Pós-Graduação em Ciência da Computação

A Theory of Motivation and Satisfaction of Software Engineers

By

Alberto César Cavalcanti França

PhD Thesis

Universidade Federal de Pernambuco
posgraduacao@cin.ufpe.br
www.cin.ufpe.br/~posgraduacao

2014
Recife

Universidade Federal de Pernambuco
CENTRO DE INFORMÁTICA
PÓS-GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO

Alberto César Cavalcanti França

A Theory of Motivation and Satisfaction of Software Engineers

Este trabalho foi apresentado à Pós-Graduação em Ciência da Computação do Centro de Informática da Universidade Federal de Pernambuco como requisito parcial para obtenção do grau de Doutor em Ciência da Computação.

ORIENTADOR: ***PROF. FABIO QUEDA BUENO DA SILVA***

2014
RECIFE

Catalogação na fonte
Bibliotecária Joana D'Arc L. Salvador, CRB 4-572

França, Alberto César Cavalcanti.
A theory of motivation and satisfaction of software engineers / Alberto César Cavalcanti França. – Recife: O Autor, 2014.
200 f.: fig., tab.

Orientador: Fabio Queda Bueno da Silva.
Tese (Doutorado) - Universidade Federal de Pernambuco. CIN. Ciência da Computação, 2014.
Inclui referências e apêndices.

1. Engenharia de software. 2. Motivação no trabalho. 3. Satisfação no trabalho. I. Silva, Fabio Queda Bueno da (orientador). II. Título.

005.1 (22. ed.) MEI 2014-107

Tese de Doutorado apresentada por **Alberto César Cavalcanti França** à Pós Graduação em Ciência da Computação do Centro de Informática da Universidade Federal de Pernambuco, sob o título **"A Theory of Motivation and Satisfaction of Software Engineers"** orientada pelo **Prof. Fabio Queda Bueno da Silva** e aprovada pela Banca Examinadora formada pelos professores:

Profa. Patricia Cabral de Azevedo Restelli Tedesco
Centro de Informática / UFPE

Prof. André Luis de Medeiros Santos
Centro de Informática / UFPE

Prof. Guilherme Horta Travassos
Programa de Engenharia de Sistemas / UFRJ

Prof. Cleidson Ronald Botelho de Souza
Faculdade de Computação / UFPA

Profa. Helen Sharp
Open University, UK

Visto e permitida a impressão.
Recife, 11 de março de 2014.

Profa. Edna Natividade da Silva Barros
Coordenadora da Pós-Graduação em Ciência da Computação do
Centro de Informática da Universidade Federal de Pernambuco.

T.: T.: G.: O.: T.: G.: A.: O.: T.: U.:

ACKNOWLEDGEMENTS

Presenting this thesis is a great individual accomplishment of will, work and wisdom. It does not mean, however, that it resulted from a lonely effort. During the twelve-years period during which I pursued this goal, I met (literally) thousands of people, hundreds of which voluntarily spent a couple of minutes of their lives listening to me talk about motivation of software engineers.Several of these individuals spent more than just a couple of minutes offering me their valuable opinion about this work. A few others, still, decided to get heavily involved and spent much more time helping me achieve my dream. My perseverance has been always reinforced by this friendly unnameable crowd.

I would like to express my special appreciation and thanks to my advisor Professor Dr. Fabio Queda, who wisely conducted me through the enlightning process of solving a research problem, but more than that, thru the ennobling process of becoming a professional scientist. There could be no better place than the Center of Informatics of the Federal University of Pernambuco for the rise of such fruitful partnership. Priceless has also been the lessons learned while with Prof. Helen Sharp, PhD, at the Open University in Milton Keynes, UK. Thanks, Helen, for being such an adorable person, and such a sharp professional. Without your precious attention and stimulation, I would never have figured out what the word motivation really means. I owe you my deepest respect.

I owe another special thanks to my colleagues Adelnei Felix, Ana Catarina Araújo, David Carneiro, Eric Sales, Pedro Santos and Danilo Monteiro, who actively participated in this research project. Without your help, this work would not have been possible. Thanks also for the availability of all participants of our research, and for the attention of the four organisations, whose the identities and names, unfortunately, I am not alowed to write here.

I extend this acknowledgement to all the colleagues with whom, some day, I had the pleasure to work, in this research group that suddenly we decided to call HASE - Human Aspects in Software Engineering. In particular my friends Cleviton Monteiro, Shirley Cruz, Tatiana Gouveia, Marcos Suassuna, Rodrigo Lopes, Elisa Cardozo, even though they may not be aware of that, I have learned a lot with each one of them. I am very proud of sharing important moments of our career development with them, and I wish that my contributions to their history were as relevantas their contributions to mine.

Thanks to my beloved family.To Keyla, my sincere admiration for your inestimable support. I understand that studying has consumed a huge portion of my time, and I fully recognize that you gave up a lot to help me to get here. However, you must know that my passion for work is far lesser than my pleasure for being with you. I feel that this effort already has been partially recompensed with the lovely moments that we spent together. The remaining reward is yet preserved for our future together, by God. To Lara, who was born in the course of this work, and may not know how her life was strongly affect by it, just thanks for being a good girl.

I am also grateful for the incentive received from my brothers, Pedro França and all the other who reconize me as such. Finally, thanks to my parents and extended family, who are the actual responsible for everything that went right in my life so far. Berto França, Antonio Tiago and John Bridges have sheltered and welcomed me whenever I needed, even in the most inconvenient moments, always blessing me with free encouragement, as in my father's words: "I don´t know what exactly are you writing about, and I don´t fully understand what you do in those international conferences, but if people want to read and listen to you, you should keep doing it!". He was the first person to buy my first book, but I am sure he has never read it.

This research work was partially supported by the Brazilian National Research Council (CNPq), process #141156/2010-4. The international cooperation with the Open University was part of the Science Without Borders program, process #245981/2012-8. The Centre of Research of the College of Philosophy, Science and Letters of Caruaru has also contributed to this project.

"One school is finished, and the time has come for another to begin."

— Richard Bach, Jonathan Livingston Seagull: A Story

A Theory of Motivation and Satisfaction of Software Engineers

ABSTRACT

Context: Previous research work in the Software Engineering field indicates that a proper management of motivation and job satisfaction at work can help software organisations to achieve higher levels of project success. However, the little concern with the adequate use of well-established theories to underpin these researches left unclear several theoretical and practical aspects of work motivation and job satisfaction in the software context. In fact, there is enough knowledge about job satisfaction factors, but not on specific characteristics of the work that motivate software engineers. **Objective:** The starting point of this research comprises the Job Satisfaction and the Job Characteristics theories, which argue that job satisfaction and work motivation are distinguishable phenomena, with distinct antecedents and different outcomes. Then, this thesis aims to clarify specifically what factors drive motivation of software engineers at work. **Method:** The initial theoretical framework was evaluated and enhanced based on findings from a multiple case study that comprised four different software organisations from Recife, Brazil. For 11 months, rich data was collected independently in those organisations, by means of semi-structured interviews, diary studies, and document analyses, and the synthesis followed a standard procedure of cross-case analysis. **Results:** The results point out that (1) practitioners are not aware of the distinction between work motivation and job satisfaction, (2) work motivation is characterized by engagement and concentration, (3) work motivation is affected by software engineering tasks characteristics and by the co-workers' engagement, workload and technical confidence, (4) work motivation improves satisfaction moderated by feedback information provided about the individual's performance, and (5) the mediating role of individual characteristics is pervasive. **Conclusion**: Based on these data, it was possible to draw up a new theory of motivation and satisfaction of software engineers (TMS-SE), which unites elements from well established theories, expands and adapts them to the software engineering specific context. The TMS-SE represents an advance on our understanding of software engineers' behaviour as well as it raises new questions and provides an organised ground for future investigations in this area.

Keywords: Software engineer, Work Motivation, Job Satisfaction, Multi-case study, Cross-case analysis

Uma Teoria de Motivação e Satisfação de Engenheiros de Software

RESUMO

Contexto: Pesquisas na área de engenharia de software indicam que o gerenciamento apropriado da motivação e satisfação no trabalho são importantes para o sucesso de projetos de software. No entanto, rara tem sido a preocupação com o uso apropriado de teorias bem estabelecidas para fundamentar tais pesquisas, o que deixa em aberto várias questões práticas sobre motivação e satisfação no contexto do desenvolvimento de softwares. Evidências apontam que o conhecimento sobre a satisfação no trabalho, neste contexto, está relativamente consolidado, mas ainda há muito a se aprender sobre as características específicas que antecedem a motivação dos engenheiros de software. **Objetivo**: O ponto de partida compreende teorias de Satisfação no Trabalho e das Características do Trabalho, que defendem que motivação e satisfação no trabalho referem-se a fenômenos distintos. Esta tese objetiva então clarificar quais são as características do trabalho que influenciam a motivação de engenheiros de software. **Método**: Este quadro teórico inicial foi evoluído baseado nos aprendizados resultantes de um estudo de múltiplos casos, executado em quatro organisações de software em Recife-PE. Durante 11 meses, dados foram coletados nestas organizações, através de entrevistas semi-estruturadas, estudos diários, e análise documental. **Resultados:** Os resultados apontam que (1) engenheiros de software não estão conscientes sobre a distinção entre os dois fenômenos (motivação e satisfação no trabalho), (2) motivação é caracterizada pelo engajamento e concentração, (3) motivação é afetada por diversas características da tarefa do engenheiro de software, mas também pela percepção sobre o engajamento dos colegas de trabalho e pela auto-confiança técnica do trabalhador, (4) motivação contribui para a satisfação no trabalho, moderada pela informação provida sobre a performance individual dos engenheiros, e (5) o papel mediador das características pessoais do indivíduo é universal. **Conclusão**: Com base nestes dados, é proposta uma nova teoria de motivação e satisfação de engenheiros de software (TMS-SE) que une elementos de teorias bem estabelecidas, expandindo-as e adaptando-as à realidade específica de engenheiros de software. A TMS-SE representa um avanço em nossa compreensão do comportamento de engenheiros de software, bem como levanta novas questões e propõe um terreno organizado para futuras investigações nesta área.

Palavras-chave: Engenheiro de Software, Motivação no Trabalho, Satisfação no Trabalho, Estudo de múltiplos-casos, Análise cross-casos

ACRONYMS

CLT	*Consolidação das Leis do Trabalho*
CMM	Capability Maturity Model
CMMi	Capability Maturity Model Integrated
SW-CMMi	Software Capability Maturity Model
DP	Data Processing
DSD	Distributed software development
EBSE	Evidence Based Software Engineering
ESE	Empirical Software Engineering
GNS	Growth Need Strength
HR	Human Resources
IS	Information Systems
IT	Information and Technology
JCT	Job Characteristics Theory
JDS	Job Diagnostics Survey
MS	Mapping Study
MPS	Job's Motivating Potential Score
MPS.br	*Modelo de Processo de Software Brasileiro*
MOCC	Motivators, Outcomes, Characteristics and Context
OSS	Open source software
PMBOK	Project Management Body of Knowledge
PMP	Project Management Professional
PS	Primary Study
R&D	Research and Development
SLR	Systematic Literature Review
SCRUM	Scrum is the name of the agile methodology, it is not an acronym
SCT	Social Cognitive Theory
SNS	Social Need Strength
TMS-i	Initial Theory of Motivation and Satisfaction
TMS-SE	Theory of Motivation and Satisfaction of Software Engineers
XP	Extreme Programming

LIST OF ILLUSTRATIONS

LIST OF TABLES

CONTENTS

Chapter 1 Introduction

Software engineering can be described as a highly social-intensive activity, because beyond the technical aspects extensively studied in this field, there is a diversity of human (WEINBERG, 1971)and social (DEMARCO and LISTER, 1987)aspects that may affect the performance of software engineers at work.

A naïve account of the software enginering work would tend to see human and technical aspects separately: the formers including forms of interaction, behaviours, and organisation of people, while the latters addressing the use that individuals and teams make of technologies, methods, processes and tools for software development. However, in practice, it is difficult to disentangle the way people do things from the methods, techniques, and computing technologies they use (GUINAN, COOPRIDER and FARAJ, 1998).

One of these human aspects, the motivation of software engineers, is "reported to have the single largest impact on productivity and software quality management, and continues to be undermined and problematic to manage" (HALL, BADOO, *et al.*, 2009, p. 10:2). This thesis presents a theory of work motivation and job satisfaction of software engineers (TMS-SE), developed initially from previously existing theories and enhanced and adapted for the software engineering context.

Motivation and job satisfaction have been objects of study for a long time, in many different fields (STEERS, MOWDAY and SHAPIRO, 2004). In software engineering, in particular, they have been studied for more than thirty years (COUGER, J.DANIEL; ZAWACKI, R. A., 1980).In the last ten years, these phenomena have increasingly attracted attention from the software engineering community, due to previous research that claimed that a proper management of motivation and satisfaction at work could help software organisations achieve higher levels of productivity, and avoid human resource turnover, budget overflow, and delivery delays (BEECHAM, BADOO, *et al.*, 2007) (FRANÇA, GOUVEIA, *et al.*, 2011). All these impacts would represent relevant contributions to the overall success of software development projects (REEL, 1999) (HUMPHREY, 2006).

Notice that researchers in the organisational behaviour field are aware of the existence of a wide range of inter-connected factors and phenomena that challenge an unequivocal understanding of what can really be useful to the management of work motivation and job satisfaction. Therefore, over time, several theories of work motivation and job satisfaction have been carefully developed, evaluated, questioned, and evolved, through a continuous cycle of interaction between theoretical and empirical research work. Some of these theories have been completely discharged, such as Maslow's Hierarchy of Needs Theory (MASLOW, 1954), while others have endured, such as the Job Characteristics Theory (HACKMAN, 1980).

However, there are two main reasons to question whether the existing theories of work motivation and job satisfaction developed in other fields are applicable in a software development environment. First, in the past, researchers have shown that software engineers hold in common specific personal characteristics, and what influence their work motivation and job satisfaction is likely to be different from other professionals (COUGER and ZAWACKI, 1980). Second, recent research has argued that the knowledge-intensive nature of the software engineering work also affects the rationale beyond the motivation and satisfaction of the professionals in this field (WALLGREN and HANSE, 2007).

Nevertheless, the number of studies on this topic is relatively small, and it is only possible to find a wispy number of isolated attempts to evaluate work motivation or job satisfaction theories or models in software engineering contexts. In general, studies on motivation and satisfaction of software engineers are characterized by little concern with an adequate use of well-established theories to underpin their research designs. Besides, empirical studies in this area are still concentrated in high HDI countries, so aspects such as international cultural differences challenge the transferability of their results to abroad.

All these issues, together, prevent the accumulation of knowledge, leaving still unclear several theoretical and practical aspects of work motivation and job satisfaction in the software context and, although some advance has been indeed achieved, all research effort so far represents only an "unconnected body of work" (HALL, BADOO, *et al.*, 2009). Besides, borrowing from Hackman and Oldham *et al.* (1975), much current writing about **work motivation and job satisfaction in software engineering** is enthusiastic, sometimes even messianic, about what it can accomplish, while relevant questions such as exactly what should be done to improve jobs, and how, tend to be ignored.

The aim of this research is to generate a more sensible and contemporaneous understanding of how the work motivation and job satisfaction of software engineers are influenced by workplace factors, and how these phenomena influence their work-related behaviour. In pursuit of this goal, the theory presented in this thesis was built in three steps which, in a bird's eye view, can be described as follows:

Step 1: Literature review, and establishment of the initial theoretical model. First, we extended a previously existing systematic literature study on motivation and satisfaction of software engineers (BEECHAM, BADOO, *et al.*, 2007), reaching a list of 140 papers published between 1980 and 2010.These papers were individually analysed and synthesized to provide a comprehensive and updated view of the state of the art in this subject (FRANÇA, GOUVEIA, *et al.*, 2011), as well as to identify research gaps. The concepts of motivation and satisfaction as in the most frequently referred theories of motivation were reviewed, and the Job Characteristics Theory (HACKMAN, 1980) emerged as a promising starting point. The Job Satisfaction Theory (LOCKE, 1969), then, was added to elucidate the phenomena under investigation, and strengthen the initial theoretical framework. Chapter 2 (p. 20) introduces the theoretical foundations of this work and details these two cornerstone theories, and Chapter 3 (p. 35) depicts a historical view of the research on motivation and satisfaction in software engineering, as well as the current state of art and research gaps.

Step 2: An empirical study of software engineers in practice. This research is broadly based on empirical software engineering methods and guidelines. Empirical Software Engineering is a research paradigm that makes use of well-proven research methods to plan and carry out investigations, enhancing the scientific nature of software engineering investigations. Empirical research explores, describes, predicts, and explains natural, social, or cognitive phenomena by using scientific methods and evidence-based experience (SJOBERG, DYBA and JORGENSEN, 2007). Evidence is any observable event that tends to establish or disprove a fact (KITCHENHAM, BUDGEN, *et al.*, 2005). Thus, in order to gather data to evaluate the initial theoretical framework, a multi-case study was carried out in four software engineering settings, in Recife, Brazil: a government organisation, a R&D not-for-profit organisation, a small startup company, and the IT department of a University. For 11 months,

rich data was independently collected in those organisations, by means of semi-structured interviews, diary studies, and document analyses. Then, a cross-case comparison was carried out, providing supporting, conflicting and complementary evidence to several aspects of our initial theoretical framework. The research method, data collection and analysis procedures are carefully explained in Chapter 4 (p. 52), as well as the rationale beyond the research strategy and the threats to validity and reliability of our research design. Chapter 5 (p. 79) reports the results of the cross-case analysis.

Step 3: The evaluation and evolution of the initial theoretical model. The last step of this research aimed to consolidate the new theory of work motivation and job satisfaction of software engineers (TMS-SE), by following a process of building theory from case study research, suggested by Eisenhardt (EISENHARDT, 1989). First, the facts witnessed in the case studies are reviewed, and discussed in light of the initial theoretical framework, which is then evaluated, expanded and adapted to the reality of software engineers. Finally, the TMS-SE is compared against similar and conflicting evidence from the studies available in the literature, enfolding the experience provided from them. Chapter 6 (p. 107) deeply describes this process, discuss some important theoretical considerations, and presents reflections about the challenges for the software engineering practice.

The output of this effort, the TMS-SE, argues that work motivation and job satisfaction are distinguishable phenomena, with distinct antecedents and outcomes. Job satisfaction refers to pleasurable emotions in reaction to the job, signalled mainly by the individuals' happiness at work, and influences attitudes towards the organisation (intention to stay, attendance, and others). Work motivation, in contrast, refers to the desire to work, is signalled by individuals' attitudes toward the work (engagement and concentration), and directly influences individual work performance. Then, this theory claims that the job satisfaction of software engineers is determined by the same workplace factors that influence other professionals, while antecedents of work motivation are slightly different. Software engineers are influenced by task characteristics such as significance, skill variety, identity and autonomy, but also by their own technical confidence, by their co-workers' engagement, and by the cognitive workload. The TMS-SE also remarks other factors such

as communication and participation, collaboration, feedback, and the role of individuals' characteristics.

This theory contributes to the state of art in three complementary ways. First, it advances the knowledge on this topic by providing a more solid framework through which the available knowledge on this field is evaluated and encompassed. Second, it enlightens the management of software engineers by clarifying what aspects of the work and the workplace are relevant to the work motivation and job satisfaction of this specific type of professional, as well as by pointing out practical challenges attached to the software development practices. Third, it suggests crucial questions, worthy of further investigation, serving, thus, as a basis to substantiate and organise future research in this area. Chapter 7 (p. 138) presents the concluding remarks of this work, and enumerates suggestions for future research endeavours.

Chapter 2 Theoretical Foundations

The study of work motivation and job satisfaction developed mainly in 1900s. Although they have been implied to refer to different concepts since the early theories, this distinction became explicit only after the 1970's and as a result, these concepts are still confusing. Additionally, the study of human motivation branched out in different theories in several fields, which also contributed to a conceptual uncertainty that is problematic to researchers and practitioners. If a research effort is set out to identify the antecedents and outcomes of work motivation or job satisfaction without a clear understanding of their distinction, its findings may be baffled.

Moreover, drawing management schemes without a clear and consistent basis may result in ineffective practices (GOLEMBIEWSKI, 2000).The risk of confusing work motivation and job satisfaction lays on the different antecedents, behavioural signs, and outcomes. Job satisfaction refers to pleasurable emotions in reaction to the job; and it influences attitudes towards the organisation (intention to stay, attendance, etc.). Work motivation is the desire to work, and has direct effects on individuals' attitudes toward the work. Job satisfaction has no proven direct effect on productivity (LOCKE, 1976)(IAFFALDANO and MUCHINSKY, 1985)(JUDGE, THORESEN, *et al.*, 2001) but it affects other aspects such as physical and mental health, absence and turnover (LOCKE, 1976). Conversely, individuals motivated to work will perform at their possible best, which in turn influence their productivity. Software engineering organisations should be aware of this difference to design effective strategies to deal with different problems related to the human resource management.

In this Chapter, Section 2.1 (p. 21) explores the origin of both concepts, the roots of the confusion, and explains how to distinguish work motivation and job satisfaction. Section 2.2 (p. 25) details the Job Satisfaction Theory (LOCKE, 1976), and Section 2.3 (p. 28) describes the Job Characteristics Theory of Motivation (HACKMAN, 1980), which together form the initial theoretical framework that guided this research, which is consolidated in Section 4.3 (p. 59).

2.1 Work Motivation and Job satisfaction are distinct concepts

According to Steers *et al.*(2004, p. 379), if it is possible to effectively synthesize the different concepts of motivation, they would have common characteristics: "They are all principally concerned with factors or events that energize, channel, and sustain human behaviour over time." Job satisfaction, in contrast, has been defined as "complex emotional reactions to the job" (LOCKE, 1969). Although both phenomena are connected, two critical characteristics make work motivation differ from job satisfaction. First, motivation is future oriented, while satisfaction is past oriented (LOCKE and LATHAM, 1990), i.e. motivation is antecedent of performance, while satisfaction is a consequence of work events, including performance. Second, *work* motivation is about individuals' perception of the work and its intrinsic characteristics, while *job* satisfaction is about the perception of a broader set of elements present in the *job*, including but not limited to the work itself (see Figure 1). Thus, work motivation and job satisfaction refer to different dimensions of the work life.

This text sticks to the following system of meanings, in compliance with *Dictionary.com*:

- *Employment* is the wider context of relationships between individuals and organisations.
    - *Employment* is the amount of *jobs* in a *profession* in a determined area
- *Profession* refers to an occupation that requires a specific knowledge, and by which a person earns a living, e.g. software engineer.
    - A *profession* limits the sorts of *works* that can be done by an individual in a *job*
- *Workplace* refers to the physical environment that configures the concrete place of work.
    - *Workplace* is the place where *work* is done
- *Job* is contractual relationship between an individual and an organisation.
    - *I.* *Job* is a *professional* post of *employment*
    - II. *Job* is a *work* contract
- *Task* is a specific thing to do, e.g. write code, test a system, etc.
    - *Task* is a piece of *work*
- *Work* is a set of tasks that are part of one's responsibilities.
    - *Work* is a set of *tasks*
    - *Work* are the *job*'s obligations

Figure 1 - Basic terminology system

The first attempts to scientifically understand and explain human motivation came from the fields of psychotherapy, psychometrics, and learning theories (PRITCHARD and ASHWOOD, 2008). In its initial stages, there were only isolated initiatives in studying human motivation, most of which were disconnected and lacked sound theoretical foundations (LATHAM and ERNST, 2006). Maslow (1943) offered the first synthetic theory of human motivation, known as Hierarchy of Needs Theory (MASLOW, 1954). However, Maslow lacked theoretical rigour as no explicit definitions for central constructs in his theory were presented: motivation and satisfaction (NEHER, 1991).However, this distinction was implied in his conceptual system (Table 1). As empirical tests have systematically failed to provide clear and consistent support to Maslow's hierarchy of needs (WAHBA and BRIDWELL, 1976), further theories kept being developed, in the psychology field, from his basic notion of motivation and satisfaction (e.g. (ALDERFER, 1969)) offering no further clarifications.

Herzberg (1964) is one of the most influential works in this field. He condensed a broad set of empirical studies with engineers and accountants in the Motivation-Hygiene Theory. This theory asserts that it is only possible to increase employees' satisfaction, and productivity as a consequence, through job enrichment activities, which are "the alteration of specific jobs (…) with the intent of improving both productivity and the quality of employee work experiences" (HACKMAN, 1980, p. 445). In fact, Herzberg uses the terms satisfaction and motivation interchangeably through his work, and it is not possible to find clear definitions of the satisfaction and dissatisfaction constructs, although he refers once to job satisfaction as "what makes people happy on the job"(HERZBERG, 1987, p. 8).

Unlike the previous theories, The Expectancy Theory (VROOM, 1964) explicitly refers to satisfaction and motivation as distinct constructs, as summarized in Table 1. This theory focuses on the variables that affect the conscious decision to apply effort to perform a specific act rather than optional others. It states that "A person is motivated to the degree that he or she believes that (a) effort will lead to acceptable performance, (b) performance will be rewarded, and (c) the value of the rewards is highly positive" (LUNENBURG, 2011).

Building up on the principles of Expectancy Theory and Motivation-Hygiene Theory, the Job Characteristics Theory (JCT) suggested five characteristics of the work likely to foster both satisfaction and work performance through internal work motivation.

According to the JCT's definition, internal work motivation refers to "being turned on to one's work because of the positive internal feelings that are generated by performing well" (HACKMAN, OLDHAM, *et al.*, 1975, p. 2).

Table 1 - Overview of human motivation, work motivation and job satisfaction theories

Theory	Conceptual system	Empirical Support
Hierarchy of Needs Theory (MASLOW, 1954)	It is not possible to find an explicit definition of **motivation** and **satisfaction** in his articles. However, he implies a semantic difference between the words motivation, which refers to a state of need, and satisfaction, which refers to a state of no need.	Maslow does not present any data. Because of the difficulty in interpreting and operationalizing its concepts, the testability of this theory is limited (HALL and NOUGAIM, 1968). Therefore, empirical assessments show generally weak or no support. (WAHBA and BRIDWELL, 1976)
Motivation-Hygiene Theory(HERZBERG, 1964)	It states that satisfied people are more productive, and job satisfaction is activated by two independent sets of factors: *motivators* (or satisfiers) are the primary cause of job satisfaction, and *hygiene factors* (or dissatisfiers) identified as primary cause of job dissatisfaction.	He shows no evidence on the relation between satisfaction and productivity (HOUSE and WIGDOR, 1967). Results are consistently supported only when Herzberg′s basic methodology is used, including his classification scheme. (LOCKE, 1975)
Expectancy Theory (VROOM, 1964)	Satisfaction given by the convergence between subjective expectations and actual outcomes of an action. Motivation is the process of deciding whether an effort to perform a specific action is worthier than its available alternatives, and it is guided by the maximization of satisfaction experiences.	Empirical evaluations generally supported the predictive power of the expectancy theory in laboratory studies, but not in real settings given the existence of excessive uncontrollable factors (WABBA, 1974). (EERDE and THIERRY, 1996)
Goal Setting Theory (LOCKE, 1968)	Motivation is the willingness to strive for the goals of a particular organisation. The four elements that represent motivated behaviour in the Goal Setting theory are: *Direction*: goals direct attention and action; *Effort*: the amount of effort mobilized in proportion to the perceived requirements of the goal or task; *Persistence*: directed effort extended over time; *Strategy development*: the development of strategies or action plans for attaining one's goals.	There have been more than 500 studies of goal setting conducted by Locke, his colleagues, and others (LOCKE, 1996). This is the longest stable theory of performance and task motivation, with the largest amount of empirical work supporting its claims.
Job Satisfaction Theory(LOCKE, 1969)	Job satisfaction is the pleasurable emotional state resulting from the subjective appraisal of one's job as achieving or facilitating the achievement of one's job values, providing these values are congruent with or help to fulfil one's basic needs. Subjective means pertaining only to individuals. *Value* is that which one acts to gain and/or to keep. *Need* refers to objective requirements to an organism wellbeing	Locke describes several empirical studies testing the existing correlation between subjective value-discrepancy and grades of job satisfaction. The results revealed a very similar level of correlation (+.70, +.69, -.61, -.81, and -.72 at $p<.01$).
Job Characteristics Theory (HACKMAN, OLDHAM, *et al.*, 1975)	Internal work motivation refers to "being turned on to one's work because of the positive internal feelings that are generated by performing well". Satisfaction is the degree to which the employee is happy with the job, or with specific aspects of the job.	This theory has found support on tests with more than one thousand people working on more than one hundred different jobs from real organisations, but relying on correlational instead causal analyses. (HACKMAN and LAWLER III, 1971)

The definitions of work motivation and job satisfaction have also stimulated researchers from several fields, resulting in different competing and complementary theories (KATZELL and THOMPSON, 1990). Examples can be found in economics and marketing (BAYTON, 1958), entrepreneurship (MCCLELLAND, 1965), education (SCHUNK, MEECE and PINTRICH, 2013) and decision-making processes (SVENSON, 2013).

