learning Organizations. *International Convention on Information and Communication Technology, Electronics and Microelectronics*, 2010.
- [34] MELO, C.; CRUZES, D. S.; KON, F.; CONRADI, R. Agile Team Perceptions of Productivity Factors. *Agile Conference*, 2011.
- [35] FRANCA, C. A. C.; FELIX, A. L. C.; DA SILVA, F. Q. B. Towards an explanatory theory of motivation in software engineering: a qualitative case study of a government organization. *International Conference on Evaluation & Assessment in Software Engineering*, 2012.
- [36] FOWDAR, C. D.; NAGOWAH, S. D. Knowledge Management Practices in IT ProjectBased Companies in an African Country. *International Conference on Computer & Information Science*, 2012.
- [37] ALNUEM, A. M.; AHMAD, A.; KHAN, H. Requirements Understanding: A Challenge in Global Software Development. *International Conference on Computer Software and Applications*, 2012.
- [38] KANKANHALLIA, A.; TANA, B. C. Y.; WEIA, K.; HOLMES, M. C. Cross-cultural differences and information systems developer values. *Decision Support Systems*, 2004.
- [39] BJARNASON, E.; RUNESON, P.; BORG, M.; UNTERKALMSTEINER, M.; ENGSTRÖM, E.; REGNELL, B.; SABALIAUSKAITE, G.; LOCONSOLE, A.; GORSCHKE, T.; FELDT, R. Challenges and practices in aligning requirements with verification and validation: a case study of six companies. *Empirical Software Engineering*, 2013.