That has become another source of confusion. Expectancy theories based on Vroom (VROOM, 1964), for example, assume that motivation is a result of a rational decision-making process, so its developments have naturally been merged to the field of decision-making over time. It is not unusual to see research work focused in decision-making processes using "motivation" as a keyword, like in Baker, Morse & Sherman *et al.*(1986). Moreover, the evaluation of expectations and outcomes is contingent on the individuals, actions and the context, so the factors that account to a specific decision may completely differ among distinct actions and organisations. To these researchers, the term "motivation" in isolation is useless and it should be used as motivation to someone [*with individual values*] to do something [*which will possibly lead to a desired outcome*] in a determined context.

Locke (1969) also developed an extensive theoretical study to redefine the construct of job satisfaction (see Table 1). Since his definition was presented, it has become a consensus between academics from the organisational behaviour field (WEISS, 2002). In contrast, the concept of work motivation remained as a fuzzy abstract concept.

Ambrose and Kulik (AMBROSE and KULIK, 1999) identified that research in the organisational behaviour field replaced the construct "motivation" with more specific measures of employee behaviour in the 1990s, so that this construct is "moving backstage as a largely unmeasured, but still theoretically relevant, mediating variable" (AMBROSE and KULIK, 1999, p. 280).

2.2 Job Satisfaction Theory

The origin of the theory

Edwin A. Locke is an American psychologist, retired as Professor Emeritus from the University of Maryland (LOCKE). Since his early career, Locke was interested in applied problems of industrial psychology. Although there had been intense academic work on this topic at that time, Locke was uncomfortable with the fact that the general ability of predicting and explaining job satisfaction was not evolving (LOCKE, 1975). Besides, the technical literature was becoming trivial, repetitive and inconclusive, because studies only reported correlational data, without much attention on explaining why the variables were (or should be) correlated. The problem with those studies, according to Locke's view, was that researchers were ignoring the epistemological roots of the term "job satisfaction", and consequently its meaning was attached to whatever their measurement instruments where measuring. Thus, most insights about the nature of job attitudes remained at the level of common sense hypotheses rather than being explicitly formulated causal principles. Given this picture, Locke decided to carry out a deep theoretical analysis in order to redefine the term "job satisfaction".

Main constructs and propositions

After reviewing the evolution of the concept of Job Satisfaction, Locke suggested that job satisfaction and dissatisfaction are complex emotional reactions to the job. An Individual's emotions are dependent upon an interaction between the person and his environment through the biological functions of *cognition* (sensations, perception), *evaluation* (consciously or subconsciously selection among alternative life-enhancing actions) and *regulation* (one's judgment of values). Therefore, job satisfaction is not contingent only on things, but it is also dependent of one's judgment. It pertains to a relationship between a person and facts of reality.

Locke defines emotion as "a super-rapid subconscious appraisal of value", and a value is "that which one acts to gain and/or to keep", "is that which one regards as conductive to one's welfare" (LOCKE, 1969). Then, he proposes:

A) Job satisfaction is the pleasurable emotional state resulting from the appraisal of one's job as achieving or facilitating the achievement of one's job values.
B) Job dissatisfaction is the unpleasurable emotional state resulting from the appraisal of one's job as frustrating or blocking the attainment of one's job values or as entailing disvalues.
C) Job satisfaction and dissatisfaction are a function of the perceived relationship between (i) what one wants from one's job, and (ii) what one perceives it as offering or entailing (value-discrepancy model).
D) Values are dynamic, because they involve the discrepancy between individual's perception and social standards. A person who attains his/hers goals will not remain satisfied indefinitely, but will seek for new goals.

Locke's concept of job satisfaction differed from other job attitudes, such as:

- *Morale* is "an attitude of satisfaction with, desire to continue in, and willingness to strive for the goals of a particular group or organization" (LOCKE, 1976). Morale is composed, in part, by job satisfaction.
- *Involvement*: "a person who is involved in his job is one who takes it seriously, for whom important values are at stake in the job, whose moods and feelings are significantly affected by his job experiences and who is mentally preoccupied with his job"(LOCKE, 1976). An involved person is likely to feel more or less satisfaction with job experiences, while an uninvolved person has less emotional reactions to analogous job experiences.
- *Expectations*: some theorists suggest that satisfaction is the difference between what is expected and what actually happens. Locke redefines that difference as a "surprise", which can be pleasant or unpleasant according to one's value judgment (LOCKE, 1976).
- *Needs*: need is a concept borrowed from Biology, and refers to "objective requirements to an organism wellbeing" (LOCKE, 1976). Conscious living organisms may or may not be aware of their needs. A need-discrepancy may cause discomfort, but not action.

Combining the most defendable aspects of other existing theories to his own definition of job satisfaction, Locke suggests (LOCKE, 1976):

> *"Job satisfaction results from the appraisal of one's job as attaining or allowing the attainment of one's important job values, providing these values are congruent with or help to fulfil one's basic needs."(LOCKE, 1976, p. 1319)*

Locke's job satisfaction theory is based on the idea that the **value-discrepancy** is what determines the individual's happiness at work. The value-discrepancy model refers to the difference between the subjective importance attributed to some aspect of the job, and the perceived characteristic of the job as facilitating or blocking the attainment of that value.

Finally, according to the Locke's review, the work life is a complex of interrelated tasks, roles, responsibilities, interactions, incentives, and rewards. He points out ten different dimensions, under which the values of job satisfaction had been investigated, and for which there is evidence to support their effect over job satisfaction. These dimensions are:

a. *The work itself*: including intrinsic interest, variety, learning, difficulty, success, autonomy, etc;
b. *Pay*: amount, fairness, equity, methods of payment, etc;
c. *Promotion*: opportunities for, fairness of, etc;
d. *Recognition*: praise for accomplishment, credit for work done, criticism, etc;
e. *Benefits*: pension, medical, annual leave, paid vacations, etc;
f. *Working conditions*: hours, rest pauses, equipment, etc;
g. *Supervision*: supervisory style, human relations, etc;
h. *Co-workers*: competence, helpfulness, friendliness, etc;
i. *Company*: management: organisational policies;
j. *The self*: self esteem, self-image, etc.

These dimensions combine in three perspectives that are more abstract: those concerning the work itself (a), environmental conditions (b-f) and human/social agents (g-j). However, in fact, this particular combination may only be useful for research purposes, rather than in terms of statistical consideration.

Regarding side effects, job satisfaction has a variety of consequences for the individual (e.g. attitudes toward life, family, and the self), health and longevity. However, job satisfaction has no reported direct effect on productivity.

Empirical evaluations and limitations

Locke describes several empirical studies, designed based on this model of value-discrepancy judgment. The studies aimed to test the existing correlation between value-discrepancy and grades of job satisfaction. The results revealed a very similar level of correlation (+.70, +.69, -.61, -.81, and -.72 at $p<.01$). Although these studies have some validity threats, they represented an improvement because past studies were inconsistent at explaining job satisfaction. However, Locke avoided presenting any precise mathematical formula or objective measurement tools. He argued that "intensity of satisfaction" and "value importance" might not be measured in terms of any known physical or psychological unit (LOCKE, 1969).

2.3 Job Characteristics Theory

The origin of the theory

J. Richard Hackman was a distinguished researcher from the Harvard Business School (HACKMAN). Hackman's initial studies focused on the task-performance relationship, and he investigated four classes of variables that formed the core of the problem: (i) the characteristics of the task; (ii) individual's intentions about how to respond to the task; (iii) work process; and (iv) the outcomes of the work (HACKMAN, 1969). Among other findings, he revealed that task characteristics are associated with employees' hypothesis to work hard on their jobs.

Then, following these initial results, Hackman saw an opportunity to develop new strategies for job enrichment, as a means to increase the motivation and satisfaction of people and, consequently, improve productivity. Hackman identified that:

> *"much current writing about job enrichment is [or was, at that time] enthusiastic, sometimes even messianic, about what it can accomplish. But the hard questions of exactly what should be done to improve jobs, and how, tend to be glossed over" (HACKMAN, OLDHAM,* et al.*, 1975, p. 57).*

Thus, Hackman teamed up students and consultants, and gradually refined his initial models, giving birth to the Job Characteristics Theory (JCT) (HACKMAN,

OLDHAM, *et al.*, 1975). The Motivation-Hygiene theory and the Expectancy Theory of Motivation provided the primal bases for the JCT.

Main constructs and propositions

The primary aspirations of Job Characteristics Theory (JCT) were (1) to explain how properties of the work itself affect people's attitudes toward the work, and (2) to identify the conditions under which these effects are likely to be strongest (HACKMAN and LAWLER III, 1971) (OLDHAM and HACKMAN, 2005). The theory specifies objective characteristics of tasks that create conditions for high levels of *internal work motivation* on the part of employees (HACKMAN, 1980).

According to the Hackman's definition, *Internal work motivation* refers to being turned on to one's work because of the positive internal feelings that are generated by performing well (HACKMAN, OLDHAM, *et al.*, 1975). It is the individuals' willingness to work hard and well (HACKMAN, 1980).

Hackman argued that behavioural scientists had then found three psychological states that are critical in determining if a person is internally motivated (HACKMAN, OLDHAM, *et al.*, 1975):

> *"Individuals will be internally motivated to perform well when they experience the work as meaningful, they feel they have personal responsibility for the work outcomes, and they obtain regular and trustworthy knowledge of the results of their work" (HACKMAN, 1980, p. 447).*

When these three conditions are present, a person tends to feel good about himself when performing well, so those good feelings will prompt him/her to continue trying to do well. The three psychological states are better described as follows (OLDHAM and HACKMAN, 2005):

- *Experienced Meaningfulness*: The degree to which the employee experiences the work as inherently meaningful, as something that "counts" in his or her own system of values (OLDHAM and HACKMAN, 2005).

- *Experienced Responsibility*: The degree to which the individual feels personally accountable and responsible for the results of the work he/she does.
- *Knowledge of Results*: The degree to which the individual has confident knowledge about how well he or she is performing.

JCT then suggests that the simultaneous presence of these three psychological states results in a set of favourable personal and work outcomes, but if they are deficient, fewer of these outcomes emerge:

- *Internal work motivation*: feel good when performing well, and feel bad or unhappy when performing poorly;
- *Satisfaction with the work*: both with the opportunities for personal growth and development at work and with the job in general;
- *Work performance*: produce work that is of high quality;
- Low absenteeism and turnover.

The three psychological states are internal to individuals and, therefore, do not represent properties of the work itself that might be changed or manipulated for purposes of job enrichment. Alternatively, the JCT identifies five objective characteristics of jobs that, when present, increase the chances that an employee will experience the three psychological states and, through them, shape the personal and work outcomes (OLDHAM and HACKMAN, 2005).

Experienced Meaningfulness is shaped by three job characteristics:

i. **Skill Variety** is the degree to which the job requires a number of different activities in carrying out the work, which involve the use of a number of different skills and talents of the individual. Work that stretches one's skills and abilities invariably is experienced as more meaningful than work that is simple and routine.
ii. **Task Identity** is the degree to which the job requires completion of a whole and identifiable piece of work, doing a job from beginning to end with a visible outcome. Putting together an entire product or providing a complete unit of service is inherently more meaningful than being responsible for only a small part of the work.

iii. **Task Significance** is the degree to which the work has a substantial impact on the lives of other people, whether in the immediate organisation or in the external environment. An activity that is consequential for the psychological or physical well-being of others is experienced as more meaningful than is work that makes little difference to anyone else.

Experienced Responsibility is shaped by the amount of autonomy the job provides:

iv. **Autonomy** is the degree to which the work is structured to provide the employee with substantial freedom, independence, and discretion in scheduling the work and in determining the procedures to be used in carrying it out.

Knowledge of Results is shaped by the degree to which carrying out job-specified work activities provide the individual with direct and clear feedback:

v. **Feedback** is the information about the effectiveness of his or her performance. When someone receives information about his or her performance from the work itself (e.g., when a salesperson closes a deal and receives payment from a customer), that feedback is direct and immediate and, therefore, contributes substantially to his or her overall knowledge of results about work outcomes.

Then, jobs high on the five core dimensions were described as having high Motivation Potential Score (MPS), given by the following formula (OLDHAM and HACKMAN, 2005):

$$MPS = [(Skill\ Variety + Task\ Identity + Task\ Significance) \div 3] \times Autonomy \times Feedback$$

Although Hackman and his associates suspected that the core propositions applied to most people, they noticed in their empirical studies that not everyone was able to become internally motivated at work, even when the MPS was high (HACKMAN, OLDHAM, *et al.*, 1975). Thus, they incorporated the concept of "Growth Need Strength" (GNS) in their theory, defined as "the degree to which an individual values opportunities for personal growth and development at work" (HACKMAN, OLDHAM, *et al.*, 1975). The JCT posits that individuals who have high GNS respond positively to jobs high on MPS, while individuals who have low GNS respond less positively, or may even feel over-stretched.

The JCT is summarized in Figure 2.

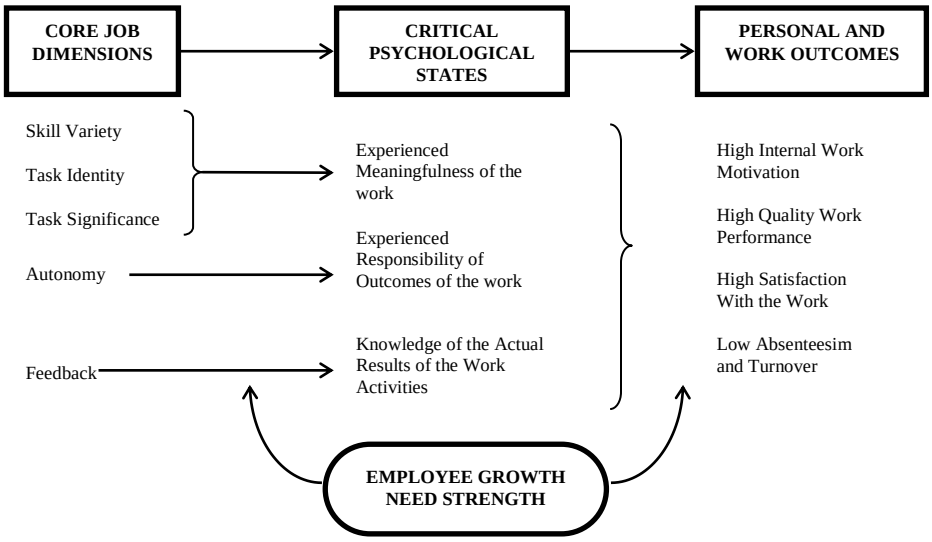


Figure 2 The Job Characteristics Model (HACKMAN and OLDHAM, 1975, p. 161)

Hackman and Oldham developed then a questionnaire tool – named Job Diagnostic Survey (JDS) to assess employees' perceptions of the five core job characteristics, their experienced psychological states, their GNS, and the affective outcomes (internal motivation and job satisfaction) (HACKMAN and OLDHAM, 1974) (HACKMAN and OLDHAM, 1975). This tool was part of their job-redesign process, and its function was: (1) evaluate the objective characteristics of the jobs (MPS); (2) assess the current levels of motivation, satisfaction, and work performance of employees on the job (JDS); and (3) identify the level of growth-need strength of the employees (GNS). The JDS does not assess work performance, absenteeism and turnover. In their studies, this data was collected from supervisors, or from the companies' records (HACKMAN and LAWLER III, 1971).

Empirical evaluations and limitations

After testing their theory with more than one thousand people working on more than one hundred different jobs from real organisations (HACKMAN and OLDHAM, 1975), their findings are summarized in the following statements (HACKMAN, OLDHAM, *et al.*, 1975):

I. People who work on jobs high on the core dimensions are more motivated and satisfied than are people who work on jobs that score low on the dimensions. The same is true for measures of actual behaviour at work (absenteeism and performance effectiveness).

II. Responses to jobs high in motivating potential are more positive for people who have strong growth needs than for people with weak needs for growth".

III. "Both groups of employees show increases in internal motivation as MPS increases, but the rate of increase is significantly greater for the group of employees who have strong needs for growth".

Hackman is generally careful regarding his empirical results, and states that

> *"It has been assumed throughout the above discussion that job characteristics actually cause the differences in employee satisfaction, motivation, performance, and absenteeism which were observed. Although the predictions which were made (and confirmed by the data) were based on a conceptual framework which includes causal propositions, the study design was correlational and at no point were the casual links in the theory directly tested." (HACKMAN and LAWLER III, 1971, p. 280).*

On one hand, the theory holds its strengths. It is grounded in a basic psychological theory of what motivates people in their work; it emphasizes that planning for job changes should be done on the basis of data about the jobs and the people who do them; and it provides an instrument, which besides being easy to use, it makes possible for the accumulation a solid body of normative data. The JCT theory aroused a large number of empirical studies. Its instruments were adapted for use with several special populations, such as students and teachers, and translated into numerous languages (OLDHAM and HACKMAN, 2005).

On the other hand, some questions remain unresolved in the JCT, such as the role of the individual different moderators in Job Characteristics Theory, and a circular reasoning argument that sustains that the core dimensions cause internal motivation to those people who can be motivated by the core dimensions. Empirical findings also show that the MPS calculation suffer from psychometric problems, and lacks predictive power (OLDHAM and HACKMAN, 2010).

2.4 Summary of this chapter

Chapter 2 presented the theories and concepts that underlie the remaining of this work. First, in Section 2.1 (p. 21), we discussed that although the terms "motivation" and "satisfaction" have roots in common studies, they have been evolved and discerned over time. Currently, there is a consensus that motivation and satisfaction refer to distinguisable, although connected, phenomena. Basically, considering a single isolated action, motivation occurs before the action, while satisfaction occurs after it. However, in fact, considering that the human life is a continuous composed by series of actions, both concepts are mutually reinforcing, because past satisfaction shapes people's perception about experiences and the world around them, which consequently affects their future motivation.

Technical literature on organisational behaviour generally agree that the most adequate definition for the "satisfaction" concept comes from Locke (1975), which states that "Job satisfaction is the pleasurable emotional state resulting from the appraisal of one's job as achieving or facilitating the achievement of one's job values." The Job Satisfaction Theory was described in details in Section 2.2 (p. 25).

However, there is still no consensus aroubd the "motivation" concept. Theories in this field focus on very specific dimensions of the life continuous. While some of the motivation theories presented in this chapter focus on motivation from a general decision-making process approach, that guide the rational choice of a determined behaviour, other theories are rather interested in describing what aspects of the workplace can make people more or less turned on to work. All these theories are equally limited by the approach they choose to look at the motivation phenomenon. In this research, we are specifically interested in the software engineers' activity. Given that individuals motivated to work will perform at their possible best, we set out to investigate what elements of a software engineering workplace turn on these individuals to work. Thus, the job characteristics approach was chosen to underpin this research work. The Job Characteristics Theory was detailed in Section 2.3 (p. 28).

Next Chapter details the current state of art, describing what has been investigated and learned about software engineers' satisfaction and motivation so far. That chapter will also present further arguments on the relevance and validity of the research problem in which we are interested.

Chapter 3 Motivation and Satisfaction in Software Engineering Research

The influence of general human aspects on individual and collective performance of software engineers has been recognized since the early days of software engineering (WEINBERG, 1971)(BROOKS, 1975)(DEMARCO and LISTER, 1987) (CURTIS, HEFLEY and MILLER, 2001).

In the software engineering field, a seminal work of Couger and Zawacki (1980) has brought light to the issue that computer personnel may be a distinctive group from the average population, regarding individual needs and, for this reason, what motivates software engineers is likely to be different from what motivates the population in general. Since then, researchers have addressed the problem of how to deal with the motivation of software engineers in several complementary manners, among more than a hundred published articles (BEECHAM, BADOO, *et al.*, 2007) (FRANÇA, GOUVEIA, *et al.*, 2011).

Based on the results of a comprehensive literature review, Sharp *et al* .(2008) proposed the MOCC model, which is the most relevant recent advance about the motivation of software engineers. Section 3.1 (p. 36) discuss the state of art of motivation in software engineering, and details the components of this model.

Nevertheless, the MOCC model combines different concepts of motivation and job satisfaction in a single synthesis, and puts together, in an abstract and superficial manner, results from research that may have been interested in one, both or none of these concepts. In fact, the general conclusions of the technical literature reviews point out that the complex relationship between the workplace factors and software engineers' work motivation is still unclear. Section 3.2 (p. 40) reviews the empirical evidence found in the literature studies, and discuss the research gaps in the current state of art.

Finally, in Section 3.3 (p.47), we discuss on how the more recent research works have not been effective in closing the gaps that we identified in the literature.

3.1 Motivation and Satisfaction in Software Engineering: state of art

Couger and Zawacki (1980) carried out a nationwide survey in the United States, interested in mapping the level of motivation of individuals in the computer field. Their survey was mainly based on the Job Characteristics Theory (HACKMAN and OLDHAM, 1976), using the Job Diagnostic Survey (HACKMAN and OLDHAM, 1975). They claimed that, given the particular characteristics of the computing personnel, specific strategies of job design could be proposed for this type of professional, to increase their job's motivational potential and consequently their performance at work (COUGER and ZAWACKI, 1980).

Their survey was replicated over a decade in several countries, such as Austria (COUGER and ADELSBERGER, 1988), Israel and Singapore (COUGER, BOROVITS and & ZVIRAN, 1989), Australia (DENGATE, COUGER and WEBER, 1990), Hong Kong (BURN, COUGER, *et al.*, 1991), Finland (COUGER, HALTTUNEN and LYYTINEN, 1991), Spain (COUGER and CALLAGHAN, 1994), Japan (COUGER and ISHIKAWA, 1995) and Egypt (KHALIL, ZAWACKI, *et al.*, 1997). After all, they accumulated enough data to argue that:

(i) Software engineering settings generally suffered from a low level of feedback, which was later explained by the fact that software engineers have specific personality traits that limit their social interactions (COUGER, BOROVITS and & ZVIRAN, 1989)(TOMPKINS and COUGER, 1991)(COUGER, OPPERMANN and AMOROSO, 1994);

(ii) Regardless of the cultural differences, populations of software engineers from all over the world exhibited similarities regarding their high growth needs (GNS)(ZAWACKI, 1992);

(iii) Although the job characteristics had some influence over the software engineers' motivation, other work-place factors should also be accounted in the equation, such as goal contents (LOCKE, 1968). However, they do not provide details on why or how other theories would add to the study of software engineers' work motivation.

After Couger and Zawacki's study, motivation kept being systematically studied in software engineering. Two recent literature reviews (BEECHAM, BADOO, *et al.*,

2007)(FRANÇA, GOUVEIA, *et al.*, 2011) searched relevant sources in software engineering, and systematically selected 140 studies about motivation and job satisfaction published in this field between 1980 and 2010, looking for answers to five research questions:

(1) What are the characteristics of Software Engineers? (Table 2)
(2) What (de)motivates Software Engineers to be more (less) productive? (Table 3 and Table 4)
(3) What are the external signs or outcomes of (de)motivated Software Engineers? (Table 5)
(4) What aspects of Software Engineering (de)motivate Software Engineers? (Table 6 and Table 7)
(5) What models of motivation exist in Software Engineering? (Table 8).

The answers are listed in Tables 2-8.

Table 2 - Characteristics of software engineers (FRANÇA, GOUVEIA, *et al.*, 2011)

Ch. 1: Need for stability
Ch. 2: Technically competent
Ch. 3: Achievement orientated
Ch. 4: Growth orientated
Ch. 5: Need for competent supervising
Ch. 6: Introverted
Ch. 7: Need for involvement in personal goal setting
Ch. 8: Need for feedback
Ch. 9: Need for Geographic stability
Ch. 10: Need to make a contribution
Ch. 11: Autonomous
Ch. 12: Need for variety
Ch. 13: Marketable
Ch. 14: Need for challenge
Ch. 15: Creative
Ch. 16: Need to be sociable
Ch. 17: Competent in Management
Ch. 18: Flexible / Team worker (easy to work with)
Ch. 19: Have fear of punishment

Table 3 – Motivators for software engineers (FRANÇA, GOUVEIA, *et al.*, 2011)

M. 1: Rewards and incentives
M. 2: Development needs addressed
M. 3: Variety of work
M. 4: Career path
M. 5: Empowerment/responsibility
M. 6: Good management
M. 7: Sense of belonging
M. 8: Work/life balance
M. 9: Working in successful company
M. 10: Employee participation
M. 11: Feedback
M. 12: Recognition
M. 13: Equity
M. 14: Trust/respect
M. 15: Technically challenging work
M. 16: Job security/stable environment
M. 17: Identify with the task
M. 18: Autonomy
M. 19: Appropriate working conditions
M. 20: Making a contribution/task significance
M. 21: Sufficient resources
M. 22: Team quality
M. 23: Creativity/Innovation
M. 24: Fun (playing)
M. 25: Professionalism (high professional environment)
M. 26: Having an Ideology
M. 27: Non-financial benefits (availability of perks)
M. 28: Penalty Policies
M. 29: Good relationship with users/customers

Table 4 – Demotivators for software engineers(FRANÇA, GOUVEIA, *et al.*, 2011)

D. 1: Risk
D. 2: Stress
D. 3: Inequity
D. 4: Interesting work going to other parties
D. 5: Unfair reward system
D. 6: Lack of promotion
D. 7: Poor communication
D. 8: Uncompetitive pay
D. 9: Unrealistic goals
D. 10: Bad relationship with users and colleagues
D. 11: Poor working environment
D. 12: Poor management
D. 13: Producing poor quality software (no sense of accomplishment)
D. 14: Poor cultural fit/stereotyping/role ambiguity
D. 15: Lack of influence/not involved in decision making/no voice
D. 16: Task Complexity (too easy or too difficult)

Table 5 – External signs of (de)motivated software engineers(FRANÇA, GOUVEIA, *et al.*, 2011)

Ext. 1: Retention
Ext. 2: Project Delivery Time
Ext. 3: Productivity
Ext. 4: Budgets
Ext. 5: Absenteeism
Ext. 6: Project success
Ext. 7: Organizational Commitment
Ext. 8: Benevolence

Table 6 – Motivating aspects of software engineering(FRANÇA, GOUVEIA, *et al.*, 2011)

Asp 1: Problem solving
Asp 2: Team working
Asp 3: Change
Asp 4: Challenge
Asp 5: Benefit
Asp 6: Science
Asp 7: Experiment
Asp 8: Development practices
Asp 9: Lifecycle
Asp. 10: Creativity
Asp. 11: Relationships with users/customers

Table 7 – De-motivating aspects of software engineering(FRANÇA, GOUVEIA, *et al.*, 2011)

De-asp. 1: Software process/lifecycle – maintenance
De-asp. 2: Boredom (repetitive tasks)

Table 8 – Models of motivation in Software Engineering(FRANÇA, GOUVEIA, *et al.*, 2011)

Explicit models of motivation
Mod. 1: Job Characteristics Theory Model
Mod. 2: Models of leadership influence on SE motivation
Mod. 3: Models of open source developer SE Motivation
Mod. 4: Model of task design influence on SE motivation
Mod. 5: Model of career progression influence SE on motivation
Implicit Models of motivation
Rel. 1: Models focusing on Software Engineer job satisfaction
Rel. 2: Models drawing on expectancy theory, goal-setting theory, and organizational behaviour specific to the software development process
Rel 3: Social support influence on Software Engineer turnover

Based on the results of the first literature review, Sharp *et al.* (2008) noticed that the pre-existing models of motivation in software engineering were being developed in isolation. Then, Sharp *et al.* (2008) delivered a proposal for an integrating model of motivation in software engineering, consistent with the findings of their systematic literature review and with other pre-existing models of motivation. The MOCC model (acronym for Motivators, Outcomes, Characteristics and Context) stands as an abstract, holistic model that enables researchers and practitioners to have a better under-standing of the landscape of motivation, and provides a coherent framework for integrating research findings(SHARP, BADOO, *et al.*, 2008).

The MOCC model (Figure 3) describes motivation in software engineering through the relationships among its four components:

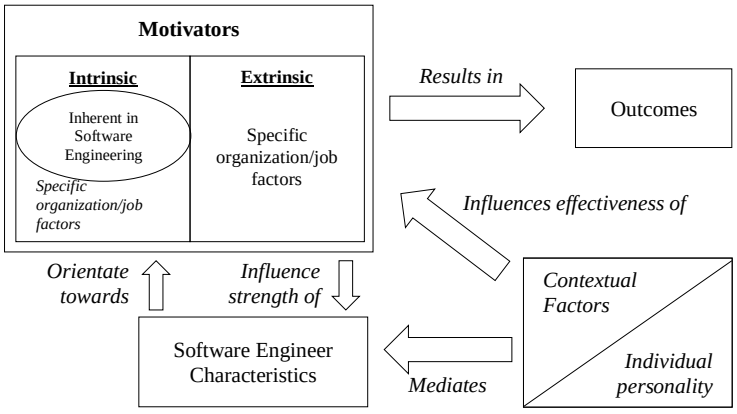


Figure 3 - The MOCC Model (SHARP, BADOO, *et al.*, 2008)

- **Motivators**: Sharp *et al.* (2008)classify the factors that motivates and demotivates software engineers in: intrinsic, which include those factors that come from the pleasure of doing the work itself (all factors inherent to software engineering were considered intrinsic); and extrinsic, related to factors external to the job, such as working conditions.
- **Outcomes**: Refers to the results caused by motivated individuals, which according to the reviewed literature might be improvements in retention, productivity, project delivery time, adherence to budgets, low absenteeism, and project success.
- **Software engineer characteristics**: The MOCC model suggests the software engineers' characteristics as one of the pillars of the motivation model. Although some previous research cast doubt on the existence of significant differences between individual characteristics of software engineers and non-software engineers, many studies retrieved in the literature review reported that specific characteristics of software engineers affect their motivation.
- **Contextual factors**: Research confirmed that some of the software engineer characteristics are indeed influenced by individual and contextual factors, most specifically personality and the environment in which they are practicing.

As shown in Figure 3, the software engineer characteristics are mediated by contextual factors and individuals' personality. The software engineering characteristics orientate towards the motivators. The motivators, which influence the strength of software engineers' characteristics, have, on the other hand, its effectiveness influenced by the contextual factors and individuals' personality.

3.2 A critical review of the available evidence

The systematic reviews and the MOCC model present a reasonable overview of factors that have been studied in this area. Overall, this work uncovered the following three issues:

Issue I. There is an increasing interest in this subject, but empirical studies are still concentrated in the developed world

The number of studies on this subject has consistently increased, as shown in Figure 4: 19 studies in the 1980's, 34 studies in the 1990's, and 87 studies in the 2000's. Research focusing on emerging contexts such as agile methods and open source represent an important portion of the studies in the 2000's (21/87); The majority of studies present empirical data with actual practitioners (103/140), which may be an indicator of how supportive and interested the software engineering industry is on this topic.

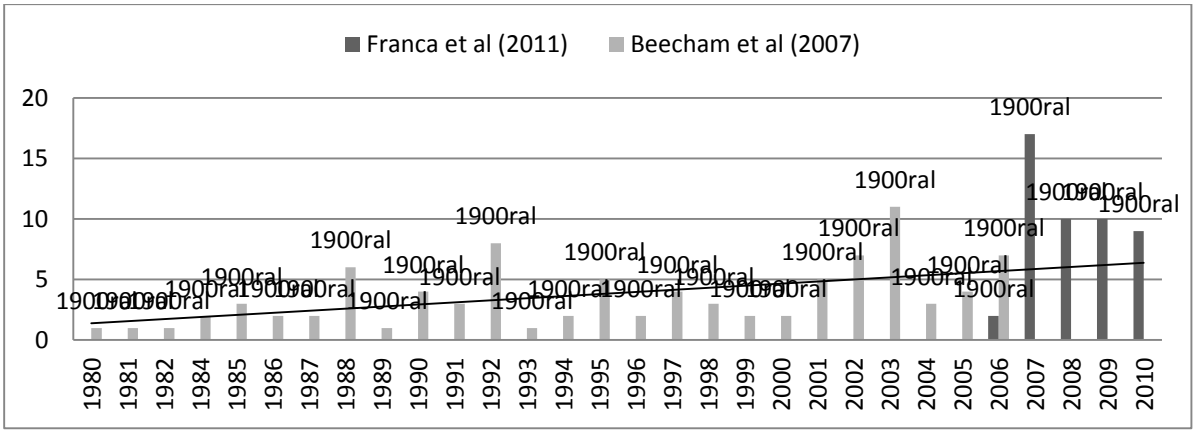


Figure 4 - Number of articles published per year

The studies have been authored by 245 researchers from 145 institutions in 30 different countries, the majority of which (206/245 researchers and 129/145 institutions) are from high HDI countries (21/30). Regarding the researched subjects, there are studies looking at the North America (57/140), Asia (27/140), Europe (14/140), Africa (9/140), Middle East (5/140), Oceania (4/140), South America (5/140), and Central America and Caribbean (1/140). The remaining studies are theoretical or do not specify the origin of their participants. These numbers show that there may be a cultural bias in the overall results, which represent a risk for drawing generalizations as pointed out in Hofstede (1980).

Issue II. There is a gap on theoretical underpinning, which obstacles the accumulation of knowledge

The first problem is related to the interpretation of the term "motivation" in these studies. In order to be able to integrate different studies, learn from the differences, and advance our knowledge on this issue, we have to make sure that research efforts at least address the same phenomena. However, unfortunately, there is no singularity in the notion of work motivation in software engineering research. From the 67 empirical papers from both SLRs that contributed to the list of software engineers' motivators (Table 3), only 35 effectively show evidence that support their claims (Table 9 – Groups A to C), while the others only mention something about motivation or satisfaction, referring to other studies or to general knowledge, without much responsibility on the factual truth. It is also possible to notice that these articles deal with completely different objects of study (Table 9).

Table 9 - Typology of studies interested in Work Motivation and Job Satisfaction

Group	Type	Instances*
Group A: empirical studies explicitly interested in job satisfaction	Type A1: papers focused on antecedents of job satisfaction.	PS005, PS007, PS016, PS065, PS102, PS105, PS116, PS120, PS134
	Type A2: papers focusing on intention to leave/stay in an organisation, as outcomes of job satisfaction.	PS007, PS014, PS036, PS037, PS045, PS050, PS052, PS066,PS076, PS086, P095, PS097, PS120
Group B: empirical studies that treat motivation as a decision-making process	Type B1: papers focused on reasons for choosing IT as a career	PS123, PS124
	Type B2: papers focused on reasons for developing open source software	PS090, PS113, PS115, PS118, PS119, PS140
	Type B3: papers focused on reasons for choosing an open source software to work for	PS113, PS115, PS136
	Type B4: papers focused on reasons for doing a specific task (e.g. refactoring)	PS131
Group C: empirical studies interested in work motivation	Type C1: papers focused on the antecedents of work motivation	PS016, PS056, PS100
	Type C2: papers focused on assumed outcomes of work motivation (performance, productivity, proactive behaviour)	PS005, PS091, PS099,PS101
Group D – theoretical accounts	Type D1: theoretical papers focused on work motivation of software engineers	PS001, PS006, PS024, PS029, PS033, PS034, PS043, PS046, PS058, PS075, PS107, PS112, PS129

**see the list of SLR primary studies (PS) on page159*

Table 10 presents a summary of the thirteen Type D1 papers. From this list, we identify three different theoretical approaches. In the first, there are the *pioneer* papers, i.e. papers limited to the re-interpretation of theories from other fields to the software engineering field, without adding or testing any adaptation (PS001, PS006, PS033, PS046, PS075, PS107). These papers generically argue about the importance of creating strategies to cope with the motivation of software engineers, and reinforce the importance that managers have on determining the effectiveness of the motivational strategies. These papers are usually outdated, with exception of PS075 and PS107. PS107, in particular, induces the

Job Characteristics Theory to the context of Open Source projects, which explains why it is relatively recent. There is a paradox in the older papers: those based on the idea that software engineers are different from other professionals rely on theories developed on other fields to underpin their recommendations about how to deal with software engineers' motivation.

Following the second approach, there are the *rhetoric and position papers* (PS024, PS029, PS043, PS058, PS129). These articles comprise argumentative structures aiming to defend individual opinions of the authors, or to propose improvements on existing models and theories without an empirical support. PS024 brings about the issue that the nature of the software engineering profession was evolving, and facing significant changes at that time, challenging a stable view of the work motivation phenomenon for software engineers. PS029 sheds light on the individual-cultural fit and misfit, which represents an alternative approach to the study of problems related to productivity and turnover. PS043 points out that human aspects may influence the quality of software products. PS058 suggest a taxonomy of HR strategies based on its concern to individuals' needs. PS129 reviews the role of individual personality in the MOCC model.

The third approach comprises the *problem-solvin*g papers. PS034 suggests a complex model to predict organisational commitment and turnover. Job satisfaction and motivation appear embedded among the other ten variables and 23 relationships of the model, which was not tested after all. PS112 proposes an objective framework to help stakeholders to identify lists of motivational factors of IT workers, and filter them according to their importance. This framework, in particular, does not help to distinguish job satisfaction from work motivation factors.

The single characteristic that all these theoretical papers have in common is that they (1) recognise the importance of motivation of software engineers to support the success of software projects, and (2) call for clarifications about the antecedents of motivation and about the mediating role of individual characteristics between work motivation and performance.

A significant part of the SLR studies (40/140) does not explicitly mention any theory of motivation or satisfaction. It does not mean that the remaining portion of studies have properly used a theory. The Job Characteristics Theory, for example, is referred in 55 papers, but 31 of which have only mentioned the JCT without using it, 17 of which were

guided by this theory to measure and describe the job characteristics in some specific context. Only seven papers effectively used the theory to either test or discuss the empirical findings. Thus, as Hall *et al.* (2009, p. 10:25) concluded: "studies of motivation in software engineering (…) should be more rigorously based on existing theory."

Table 10 – Theoretical essays addressing work motivation of software engineers

Approach	Papers* (Year)	Overview of the Study
Pioneer papers	PS001 (1980)	Based on the distinction between internal and external motivation, this paper reviews motivation theories and empirical studies conducted in the software engineering field. It concludes that programming managers must (1) establish and encourage an atmosphere in which people are motivated by the work itself, an atmosphere that values achievement and challenge; (2) take care of people's level of satisfaction; (3) identify and encourage all opportunities for growth; and (4) build a sense of responsibility in the programming staff.
	PS006 (1985)	Reviews classic theories of motivation, and draws isolated recommendations based on each individual theory. It concludes that supervisors have a great deal of influence on the general morale and attitudes of his subordinates.
	PS033 (1992)	Presents the idea of Social Needs Strength, as a follow up from his original Couger and Zawacki's study.
	PS046 (1997)	Based on his own experiences, the author argues that the vast majority of problems encountered while developing software are more people oriented rather than technology based. The author, then, elaborates solutions for common people-related problems that a software production unit may face. He also reinforces the importance of the management to assure a motivating environment for software engineers.
	PS075 (2003)	Based on the assumption that IT professionals are distinct from professionals from other occupations, this paper catalogs motivational techniques drawn from classic motivation theories and studies from other fields.
	PS107 (2007)	Based on the Job Characteristics Theory, this paper argues that the precepts of the job redesign theory can also be applied to OSS projects.
Rhetoric and Position papers	PS024 (1991)	Discuss on the contradiction between researchers that argue that IS people differ from non-IS people and researchers that claim to prove the opposite. It argues that both are valid scientific observations of a phenomenon, each of which shapes a piece of a IS motivation. Then, it concludes that this picture is a moving target, because the profession is rapidly evolving and maturing. It also pointed out that, at the time, there were many opinions on what motivates IS people, but there was very little research evidence.
	PS029 (1992)	Discusses on the usefulness of the concept of cultural misfit between a corporate culture and the characteristics of IS personnel as an alternative approach to address the problems of stagnant productivity and high turnover rates.
	PS043 (1996)	This paper raises questions about software quality management and motivation of software engineers, in order to establish a research agenda.
	PS058 (2001)	In this paper, the authors suggest a taxonomy of HR strategies. It classifies different strategies based on the level of concern for the individuals' needs.
	PS129 (2009)	This paper draws on personality theories to create a model of individual motivation based on the MOCC model. It basically argues that contextual factors affect software engineers' extrinsic motivation, while personality factors affect their intrinsic motivation.
Problem-solving approach	PS034 (1992)	Suggests a multi-variate model to predict organisational commitment and turnover of IS professionals, as an attempt to integrate a variety of organisational theories.
	PS112 (2008)	This paper presents a self-formulated framework, called 'Imperative Motivational Factors Framework' whose the aim is to assist stakeholders to identify core motivational factors of IT workers. The initial list of motivational factors is based on classic motivation theories and literature review, and then they provide a working example in the Pakistan industry.

**see the list of SLR primary studies (PS) on page 159*

Issue III. There is enough knowledge about job satisfaction factors, but not on specific characteristics of the work that motivate software engineers

Roznowski and Hulin commented in 1992:

> *"Job satisfaction... has been around in scientific psychology for so long that it gets treated by some researchers as a comfortable 'old shoe'. (...) Many organizational researchers assume that we know all there is to know about job satisfaction" (ROZNOWSKI and HULIN, 1992, p. 124) .*

Indeed, the ten dimensions of job satisfaction studies pointed out by Locke (LOCKE, 1976)have not changed over time. In software engineering, the factors assessed by studies explicitly interested in antecedents of job satisfaction (Type A1, Table 9) fit perfectly in that classification (Table 11).

Table 11 - What is known about Job Satisfaction factors in software engineering (Type A1 papers)

Factors	What the literature says*
Characteristics of the work	Seven studies [P005, P016, P102, P105, P116, P120, P134] support the relationship between the characteristics of the work (autonomy, identity, variety, significance, and feedback) and job satisfaction of software developers. The relationship between autonomy and satisfaction is disputed in two articles [P102, P116], the relationship between task identity and job satisfaction was not supported in two [P005, P065]. P065 also found no support for the effects of task significance and feedback over job satisfaction.
Pay & Benefits	Three articles provide evidence showing that both a good salary [P016, P134] and a good variable remuneration [[P102] relates to the job satisfaction of software engineers.
Recognition	Two articles [PS016, PS102] support the relationship between recognition and job satisfaction of software engineers.
Promotion	Three papers [PS016, PS105, PS120] support the relationship between opportunities for promotion and job satisfaction. In only one study [P102] this relationship was not supported.
Working conditions	Only one paper [PS016] addressed the relationship between working conditions and job satisfaction, and found support for this relationship on the data. If the relationship between the developer and the users is included in this category, then there is another study [P102] that supports it.
Company	One study [P016] finds a positive relationship between job security and job satisfaction, while another study [P102] does not. In the second article, the authors asked the participants from what job aspects they get most satisfaction. Therefore, it is understandable that job security, being a hygienic factor, does not appear in their list.
Supervisors	Four studies [PS005, P065, PS120, PS134] support the relationship between the satisfaction with supervisory behaviour and job satisfaction.
Co-workers	Sense of belonging appears related to job satisfaction in one article [P120], while working with other people in a team appear related to job satisfaction in two other [P102, P134].
The self	*None*

**see the list of SLR primary studies (PS) on page 159*

The effects of *the self* seem to have been neglect by researchers in software engineering. Papers that assessed elements such as *self-esteem* or *self-confidence* focused only on comparing software engineers with other professionals (see P051, for instance).

More research is needed to clarify the effects of factors such as autonomy, task identity, and promotion over software engineers' job satisfaction. There may be other remaining drives to study, but, according to Maslow (MASLOW, 1943), it is unfruitful to make atomistic lists of drives, because they do not range themselves in an arithmetical sum of isolated, discrete members.

In contrast, there are only three papers focusing on the antecedents of work motivation for software engineers (Table 12), two of which (PS016, PS056) are not underpinned by any classical theory of motivation. PS016 presents survey data from 339 people from 11 high tech companies, and investigates factors associated to work excitement and job satisfaction. PS056 also presents survey data collected from 118 people from a North American company. PS100 presents a rich description and an analysis of the FreeBSD open source project.

Table 12 - What is known about work motivation factors in software engineering (Type C1papers)

Factors	What the literature says*
Skill variety: The degree to which a job requires a variety of different activities in carrying out the work, which involves the use of a number of different skills and talents of the person (HACKMAN, 1980).	In P100, skill variety contributes to motivation. P016 is not based on the JCT, but the "ability to make full use of skills" appears correlated with work excitement.
Task identity: The degree to which the job requires completion of a 'whole' and identifiable piece of work; that is, doing a job from beginning to end with a visible outcome (HACKMAN, 1980).	In P100, task identity contributes to motivation.
Task significance: The degree to which the job has a substantial impact on the lives or work of other people, whether in the immediate organisation or in the external environment (HACKMAN, 1980).	P016 is not based on the JCT, but relates "feeling useful" with work excitement.
Autonomy: The degree to which the job provides substantial freedom, independence, and discretion to the individual in scheduling the work and in determining the procedures to be used (HACKMAN, 1980).	In P100, autonomy is responsible for motivation.
Feedback: The degree to which carrying out the work activities required by the job results in the individual obtaining direct and clear information about the effectiveness of his or her performance (HACKMAN, 1980).	P016 is not based on the JCT, but relates the "awareness of making a real contribution to success of company" with work excitement.
Interesting work (atheoretical)	In P016, "interesting work" is related with work excitement.
Challenge (atheoretical)	In P016, a technically challenging work is associated to motivation.
Recognition (atheoretical)	In P016, "recognition for doing a good job" is related with work excitement.
Remuneration (atheoretical)	P016 relates fixed remuneration and benefits with motivation.
Career prospects (atheoretical)	Two studies associate career development to motivation [P016, P056]. In P016, "opportunity for advancement" is related to work excitement.

**see the list of SLR primary studies (PS) on page 159*

The results of the other four studies that assume outcomes, and use them as proxies to draw conclusions about work motivation (Type C2) are summarized in Table 13.

However, it is not actually possible to infer from these studies how those factors are responsible for the motivation of the software engineers. According to Maslow (MASLOW, 1943), the human behaviour is determined by a set of antecedents, of which motivation represents only one.

Table 13 - What is known about work motivation factors in software engineering (Type C2 papers)

Factors	What the literature says*
Skill variety: The degree to which a job requires a variety of different activities in carrying out the work, which involves the use of a number of different skills and talents of the person (HACKMAN, 1980).	One study [P005] finds no support for the relationship between skill variety and productivity.
Task identity: The degree to which the job requires completion of a 'whole' and identifiable piece of work; that is, doing a job from beginning to end with a visible outcome (HACKMAN, 1980).	In P005, task identity relates with productivity.
Task significance: The degree to which the job has a substantial impact on the lives or work of other people, whether in the immediate organisation or in the external environment (HACKMAN, 1980).	One study [P005] finds no support for the relationship between task significance and productivity.
Autonomy: The degree to which the job provides substantial freedom, independence, and discretion to the individual in scheduling the work and in determining the procedures to be used (HACKMAN, 1980).	Autonomy is related with productivity [P005] and performance [P101].
Feedback: The degree to which carrying out the work activities required by the job results in the individual obtaining direct and clear information about the effectiveness of his or her performance (HACKMAN, 1980).	Feedback is related with productivity [P005] and performance [P101].
Challenge (atheoretical)	One study [P091] finds challenge related with performance.
Participation (atheoretical)	One study [P005] finds participation related with productivity.
Learning (atheoretical)	One study [P101] finds learning related with performance.
Recognition (atheoretical)	One study [P091] finds recognition related with performance.
Remuneration (atheoretical)	One study [P091] finds remuneration related with performance.
Job Security (atheoretical)	One study [P091] finds job security related with performance.
Managerial support	Two studies [P005, P091], based on two different theories, associate managerial support to productivity and performance.
Self-efficacy	In one study [P099], self-efficacy is associated to proactivity.

**see the list of SLR primary studies (PS) on page 159*

Types C1 and C2 studies, together, present only weak evidence on the effect of the job characteristics on work motivation of software engineers, and wispy knowledge about this phenomenon. At the same time, they reveal other factors not covered in the JCT. However, they lack explanations about how or why these elements affect or relate to the work motivation of software engineers. Therefore, there is much more to learn about work motivation than about job satisfaction in software engineering.

3.3 Other recent developments

While this thesis was being developed, other researchers kept investigating issues related to work motivation and job satisfaction of software engineers all over the world. In this subsection, we review some of recent developments that became public after the time period covered in the systematic reviews (Table 14).

Table 14 - Recent studies on motivation and satisfaction of software engineers

Focus	References	Overview of the Study
Impacts of motivation in distributed software development environments	De Farias Junior*et al.*(2012)	This study presents an adhoc literature review on motivational factors that act in distributed software development. It filters factors from the MOCC that have been argued to be relevant in this context, and adds four new factors: setting standards, sharing leadership, attention to cultural differences and attention to individualities.
	Šteinberga(2012) and Šteinberga and Šmite(2013)	The authors carried out a case study on a Swedish software organisation and its offshore site in Russia, in 2011. They collected survey data from 16 of the 18 employees, and recorded semi-structured face-to-face interview with an employee who resigned the job. They found that the offshore employees were dissatisfied with their personal growth and development, and then delivered recommendations regarding how to deal with the limited variability of the work in the organisations, responsibility assignment, supportive relationships, and to enhance communication and reward strategy.
	El Khatib *et al.*(2013)	This study surveyed 206 professionals working in DSD projects in 49 different conuntries, aimed at testing a model of motivation in virtual teams, the VIST model (HERTEL, 2002). After a multiple regression analysis, they conclude that the model holds in a DSD context, and that individual motivation has significant predictive power for the success of the team.
Impact of motivation in agile software projects	Melo *et al.* (2012)	This paper presents a two-phase study: First, the authors filtered factors from our updated SLR (FRANÇA, GOUVEIA, *et al.*, 2011) that have been argued to be relevant in the context of agile teams. Then, they conducted amultiple case study in three organisations in Brazil, which lead them to suggest that motivation in the agile context is slightly different from the MOCC's general view of motivation in software development.
	Jansson(2013)	The author designed a SLR protocol specific for seeking for studies on motivation in agile projects, in order to evaluate their theoretical bases. It found only one study, based on the MOCC model, showing that very little is actually known on this subject.
Antecedents of satisfaction and motivation of software engineers	Hernández-López(2012)	Describes an exploratory case study, carried out in Spain,based on semi-structured interviews data from 14 software professionals. The result is a list of several items that improve job satisfaction and individual productivity.
	Da Silva and França (2012)	This study analised survey data collected from a semi-random sample of 176 software engineers from 20 software companies located in Recife-PE, Brazil. The results show the actual level of motivation for each motivator in the target population. Using principal component analysis on the set of all motivators, a five factor structure was identified and used to propose a guideline for the creation of motivational programs.
	Verner *et al.*(2014).	This study collected data from 312 practitioners from the USA, Australia, Chile and Vietnam. Their statistical analyses indicate that team motivation and overall project success, as perceived by the respondents, are significantly related in whole data (U=4961.5, p=0.000, N=298), but different national contexts mediate this relationship.
	Sach(2014)	The author investigates the impact of feedback on software engineers job satisfaction. A case study was conducted in a British company. Over a period of four weeks, 24 software engineers, using a diversity of agile practices, participated from observations, recorded interviews and diary studies, through several iterations of data colletion. The results evidenced that positive feedback reportedly influenced the job satistacion of software engineers. The study also identified several other characteristics of the feedback that influence the strength of this effect.
Extensions of the MOCC model to encompass individual characteristics	Rehman *et al.*(2011) and Rehman *et al.* (2011)	The authors measured personal characteristics of 80 Malaysian software professionals, and compared the most frequent characteristics with the number of studies found in the MOCC's base systematic review, to conclude that there is a slight difference between the importance of characteristics of Western and alaysian based software engineers.
	Asghar and Usman(2013)	This study surveyed 306 Pakistanis software engineers for the motivating characteristics of their jobs, and compared the results to the number of studies found in the two SLRs. They conclude that different national culture, as compared to western countries, influences the importance attributed to the main motivators at work.

These articles were selected according the following criteria: (I) it is not part of the work presented in this thesis, (II) it refers to one or more of our published work (see p. 172) according to the Google Scholar engine, (III) it reportedly focuses on motivation or satisfaction of software engineers, (IV) it was published after 2010 and before the submission of this thesis, and (V) it is written in English, which enables universal readability for the international software engineering research community.

These studies in general provide more evidence for two of the issues discussed in the previous subsection. First, new organisational challenges constantly faced by software engineering companies keep attracting research about motivation and satisfaction of software engineers. Second, the lack of theoretical and methodological concerns is evidenced through (i) the use of outdated theories underlying recent studies, (ii) the adoption of inappropriate research procedures and (iii) the lack of concern with what exactly motivation and satisfaction mean.

Work motivation and job satisfaction of software engineers in the context of agile methods, and more recently in the context of distributed software development (DSD) practice, are two common focal problems. Šteinberga (2012) and Šteinberga and Šmite (2013) are mainly concerned to employee turnover resulting from lack of job satisfaction of software engineers in the context of offshore projects, because of the additional complexity that globally distributed projects brings to managers. El Khatib et al. (2013), in contrast, focus on the subjective characteristics of the DSD practice that influence work motivation and, as a consequence, performance.

The agile manifesto (BECK, BEEDLE, *et al.*, 2001) suggests that successful projects must be built around motivated individuals, but the technical literature seems to have ignored this proposition for a long time. Melo *et al.* (2012)argue that motivation in the agile context is slightly different from the MOCC's general view of motivation in software development. It may alternatively mean that the MOCC is not very representative for software projects, or at least for the agile ones. Jansson (2013) evidences that more studies are needed to have a better and understanding of how motivation is influenced in agile contexts.

With respect to theoretical frameworks, De Farias Junior *et al.* (2012), Hernández-Lopez (2012), and Sach (2014) underpin their work with outdated theories, such as Maslow's Hierarchy of Needs and Herzberg's Motivation-Hygiene Theory. The use of these

theories is discouraged in the organisational behaviour field, partly because of their limited validity and partly because of the large deal of knowledge that has been developed after these theories were first delivered (MALIK and NAEEM, 2012).

Hernández-Lopez (2012), in particular, seeks for items that improve job satisfaction, and for items that improve productivity, separately, without explaining exactly the "why" of this duality. As a result, his list of job satisfaction items is largely compatible with the Job Satisfaction Theory, but the resulting list of productivity items displays sparse results containing some elements such as "more motivation". De Farias Junior *et al.*(2012) and Melo *et al* .(2012) provide lists of factors but are not concerned to explaining *why* those elements affect the motivation or satisfaction of software engineers in those specific contexts. These studies also do not seem to be concerned with the internal consistency with these lists, so several factors are ambiguous and overlapping.

Rehman and Mahmoud (2011) and Asghar and Usman (2013) are concerned to the transferability of the MOCC model, regarding the cultural specificities of their countries. However, they compare survey data collected from the field to the number of studies found in the MOCC's SLR, as if it reflected the *importance of motivators for western countries*. First, among the 79 empirical papers found in the MOCC's base SLR, 14 have collected data with subjects from eastern countries, while considering both SLRs, this number increases to 28 out of 118 empirical papers. Therefore, the motivators contained in the MOCC model do not reflect the reality of western countries only. Second, the paper count may only reflect a scientific bias caused by the prevailing adoption of the Job Characteristics Theory among software engineering studies, possibly influenced by the pioneer work of Couger and Zawacki, without necessarily reflecting cultural values of the studied subjects. Thus, although the problems addressed in those papers are reasonable, their analyses are not valid enough to support their conclusions.

Although Šteinberga (2012) and Šteinberga and Šmite (2013) explicitly adopted the Job Characteristics theory to underline their investigation, they proposed an adaptation of the JDS questionnaire. As we discussed in Chapter 2 , the original JDS has already been disregarded because of its psychometric limitations. Verner *et al.*(2014) conducted a survey in four countries aimed at exploring both the hypothetical relationship between team motivation and project success,as well as the hypothetical impact of national cultures in this relationship. However, their exploratory stance assumed a bottom-up perspective of "team

motivation", i.e. it is defined as whatever the practitioners think it is. They did not provide a top-down definition for the term, and did not underpin their work with any classic theory of motivation. Chen and Kanfer (2006) present a multilevel conceptualization of motivated behaviour in teams. According to these authors, team motivation is "the collective system by which team members coordinate the direction, intensity, and persistence of their efforts." (CHEN and KANFER, 2006, p. 233). It is a dynamic phenomenon that takes into account the mutual influences of individuals in the team, and the cross-level processes that include contextual influences of team motivation on individual motivation and emergent effects of individual motivation on team motivation. Given the complexity of this phenomenon, the study of team motivation is out of the scope of this thesis.

Da Silva and França (2012) and El Khatib *et al.* (2013), in contrast, orientate their work toward well proven theories. Da Silva and França (2012), in an attempt to consolidate the motivators of the MOCC model, proposed a higher level structure of five factors: *Use of competencies in SE*, *Power*, *Work/life balance*, *Career*, and *Actualization*. El Khatib *et al.* (2013) carried out an evaluation of a previously existing model, named VIST as an acronym for *Valence, Instrumentality, Self-efficacy and Trust*. The model predicted that these four elements would affect motivation, and that motivation would affect the team effectiveness of a virtual team. All the relationships suggested in the model were confirmed in that study. However, unfortunately, we had no access to the questionnaires used in that study, and the original VIST publication was in German, so we were not actually able of doing a more sensible evaluation of that work. Nevertheless, these two studies have roots on the Expectancy Theory (VROOM, 1964) and, as discussed in Chapter 2 , the Expectancy Theory has a very limited practical use in the study of work motivation and job satisfaction. It is more suited to studies that deal with decision-making processes. This issue limits the generality of both models.

On one hand, as discussed in the previous section, software engineering as a discipline keeps maturing, and unpredictable impacting changes are still likely to occur in years to come. Both academy and industry are calling for more focus on human aspects of software engineering research. On the other hand, we have not been able to effectively answer the research questions because the lack of an appropriate theoretical framework, tools, and concern with the scientific discipline. More than being a simple warning about all these problems, though, this thesis is an attempt to contribute to their solution.

3.4 Summary of this chapter

In this Chapter, we review thirty years of research on Satisfaction and motivation of software engineers. In Section 3.1 (p. 36), we summarized the most relevant advances made so far in this field. First, Couger and Zawacki (1980) pointed out that the Job Characteristics theory, as is, was not enought to explain the software engineers' behaviour. The systematic reviews conducted by Beecham et al. (2007) and updated by França, Gouveia et al (2011) synthesized lists of elements (concepts, factors, variables, etc) which have been studied in this area since the 80's, and have been related to the satisfaction and/or motivation of software engineers. Sharp et. al (2008) presented the MOCC model, which is an evidence-based model that describes motivation in software engineering through the relationships between software engineer characteristics, contextual and individual factors, motivators, and outcomes.

Then, in Section 3.2 (p. 40) we presented a deep review of the evidence reported in the 140 studies referenced in the systematic reviews. This section reveals three issues: (I) both industrial and academic interests on understanding what factors influence the satisfaction and motivation of software engineers are increasing, but empirical studies are still concentrated in the high HDI countries; (II) in the software engineering field, studies on this topic suffer from a general lack of theoretical rigour, which represents not only a threat to the validity of what these studies claim, but also a hindrance to the accumulation and advance of knowledge; and (III) the job satisfaction of software engineers is guided basically by the same factors of any professional from other areas, while our empirical knowledge on motivation of software engineers is actually very limited.

Finally, Section 3.3 (p. 47) illustrate that even more recent research developments keep using outdated theories, adopting arguable research procedures, and failing on identifying what phenomenon they are actually approaching. Nevertheless, recent studies reinforce the emergence of new trends in the software engineering practice that challenge our current knowledge on motivation of such type of worker.

Given the growing relevance of the problem, and based on the limitations of the state of art pointed out in this Chapter, in the next Chapter we present our methodological approach, designed therefore to advance and solidify our current knowledge in this area, and further to contribute to solve this problem.

Chapter 4 Methods

The research on motivation in software engineering has been largely dominated by quests for general results that would apply across a large number of different organisational contexts, technological contingencies, and types of individuals (BEECHAM, BADOO, *et al.*, 2007) (FRANÇA, GOUVEIA, *et al.*, 2011). Consistently, the preferred research method used has been survey research, with emphasis on quantitative analysis (84/140). Besides, the portion of these studies that comprehend relational studies (40/84) are not necessarily interested in explaining why the variables are related.

It is natural that objective research approaches adopt clear-cut simplifications of the phenomenon, ignoring relevant contextual elements, or the complex relationship that may exist among these variables. In a different direction, the present research is interested in understanding how individual software engineers interpret their experiences in the workplace, how these interpretations shape the meaning of motivation, and why certain combinations of workplace factors lead to more or less motivated behaviour.

This research calls, therefore, for a qualitative and interpretive research method. According to Denzin and Lincoln (DENZIN and LINCOLN, 2005) and Merriam (MERRIAM, 2009), qualitative researchers study things in their natural settings, attempting to make sense of, or interpret, phenomena in terms of the meanings people bring to them. Consistently with this intention of investigating individuals in their natural working settings, the case study method was selected as the main research procedure. Case studies are understood as ‘in-depth description and analysis of a bounded system’(MERRIAM, 2009). Four independent case studies were conducted, using a multi-case replication design (YIN, 2009), choosing cases of contrasting nature to increase the strength of the results.

This chapter aims to describe the general research strategy, in terms of research goals (Section 4.1 , p. 53), research design (Section4.2 , p.. 54), theoretical framework and the research focus (Section4.3 , p. 59), selection of the cases (Section 4.4 , p. 61), data collection procedures and tools (Section4.5 , p. 67), analysis procedures (Section4.6 , p. 70), theory building process (Section 4.7 , p. 75) and threats to validity and reliability (Section4.8 , p. 77).

4.1 Objective statement and Research question

The main goal of this research project was to generate a sensible and contemporaneous theory of how the work motivation and job satisfaction of software engineers are influenced by workplace factors, and how these phenomena influence their work-related behaviour. According to Marconi and Lakatos (2004), a theory is not a reckless speculation. Instead, it is a set of basic principles, which constitute an appropriate scientific mechanism to guide the search and the explanation of facts (MARCONI and LAKATOS, 2004, p. 100). A theory is useful because it helps to organise and narrow down the amplitude of the phenomena, it helps to predict new facts and relationships based on previously known facts and relationships, and it indicates facts that have not been convincingly explained. Given the current state of the research on motivation and satisfaction in software engineering pictured in Chapter 3 , we believe that a solid theory of work motivation and job satisfaction for software engineers would represent a relevant contribution at this moment.

However, why not taking one of the classic theories of motivation from the organisational behaviour as granted for software engineers? Although we have enough evidence to believe that the job satisfaction of software engineers are influenced by the same workplace aspecs that affect other professionals, there are reasons to believe that the classical theories of motivation may not function effectively in the software engineering context. Previous researchers have claimed that software engineers are significantly different from the overall population with respect to values that drive motivation and satisfaction (COUGER and ZAWACKI, 1980). This claim has been disputed over time (FERRATT and SHORT, 1986), and is still source of much discussion (WYNEKOOP and WALZ, 1998) (CAPRETZ, 2003). Nevertheless, theories of job redesign generically agree with the fact that the characteristics of the tasks are the core source of motivation at work, but the software engineering work *per se* holds a high knowledge-intensive nature, whichcharacterize a relatively new type of work not covered by antique theories(WALLGREN and HANSE, 2007).Furthermore, the available theories do not give any hints about what and how software engineering practices may influence the software engineers' motivation to work. Therefore, we set off this research towards the following problem:

4.2 Multi-case study design

The general purpose of this research is to pursue answers for a question that is, in its essence, exploratory. Exploratory questions are designed to gain deeper knowledge about some phenomenon, and discuss useful issues that help to clarify our understanding about that phenomenon (EASTERBROOK, SINGER, *et al.*, 2008). The phenomenon in question is the work motivation of software engineers. According to Easterbrook *et al.*(2008), Yin (2009) and Merriam (2009), the most suitable research methods for exploratory questions tend to be those that offer rich, qualitative data, such as case studies, because they help researchers to build tentative theories.

A case study is formally defined as "an empirical inquiry that investigates a contemporary phenomenon in depth and within its real life context, especially when the boundaries between the phenomenon and context are not clearly evident"(YIN, 2009, p. 18). According to Merriam (2009), case studies have special features that differ them from other methods: they focus on a particular situations, events, programs, or phenomena; they provide rich description of the phenomenon under study; they are more sensory than abstract; reported experiences are rooted in the context; and they illuminate the reader's understanding of the phenomenon with episodes that may be merged with their own previously existing experiences.

Case studies can be used to investigate complex social issues, consisting of multiple variables of potential importance in understanding the phenomenon (MERRIAM, 2009), as well as experiments and history. However, experiments purposively divorce a phenomenon from its context, while history cannot deal with contemporary events. Therefore, case studies are powerful methods to when it regards either contemporary events that cannot be controlled by the researcher (YIN, 2009), or situations where the context is expected to play an important role, so that the reductionism of controlled experiments would sound inappropriate (EASTERBROOK, SINGER, *et al.*, 2008).

Case studies have been used in a diversity of research fields, such as psychology, sociology, political science, anthropology, social work, business, education, nursing, community planning, and economics (YIN, 2009). In software engineering literature, case studies are often referred as a working example, in which the method definition does not seem to fit correctly (EASTERBROOK, SINGER, *et al.*, 2008) (RUNESON and HÖST, 2008).Nevertheless, software engineering is a multi-disciplinary field, crossing many social and technological boundaries. Runeson and Höst (2008) argue that:

> *"research on software engineering is to a large extent aimed at investigating how this development, operation, and maintenance are conducted by software engineers and other stakeholders under different conditions"(RUNESON and HÖST, 2008, p. 137).*

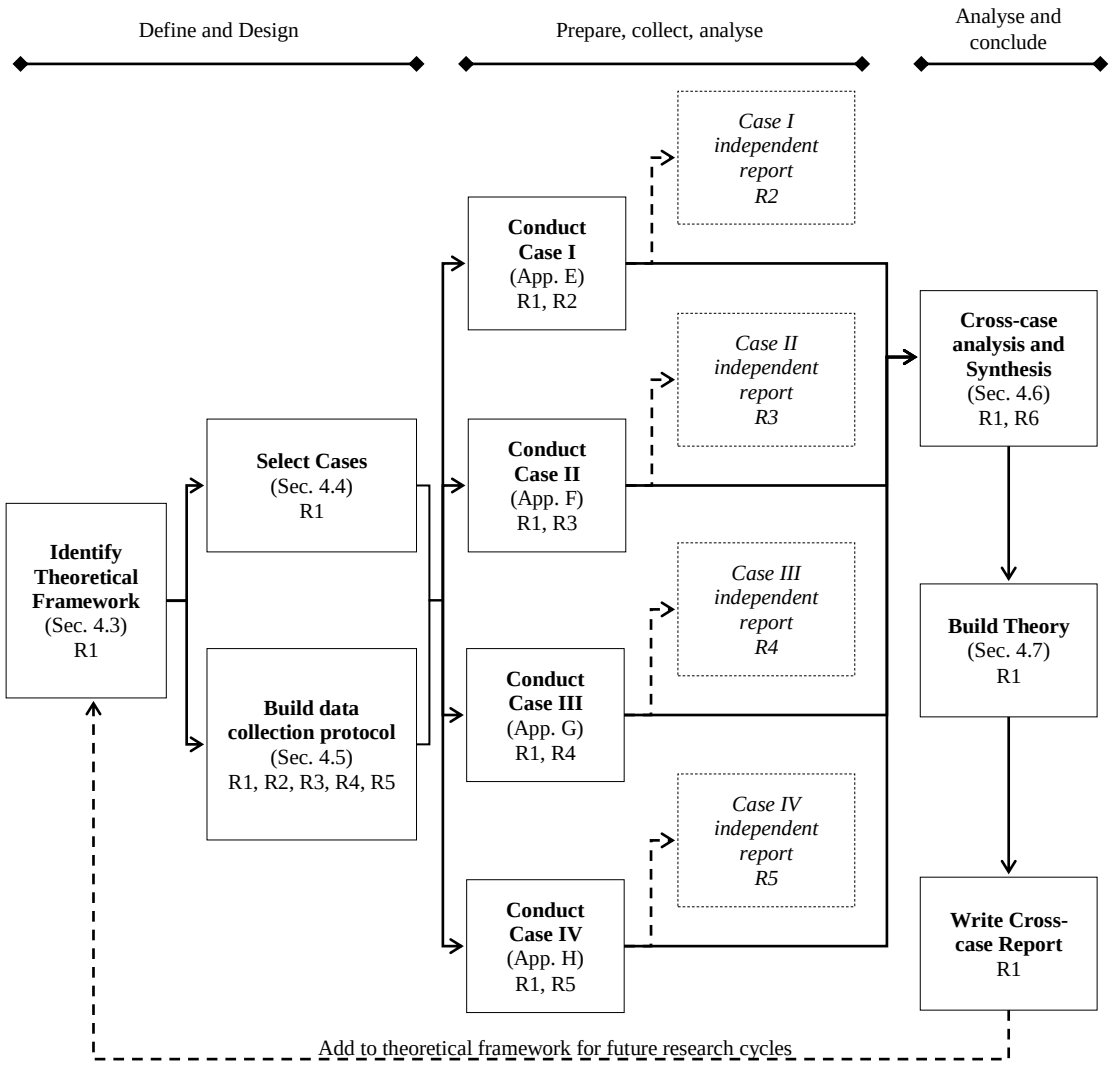
Thus, human and social activities should be investigated in their natural settings to achieve a better understanding ofpractical software engineering issues (EASTERBROOK, SINGER, *et al.*, 2008), taking in to account contextual elements that encompass the software development processes under study. Sjoberg and Dyba (2007) complements by claiming that:

> *"in SE [Software Engineering], case studies are particularly important for the industrial evaluation of SE methods and tools, because they can avoid the scale-up problems that are often associated with experiments."(SJOBERG, DYBA and JORGENSEN, 2007, p. 361).*

It is possible to find examples of case studies in software engineering that study different software development settings, such as Open Source Software development (MOCKUS, FIELDING and HERBSLEB, 2002), agile methods (MULLER and FASANENGARTEN, 2001) and software developed for specific purposes (CARVER, 2006), all of which providing real examples and lessons learned that could serve to improve general software engineering processes.

Runeson and Höst (2008) proposed guidelines for conducting and reporting case study research in software engineering. They recommended that case studies should be conducted in four iterative phases: (i) *design*, in which the objectives of the case study are clearly stated; (ii) *data collection*, in which the collection techniques and instruments are designed, the data sources defined, and the collection conducted in practice; (iii) *analysis*, in which the data is interrogated for patterns that answer the research questions; and (iv) *report*, that should include sufficient data to allow the reader to understand the chain of evidence. This research strictly followed these phases.

Case study research can be single and multiple-case studies, but both are actually variations of case study designs. We designed an instrumental multi-case research (YIN, 2009),using replication rationale, to be carried out in different software organisations.The evidence from multiple cases is more compelling and the overall results are more robust, because single-case studies may be limited by the uniqueness conditions surrounding the case(YIN, 2009). Then, four independent holistic case studies were performed, following a single standard protocol. Figure 5 pictures the overall research steps, and refers to the sections and chapters of this document where more details can be found for each step.



Sec. – Section,
App. – Appendix,
R – Researcher (see Table 15)

Figure 5 - Multi-case study design - adapted from Yin (2009, p. 57)

According to Yin (2009), multiple-case studies also have disadvantages in comparison to single-case designs. The conduct of a multiple-case study can require extensive resources. In order to be able to study these four cases, four additional researchers (master students) participated in the present research. The five master students participated in the design of the data collection protocol,each one also conducted an independent case study under the supervision of the PhD student, and reported the analysis of the case in their master's dissertation. Table 15 brings details about the participation of each researcher.

Table 15 - Researchers participation

Researcher	Task(s)*	Outcomes
R1 César França (*PhD student*)	• Identify Theoretical Framework • Select Cases • Build data collection protocol • Cross-case analysis and synthesis • Build theory • Write cross-case report	• Theoretical Framework (FRANÇA, GOUVEIA, *et al.*, 2011) • Research protocol (DA SILVA, FRANÇA, *et al.*, 2011) • Research papers (FRANÇA, FELIX and DA SILVA, 2012) (FRANÇA, CARNEIRO and DA SILVA, 2012) (FRANÇA, ARAÚJO and DA SILVA, 2013)(FRANÇA, DA SILVA, *et al.*, 2013) • Web diary system (Appendix D) • Cross-case report and theory building (PhD Thesis)
R2 Adelnei Felix (*MSc student*)	• Build data collection protocol • Conduct case study I • Write case I independent report	• Research protocol (DA SILVA, FRANÇA, *et al.*, 2011) • Master's dissertation (FELIX, 2011) • Research papers (FRANÇA, FELIX and DA SILVA, 2012)
R3 Ana Araújo (*MSc student*)	• Build data collection protocol • Conduct case study II • Write case II independent report	• Research protocol (DA SILVA, FRANÇA, *et al.*, 2011) • Master's dissertation (ARAÚJO, 2011) • Research paper (FRANÇA, ARAÚJO and DA SILVA, 2013)
R4 David Carneiro (*MSc student*)	• Build data collection protocol • Conduct case study III • Write case III independent report	• Research protocol (DA SILVA, FRANÇA, *et al.*, 2011) • Master's dissertation (CARNEIRO, 2011) • Research papers (FRANÇA, CARNEIRO and DA SILVA, 2012)
R5 Eric Sales (*MSc student*)	• Build data collection protocol • Conduct case study IV • Write case IV independent report	• Research protocol (DA SILVA, FRANÇA, *et al.*, 2011) • Master's dissertation (SALES, 2011)
R6 Danilo Monteiro (*MSc student*)	• Cross-case analysis	*None*

**see Figure 5*

According to Yin (2009), the main concerns against case study research have been over the lack of rigor of case study research. However, as Perry *et al.* (2004) argues, a case study: is not a toy example; is not an experience report; is not a quasi-experiment with n=1. It is important to reinforce that case studies are as systematic and rigorous as any other scientific method. Table 16 summarizes the most common misunderstandings about case study research.

Table 16 - Five common misunderstandings about case study research (MERRIAM, 2009, p. 53)*.

Misunderstanding	Restatement
1. General knowledge is more valuable than context-specific knowledge	Universals can't be found in the study of human affairs. Context-dependent knowledge is more valuable.
2. One can't generalize from a single case so a single case doesn't add to scientific development.	Formal generalization is overvalued as a source of scientific development; the force of a single example is underestimated.
3. The case study is most useful in the first phase of a research process; used for generating hypotheses.	The case study is useful for both generating and testing hypotheses but it is not limited to these activities.
4. The case study confirms the researcher's preconceived notions.	There is no greater bias in case study toward confirming preconceived notions than in other forms of research.
5. It is difficult to summarize case studies into general propositions and theories	Difficulty in summarizing case studies is due to properties of the reality studied, not the research method.

* Table built based on Flyvbjerg (2006)

Additionally, there are concerns regarding other two appropriate questions: (i) the case, *per se*, characterize the whole study, so that the selection has to be done very carefully (EASTERBROOK, SINGER, *et al.*, 2008) (MERRIAM, 2009); and (ii) the context has a single large impact on case studies conclusions. We address the former issue by using the maximum variation sampling approach, in order to purposively select cases that are relevant to the study, which is better explained in Section4.3 . The later one, we address by adding as much details as possible about the contexts in our research reports (MERRIAM, 2009), and also following the guidelines suggested in Petersen and Wohlin(PETERSEN and WOHLIN, 2009) for reporting details of the context in industrial empirical studies in software engineering.

4.3 Initial Theoretical Framework

The first step of our multi-case design is the statement of the initial theoretical framework. According to Yin (2009), the framework needs to state the conditions under which a particular phenomenon is likely to be found. The theoretical framework later becomes the vehicle for transferring the research results to other cases.

Our initial theoretical framework is, then, a combination of the Job Characteristics and Job Satisfaction theories. The Job Satisfaction Theory was chosen because it is referred to as the consensual definition of the term, in the organisational behaviour field (WEISS, 2002). Job Characteristics theory, on the other hand, was chosen for three main reasons. First, in the core of our argument is the fact that the nature of software engineering tasks are special, so a theory of motivation that focus on characteristics of the tasks would help us to appropriately evaluate this issue. Second, it evolved from a combination of theoretical and empirical studies, so its consistency has been partially supported over time. Main criticism to this theory focus on its measurement instruments, which we opted for not using in this research. Third, it is the theory of motivation most frequently referred in software engineering studies.Figure 6 presents a summarized model of the combination of these two theories, and more details are provided below.Hereafter, this initial model is referred to as the initial Theory of .Motivation and Satisfaction(TMS-i).

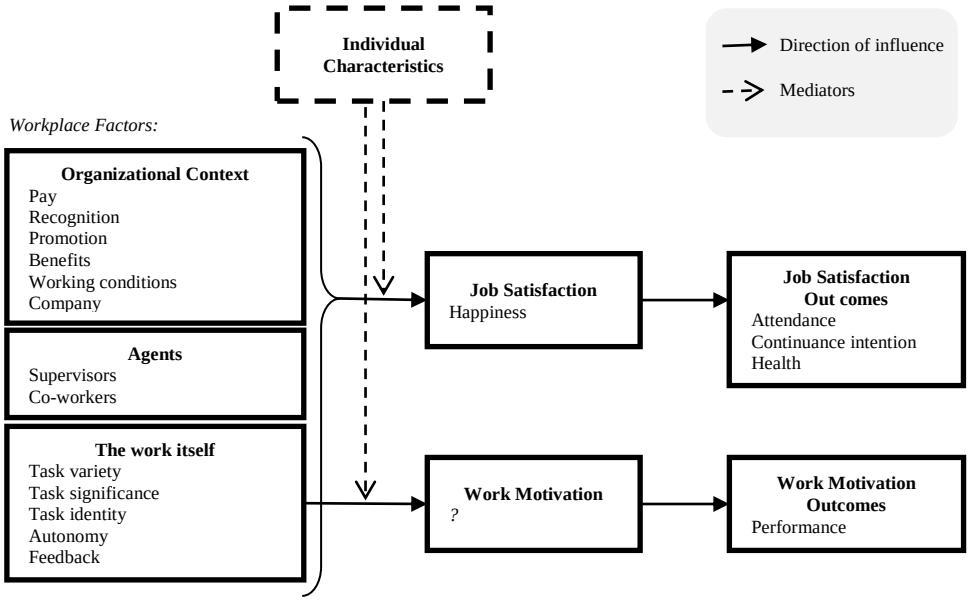


Figure 6–TSM-i: the initial frame of reference

According to the Job Satisfaction Theory, there are several known workplace factors that affect job satisfaction, which is signalled by the happiness of the individuals at work. Job satisfaction, in turn, affects employees' attendance, continuance intention, and health. Although it is not clear exactly how these workplace factors combine to determine job satisfaction, it is known that the individual subjective appraisal of these workplace factors against their own values and expectations is what accounts for their job satisfaction. Consequently, the adoption of a third-party objective assessment of workplace factors may be unsound to infer one's job satisfaction.

The JCT is compatible with this definition of job satisfaction, as the JCT defines it as "the degree to which the employee is happy with the job, or with specific aspects of the job." In addition, the JCT identifies characteristics of the work itself that are likely to foster job satisfaction and work motivation of high GNS individuals, which tends to be the case of software engineers (COUGER and ZAWACKI, 1980). Work motivation is understood as the desire to work, and is postulated to result in performance, which in turns affects individual productivity at work.

However, the JCT relies on the notion of 'internal work motivation', which refers to a set of internal feelings. We also found no relevant technical literature work about what exact behavioural traits constitute high and low motivated behaviour. Consequently, the external signs of a motivated behaviour remained incognito in the TMS-i. Then, we understood that these external signs should be documented in order to enable a data-driven investigation of the influence of workplace factors. Expliciting these external signs would also improve the traceability and, as a consequence, the credibility of the analysis of the empirical data. In practical terms, this theoretical framework guided us to break up the main research question in two more operational questions, as shown in Figure 7.

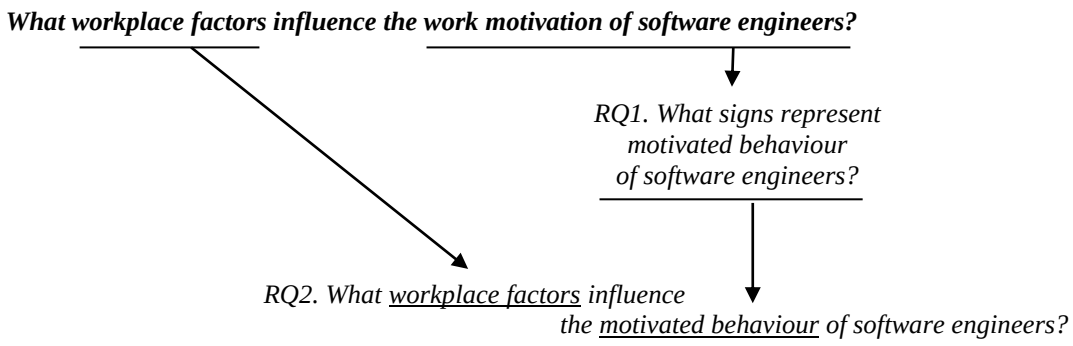


Figure 7 - Research Questions

4.4 Case Selection

Two levels of data points selection are necessary in multiple-case studies: the cases, and the participants. The former is the selection of the cases that will be investigated or, in other words, what constitute the bounded system of interest for the research. In the last, we must sample participants and other sources of data within each case.

In this research, the bounded system of interest for each case study is a software organisation. According to Yin (2009), the replication logic in multiple case-studies is analogous to that used in multiple experiments:

> *"Some of the replications might attempt to duplicate the exact conditions of the original experiment. Other replications might alter one or two experimental conditions considered uninportant to the original finding, to see whether the finding could still be duplicated. Only with such replications would the original finding be considered robust." (YIN, 2009, p. 54)*

Therefore, the cases must to be carefully selected in order to either (a) predict similar results or (b) predict contrasting results for anticipatable reasons. Following this rationale, we chose four distinct software organisations in the same business environment (although in different business sectors) and in similar social and educational contexts. We chose organisations with different business nature (public and private), different sizes and maturity, and distinct business goals, but kept similar overall context (social, cultural, economic, and educational) to facilitate comparison and integration of findings. In this way, we could achieve large variation regarding factors that are supposedly unimportant for the motivation of software engineers, according to the TMS-i. The four studied organisations are detailed as follow:

- **Case I – The government organisation**: The first case study was carried out in a government software organisation situated in Recife, Brazil, established in 1969 by Government of the State of Pernambuco. Its core mission is to provide Information Technology services to internal customers in several levels of the State Government administration and to the citizens of the State. As a government owned organisation, it is regulated under the laws and norms of

the Brazilian public sector, which have two characteristics that are relevant for this study. First, since the Brazilian Constitution of 1998, public employees must be hired through an open process with universal access, based on objective criteria. This rules out subjective interviews, personality and behavioural assessment, peer indication, and other forms of employee selection found in the private sector. On the other hand, it slows down the process of hiring new employees and, therefore, makes it difficult to produce timely replacement when someone leaves the organisation. Second, all public employees have job stability after a probation period of 3 years of work in the public sector (State Law Nº. 6.123/68). Currently, the organisation is structured in 14 main unities distributed in different locations throughout the State. Its employees, including software engineers, are distributed in the main unities and in over 60 other public administration buildings. At the time this research was performed, the organisation held 2,580 employees. Regarding software development methods and practices, it uses traditional, process-oriented methods, with command and control style of management in most software projects, although some small and isolated agile initiatives could also be found. The organisation explicitly stimulates the adoption of open source software in the State administration, and there is one open source project being currently developed.

- **Case II – Private not-for-profit organisation**: The second case study was carried out in a private and not-for-profit software development organisation, which has unities in three states of Brazil.The organisation's headquarters are located in the Porto Digital Science Park (Porto Digital, 2000), in Recife, Brazil. This organisation was created through the merging of two foundations, the first one created in 1994. It operates in many different areas, such as Information Technology, Telecommunications, Industrial Automation, Solutions for the Public Sector, and Energy, by providing support services, workforce supply for third-parties, development of software and hardware products, software factory, product certification tests, and research and development of technological innovative products. The organisation had a SW-CMMi level 2 certificate and was targeting the SW-CMMi level 3 at the time of the development of the case study. The management processes broadly followed the PMBOK guide, and managers were certified Project Management

Professionals (PMP), but some projects have already been adopting SCRUM agile management practices.At the time that this research was carried out, the organisation had about 300 professionals, 85% part of the technical workforce and 15% allocated in administrative tasks. This case study was limited to the Recife unity, with 40 professionals. This unity had both hardware and software development projects, but only software professionals were selected to participate in this research, which included people working on web, mobile and embedded systems, using technologies such as .NET and Java. In this unity, there was no specific human resource management, and project managers performed the activities related to human resources management.

- **Case III – Small software development company**: The third case study was carried out in a software company, formally established in 2006 by the initiative of five entrepreneurs from the Information Technology sector, in Recife, Brazil. Its core mission is to support the development of people and organisations with software tools, by means of technical excellence and innovation. This company is specialized in software development for different platforms, with expertise in different programming languages (such as .NET Framework, Java family, LUA, and others). It focuses on the on-demand development of information systems, operating in areas such as management, finance, mining, health, and others. In addition, it also develops its own products. Its flagship product, a corporative social network, stands for intra-organisational innovation management. Currently, it serves national and international customers, usually medium and big companies. Internal products and external projects significantly differ in terms of requirements management process and time pressure. People from both types of projects participated in this research. The company follows an agile-like software development process, broadly adopting practices such as regular delivery of software, adaptive management style (SCRUM based), small teams, face-to-face meetings, and customer authority. The organisational structure is flat, and the directors eventually act as part of the development teams. The directors themselves, who have software engineering background, instead of management, administer all organisational issues, including the human resource management. At the time that the case study was carried out, the company was composed of 27 people, everyone younger than 30 years

(directors included), occupying functions in one of the three types of teams: software development, research and design areas. Some of these people were in the organisation for less than six months, while others had more than 3 years along with the team. As an organisational strategy, the company is closely tied to the academy, both physically (its location is near a University) and operationally, since its staff is composed of undergraduate students (trainees) as well as graduated students in software engineering. We sampled participants representing all clusters.

- **Case Study IV – IT Department of a University**: The last case study was carried out in the Information Technology department of a federal university in Recife. The department is responsible for the maintenance and evolution of the software systems that holds all the valuable information of that organisation (such as academic and patrimonial info). Its core product was released in the early 2000's, and since then has continuously evolved and adapted. The product is a web-based system, written in Java, with about 840 functionalities, more than one million Lines of Code, and at the time this case study was carried out, its website received about four thousand hits per month. The department is mainly organized in three sectors: one responsible for the inception of new projects and products to improve the informational procedures in the university; another is exclusively responsible for the maintenance of the academic module of the system; and a third one is responsible for the elaboration and development of a new module. Regarding the software development process, this department follows an agile SCRUM-based approach. Internal procedures are defined and continuously improved by a study group, which aims to make these internal processes adherent to the MPS.br model(SOFTEX, 2007). The development process was already stable regarding the configuration management, project management, requirements management, portfolio management and quality assurance. Some initiatives were serving as pilot studies for procedures such as acquisition, measurement, validation and verification. Regarding the human management procedures, this department had 37 professionals, working under three different types of contracts: eighteen public employees, eleven third-parties workforce and eight internships. The former category is composed of government employees, and therefore have the same rights as described in Case I. Third-party workforce

are regular employees of another organisation that is responsible for supplying workforce for many departments in the university, so they have a regular private employment contract with the third-party organisation, but they are fully allocated in the studied department. Interns are contracted under a standard educational internship contract, with (supposedly) less responsibilities and less work time in the organisation. The data collection covered professionals with the three different types of job contract.

Apart from the facts that the four organisations are located in Recife, worked centrally on information systems development and maintenance, and adopted Java® among their programming languages portfolio, they have only very few other common characteristics (see Table 17for a summary).

Table 17 - summary of the characteristics of the studied organisations

Characteristics	Case I	Case II	Case III	Case IV
Established in	1969	1994	2006	2000
# of employees	2,580	300	27	37
Distribution	14 places	2 places	Co-located	Co-located
Certifications	None	CMMi 2	None	MPS.br-oriented
Social nature	Public	Private	Private	Public
End-product	Information Systems	Information Systems	Information Systems	Information Systems
Domains	Government Administration	Information Technology, Telecommunications, Industrial Automation, Government Administration, and Energy	Management, Finance, Mining, Health, and others.	University administration
Consumers	Government/Citizens	Private organisations	Private organisations	Public University
Selection process	Public selection	Ad-hoc	Ad-hoc	Public selection
Types of Contract	Public contract	Private contract (CLT) Educational contract	Private contract (CLT) Educational contract	Public contract Private contract (CLT) Educational contract
Technologies	Java	.NET, Java	.NET, Java, LUA	Java
Development processes	Traditional (RUP-like)	Traditional (RUP-like) Small SCRUM initiatives	SCRUM-like	Traditional (RUP-like) Small SCRUM initiatives

Specifically, with respect to the size and nature of the organisations, we developed case studies on two polar opposite organisations, as shown in Figure 8.

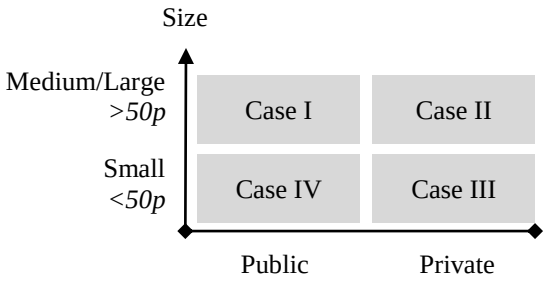


Figure 8 - Contrasting characteristics of the chosen organisations

Patterns that emerge based on great variation are likely to have more value in capturing the central perceptions of the phenomenon of interest (MERRIAM, 2009). Thus, in the second level of selection, we aimed for a good coverage of age, background, education, years of employment in the organisation, participation in different projects in the organisation, work on different activities in software development and maintenance, among other factors, to ensure a fertile sample. The participants from the four case studies clearly differed in terms of their work experience (Figure 9), as the more recent organisations (Cases III and IV) tended to have less experienced engineers.

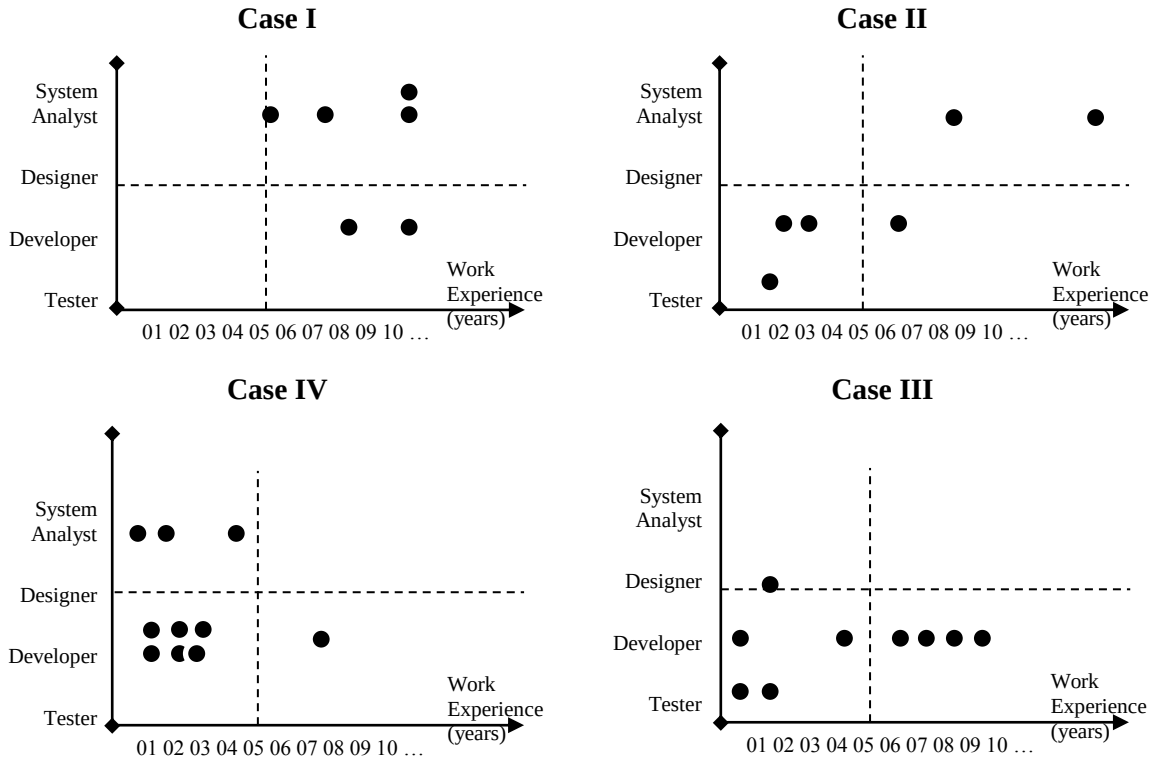


Figure 9 - Technical profile vs. work experience of participants

4.5 Data collection tools and procedures

Qualitative data are usually obtained using one or a combination of more than one technique. Several factors influence the choice of technique, including the nature of the phenomenon that is being studied. The previously discussed obstacles in observing both motivation and personal feelings and opinions led to the decision to not use observation in this study. We therefore used interviews, complemented by diary studies, and document analysis for data collection.

One important issue in qualitative research is the language in which the data is collected. In our cases, the native language of the participants was Brazilian Portuguese. We, thus, conducted all data collection in this language. This is consistent with obtaining data that is richer in opinions, feelings, and emotions that is easier to express in one's native language.

Interviews

According to Runeson and Höst (2008) and Merriam (2009), interviews are effective at eliciting information about things we cannot observe (such as feelings, thoughts, and intentions). We used semi-structured interviews, which is a type of interview in which a script, composed of open or less structured questions, is designed to guide the interviewer, but some flexibility on the sequence of the questions and their exact wording is allowed so that the interviewer can extract as much useful and rich information as possible.

A key aspect for the quality of the data is the design of the interview script. We interviewed software engineers using an interview guide composed of 43 open-ended questions (see Appendix C for the full script). The interview guide included questions aimed at exploring experience and behaviour, opinion and values, feelings, knowledge and background (Table 18). We also included demographic questions at the end of the script.

Our questions were presented in a funnel format, beginning with general questions and moving towards more specific ones (RUNESON and HÖST, 2008). All positive questions (e.g. 'what do you like about…?') had a corresponding negative one (e.g. 'what do you dislike about …?'). The interview guide was pre-tested with two pilot interviews. As a result, only minor changes were identified as necessary to improve the interview guides, such as better wording of some questions.

Table 18 - A typology of interview questions(MERRIAM, 2009, p. 96)

- **Background/demographic questions** = "questions that refer to the particular demographics of the person being interviewed as relevant to the research study"
- **Experience/behavior questions** = "gets at the things a person does or did, his or hers behaviors, actions, and activities"
- **Opinion and values questions** = "interested in a person's beliefs and opinions, what he or she thinks about something"
- **Feeling questions** = "these questions tap the affective dimension of human life (…) The interviewer is looking for adjective responses"
- **Sensory questions** = "these are similar to experience and behavior questions, but try to elicit more specific data about what is or was seen, heard, touched, and so forth"
- **Knowledge questions** = "these questions elicit a participant's actual factual knowledge about a situation"

As motivation is not an observable phenomenon, we used peer data to triangulate only factual information, or events, such as the characteristics of the organisation, software engineering processes, etc.

Diary Study

This is a method in which "participants are asked to record their daily activities on a pre-printed log form" (RIEMAN, 1993).It is a method of understanding participant behaviour and intent, *in situ*, which minimizes the effects of observers or interviewers on participants (SCOTT and MANKOFF, 2005).

Sohn *et al*(2008) highlights that interviews rely on participants' memories, and there is always the risk of the interviewee not remembering some relevant thing. Diary studies, being present right in the moment when relevant events happen, overcome this problem. Diary studies, then, can be used to complement and/or to triangulate data collected using other techniques. Our diary study data collection was designed to occur in three steps:

1. In the beginning of each week, participants received a blank notepad (Appendix D), in which they should annotate informations about any event that affected (positively or negatively) his/her motivation at the moment the event occurred (Table 19).
2. Every day, participants would fill out an on-line form (Appendix D), in which they would list all relevant events occurred in that day, and provide more

detailed information about how and why the events affected their work (Table 19).

3. At the end of each week during the study, the researchers carried out a retrospective interview to clarify and complement information submitted on the online form.

Table 19 - Information structure of diary events

Diary	*Question*	*Answer format*
Notepad	When?	Day/month/year
	What time?	Morning/afternoon/night
	What happened?	Short description of the event
	What was the effect?	Positive/neutral/negative
Web	What happened (in details)?	More detailed information about the event
	Why did it positively/negatively affected you?	More detailed information about the effects of the event

The main setback of diary studies is that it demands much more effort from the participants than other traditional methods of data collection, such as interviews or questionnaires. Therefore, resignation is a common problem in diary studies. Dearman, Kellar, and Truong (2008) suggest strategies to help avoid evasion of participants. First, the researcher must introduce the tools to the participants appropriately, so they will not miss important events just because they do not know how to fulfil the forms. Second, the participants must receive constant and encouraging feedback about their progress. Other complementary actions can be performed to assure that the participants will not forget the activity, such as sending short email reminders, and being in touch to assure that the participants have no doubt about how to proceed. In our case studies, we followed strictly these recommendations.

Document analysis

Documents are an important source of data for qualitative research because they are usually produced for reasons other than the research and therefore are not subject to the same limitations and biases. Further, the analysis of documents does not intrude on the daily activities of participants in the same way as interviews and diary studies do (MERRIAM, 2009). In this study, we mined documents related to human resources and norms that regulate employee-organisation relationships, among others. The main purpose was to perform triangulation, to verify and corroborate evidence raised in interviews and diary studies.

Data collection process

Potential participants were initially contacted by e-mail, and invited to participate. The interviews were scheduled and conducted individually, on the own organisation's premises. All interview sessions were audio recorded with the consent of the participants.Participants in the diary studies were chosen from those that participated in the interviews.

4.6 Data analysis procedures

The objective of qualitative analysis is to systematically consolidate, reduce, and interpret data obtained from various sources, and make sense of them. According to Merriam (2009), in qualitative research, data collection, analysis, and reporting often occur simultaneously, in incremental and iterative steps that are adapted as the research proceeds and results emerge. In this study, several iterations of data collection and analysis were performed, using the techniques described in this and the previous sections.

Individual case studies analysis

The specific aim of each individual case study was to answer the following question: *How do organisational, individual, and task related factors affect the motivation of software engineers in the workplace, and what are the perceived outcomes of motivated behaviour?*. Thus, we used the methods and techniques of grounded theory (STRAUSS and CORBIN, 2008) to code, categorize, and synthesize data, towards the construction of a theory of motivation in each case study. Initially, all audio from the interviews was transcribed verbatim. We used QSR NVivo 8 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. Then, the codes arising from each interview (and diary study of the same participant, when applicable) were constantly compared to codes in the same interview and from other interviews. From the constant comparisons of the codes, we grouped codes into categories that represent factors affecting motivation and the outcomes of high and low motivated behaviour. Figure 10 shows an example of the category building process.

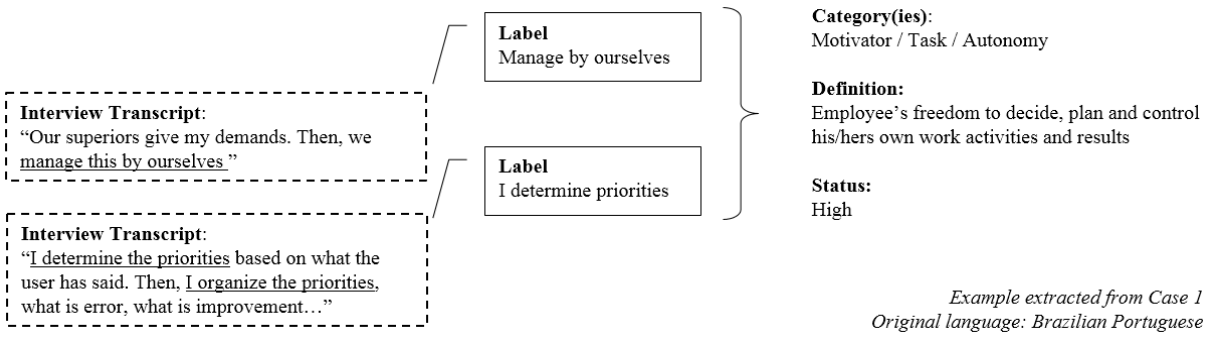


Figure 10: Illustration representing the coding process in the individual case studies

As the process of data analysis progressed, relationships among categories were built, leading to explanatory propositions. 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. The motivators, the propositions, and the central story constitute the elements of a grounded theory for each case study.

The analysis procedures of each individual case study is better explained in the research protocol (DA SILVA, FRANÇA, *et al.*, 2011), as well as in their individual reports (FELIX, 2011) (ARAÚJO, 2011) (CARNEIRO, 2011) (SALES, 2011). The individual results of each case study arealso independently detailed in their specific publications. These details are not included in this text because the cross-case analysis was based directly on the collected data, rather than in the secondary analysis of the cases reports (as pictured in Figure 5, p. 56), as will be better explained in the next section. The characteristics of the cases that are relevant to the present work can be found in Appendices E-H.

Cross-case synthesis and procedures

Research synthesis is a term used to describe a family of methods for summarizing, integrating, combining, and comparing the results of different studies which are mainly interested in similar or related research questions or topics (CRUZES and DYBÅ, 2010). There is a debate among qualitative researchers on whether qualitative synthesis is appropriate or even feasible, without subverting the very principles of qualitative inquiry (SANDELOWSKI and BARROSO, 2003). As explained before, our position follows the view that the findings of qualitative research that uses multiple case studies are likely to be stronger than those of only one case as long as the research synthesis is properly and carefully conducted (SANDELOWSKI and BARROSO, 2003)(YIN, 2009).

Cross-case syntheses can be performed whether the individual case studies have previously been conducted as independent research studies (authored by different persons) or as a predesigned part of the same study. In either situation, the cases must be treated as separate studies (YIN, 2009).

There are mainly two methodological approaches to cross-case analysis: case-surveys and case-comparison (YIN, 1981). The case-survey approach requires a large enough number of case studies to substantiate a conversion from qualitative to quantitative data, using the cases as if they were data points. When this condition cannot be fulfilled, case-comparison is the preferred approach. However, choosing the most suitable method for synthesizing case studies is not straightforward (CRUZES, DYBA, *et al.*, 2011), because several operational procedures have been developed to support the systematic execution of cross-case comparisons, such as: *thematic synthesis*, *cross-case analysis*, *qualitative comparative analysis*, and *meta-ethnography*.Cruzes, Dyba *et al* (2011) suggest four criteria to select an appropriate synthesis method:

A. *Goals and research questions*: A fundamental distinction regarding the objective of the synthesis is whether it attempts to provide knowledge support or decision support. A synthesis directed to knowledge support will typically bring together and synthesize evidence on a particular topic. The objective of our synthesis is directed to knowledge support, so *thematic synthesis*(CRUZES and DYBÅ, 2011) or *cross-case analysis* would be candidate techniques.
B. *Number of case studies*: a *case survey* cannot be meaningfully performed with a small number of cases, as the goal is to have statistically significant results.
C. *Temporal and Spatial variation*: Temporal variation refers to development over time. Spatial variation refers to case studies that were run in different contexts. As previously discussed, our selection of cases controlled these two variables, so it does not represent a risk for any specific type of technique. Nevertheless, *cross-case analysis* are more effective than *thematic analysis* to take variations of contexts to account(CRUZES, DYBA, *et al.*, 2011).
D. *Access to raw data*. The challenge faced by synthesis techniques that deal with secondary sources (such as *meta-ethnography*) is that the factors of interest may carry different meanings in different contexts, and limited access to raw data may represent a relevant obstacle to avoid this trap. In our case, all the raw data was available for the cross-case synthesis.

In fact, in our first attempt to integrate and synthesize two case studies, we have followed the meta-ethnography (NOBLIT and HARE, 1988) technique. The results are reported in França, da Silva *et al*(2013). Although this study was of some usefulness, the synthesis of the four cases raised an unmanageably large variety of factors that supposedly affected the work motivation of software engineers. This led us to question whether they were giving consistent meaning to the term 'motivation' or were they in fact expressing their opinions and experiences about several phenomena rather than a single one? Then, the present research was redesigned in order to answer different research questions, as explained below:

RQ1. What signs represent motivated behaviour of software engineers?

RQ2. What workplace factors influence the motivated behaviour of software engineers?

The first question aims to establish the empirical basis that would enable the investigation of the second question, which is the focus of this synthesis.The cross-case analysis method was chosen to guide our case comparison. Cross-case analyses include a variety of devices, such as tabular displays and graphs, to manage and present qualitative data, maintaining the original meanings of it (CRUZES and DYBÅ, 2011). Commonalities and differences between the studies are explored, and evidence is then summarized within themes across studies with a brief citation of primary evidence.

For the first research question, the analysis was conducted using only the data collected from the interviews. The most obvious path to answer RQ1 would be asking the participants how they define the term 'motivation'. The participants in the case studies were informed that the research focused on motivation, and variations of the word "motivation" were used in the interviews and in the diary studies (e.g. motivate, motivation, motivated). However, we did not suggest any definition of the term before the interview, so that the individuals' own interpretations of these words could affect the way that they answered the interview questions

However, as discussed in Argyris & Schon (1974), there could be a gap between how people define a concept (in this case "motivation") and how they actually perceive, describe and react to it in practice. Therefore, this first analysis was focused on the two questions in the interview script that asked: "How would you describe a clearly motivated

colleague?" (Q25 in the interview script) and "How would you describe a clearly demotivated colleague?" (Q31 in the interview script) to identify adjective sets used to describe motivated engineers (see Appendix C for the full script).

Then, two researchers (R1 and R6) coded the participants' answers independently, one or more behavioural descriptor (adjectives only) to analyse the answersprovided to Q25 and Q31. The codes were merged, and the agreements between the two researchers were classified in one of the three types specified in Table 20. Then, the conflicts were discussed in face-to-face meetings until agreement was achieved. Table 20 illustrates this process.

Table 20 - Examples of coding and conflict resolution

Q25 – Participants' answers (PT_br)	**Independent coding**		**Conflict resolution**	
	Researcher 01	**Researcher 02**	**Type of Agreem.**	**Final Solution**
{Case III - P028} "Ele é um cara que procura o trabalho, é o cara que olha pra aquela task que todo mundo diz ih isso aqui vai dar *problema*, ele vai lá e pega essa"	- Hard-working - Proactive	- Proactive	*Partial*	- Proactive
{Case III – P031} "Ele geralmente está propondo novas soluções, está insatisfeito com os problemas, acho que reclamar e propor soluções é o principal."	- Proactive	- Proactive	*Total*	- Proactive
{Case III – P033} "Deixa eu pensar em um adjetivo, num tinha pensado em um adjetivo ainda não"	*N/A*	*N/A*	*Total*	*Invalid answer*
{Case IV – P042} "(...) não se abalam à pressão, eu não sei. (...), eu acho que eles se sentem bem com as responsabilidades que são colocadas."	- Hard-working - Responsible	- Calm	*Conflict*	- Calm - Responsible

Types of agreement:

- **Total** – both have coded the answer using exactly the same adjectives, or adjectives that pertain to the same category;
- **Partial** – there is an intersection between the sets of codes used, but there are also adjectives that pertain to distinct categories;
- **Conflict** –the codes used by the two researchers pertain to disjoint sets.

For the second research question, the full interviews and diaries data were scrutinized. *Workplace characteristics* is the term that we generically use to refer to any characteristic perceived in the workplace or in the job, which people mention as influent over their behaviour. Hackman (HACKMAN, 1980) and Locke (LOCKE, 1976) do not use a standard term for that. Instead, they use terms such as "job characteristics", "job dimensions", "job characteristics", and other variations. We opted for using a standardized term to improve the understandability of this text.

In this second step of the analysis, the resulting sets of adjectives from the RQ1 analysis were used as pre-formed codes, to identify the useful chunks of data that contribute to the answer of our research question. This step was also carried out by the same two researcher (R1 and R6) and passed through a similar conflict resolution process. After that, cross-case tables were constructed, and the categories were analysed according to the

specificities of each case. Then, the representative categories of each case were identified. Representative means that more than half of participants in a case mentioned the category.

However, representativeness and relevance are distinct things. While representativeness communicates that at least half of participants in a case study mentioned a factor, relevance communicates the possible influence of that factor in a case. Representative does not necessarily assure importance to a factor. Accordingly, a factor that was not mentioned, or was mentioned but was not representative for a case, is not necessarily unimportant. Thus, as the last step, we review the relevance of each category case a case based on similarities and differences raised in the cross-case comparison.

4.7 Theory Building procedure

Glaser and Strauss (1967) argue that it is the intimate connection with the empirical reality that permits the development of testable, relevant, and valid theory. However, this process of theory generation must be systematic and explicit (EISENHARDT, 1989). Eisenhardt (1989) provides a full roadmap for building theories from case study research (Table 21). This roadmap is largely consistent with Yin's(2009) approach for case study research.

After finishing the cross-case analysis, we conducted two final steps proposed by Eisenhardt (1989) (Table 21). In order to shape the theory, Eisenhardt (1989) reccomended that researchers constantly compare theory and data. The TMS-i was used as the starting point to construct the theory. At this point, we referred back to the qualitative data, whenever needed, to look for the "why" behind the constructs and the relationships. According to Eisenhardt (1989), just as in hypothesis testing research, apparent relationships may simply represent spurious correlations, so it is important to discover the underlying theoretical reasons for the relationships to assure the internal validity of the findings.

Finally, the resulting theory is compared to conflicting and similar studies available in the technical literature. The studies reviewed in Chapter 3 served as a basis to this comparison. First, we compare our findings with the Couger and Zawacki (1980)'sstudy and with the MOCC model(SHARP, BADOO, *et al.*, 2008)'. Then, we reviewedthe relevant papers identified in the systematic literature studies(BEECHAM, BADOO, *et al.*, 2007) (FRANÇA, GOUVEIA, *et al.*, 2011) as well as more recent developments.

Table 21 - Process of building theory from case study research (EISENHARDT, 1989, p. 533)

Step	Activity	Reason
✓Getting Started	Definition of research question	Focuses efforts
	Possibly a priori constructs	Provides better grounding of construct measures
	Neither theory nor hypotheses	Retains theoretical flexibility
✓Selecting cases	Specified population	Constrains extraneous variation and sharpens external validity
	Theoretical, not random, sampling	Focuses efforts on theoretically useful cases, i.e. those that replicate or extend theory by filling conceptual categories
✓Crafting Instruments and Protocols	Multiple data collection methods	Strengthens grounding of theory by triangulatino of evidence
	Qualitative and quantitative data combined	Synergistic view of evidence
	Multiple investigators	Fosters divergent perspectives and strengthens grounding
✓Entering the Field	Overlap data collection and analysis including field notes	Speeds analyses and reveals helpful adjustments to data collection
	Flexible and opportunistic data collection methods	Allows investigators to take advantage of emergent themes and unique case features
✓Analyzing Data	Within-case analysis	Gains familiarity with data and preliminary theory generation
	Cross-case pattern search using divergent techniques	Forces investigators to look beyond initial impressions and see evidence thru multiple lenses
✓Shaping Hypotheses	Iterative tabulation of evidence for each construct	Sharpens construct definition, validity and measurability
	Replication, not sampling, logic across cases	Confirms, extends and sharpens theory
	Search evidence for "why" behind relationships	Builds internal validity
✓ Enfolding Literature	Comparison with conflicting literature	Builds internal validity, raises theoretical level, and sharpens construct definitions
	Comparison with similar literature	Sharpens generalizability, improves construct definition, and raises theoretical level
×Reaching Closure	Theoretical saturation when possible	Ends process when marginal improvement become small

✓ - step fully executed in this research

× - step partially executed in this research

4.8 Threats to validity and reliability

Being able to trust results from research is important for the adoption and dissemination of the results both in academia and in the industrial practice. We address below the validity and reliability of our results from the three perspectives proposed by Merriam (2009):

1) *Credibility* (*instead of Internal Validity*): the central problem is how to provide evidence that the findings are credible as the data is presented. To increase credibility, we used triangulation by having data collected from participants with different roles and by using multiple data collection techniques inside each case. We then used member checking, also known as respondent validation, to avoid misinterpretations of what participants said.

2) *Consistency* (*instead of Reliability*): an important question in qualitative research is whether the findings are consistent with the data collected. To increase consistency, we used triangulation in data collection and analysis inside each case. We also kept research diaries and process logs that can be used as audit trails by external reviewers.

3) *Transferability* (*instead of External Validity*): as discussed before, it is a common understanding in qualitative research that generalization of research findings should be performed by the reader or user of the study. In this sense, reader or user can decide to what extent the findings can be applied to other situations. The researcher has to enhance the possibility of someone else "transferring" the results. Two strategies were employed to enhance transferability. First, we tried to provide rich description of the research method, context in which the research was performed, and the results themselves. Second, we sampled the participants to achieve maximum variation since this would help to provide richer data and a more robust resulting theory.The theoretical saturation was not tested, and although we are confident that an adequate level of saturation has been achieved, it represents a threat to the validity of this study. We recommend future research to address this issue further.

A Term of Authorization and Commitment to the Research was also designed to meet ethical requirements of this type of research, and granted the researchers access to facilities, to the participants, and to necessary documents. Additionally, it authorized the participants to use work hours for the interviews and diary studies. Each participant would be also required to sign an Informed Consent Form (Appendix B) explaining the overall

objective and importance of the research, which guaranteed confidentiality of the data provided, the anonymity of the participation, and the right to withdraw from the research at any moment.

4.9 Summary of this chapter

This chapter describes the general research strategy of this work. Following an interpretive exploratory approach, we designed an instrumental multi-case study (YIN, 2009) aimed at answering our main research question: "What workplace factors influence the work motivation of software engineers?". Section 4.1 (p. 53) describes this objective in details, and the research design was detailed in Section 4.2 (p. 54).

Departing from an initial theoretical framework composed of the Job Satisfaction and the Job Characteristics theories (Section 4.3, p. 59), four software engineering organisations were chosen following a replication logic, and seeking for a large variation of size and maturity of the organisations at the same time.

The four organisations were described in details in Section 4.4 (p. 61). In Section 4.5 (p. 67), we provided details on how the interviews, diary studies and document analysis were designed and conducted. Then, we proposed in Section 4.6 (p. 70) the data analysis procedures, for the individual case studies and for the cross-case synthesis. The cross-case analysis method was chosen to guide our case comparison. Then, we proposed to follow Eisenhardt's (1989) of building theory from case study research, which was explained in Section 4.7 (p. 75).

The strength of the designed method is that case studies are powerful methods to deal with contemporary events that cannot be controlled by the researcher, which is the case of our phenomenon of interest: motivationof software engineers. As a weakness, on the other hand, the transferability may have been harmed by the fact that we did not look for theoretical saturation. Section 4.8 (p. 77) explains how we dealt with the threats to validity of this method.

In the next Chapter, the results are presented and, based on those data, we start drafting answers for our research question, which are discussed in depth in the following chapters.

Chapter 5 Cross-Case Results

This chapter presents the cross-case analysis of the four studied cases: a large government organisation (Case I); a private not-for-profit R&D organisation (Case II); a small software company (Case III), and an IT department of a university (Case IV). The methods, tools and procedures used to collect data are detailed in the Chapter 4 . The audio-recorded interviews with the 32 participants from the 4 cases summed up to 24h42', and 91 diary entries were reported.The amount of datacollected over the period of 11 months between 2010 and 2011 is briefly summarized in Table 22.

According to Yin (2009), multiple case study evidence does not need to be presented in traditional form, i.e. a case-by-case narrative. An alternative form for presenting the same evidence is to write the narrative in a question-and-answer form. In this chapter, the results of the cross-case analysis are presented following the order of the research questions. First, the data from the four cases were interrogated in order to answer the first research question: "*RQ1. What signs represent motivated behaviour of software engineers?*". Our results pointed out the most representative signs for the motivated behaviour of software engineers are: engagement, happiness, concentration and collaboration (definitions presented in Section 5.1 , p. 80). After that, the findings of the four cases are reviewed and synthesized in order to answer the second research question: "*RQ2. What workplace factors influence the motivated behaviour of software engineers?*". Then, these results are presented in Section 5.2 (p. 90)

Table 22 - Summary of the collected amount of data

Case Study	Researcher	Interviews			Diary Studies		
		# of Participants	Period	Audio recorded	# of Participants	Period	# of events
I	R1	6	Aug. - Dec/2010	4h 57min	1	Feb. - Mar/2011	17
II	R2	6	Sep. – Nov./2010	4h 05min	2	Feb. - Mar/2011	32
III	R3	10	May/2011	6h 40min	3	May – Jun./2011	10
IV	R4	10	Feb. – Mar./2011	8h 58min	2	Mar. – Apr./2011	32
Total		***32***		***24h 42min***	***8***		***91***

5.1 Behavioural traits of motivated engineers

Our first research question was designed to elicit how the participants in the case studies interpret the term "motivation" in the interviews: *"RQ1. What signs represent motivated behaviour of software engineers?"*

Several answers communicated the lack of confidence participants had regarding the precise words to use to describe a high or low motivated colleague, but only answers such as "I do not know" have been considered invalid for analysis purposes, because they are not useful to make any further inference.

Tables 23-26 present the results of each independent case study. Tables 23-26contain the descriptors attributed to high and low motivated engineers raised from the analysis of the raw data from the transcribed interviews. Beside the adjectives that represent the behavioural descriptors, we identify the participants that mentioned each descriptor. A more detailed profile of each participant is in Appendices E-H. These tables also contain examples of interview excerpts, showing underlined the coded passages from the interviews. These examples were chosen because they are the best illustrations for the meaning of each descriptor.

For each case, we identified the most representative descriptors, i.e. those that represent the best the opinion of the participants in that case. Representative descriptors are those mentioned by at least a half of the participants. The representative descriptors of each case appear marked with a star in Tables 23-26. In case I, motivated engineers were described as focused, communicative, involved and interested; in case II, involvement is the only representative trait; in case III they are careful, involved and proactive; and in Case IV they are communicative, hard-working, and interested. In contrast, engineers with low motivation were described as distracted, reserved, indifferent, absent, troublemaker in Case I; in Case II they are careless, uninvolved, lazy; in Case III, distracted, careless, uninvolved; and in Case IV distracted, reserved, lazy, bored.

Table 23 – Behavioural descriptors and illustrative/representative excerpts (Case I)

High motivated behaviour	Low motivated behaviour
	⁎**Distracted** {P006, P007, P009} *"Seria uma pessoa que a todo o momento alguma outra coisa chama mais atenção, a pessoa que está dispersa. (P009)"*
⁎**Focused** {P006, P009, P010, P11} *"Totalmente dedicado, persistente, focado em seus objetivos." (P011)*	**Unfocused** {P009} *"Indícios fortes disso seria a falta de foco, de interesse nos assuntos, nos temas, no contexto dos projetos. (P009)"*
⁎**Communicative** {P009, P010} *"Se uma pessoa está ali falando das possíveis soluções, se a pessoa está mostrando pra você, tá entusiasmada com os resultados que está sendo atingidos." (P009)*	**Reserved** {P007} *"Tá ali abusado, não conversa com ninguém, no canto quieto." (P007)*
⁎**Involved**{P007, P009, P010, P011} *"O envolvimento dele com o projeto, o compromisso" (P009)*	**Uninvolved** {P007} *"O que acontece que uma pessoa dessa chegou tarde, ta ali... vai ler o email, ou então chega de 8:00 da manhã e passa 3 horas lendo email".(P007)*
Hard-working {P011} *"Totalmente dedicado, persistente, focado em seus objetivos." (P011)*	**Lazy** {P006} *"Não entrega as atividades ou deixa as atividades para o último dia da entrega. Vai postergando."* (P006)
⁎**Interested** {P009, P010} *"a pessoa tá interessada em ver depois os resultados daquele trabalho" (P009)*	**Indifferent** {P006, P009} *"Ela não dá valor ao trabalho. Ela não realiza o trabalho da forma que deveria. (P006)"*
Proactive {P006, P007} *"Você vai atrás para resolver as coisas, você não espera."(P006)*	
Excited {P009} *"a pessoa está mostrando pra você, tá entusiasmada com os resultados que está sendo atingidos". (P009)*	
Good mood {P006} *"Sorridente" (P006)*	**Bad mood** {P006} *"ela já vem de mau-humor" (P006)*
	Resented {P008, P011} *"Só considero quando a pessoa tá reclamando do trabalho."(P008)*
Punctual {P006} *"Começa que ele já chega cedo."(P006)*	⁎**Absent** {P006, P007, P010} *"Ele já chega tarde" (P006)*
	⁎**Troublemaker** {P006, P008, P010} *"ele começa a reclamar muito de tudo. Tá tudo ruim nem a água presta" (P010)*
Productive {P008, P009} *"ele ta com uma produção boa."(P008)*	

Table 24 – Behavioural descriptors and illustrative/representative excerpts (Case II)

High motivated behaviour	Low motivated behaviour
Concentrated {P020} *"mostra que ele ta concentrado naquele trabalho dele" (P020)*	
	⁎ **Careless** { P018, P019, P021} *"entregam sua parte específica, mas mal implementada, aí acaba impactando em toda a equipe" (P019)*
Communicative {P023} *"A participação dele mesmo no projeto, assim, em ta querendo ajudar, em ta perguntando." (P023)*	**Reserved** {P021, P023} *"mal conversa, por exemplo, na hora de sair não fala com ninguém e vai embora" (P021)*
Helpful {P018} *"procura ajudar os seus companheiros."(P018)*	
⁎**Involved** {P018,P019,P023} *"Meio que abraçar a causa, mesmo, segurar o peso junto" (P019)*	⁎**Uninvolved** { P019, P020, P021} *"Às vezes você pega um código de um sistema que você vê que faltou aquele ânimo pra se fazer da melhor maneira."(P021)*
Hard-working {P019, P020} *"ele ainda fica até mais tarde, um dia ou outro, pra poder terminar aquilo ali"(P020)*	⁎**Lazy** {P018, P019, P020, P021} *"o pessoal vai esticar e comunica a todo mundo. Aí tem algumas pessoas que saem mais cedo."(P019)*
	Indifferent {P020} *"não consegue fazer as atividades naqueles dois dias e não ta nem aí pra isso" (P020)*
Proactive {P021, P023} *"ta sempre buscando mais coisas. Se acabou algo, você vai atrás de mais coisas."(P021)*	**Passive** {P021} *"Você vê que foi meio que empurrado com a barriga."(P021)*
Good mood {P020} *"uma pessoa que não ta chateada" (P020)*	**Bad mood** {P021} *"Às vezes, o humor da pessoa, não sei, a gente percebe."(P021)*
	Absent {P018} *"quando o cara começa a perder compromisso de carga horária"(P018)*
Responsible {P018} *"executa as atividades planejadas na expectativa que se definiu a ela" (P018)*	**Irresponsible** {P018} *"perder o compromisso de execução de prazo, de qualidade do que ta fazendo" (P018)*

Table 25 – Behavioural descriptors and illustrative/representative excerpts (Case III)

High motivated behaviour	Low motivated behaviour
Concentrated {P036} *"tá concentrado no que tá fazendo, que num fica muito disperso" (P036)*	***Distracted**{P032, P033, P035, P036, P037} *"ele tá disperso assim num tá concentrado na atividade dele, fica fazendo outras coisas" (P036)*
Careful {P034} *"se preocupa em fazer da melhor forma" (P034)*	* **Careless** {P028, P029, P030, P032} *"não se importa com o resultado final, ele só faz funcionar" (P028)*
Focused {P029, P034, P036} *"tá fazendo as atividades dela e não tá perdendo o foco" (P034)*	**Unfocused** {P036, P037} *"sonolento trabalha pouco, visivelmente enrola, sai muito do ambiente, fica meio aéreo" (P037)*
Communicative {P032, P037} *"conversa e desenvolve tranqüilo" (P037)*	**Reserved** {P031, P032, P037} *"com pouca interação com as pessoas" (P031)*
Helpful{P035} *"tá tirando as dúvidas, auxiliando também" (P035)*	
***Involved**{P029, P032, P034} *"Ele não conta quantas horas faltam pra ele ir embora, ele tá ali fazendo aquilo, ele tem um objetivo"(P029)*	**Uninvolved**{P028, P032, P033} *"as pessoas ficavam fugindo de coisas que comprometessem mais o tempo" (P032)*
	Lazy {P029, P032, P037} *"ele não tem porque entregar as coisas antes do prazo, se ele pode entregar até sexta, que ele enrole a semana inteira e entregue na sexta então" (P029)*
	Indifferent {P028, P034} *"não tem prazer no que ele tá fazendo (...) não contribui com a melhoria do projeto"(P034)*
***Proactive**{P028, P029, P031, P032, P035} *"Ele é um cara que procura o trabalho"(P028)*	**Passive** {P029} *"faz o que faz por obrigação" (P029)*
	Bored {P029, P034} *"ele não tem ânimo"(P029)*
Good mood {P037} *"é um cara sorridente, que desenvolve ou sei lá, trabalha de forma natural, sem estresse aparente" (P037)*	**Bad mood** {P029, P036} *"num tá satisfeito com o que tá fazendo."(P036)*
	Absent {P029} *"Ele chega tarde (...)ele não tem vontade de tá ali" (P029)*
Responsible {P034} *"demonstrando algum comprometimento com a equipe, com os marcos, com as entregas" (P034)*	
Productive {P030 } *"demonstra no tempo de produção que ele implementa" (P030)*	**Unproductive** { P031} *"trabalhando pouco ou rendendo pouco" (P031)*
Calm {P035, P037} *"não perder a paciência na hora que encontrar alguns problemas"(P035)*	

Table 26 – Behavioural descriptors and illustrative/representative excerpts (Case IV)

High motivated behaviour	Low motivated behaviour
Concentrated {P049} *"não perde muito tempo com besteira. (...)ta concentrado naquilo)" (P049*	***Distracted**{P041, P042, P043, P046, P048, P049} *"pouca concentração, não consegue manter a cabeça no trabalho, se distraindo, vendo algumas besteiras" (P043)*
Focused {P041} *"ele quer terminar aquilo ali, ta desejoso de terminar aquilo" (P041)*	**Unfocused** {P041, P042} *"sempre desvia do que precisa fazer" (P042)*
Careful{P040, P043, P045, P046} *"costuma ser mais detalhista do que necessário" (P040)*	
***Communicative**{P041, P044, P047, P048} *"ela ta interagindo com outras pessoas (...)ta querendo participar das atividades" (P044)*	**Reserved**{P044, P046} *"não interage com os outros." (P044)*
Helpful{P044} *"tá realmente ajudando e levando dúvidas para todas as outras pessoas"(P044)*	**Unhelpful**{P044} *"não quer ajudar, não gosta de ajudar essas pessoas."(P044)*
Involved {P044} *"sabe as atividades que tem que fazer e que não fica deixando para o outro dia" (P044)*	
***Hard-working**{P041, P048} *"as vezes ele fica um poquinho mais do que o horário devido, por que ele quer terminar aquilo ali" (P041)*	***Lazy**{P041, P045, P047, P048} *"Acho que é aquele que vai muitas vezes ao cafezinho, costuma sair da sala."(P048)*
***Interested**{P041, P043, P044, P047, P048, P049} *"A pessoa chega e faz o que tem que faze, com vontade."(P043)*	**Indifferent** {P045, P046} *"Ele procura qualquer outra coisa, menos exercer as atividades dele do dia a dia"(P046)*
Proactive {P045, P047, P049} *"Não espera que você vá ate ela com o problema. Ela já identifica o problema e já diz pra você que vai resolver aquilo ali" (P045)*	**Passive** {P045, P047} *"falta de atitude proativa"(P045)*
Excited {P040, P041, P048} *"Costuma falar mais, se empolgar mais quando comenta o que faz."(P048)*	***Bored**{P040, P041, P042, P047, P048, P049} *"a pessoa talvez não esteja entusiasmada com aquela atividade."(P041)*
Good mood {P040} *"fala contigo sempre de bom humor" (P040)*	**Bad mood** {P040, P046} *"Não dá um bom dia pros amigos de trabalho, baixa a cabeça"(P046)*
	Resented {P049} *"a pessoa não ta a fim, não ta gostando daquilo" (P049)*
	Absent {P044} *"ele não cumpre horário, não cumpre as atividades que foram planejadas" (P044)*
Responsible {P042, P044} *"eu acho que eles se sentem bem com as responsabilidades que são colocadas" (P042)*	**Irresponsible** {P044, P047} *"ele não cumpre horário, não cumpre as atividades que foram planejadas" (P044)*
Productive {P043, P045, P046} *"costuma produzir mais" (P043)*	**Unproductive** {P041, P043} *"a produtividade cai bastante."(P043)*
Calm {P042} *"os motivados eu acho que não se abalam à pressão" (P042)*	
	Pessimistic {P045} *"o camarada coloca empecilho pra tudo" (P045)*

It is possible to notice that these tables include four different types of descriptors: descriptors that represent individual attitudes towards the task (e.g. careful/careless, hard-working/lazy) and towards the organisation (e.g. punctual/absent, responsible/Irresponsible), emotions (e.g. good mood/bad mood, excited/bored), and work outcomes (e.g. productive/unproductive). Based on the semantic similarity between the adjectives, the behavioural descriptors were grouped in more general categories, by following a standard process of content analysis.

Table 27 - Behavioural descriptors for motivated and unmotivated behaviours

Categories	**Positive Adj. (motivated)**	**Case I**	**Case II**	**Case III**	**Case IV**	**Negative Adj. (unmotivated)**	**Case I**	**Case II**	**Case III**	**Case IV**
Concentration	Careful			✓*	✓	Careless		✓*	✓*	
	Concentrated		✓	✓	✓	Distracted	✓*		✓*	✓*
	Focused	✓*		✓	✓	Unfocused	✓		✓	✓
Collaboration	Communicative	✓*	✓	✓	✓*	Reserved	✓*	✓	✓	✓*
	Helpful		✓	✓	✓	Unhelpful				✓
Engagement	Involved	✓*	✓*	✓*	✓	Uninvolved	✓	✓*	✓*	
	Hard-working	✓	✓		✓*	Lazy	✓	✓*	✓	✓*
	Interested	✓*			✓*	Indifferent	✓*	✓	✓	✓
	Proactive	✓	✓	✓*	✓	Passive		✓	✓	✓
Happiness	Excited	✓			✓	Bored			✓	✓*
	Good mood	✓	✓	✓	✓	Bad mood	✓	✓	✓	✓
	-					Resented	✓			✓
	Upbeat				✓	-				
Professionalism	Punctual	✓				Absent	✓*	✓	✓	✓
	-					Troublemaker	✓*			✓
	Responsible		✓	✓	✓	Irresponsible		✓		✓
Productivity	Productive	✓		✓	✓	Unproductive			✓	✓
Stability	Calm			✓	✓	-				
Optimism	-					Pessimistic				✓

✓*- representative in the case

✓ - mentioned in the case, but not representative

Table 27 lists the categories, illustrating the representative descriptors in each case study. The categories were carefully labelled in order to keep consistency with the most representative behavioural traits, but the other also were taken to account. For example, C2 – Collaboration means mainly communication between the peers, but communication about their obstacles and achievements related to the work the engineers are carrying out as a team, in order to maintain the teams' synchronisation and to be able to help, or ask for help of, other engineers. C3 – Engagement isrepresents the union of the four adjectives in that category: involved, hard-working, interested and proactive.

This table shows that some categories, such as concentration, collaboration and engagement are visually more dense in the table than others, which are rarely mentioned, such as productivity, stability and optimism. Figure 11 shows the most representative categories of each independent case, and it reveals that:

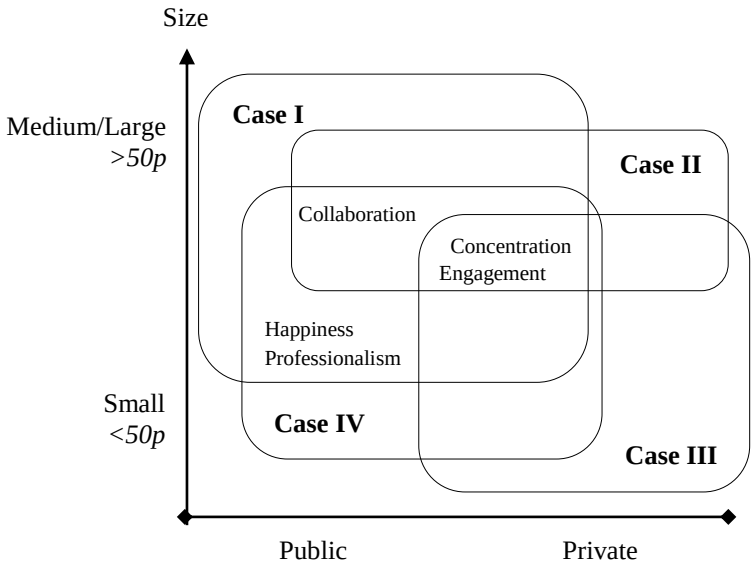


Figure 11 - Categories of descriptors of motivated behaviour (Size vs. nature of the organisation)

Observation #1 Concentration and Engagement are representative descriptors in all cases. Both descriptors comprise attitudes toward the work that are perceivable before and/or during an execution of a task. Joint views of the data from the four cases (Figure 12 and Figure 13) show that Concentration and Engagement keep representative regardless possible interfering variables: work experience, education, role or gender.

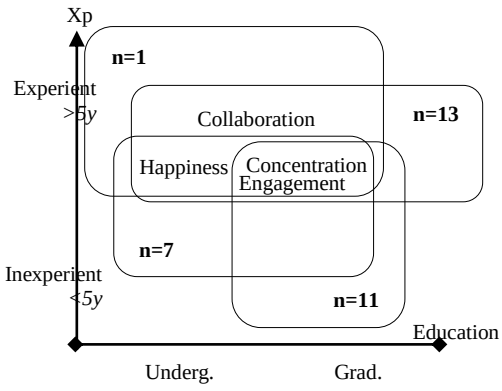


Figure 12 - Categories of behavioural descriptors (work experience x education)

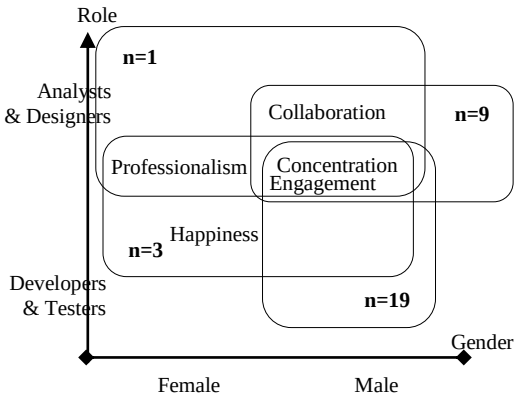


Figure 13 - Categories of behavioural descriptors (technical role x gender)

Observation #2 Happiness and Professionalism are representative only in the two public organisations. In both cases, the type of employment contract allowed people to be absent and/or unproductive without being fired. Therefore, it makes sense that professionalism is seen as a matter of motivation, i.e. only if people are motivated, will they maintain their attendance and responsibility at work. In contrast, in the private organisations, being professional is a matter of keeping their jobs, regardless they are more or less motivated. Nevertheless, as shown in Table 27, variations in professionalism are also perceived in the private sector, although it is not necessarily representative. Anyway, Professionalism can be better classified as an outcome of motivation rather than a descriptor, because it represents an attitudes towards the organisation that may be influenced by the level of one's motivation, but do not necessarily describe one's level motivation.

Observation #3 Collaboration is not a representative trait of motivated engineers only in Case III.Case III is also the only organisation among the four that effectively uses agile methods, so collaboration is part of their software engineering process.Thus, whether a person,in this organisation, is more or less motivated, his/her level of collaboration may not be very affected since that to collaborate is part of their work. The other three cases follow a more traditional-oriented engineering process, so collaboration is not embedded in their development process, i.e. to help others is not an obligation. Thus, motivation may be more clearly perceived through the collaborative behaviour.

Observation #4 Productivity, Stability and Optimism are not representative in any case. Productivity in particular is an outcome that, although may be related to the motivation, can be influenced by several other aspects of the work, such as technical knowledge or work experience. The findings of Case II (see Appendix F) reveal that given one's potential performance determined by factors other than motivation, high motivation would lead software engineers to perform as well as they can. Therefore, it is indeed not safe to infer motivation based on the productivity of an engineer. Stability and Optimist can easily be confounded to personality traits. For example, people with high neuroticism tend to be less stable and less optimistic than others,

they are not necessarily less likely to develop positive attitudes toward their work (BOZIONELOS, 2004).

The analysis of the answers to Q25 and Q31, together, also enabled us to check how consistent the participants were in describing high and low motivated behaviours.The data is shown in Table 28. Consistency, here, means that the participant used opposite adjectives from the same category to describe high and low motivated behaviours. Engagement was far the most consistently used category.

Table 28 - Analysis of the engineers' discourse consistent use of the behavioural descriptors

Categories	Motivated Behaviour				Motivated AND Demotivated Behaviour				Demotivated Behaviour			
	Case I	Case II	Case III	Case IV	Case I	Case II	Case III	Case IV	Case I	Case II	Case III	Case IV
Concentration	P006 P009 P010 P011	P020	P029 P034 P036	P040 P041 P043 P045 P046 P049	P006 P009	-	P029 P036	P041 P043 P046 P049	P006 P007 P009	P018 P019 P021	P028 P029 P030 P032 P033 P035 P036 P037	P041 P042 P043 P046 P048 P049
Collaboration	P009 P010	P018 P023	P032 P035 P037	P041 P044 P047 P048	-	P023	P032 P037	P044	P007	P021 P023	P031 P032 P037	P044 P046
Engagement	P006 P007 P009 P010 P011	P018 P019 P020 P021 P023	P029 P032 P034	P041 P043 P044 P045 P047 P048 P049	P006 P007 P009	P018 P019 P020 P021	P029 P032 P034	P041 P045 P047 P048	P006 P007 P009	P018 P019 P020 P021	P028 P029 P032 P033 P034 P037	P041 P045 P046 P047 P048
Happiness	P006 P009	P020	P037	P040 P041 P048	P006	-	-	P040 P041 P048	P006 P008 P011	P018 P021	P029 P034 P036	P040 P041 P042 P046 P047 P048 P049
Professionalism	P006	P018	P034	P042 P044	P006	P018	-	P044	P006 P007 P008 P010	P018	P029	P044 P047
Productivity	P008 P009	-	P030	P043 P045 P046	-	-	-	P043	-	-	P031	P041 P043
Stability	-	-	P035 P037	P042	-	-	-	-	-	-	-	-
Optimism	-	-	-	-	-	-	-	-	-	-	-	P045

This analysis assisted three additional observations:

Observation #5 Negative descriptors related to Concentration (Cases II and III), Happiness and Professionalism (in Case I) categories are more easily perceived than their opposites, i.e. distraction and unhappiness describe demotivated engineers better than concentration and happiness describe motivated engineers. In contrast, positive Collaboration (Cases I and IV) descriptors are more common when describing motivation than the opposite.

Observation #6 In all cases, few software engineers consistently used some category to describe motivated and demotivated colleagues. That may have occurred for two possible reasons: either 'motivated' and 'demotivated' behaviours were actually perceived to be two independent phenomena rather than two opposite of the same construct, or it may simply illustrate the difficulty that some participants had to effectively communicate their ideas.

Observation #7 Although some individual participants tended to be polar, i.e. to mention more positive than negative descriptors (e.g. Case I-P009 and Case III-P035) or vice-versa (e.g. Case I-P007 and CaseII-P021), when it comes to the cases as a whole, there is a balance: the number of positive and negative descriptors are balanced among the cases. While in Case I, the positive descriptors were slightly more common, in Case III the negative were so. It was also not possible to find any relevant interference of specific individual characteristics (gender, education, work experience and technical role) over the tendency to mention more positive or negative descriptors. However, we have not assessed participants' personalities, which may be influent over this aspect.

In face of these observations, it is possible to conclude that Engagement, Concentration, Collaboration,and Happiness are the most representative and consistent descriptors among all four case studies, and thus are the characteristics that best describe the motivation of software engineers. In the next section, we investigate the antecedents for each one of these behavioural traits, based on the interview and diary data.

5.2 Antecedents of motivated behaviour of software engineers

The second research question was designed to collect workplace factors that influenced the motivation of participants: "*RQ2. What workplace factors influence the motivated behaviour of software engineers?*". As presented in Section 5.1 , the answer for our first research question pointed to four behavioural traits that best described motivation in our data: engagement, concentration, collaboration, and happiness. Then, in order to be able to answer the second question, it was split in the following four questions:

*RQ2. What workplace factors influence the **motivated behaviour** of software engineers?*

- *RQ2.1: What workplace factors influence **engagement** of software engineers?*
- *RQ2.2: What workplace factors influence **concentration** of software engineers?*
- *RQ2.3: What workplace factors influence **collaboration** of software engineers?*
- *RQ2.4: What workplace factors influence **happiness** of software engineers?*

All participants contributed to answer at least one of the four questions. Consistently to what has been described in Observation #1 (p. 86), there were much more contributions to antecedents of Engagement (RQ2.1) and Concentration (RQ2.2) than to the others (RQ2.3 and RQ2.4). Table 29 summarize the antecedents found for each behavioural trait of motivation..

Table 29 - Summary of the antecedents of motivation

Motivated behaviour trait	Antecedents	Case I	Case II	Case III	Case IV
Engagement	Monetary Rewards	✓	✓	✓	✓
	Acquisition of useful knowledge	✓*	✓*	✓*	✓*
	Social Impact	✓*	✓	✓	✓*
	Work variety	✓	✓*	✓	✓
	Creative work	✓*	✓*	✓*	✓*
	Engagement of co-workers	✓*	✓*	✓	✓
	Technical confidence	✓	✓	✓*	✓*
Concentration	Well defined work	✓*	✓	✓	✓*
	Workload and pressure	✓*	✓	✓	✓
	Artifacts and tools	✓			✓
	Work environment		✓	✓	✓
Collaboration	Communication and participation		✓*	✓*	✓
	Knowledge exchange		✓	✓*	✓
	Interdependence			✓	✓
	Team competence	✓	✓		✓
Happiness	Performance	✓*	✓*	✓*	✓*
	Feedback	✓	✓*	✓*	✓

✓ - mentioned, but not representative
✓* - representative for the case study

The following subsections detail the data that answer RQ2{1-4} and explore the other relevant observations in this analysis.

5.2.1 Workplace factors of Engagement (RQ2.1)

In this Section, we explore the answers raised for the following research question: "*RQ2.1: What workplace factors influence **engagement** of software engineers?"*

It was possible to identify workplace factors that influence engagement in all participants' interviews in all case studies. In general, personal identification with the task was a recurrent theme in all cases, because there were questions in the interview script explicitly asking for the interviewees' favourite tasks (e.g. Q12, Q14, Q16, Q18). However, it was not considered explicitly as an antecedent of motivation, because knowing what the engineers "like" or "do not like" do not clarify what parameters, or characteristics of the work, are relevant to determine when they feel identified or not. The interview script was complemented with "why"-questions (Q13, Q17) seeking for the reasons for they do like or dislike those tasks, and these questions contributed to the factors that appear in Table 30and Table 31.

Table 30 and Table 31 showthe factors that appeared in each case, the participants that mentioned that factor, and an interview excerpt that is representative to describe each factor. The most representative factors that influence engagement across the four cases are *Acquisition of useful knowledge* and the *Creative work*. This, and other observations, are detailed below.

Observation #8 Monetary rewards may not be a central problem for software engineers. Monetary rewards have been mentioned in all four case studies. In Case I, both participants that mentioned 'monetary incentives' are System Analysts, and reported that they do not like to do the work they are currently doing, for different reasons. In their case, monetary incentives could be a compelling reason to keep doing their work. In Case III, P030 became software engineer because he believed that could get rich in a short time. In Case IV, P046's work experience began in private companies, and he also has started his own company. These two participants seem to hold natural drives for money. In P023 and P046's discourses, the bonuses and rewards

did not seem to be a central issue in their discourses. These individual characteristics illustrate that monetary rewards may be perceived as an indirect or a direct source of motivation. However, in fact, none of the four studied organisations offered monetary incentives in a systematic basis. Therefore, on one hand, the evidence that we have collected on this factor may only be a collection of isolated assumptions. On the other hand, it reveals that monetary rewards is not a central issue in the studied organisations.

Observation #9 Acquisition of useful knowledge is a representative antecedent of engagement in all four cases. Being updated, either to apply new technologies to work in order to be more productive (Cases I and II), or to expand one's possibility to find other job opportunities (Cases III and IV), is a relevant drive of engagement. The knowledge acquisition can occur through training, or working with different people or different things (technologies, projects, problem domains, etc.). It is important to highlight the fact that the knowledge being offered to software engineers must be useful, i.e. can be converted on to productivity or money (job opportunities), otherwise it would not be valuable for them. Therefore, the engineers are more likely to be engaged in tasks that yield useful knowledge.

Observation #10 Software engineers are likely to be engaged in creative work. This observation points out two things. First, it is not only the brainwork characteristic of the work (as in Case II) that is motivating, but more challenging intellectual tasks, involving problem solving(Case I), research (Case II) and creation (Case IV). Second, even though writing software is a knowledge-intensive work, not all tasks in a development process are considered challenging and creative. Besides, the meaning of "challenge" varies to different people. Among our participants, there are a few examples of participants saying that testing is boring, while others refer to testing as a challenging and creative activity. The same conflict of opinions happens to other activities such as requirements elicitation, documentation, and programming. Thus, other subjective variables, not identifiable in our data, may influence the individuals' perception of challenge and creativity in different tasks.

Observation #11 Social impact appeared as a representative antecedent of engagement in the two public organisations. It has also been mentioned by isolated participants in the private organisations cases, but it was not representative for those cases. In Cases I and IV (the public organisations) the social impact of their work is clear, because they are developing information systems that are used by a social group in which they are personally inserted. Besides, they do not have the burden of having to struggle to help the organisation to survive at any cost, to maintain their clients, or to keep their jobs, which are three common pressures in the private environment. Thus,so the social impact becomes a compelling reason to be engaged to their work. Furthermore, although analysts are the ones that work closer to clients, the social impact is also equally important to both analysts and developers.

Observation #12 Work variety is representative only in Case II. However, the eight participants that mentioned 'work variety' in all four organisations have a strong technical orientation (P009 in Case I, P020, P021 and P023 in Case II, P032 and P034 in Case III, P040 and P044 in Case IV). Therefore, this factor may be relevant to engineers with this specific characteristic. In general, personal identification with the task was a recurrent theme in all cases, because there were questions in the interview script explicitly asking for the interviewees' favourite tasks (e.g. Q12, Q14, Q16, Q18). However, it was not considered explicitly as an antecedent of motivation, because it is rather an abstract condition, because knowing what the engineers "like" or "do not like" do not clarify what parameters, or characteristics of the work, are relevant to determine when they feel identified or not. The interview script was complemented with "why"-questions (Q13, Q17) seeking for the reasons for they do like or dislike those tasks, and these questions contributed in several complementary ways with the factors that appear in in Table 30 and Table 31

Table 30 – Workplace antecedents of Engagement

Cross-case labels	Case I	Case II	Case III	Case IV
Monetary Rewards *Definition:* *monetary incentives offered before the task, and which are worthy of the individuals' pursuit.*	**Monetary incentives** {P010, P011} *"Eu acho que o incentivo financeiro (...)Talvez você não busque por que a empresa não te dá absolutamente nenhuma motivação pra isso. (...) Não tem nenhuma recompensa por aquilo. (...)" (P010, in: answer to Q40)*	**Bonus** {P023} *"acredito que sim. Tanto no aspecto financeiro, eu acho que na medida do possível eu acho que ta sendo feito, a questão de bônus, esse tipo de coisa." (P023, in: answer to Q36)*	**Payment** {P030} *"o salário que eu recebo, querendo ou não, não é questão de valor, mas só pelo fato do funcionário tá recebendo ele tem que produzir, isso daí já é a motivação natural da coisa," (P030, in: answer to Q6)*	**Rewards** {P046, P048} *"incentivo eu sei que estimula um pouquinho, né, tipo... tanto financeiro como tentaram e não conseguiram a questão de folga ou dispensa de horário" (P048, in: answer to Q40)*
Acquisition of useful knowledge *Definition:* *the knowledge that the individual believes that he/she can acquire as part of, or a reward for, a task execution, and which can be useful for their life, career, performance, etc..*	⁎**Being updated** {P007, P010, P011} *"A possibilidade de participação motiva pois este evento é enriquecedor e transmite experiências que são úteis no dia a dia do trabalho" (P010, in: diary data 03/03/2011)* -	⁎**Technical skill development** {P020, P022, P023} *"P022: Então, aprender novas coisas, modificar algumas coisas dentro do sistema, fazer mudança, acho que essa dinâmica, seja num projeto só, seja em vários projetos, sempre me atraiu muito."(P022, in: answer to Q3)* -	⁎**Continuous learning**{P028, P033, P034, P035, P036} *"você aprender com o outro no dia-a-dia em si, isso é o que mais me deixa estimulado"(P28, in: answer to Q13)* -	⁎**Learning opportunities** {P041, P042, P043, P045, P047, P048, P049} *"o que me estimula é sempre ter alguma novidade, por menor que seja, que eu converso, que eu aprendo, que eu descubro porque ouvi falar e vou procurarinformação mais aprofundada." (P042, in: answer to Q6)* **Learning about the product** { P044, P046} *"o que me motiva mais no dia-a-dia é tá descobrindo um pouco mais do negocio e assim qualquer problema ou dúvida do usuário, a gente já tem como solucionar isso rapidamente" (P044, in:answer to Q12)*
Social Impact *Definition:* *How the individuals perceive that their products have a social impact, i.e. benefits other people's lives, is useful or significant.*	⁎**Social role of the organisation**{P006, P007, P009, P010} *"Eu acho que o papel da instituição também é um papel importante, é uma coisa que me estimula." (P006, in: answer to Q6)*	**Useful products** { P019, P020} *"Houve mudanças de requisitos, o fato positivo é que agora o sistema ao qual estou implementando faz mais sentido, dentro do processo de negócio do cliente.," (P019, in diary data, 16/02/2011)*	**Helping others** {P035} *"A questão de envolver tecnologia, tá sempre lidando, tá sempre se atualizando e o fato de tá usando isso pra um bem maior de tá ajudando a população em geral" (P035, in: answer to Q2)*	⁎**Useful products**{ P040, P041, P044, P047} *"uma coisa é que você ta contribuindo pra facilitar o serviço de alguém. Isso eu acho que é o que satisfaz, é o que incentiva qualquer pessoa que faz o serviço. Você faz o serviço pra alguém e você quer que saia realmente com utilidade pra essa pessoa." (P047, in:answer to Q13)*
Work Variety *Definition:* *The individuals have contact with different tasks, business domains, rules and challenges.*	**Different domains** {P009} *"essa idéia de ta conhecendo um pouco do domínio de negócio de outras áreas e tá apresentando soluções de TI pra essas áreas ai, sempre foi atraente." (P009, in: answer to Q2)*	⁎**Project Variety**{P020, P021, P023} *"Então, você sabia que quando chegasse no final daqueles dois anos você taria noutro projeto. Seria outras regras de negócio, outro aprendizado e outros desafios"(P020, in: answer to Q3)*	**Domain variety** {P032, P034} *"quando um projeto acaba começa um novo e é tudo novo dali pra frente, ou seja, você num tem um problema assim que é muito de rotina" (P034, in:answer to Q3)*	**Project variety** {P040, P044 } *"o negócio é muito grande e a gente tá tendo, aos poucos, a oportunidade de conhecer um pouco de cada parte do negócio e isso motiva." (P44, in: answer to Q36)*

'P' – refers to participants. See detailed profiles at Appendices E-H.
'Q' – refers to interview script's questions (Appendix C)
⁎-indicates the representative factors

Table 31–Workplace antecedents of Engagement (continuation)

Cross-case labels	Case I	Case II	Case III	Case IV
Creative Work *Definition:* *The tasks involve creating new solutions for new and challenging problems*	* **Creativity** {P006, P007, P008, P011} *"eu gosto é justamente participar da parte de desenvolvimento da parte de criação, de discutir solução, discutir arquitetura" (P006, in: answer to Q3)*	* **Brainwork** {P019, P020, P021, P022, P023} *"eu não gosto quando a coisa fica muito mecânica. (...) Essa parte de só clicar, eu... eu sou muito inquieto" (P021, in: answer to Q16)*	* **Intellectual challenge** {P028, P031, P033, P034, P036, P037} *"eu gosto de desafios, eu gosto de coisas diferentes que fazem a pessoa pensar e debater sobre isso, me deixa estimulado e afim de resolver aquele problema" (P031, in: answer to Q13)*	* **Intellectual challenge** {P040, P041, P043, P045, P046, P047, P048, P049} *"você ter uma interrogação e tem que descobrir como é que se faz pra chegar no local." (P049, in: answer to Q13)-*
	Problem solving {P006, P009} *"eu gosto é justamente participar da parte de desenvolvimento da parte de criação, de discutir solução, discutir arquitetura" (P006, in: answer to Q3)*	**Research** {P022, P023} *"acho que é quando uma atividade requer pesquisa" (P022, in: answer to Q12)*	-	**Authorship** {P040,P042, P045 } *"Cria uma solução pra um problema, senão 100%, pelo menos 80% originário do seu trabalho, do seu esforço" (P040, in: answer to Q13)*
	-	-	-	* **Variety of work**{P040, P043, P044, P045, P046, P048} *"você ficar só com documentação, sempre com documentação(...), vai me desmotivar porque fica muito monótono." (P044, in: answer to Q35)*
Technical Confidence *Definition:* *the individual belief that he/she is technically able to execute a given task*	**Confidence** {P006, P008} *"Fazer coisas que sei fazer melhor," (P008, in: answer to Q13)*	**Team Expertise** *{P018}* *"o know-how das pessoas, em relação às atividades que elas fazem, é bom. (...) De um jeito ou de outro você termina aproveitando essa expertise." (P018, in: answer to Q22)*	* **Technical confidence**{P029, P032, P034, P035, P037} *"Atividades que eu já tivesse experiência em trabalhar, já soubesse o roteiro de como seguir, ou possíveis erros que possam acontecer, já saber mais ou menos onde tá e como lidar" (P035, in: answer to Q11)*	* **Self-Confidence**{P040, P041, P043, P044, P047} *"é desafiador demais. Mas, assim, o que vai acontecer é que vão surgir bugs ou problemas, conflitos lá que eu vou demorar horas e horas e horas e horas pra conseguir encontrar a solução. E, muito provavelmente, não vou encontrar." (P041, in: answer to Q18)*
Engagement of Co-workers *Definition:* *How engaged the co-workers are, according to the each one's individual perception*	* **Co-workers' engagement** {P006, P007, P010} *"você vê uma pessoa trabalhando mostrando resultado (...) talvez até estimule você a seguir o comportamento dele." (P006, in: answer to Q27)*	* **Co-workers' commitment** {P018, P019, P020} *"o que me desmotiva é muito a questão das mesmas atividades e a falta de comprometimento de outras pessoas," (P019, in: answer to Q16)*	**Co-workers' motivation** {P033, P034} *"de uma forma indireta as pessoas que estão aqui são bem animadas, acho que isso é legal e acaba motivando bastante." (P033, in: answer to Q36)*	**Co-workers' engagement** {P046, P043, P045} *"Como toda equipe sempre tem alguém que tá desmotivado um pouco mais do que deveria e acaba contagiando as outras pessoas com essa desmotivação" (P045, in: answer to Q28)*

'P' – refers to participants. See detailed profiles at Appendices E-H.
'Q' – refers to interview script's questions (Appendix C)
**-indicates the representative factors*

Observation #13 Technical confidence is a representative factor only in the small organisations (Cases III and IV), while the co-workers' engagement is representative only in the large organisations (Cases I and II). It is not clear how the size of the organisation can have such influence onto these factors. Analysing the profile of the organisations, it is possible to notice that the engineers in Cases I and II are generally more experienced than those in Cases III and IV. Thus, it would make sense to think that inexperienced engineers are still struggling to prove to themselves their own competence, so the technical confidence is likely to be more easily perceived as an antecedent of engagement. In contrast, more experienced engineers deposit more value on equity at work (ADAMS, 1963), which is the ratio between their effort and the rewards they get, in comparison to those of their co-workers.

Figure 14 summarizes the representative antecedents of engagement across the four cases.

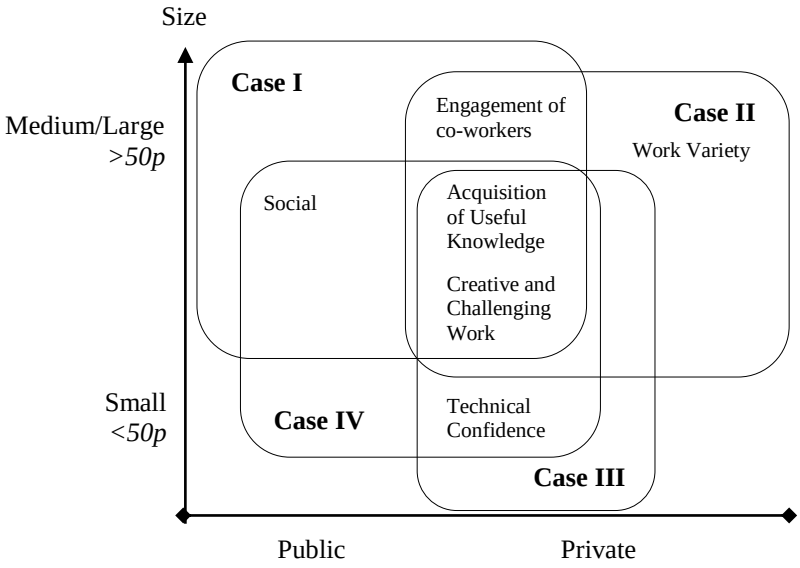


Figure 14 - Antecedents of Engagement – a cross case view of the representative factors

5.2.2 Workplace factors of Concentration (RQ2.2)

In this Section, we explore the answers given to the question: "*RQ2.2: What workplace factors influence **concentration** of software engineers?"*

In Case I, all participants contributed to this topic. Threeparticipants in Case II, six in Case III and seven in Case IV also contributted to answer this question. Table 32summarizes the cross-case themes and the categories attributed to the identified antecedents of concentration.The answer for this question brought up factors that are different in nature from those in the previous section. With respect to the antecedents of engagement, participants tended to mention only elements that determine or enhance engagement, while in respect to antecedents of concentration, participants also revealed factors that wiped out concentration, i.e. negatively influenced it. These negative factors were actually more frequently mentioned than the positive ones. InTable 32, these factors are marked with a negative symbol ("-"). This, and other relevant observations, are discussed below.

> *Observation #14* There was no single antecedent of concentration representative across all the cases. In particular, in cases II and III, there were no representative antecedents of concentration, in spite of the data in Section 5.1 showing that concentration is a representative trait of motivated behaviour for both case studies. In general, consistently with *Observation #5*(p. 89), it is possible to notice in all cases, a tendency for participants to focus on isolated factors that disturb concentration(21 contributions) rather than factors that enhance it (13 contributions), such as the poor quality of artefacts, unhelpful tools, disturbing work environment or problematic infrastructure. Besides, these isolated factors are not representative in any of the four cases.

Table 32 - Workplace antecedents of Concentration

Cross-case labels	Case I	Case II	Case III	Case IV
Well defined work *Definition:* *Working in systematic tasks with clear goals, well defined requirements and predicable results*	* **Well defined work** {P006, P009, P010, P011} *"você ta lidando com um número finito de atividades que vão chegar num resultado." (P009, in: answer to Q13)*	-	-	**(-) Unlear processes** {P042} *"esses processos mal definidos a gente não tem clareza do que executa, (...) Aí a gente não sabe como agir corretamente dentro desse ambiente bagunçado " (P042, in: answer to Q38)*
	-	**Clear requirements** {P018} *"O cliente participou ativamente da discussão do método de trabalho o que facilita oentendimento das partes."(P018, in diary data, 15/02/2011)*	**(-) Unclear requirements** {P032, P035, P037} *"o que mais me desmotiva seria as funcionalidades surgirem do nada e novos projetos surgirem do nada sem nenhuma especificação boa" (P037, in: answer to Q38)*	**Clear customer needs**{P043, P046, P047} *"Quando você entra em contato com o seu cliente, você é capaz de perceber exatamente o que ele quer." (P047, in: answer to Q14)*
	-	-	-	**Clear goals** {P042} *"antes de aceitar, antes de botar pra moer, como a gente diz, eu ia tentar entender por que tinha que ser daquele jeito." (P042, in: answer to Q35)*
Cognitive Workload *Definition:* *Refers tohow fair and balanced the cognitiveworkload and the responsibilities are, at work*	*__Fair workload__ {P007, P008, P009} *"Se aparece muito ninguém gosta, se tem pouco a pessoa fica sem saber o que fazer no dia" (P008, in: answer to Q11)*	**(-) Work overload** {P020} *"Se a equipe não faz um trabalho de qualidade, você precisa sempre trabalhar mais, pra colocar qualidade naquele projeto" (P020, in? Answer to Q7)*	**Goal-driven responsibilities {**P030} *"você sabe que tem as suas responsabilidades, sabe que tem prazos, muitas vezes prazos curtos, mas o pessoal não fica te prendendo aquilo, te pressionando" (P030, in: answer to Q36)*	**(-) Work overload** {P041, P042} *"não é priorizada corretamente, então, vira um bola de neve de demandas que não são solucionadas, não são atendidas, e aí gera um estresse desnecessário" (P042, in: answer to Q28)*
	-	**(-)- Project variety** {P018} *"O cliente mudou minha alocação para resolver uma pendência de outro projeto." (P018, in diary data, negative event, in 17/03/2011)*	**(-) Simultaneous projects** { P032, P037} *"eu termino um projeto e tenho que começar outro e esse outro é totalmente diferente, aí eu tenho que mudar todo o meu pensamento, mudar a forma de raciocinar" (P037, in: answer to Q16)*	-
Artifacts and tools *Definition:* *Refers to the quality of the artifacts and software tools produced and used in the development process*	**(-) Poor tools** {P009, P010} *"algumas ferramentas me tiram um pouco a paciência. (...) Nesse projeto particular eu fui liberado a não usar" (P009, in: answer to Q13)*	-	-	**(-) Poor code** {P041, P046, P048} *"Código bagunçado. Sem comentários. Você ter que... Você passa muito tempo pra entender uma besteira." (P048, in: answer to Q18)*
Work environment *Definition:* *Refers to the overall workplace physical environment, including ergonomy,noise, and hardware tools.*	-	**(-) Constant Interruptions** {P019, P020} *"era um negócio muito complicado, porque todo mundo ficava me perguntando: [" Eu faço o que, agora? _Eu faço o que, agora?"]." (P020, in: answer to Q20)*	**(-) Disturbance** {P028, P034} *"preciso obrigar as pessoas ao meu redor a me deixarem trabalhar, não ter tranqüilidade suficiente pra exercer a minha função" (P028, in: answer to Q15)*	(-) **Problematic infrastructure** {P048, P049} *"Todo o trabalho que levei 2 dias para terminar foi perdido por problema na máquina" (P049, in diary data, 14/03/2011)*

'P' – refers to participants. See detailed profiles at Appendices E-H.
'Q' – refers to interview script's questions (Appendix C)
(-) –exerts a negative influence
**-indicates the representative factors*

Observation #15 *Well defined work* was mentioned in all case studies, but it is representative only in Cases I and IV. Case I followed no particular development process, so it is reasonable that four participants demanded a better defined work process. However, the organisation in Case IV apparently followed well defined work procedures, but according to the participants of that case, lacked clarity about the customer needs in the documents. Thus, the category labelled "well defined work" regards not only the working process, but also the content of work in terms of requirements and/or specific goals. Furthermore, "Well defined work" is different from the other factors of concentration, because the participants refer to the other three factors (cognitive workload, artefacts, and work environment) as characteristics of the work without which they could not get concentrated. In contrast, a well defined work leads the engineers to concentration.

Observation #16 *Cognitive Workload* was mentioned primarily by experienced engineers in Cases I (P007, P008, P009), II (P018, P020) and III (P030, P032). It may mean that the influence of workload over the engineers' concentration depends on their stage of the career, or it may only mean that the engineers take some time, in terms of work experience, until they figure out that cognitive overload is not benefical for their concentration. It would be frivolous, however, to assume that cognitive overload is not important or does not affect the less experienced engineers, only because they have sporadically mentioned it.

Observation #17 Project variety is a delicate characteristic of the work. As shown in *Observation #11* (p. 93), it is important for software engineers to have contact with different tasks, business domains, rules and challenges. However, this diversity of projects and tasks must be managed in a way to avoid distractions and workload problems. As shown inTable 32, working in simultaneous projects may not be an effective strategy to assure work variety, because shifting the knowledge context between completely different projects may prevent the worker's concentration.

Given the low density of the data regarding the antecedents of concentration, it was not possible to draw any further inference about the influence of other individual

characteristics (education, gender, role) over the individuals' perception of the antecedents of concentration.

5.2.3 Workplace factors of Collaboration (RQ2.3)

This subsection present the results of the cross-case comparison with respect to the research question "*RQ2.3: What workplace factors influence collaboration of software engineers?*". It was possible to find only wispy and diffuse contributions on the antecedents of collaborations. Only one participant in Case I contributed to this topic, four participants in Case II, nine in Case III, and six in Case IV. Table 33 summarizes the analysis of the answers for this question. Nevertheless, it is possible to remark that:

Observation #18 Communication and participation was representative in Cases II and III. The most relevant characteristic that these cases hold in common is that the engineers work in small teams. As proposed in Case II: "*Small teams, within a sociable environment, are likely to create the conditions to a high level of team members' commitment and cohesion*" (FRANÇA, ARAÚJO and DA SILVA, 2013). According to *Observation #13* (p. 96), the team members' engagement would more easily influence the engineers engagement in small teams.

Observation #19 knowledge exchange was representative in Case III. However, as detailed in *Observation #3* (p. 87) collaboration is not a representative trait of motivated behaviour for the engineers in that organisation. This case holds the most inexperienced software engineers in the four cases. The participants are generically in the beginning of their careers and they aim to learn as much as possible. However, at the time that the case study was conducted, that organisation was not used to offer formal training opportunities. Then, knowledge exchange indeed was a representative antecedent of collaboration, but collaboration was seen as part of their work, whose main reason to engage to was the *acquisition of useful knowledge* (Observation #9).

Observation #20 Interdependence, by definition, is a necessary condition to cooperation. However, the collaborative behaviour that characterizes a motivated engineer involves more than being cooperative. It demands being

communicative and helpful, as shown in Section 5.1 . Therefore, interdependence does not seem to be an antecedent of work motivation. Besides, interdependence was not representative in any of the case studies.

Observation #21 Team competence was occasionally mentioned in Cases I, II and IV, and it is not representative in any of these cases. Participants P019 and P042 have a short work experience, so for them, this factor, *team competence*, may only be an expansion of their *technical confidence* (Observation #13, p.96). According to Vroom (1964), it is common that some people blame the environment when things go wrong. Thus, one's judgement of *technical confidence* may also include the capabilities and collaborative support that he/she finds in the team.

Table 33 - Workplace antecedents of Collaboration

Cross-case labels	Case I	Case II	Case III	Case IV
Communication and Participation *Definition:* *Refers to the degree of interaction and participation of the team members on decisions that will impact their work.*	-	* **Team integration**{P019, P021, P022, P023} *"Eu vi um pouco de liberdade também de dizer: [" Não, isso aqui eu tenho dificuldade"] ou [" Tenho mais afinidade"]. Então, eu senti a equipe bem aberta nesse ponto, assim." (P023, in: answer to Q21)*	* **Participation**{P028, P029, P030, P032, P033, P034} *"você vê que todos os projetos lá dentro eles são discutidos e debatidos entre todo mundo, não é alguém que chega e te diz o que é que tem que ser feito," (P029, in: answer to Q34)*	**Team interaction** {P044, P048} *"Como cada um sabe suas responsabilidades e tá relativamente motivado no seu trabalho, cada um fica trabalhando individualmente, fazendo seu trabalho, a gente tem poucos momentos de interação." (P044, in: answer to Q28)*
Knowledge exchange *Definition:* *Opportunity to share knowledge and learn from coworkers' experiences*	-	**Knowledge exchange** {P022} *"essa troca de informação é muito, muito interessante. A gente consegue dar passos mais largos" (P022, in: answer to Q22)*	* **Learning with others**{P028, P029, P034, P035, P037} *"o que me deixa estimulado é que primeiro você tá passando seu conhecimento (...) mostra que é bom ele trocar experiências" (P034, in: answer to Q13)*	**Coworkers' technical knowledge** {P041, P049} *"a gente não tem pessoas maduras em determinadas áreas com as quais a gente possa aprender muito com elas" (P041, in: answer to Q7)*
Interdependence *Definition:* *Tasks are organized in a way that the work of other people depends on what the individual is doing, and vice-versa*	-	-	**Interdependence** {P031} *"eu preciso realizar alguma tarefa pra outra pessoa poder concluir aquela certa etapa do sistema," (P031, in: answer to Q24)*	**Interdependence** {P045} *"Eu to desenvolvendo, me deparo com problema, a minha tendência é já querer resolver, eu quero alterar aquela tabela, eu quero mexer (...) essa divisão é necessária para organização" (P04, in: answer to Q145)*
Team Competence *Definition:* *The individual trusts on the team effectiveness*	**Team effectiveness** {P011} *"quando eu passo uma demanda e a demanda volta, perfeitamente funcional sem erros. (...) Tem uma confiança que é importantíssimo" (P011, in: nswer to Q21)*	**(-) Incompent team** {P019} *"eu me sinto meio chateado porque eu faço a minha parte e às vezes alguns componentes da equipe não fazem da melhor forma e acaba impactando no time como um todo." (P019, in: answer to Q15)*	-	**(-) Ineffective team** {P042} *"eu não disponho de uma equipe que pudesse fazer essas melhorias mais rapidamente." (P042, in: answer to Q9)*

'P' – refers to participants. See detailed profiles at Appendices E-H.
'Q' – refers to interview script's questions (Appendix C)
(-) –exerts a negative influence

∗-indicates the representative factors

Overall, this data is not enough to support any further assertion about the antecedents of collaboration across the cases. Besides, this analysis indicates that, if "to collaborate" was rather seen as a task, its antecedents are largely consistent with those that drive engagement to do some task. It reveals, on the other hand, specific team-level conditions that mediate the collective engagement to do some task.

5.2.4 Workplace factors of Happiness (RQ2.4)

This subsection addresses the question "*RQ2.4: What workplace factors influence happiness of software engineers?*". Five participants from Case I, six (all) in Case II, seven in Case III, and eight in Case IV contributed for answering this question. The categories generated from the cross-case analysis areshown in Table 34. Two observations are highlighted below:

Observation #22 *Performance* was a consensus among almost all participants. This factor was representative in all four case studies. According to it, participants achieve happiness when they are able to perform the activities just as, or better than, the plans. This observation evidences how important the planning activity is for the happiness of engineers, showing that the plans are the primary source for the establishment of the individuals' expectations, whichare in turn responsible for their value judgement about their own performance.

Observation #23 *Feedback*, which isthe trustworthy information that the individuals get about their results and about the impacts of their work, is the second relevant category of antecedents of happiness. It is representative in Cases II and III, and comprises both recognition, which are the compliments received from peers and supervisors after the work done, as well as the visual evidence, collected by the own individuals, that the product is being useful. Feedback is a process that complements the information that is needed to the engineers evaluate their performance against the initial expectations. Feedback was not a representative factor for Cases I and IV. In both Cases I and IV, feedback was only mentioned by System Analysts, while in cases II and III it was only mentioned by

developers and testers. Thus, it may be the case that the developers from the private companies are more susceptible to the effects of feedback, because they are more frequently in contact with the end-users of their systems. Less experienced engineers also mentioned feedback more frequently than more experienced engineers.

Table 34 - Workplace antecedents of of Happiness

Cross-case labels	Case I	Case II	Case III	Case IV
Performance *Definition:* *Being able to conclude the activities just as (or better than) the plans*	* **Performance** {P006, P007, P009, P010, P011} *"As demandas de melhorias da aplicação seriam entregues todas corretas. (...) No final do dia pra concluir, deu tudo certo" (P007, in: answer to Q11)*	* **Work success**{P018, P019, P020, P021, P022, P023} *"dar tudo certo, assim, praticamente é entregar tudo, tudo o que eu tinha pra fazer naquele dia, realizar no tempo e conseguir largar no horário certo" (P019, in: answer to Q11)*	* **Performance** {P028, P030, P031, P032, P037} *"quando eu consigo produzir várias coisas num dia isso é muito importante" (P031, in: answer to Q11)*	* **Performance** {P040, P042, P043, P044, P047, P048, P049} *"pegar alguma atividade que tenha, talvez, algum desafio, alguma coisa nova, mas que seja possível fazer. Se conseguir terminar ela, foi um bom dia." (P043, in: answer to Q11)*
Feedback *Definition:* *Trustworthy information that individualsget about the results and about the impacts of their work*	**Recognition** {P010} *"As pessoas agradecem por alguma coisa que você fez e ficou legal, eu acho isso bom." (P010, in: answer to Q11)*	**Recognition**{P019} *"A partir do momento que a gente consegue fazer a liberação, que logo depois vem o elogio em cima e tal (...)aí eu me sinto satisfeito" (P019, in: answer to Q12)*	**Recognition** {P030, P036, P037} *"reconhece quando você faz uma coisa bem feita," (P030, in: answer to Q40)*	**Recognition** {P044} *"quando o cliente passa a interagir e a dar o feedback positivo do seu trabalho, isso aí também motiva" (P044, in: answer to Q36)*
	Useful products {P010} *"Esse tipo de coisa é gratificante você ver alguém realmente tá usando aquilo que você fez e tá sendo bom pra ele." (P010, in: answer to Q11)*	**(-) Unuseful product** {P022, P023} *"não me agradava muito, porque você queria ver a coisa sendo publicada , o cliente usando, já o feedback do cliente e lá não tava vendo muito isso." (P022, in: answer to Q5)*	**Customer feedback** {P032, P034} *"então é mais como uma visualização assim do pessoal usando a funcionalidade, achando aquela coisa nova, achando interessante talvez." (P032, in: answer to Q13)*	**(-) Useless product** {P040, P044, P046, P049} *"Tu passou, acho, né, que um tempão implementando altas coisas, eu cheguei no finalzinho assim, e você não vê o trabalho funcionando, ninguém usando, isso é horrível." (P049, in: answer to Q39)*

'P' – refers to participants. See detailed profiles at Appendices E-H.
'Q' – refers to interview script's questions (Appendix C)
(-) –exerts a negative influence
**-indicates the representative factors*

5.2.5 Cross-case analysis: a review of thecases

In the last four subsections, the data from the four case studies was progressively presented, and the important observations revealed by each step of the cross-case comparison was highlighted and briefly discussed. In light of these observations and arguments, this Subsection reviews the relevance of each factor in the four cases studies. We identified four situations:

A) *Not representative, and irrelevant factor:* A factor was not mentioned in a case, or it was seldom mentioned (not representative), and there is no reason to think that it is relevant for that case.
B) *Representative and relevant*: A factor is representative and relevant;
C) *Not representative, but relevant*: A factor was not mentioned in a case, or was seldom mentioned (not representative), but there are reasons to think that it is relevant;
D) *Representative, but irrelevant:* A factoris representative, but there are reasons to believe that it is not relevant.

Situations A and Bdemand no transformation in the original propositions raised from the case studies data. Situations C and D, on the other hand, represent that the relevance of a factor should be reviewed in light of the cross-case analysis. Table 35 shows the results of this review, and refers to each piece of text that presents the arguments that underpin the suggested reviews. Finally, it is important to remark two additional things about the mediating role of individual characteristics :

Observation #24 *Work experience* mediates the effects of technical confidence and engagement of co-workers. More experienced the software engineers are, less relevant is the technical confidence to their engagement, possibly because they acquire enough technical experience to deal with problems with no anticipated solution, and without fear of the failure. The more experienced the software engineers are, the more relevant is the engagement of co-workers. Experienced engineers also noticed more easily the negative impact of the cognitive workload over their concentration.

Observation #25 Our data reveal that there are other individual characteristics that mediate the effect of some antecedents of motivation, but these data do not uncover these characteristics. *Personal identification* was the label used to refer to the unknown factors that determines one's perception of what he/she likes or dislikes to work with. However, there may be other implicit characteristics that influence one's individual perception about what useful knowledge is (and what it is not), about what challenge is (and what it is not), and about what trustworthy feedback is (and what it is not).

Table 35–Review of the relevance of antecedents of motivation

Motivated behaviour traits	Antecedents	Case I	Case II	Case III	Case IV
Engagement (Observation #1)	Monetary Rewards	A (Observation #8)	A (Observation #8)	A (Observation #8)	A (Observation #8)
	Acquisition of useful knowledge*	**B** (Observation #9)	**B** (Observation #9)	**B** (Observation #9)	**B** (Observation #9)
	Social Impact*	**B** (Observation #11)	**C** (Observation #11)	**C** (Observation #11)	**B** (Observation #11)
	Work variety*	C (Observation #11)	B (Observation #11)	C (Observation #11)	C (Observation #11)
	Creative work*	**B** (Observation #10)	**B** (Observation #10)	**B** (Observation #10)	**B** (Observation #10)
	Engagement of co-workers*	**B** (Observation #13)	**B** (Observation #13)	**C** (Observation #13)	**C** (Observation #13)
	Technical confidence*	**C** (Observation #13)	**C** (Observation #13)	**B** (Observation #13)	**B** (Observation #13)
Concentration (Observation #1)	**Well defined work***	**B** (Observation #15)	**C** (Observation #15)	**C** (Observation #15)	**B** (Observation #15)
	Cognitive Workload*	**B** (Observation #16)	**C** (Observation #16)	**C** (Observation #16)	**C** (Observation #16)
	Artifacts and tools	A (Observation #14)	A (Observation #14)	A (Observation #14)	A (Observation #14)
	Work environment	A (Observation #14)	A (Observation #14)	A (Observation #14)	A (Observation #14)
Collaboration (Observation #3)	**Communication and participation***	C (Observation #18)	**B** (Observation #18)	**B** (Observation #18)	C (Observation #18)
	Knowledge exchange	A (Observation #19)	A (Observation #19)	D (Observation #19)	A (Observation #19)
	Interdependence	A (Observation #20)	A (Observation #20)	A (Observation #20)	A (Observation #20)
	Team competence	A (Observation #20)	A (Observation #20)	A (Observation #20)	A (Observation #20)
Happiness (Observation #2)	**Performance***	**B** (Observation #22)	**B** (Observation #22)	**B** (Observation #22)	**B** (Observation #22)
	Feedback*	**C** (Observation #23)	**B** (Observation #23)	**B** (Observation #23)	C (Observation #23)

***** - relevant factor

5.3 Summary of this chapter

In this chapter, data from the four case studies were presented in order to seek answers to our research questions.

Regarding the first research question "RQ1. What signs represent motivated behaviour of software engineers?", our cross-case analysis pointed out that the studied software engineers used four main adjectives to describe motivated and demotivated behaivours: engagement, concentration, collaboration and happiness. This analysis was presented in Section 5.1 (p. 80).

The second research question "RQ2. What workplace factors influence the motivated behaviour of software engineers?" was then breaken down in four, seeking for the workplace factors influencing engagement, concentration, collaboration and happiness.

After cross-comparing the four cases, the seventeen factors that were mentioned among the cases were reduced to a list of eleven relevant factors. Engagement is influenced by the following characteristics of the work: acquisition of useful knowledge, social impact, work variety, creative work, engagement of co-workers, and technical confidence. Concentration is influenced by a well defined work, and the cognitive Workload at work. Collaboration is influenced by the communication and participation atmosphere at work. Finally, happiness is influenced by feedback and performance. These data are detailed in Section 5.2 (p. 90).

However, notice that the TMS-i, presented in our methods section (p. 59) states that motivation happens before the action while satisfaction happens after. According to the TMS-i, happines is clearly a sign of satisfaction rather than motivation. Thus, it is still important to reflect and discuss on what these raw data actually mean, in terms of our initial theoretical framework. In the next chapter, we discuss this issue and others, which will help us to delineate our contribution to the initial problem, by reviewing the TMS-i and giving birth to our theory of motivation and satisfaction of software engineers.

Chapter 6 Discussion

Chapter 5 presented the cross-case analysis of the data collected in the four case studies, aimed at answering our two research questions: "*RQ1. What signs represent motivated behaviour of software engineers?*" and "*RQ2. What workplace factors influence the motivated behaviour of software engineers?*". That analysis pointed out four behavioural traits of motivated behaviour: engagement, concentration, collaboration and happiness. Then, representative and relevant antecedents of each one of these traits were identified across the cases.

In this Chapter, these results are re-discussed in light of the TMS-i, presented in Chapter 4. As previously stated, the primary objective of this research is to generate a sensible and data-driven theory of work motivation and job satisfaction for software engineers. Section 6.1 (p. 108) comments on how the answers provided for our research questions help to enhance the initial theoretical framework, and to shapethe new theory of work motivation and job satisfaction of software engineers (TMS-SE).

Nevertheless, this research is not the first academic effort towards understanding the motivation of software engineers, as discussed in Chapter 3. In Section 6.2 (p. 121), Couger and Zawacki's work, the MOCC model, similar exploratory accounts of motivation in software engineering, other empirical work available on technical literature, and even the most recent developments are thoroughly examined and compared to our enhanced theory of work motivation and job satisfaction of software engineers. In Section 6.3 (p. 131), the falsifiability and utility of the theory are evaluated, and in Section 6.4 (p. 134) we present some challenges for the software engineering practice.

6.1 Shaping the theory

Our initial theoretical framework (TMS-i) proposes that job satisfaction and work motivation refer to distinct phenomena (see Section 2.1 for a more detailed theoretical discussion). Job satisfaction is the pleasurable emotional state resulting from the appraisal of one's job as attaining, or allowing the attainment, of one's important job values, while work motivation refers to the desire to work. Motivation happens before the action, while satisfaction happens afterwards. The TMS-i suggests that job satisfaction is signalled by the happiness of the individuals at work, while the external signs of a motivated behaviour remain unknown.

In order to contribute to close the first gap in the TMS-i, our research addressed the following question: "*RQ1. What signs represent motivated behaviour of software engineers?*". The cross-case analysis, detailed in Section 5.1, concluded that Engagement, Concentration, Collaboration and Happiness are the traits that best describe the motivated behaviour of software engineers, according to the data. Following the discussion, we focus on the answers provided to the second research question: "*RQ2. What workplace factors influence the motivated behaviour of software engineers?*". The results presented in Section 5.2 are compared to the initial theoretical framework (TMS-i), which is finally enhanced to reflect our data, substantiating a new theory of motivation and satisfaction for software engineers (TMS-SE).

6.1.1 Software engineers conflate motivation and satisfaction

As a first step of the discussion, we argue that the two core concepts of the TMS-i, job satisfaction and work motivation, were conflated by the participants of our case studies. These participants were informed that the research focused on "motivation", but no definition of the term was mentioned before or during the interviews. The word "motivation" was also used in the interview script (see Appendix C). Nevertheless, happiness resulted as one of the most representative and consistent descriptor of motivated behaviour.

Happiness, in the data, is an artificial label attributed to a list of positive and negative adjectives, namely: excited/bored, good mood/bad mood, upbeat, resented. These

adjectives clearly refer to emotional states resulting from an evaluative process about some aspect of the work. However, if happiness were a true external sign of work motivation, it would be reasonable to expect that these adjectives referred to behaviours that occur before or during the action, such as the other adjectives that describe engagement, concentration and collaboration.

Table 36 - Patterns of participants' interpretations for the term 'motivation'

Definitions		**Case I**	**Case II**	**Case III**	**Case IV**
Group I	Def. 01 – It is the overall welfare in the job	P006, P007, P008	-	P031, P033, P034	P042, P046
	Def. 02 – It is the pride for working in a meaningful product	P009, P010	P020	-	P044
	Def. 03 – It is to make people feel valuable/useful/happy	-	-	P030	-
	Def. 04 – It is the pleasure/happiness from doing the work	-	P021, P023	P028, P029	P043, P044
Group II	Def. 05 – It is a voluntary effort, or extra-effort applied to a task	-	-	-	P041
	Def. 06 – It is the willingness or desire to do the work	P008, P009	P018, P020	P037	P040, P047, P048
	Def. 07 – It is the stimulus that drive the action	P019	-	-	P045
Group III	Def. 08 – It is the willingness to attain rewards available in that environment	-	-	P028, P032	-
	Def. 09 - it is the willingness to grow, to advance in the career	-	-	P031	-
Group IV	Def. 10 – It is the overall productivity	-	-	P035	-
	Def. 11 – It is having no barriers to accomplish the work	P007	-	-	-
	Def. 12 – It is the set of factors that help to achieve the organisation's goals	P011	-	-	-

In order to double check the plausibility of this conflation hypothesis, we interrogated the data to check the answers attributed to one specific question of the interview script: "Q43. How would you define the term 'motivation'?". The twenty nine valid answers (i.e. participants that did not say 'I don't know') converged to 12 patterns of interpretations for the term 'motivation', which were grouped in more general categories. Table 36 shows that interpretations focusing on emotional reactions (Group I) such as feelings, pride, happiness and pleasure, which are consistent with the definition of job satisfaction encompassed bythe TMS-i, are predominant in all case studies.This complementary analysis supported the fact that the participants conflated job satisfaction and work motivation as if they were a single phenomenon, either because (a) they do not think that they are distinguishable phenomena, or (b) they are not aware of the distinction.

According to our referential theories, Happiness, thus, is an external sign of job satisfaction, rather than work motivation.

6.1.2 Collaboration is an outcome of motivation

As a second step in the discussion, we claim that the data collected for this research is not compelling enough to argue that collaboration is an external sign of motivation (see *Observation #3*, p. 87). Then, Collaboration could, or could not, be a trait of motivated behaviour.

According to Whitehead(2007), any software development process is inherently cooperative and, through collaboration, several limitations of the process can be addressed. In fact, the word 'collaborate' is derived from a Latin word which means 'to work together'(HENNEMAN, LEE and COHEN, 1995, p. 103).Although in our data Collaboration simply represents the label attributed to the set of adjectives: communicative/reserved and helpful/unhelpful, our analysis pointed out that "to collaborate" canindeed be seen as an activity embedded in the software development work, and that its antecedents are consistent with those that drive engagement. Therefore, it would be plausible to argue that collaboration is an element of performance, and consequently an outcome of motivation, rather than an external sign.

Whitehead (2007)also shows that research in software engineering has been evolving towards the development of communication tools to support as much as possible the collaboration of software engineers. Saeki (SAEKI, 1995) also argues that communication among the team members of a development team is one of the most important characteristics for the collaborative work. Santana *et al.* (2013)evidence how the structure of communication and collaboration of a team affect the architectural design of a software artefact. Therefore, communication is already recognized as a relevant environmental condition to collaboration in software engineering, and, in this sense, our data does not add anything new to it.

6.1.3 Motivated engineers are engaged, demotivated engineers are distracted

The third argument addresses Engagement and Concentration. According to *Observation #1* (p. 86), Engagement and Concentration are the most representative descriptors of motivated behaviour across the four studies. Both descriptors are compatible with the concept of work motivation in the TMS-i, because they comprise attitudes toward the work that are perceivable before and/or during an execution of a task. Engagement was the label attributed to the following set of adjectives: Involved/Uninvolved, Hard-working/Lazy, Interested/Indifferent, Proactive/Passive; while the label Concentration was attributed to: Concentrated/Distracted, Careful/Careless, Focused/Unfocused.

Engagement and Concentration are behavioural traits consistent not only with the definitions proposed by the TMS-i, but also consistent with other concurrent theories of motivation. As Steers *et al.*(2004, p. 379) pointed out, the theories of human motivation are generally concerned with factors or events that energize, sustain (engagement) and channel (concentration) human behaviour over time. A more concrete parallel could be drawn towards the Goal Setting Theory (LOCKE, SHAW, *et al.*, 1980), which consists of the Job Satisfaction Theory author's approach to work motivation. It suggests three mechanisms through which the goal-setting practice affect performance (goal mechanisms), namely: (i) *direction*: which refers to the individual's attention; (ii) *effort*, which refers to the amount of effort mobilized in proportion to the perceived requirements of the goal or task; and (iii) *persistence*, which refers directed effort extended over time. These three goal mechanisms are directly comparable to components of engagement and concentration as appeared in our data.

However, *Observation #5* (p. 89) argues that Distraction (the negative side of concentration) described demotivated engineers better than concentration described motivated engineers. Accordingly, in *Observation #14* (p. 97), our analysis noted a tendency for participants from all cases to focus on isolated factors that disturb concentration rather than factors that enhance it, as they did when referring to engagement. *Observation #6* (p. 89) also uncovered that only few software engineers consistently used the same category to refer to motivated and demotivated behaviours. That may have happened because we included the word 'demotivated' in the interview script, so

‘motivated’ and ‘demotivated’ behaviours may not have been perceived as two opposites of the same construct. The TMS-i does not refer explicitly to the ‘demotivation’ phenomenon, so itis elaborated in the following fourarguments:

i. The Job Satisfaction Theory defines *job dissatisfaction* asthe “unpleasurable emotional state resulting from the appraisal of one’s job as frustrating or blocking the attainment of one’s job values or as entailing disvalues” (LOCKE, 1969, p. 316).Therefore, satisfaction and dissatisfaction are concurrent forces instead of opposite extremes of the same construct. In the Herzberg’s theory of job redesign (HERZBERG, 1964), this fact is interpreted through the Hygiene metaphor: some aspects of the work, when present, influence dissatisfaction, but when absent do not increase satisfaction. Herzberg (1964) called those aspects Hygienic factors because they should be sanitized to avoid dissatisfaction, even though their sanitization would not assure any satisfaction.
ii. The Job Characteristics Theory defines motivation as the desire to work. Therefore, logically, the opposite of motivation would rather be “no motivation” or “the lack of desire to work”. The term ‘demotivation’, in contrast, communicates the idea of “the desire to not work”. Nevertheless, our data illustrates that the Hygiene metaphor could also be applied to work motivation.
iii. According to our data, while motivated engineers are engaged, demotivated engineers are distracted. Both motivational and de-motivational forces co-exist in the environment, so the combination of the engagement and concentration states reveals two other situations, illustrated in Figure 15.

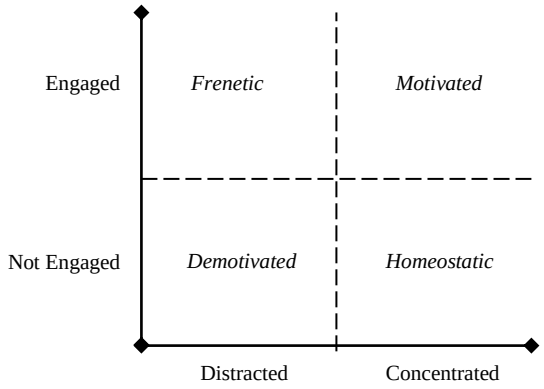


Figure 15 - Engagement x Concentration: states of motivation

iv. “Not-engaged and concentrated”represents a state defined as Homeostasis (MASLOW, 1943), or a state of balance that results in no action. Engaged but distracted indicates the influence of non-hygienic forces on the ability of the individual to get concentrated, such as constant interruptions, noise, discomfort, health conditions, among others. Following this metaphor, if the de-motivational elements were sanitized from the environment, it would not assure any motivation to the engineers

Figure 16 illustrates this interpretation of the relationship between ‘motivation’ and ‘demotivation’ concepts.

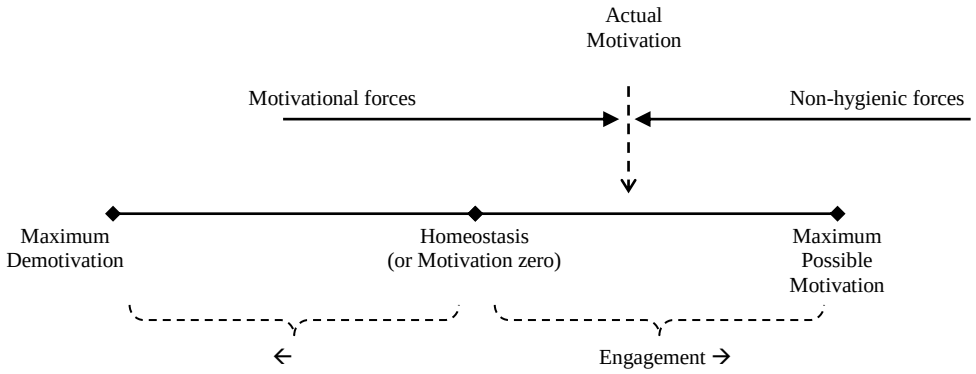


Figure 16 - Software engineers’ interpretation of motivated and demotivated behaviours

6.1.4 Other extensions to the TMS-i

The fourth step in this discussion consists of drawing a parallel between the antecedents of motivation found in the cross-case analysis and the antecedents of motivation suggested in the TMS-i.

The analysis presented in Section 5 directly supported three of the five elements from the job characteristics theory (Task identity, Task Significance, and Skill

variety).However, it presents conflicting evidence in three specific points.First, with respect to Feedback, our results suggest that it does not directly affects work motivation, instead, it appears as an antecedent of job satisfaction.Second, the idea of autonomy has appeared in a completely different way among the antecedents of motivation identified in this research. Third, there are other elements from the context, agents and the self that influence work motivation of software engineers, beside the characteristics of the tasks.

Table 37 - Job Characteristics Theory in comparison to the results of the present research

Job Characteristics Theory	Results from the present research
Task Characteristics	
Task Identity	Well defined work
Task Significance	Social Impact Acquisition of useful knowledge
Skill Variety	Work variety
Autonomy	Creative work
Feedback	-
Context	
-	Cognitive workload
Agents	
-	Engagement of co-workers
The Self	
-	Technical confidence

This parallel is summarised in Table 37, and commented below.

- ***Task identity***, in the JCT,is defined as "the degree to which the job requires completion of a whole and identifiable piece of work, doing a job from beginning to end with a visible outcome". In the JCT, it appeared as a job characteristics that shaped the *experienced meaningfulness* of the worker. In the studied software engineering contexts, the idea of *Well defined work* is partially compatible to this concept. *Well defined work*,here, means "Working in systematic tasks with clear goals, well defined requirements and predicable results". However, in the present research, instead of referring to something that "counts in his or her own system of values", a well defined work referred to something that is needed to enable their concentration.

- ***Task significance***, in the JCT, is defined as "the degree to which the work has a substantial impact on the lives of other people, whether in the immediate organisation or in the external environment." It fits perfectly in the idea of *Social Impact* that is communicated in our data, which was defined as "How the individuals perceive that their products have a social impact, i.e. benefits other people's lives, is useful or significant". Our data complements this concept, showing that the task is also perceived as significant when it has an impact on the own individual's life, through the acquisition of knowledge that may be useful to enhance their performance at work, or to expand their career opportunities.
- ***Skill Variety***, in the JCT, is defined as "the degree to which the job requires a number of different activities in carrying out the work, which involve the use of a number of different skills and talents of the individual". In the software engineering context, it is translated directly into the concept of *Work variety* as raised in our data: "The individuals have contact with different tasks, business domains, rules and challenges".
- ***Autonomy***, in the JCT,is defined as "the degree to which the work is structured to provide the employee with substantial freedom, independence, and discretion in scheduling the work and in determining the procedures to be used in carrying it out". Autonomy implies that the individuals establish an emotional connection with the product on which they are working, because the JCT assumes that being autonomous will make the individuals feel personally accountable and responsible for the results of the work. That experienced responsibility is what actually is critical in determining the individual's motivation. In our case studies, the experienced responsibility was communicated through the factor *creative work*, whose definition has apparently no relation with autonomy: "tasks that involve creating new solutions for new problems". However, the notion of creative work here has a subtle connection with the JCT's experienced responsibility. As discussed in Observation #10 (p. 92), this idea of creative work does not include only brainwork, but refers to a more refined intellectual work, one in which the individuals have the autonomy to create or suggest their own solutions, or contribute to the solution in their own way. In Case IV, for example, the participants talkedabout "authorship", i.e. something that they have created by themselves. Thus, for software engineers, autonomy is translated in

the freedom to think and create solutions, instead of the freedom to schedule and determine their own procedures to carry out the work.

- ***Feedback***, in the JCT, is defined as "the degree to which carrying out job-specified work activities provide the jobholder with direct and clear information about the effectiveness of his or her performance". In its origins, the JCT suggests that feedback may come from doing the task itself, or may also come after the performance, from other people, but "*the crucial condition is that feedback be present in form that is believable to the worker, so a realistic basis exists for the satisfaction*" (HACKMAN and LAWLER III, 1971, p. 264). Consistently, reliable feedback appeared in our data as a condition of satisfaction, instead of motivation, and it was defined as "Trustworthy information that individuals get about the results and about the impacts of their work". Nevertheless, according to the JCT and to other related theories (VROOM, 1964) (LOCKE and LATHAM, 2002), the knowledge of past work results shapes the individual's future expectations as well as their perceptions about the workplace factors, influencing, thus, indirectly their motivation in a feedback loop.

As discussed in Section 3.2 , previous research on motivation of software engineers has systematically found antecedents of work motivation that are not covered in the JCT. Consistently, and as expected, our data pointed out three workplace factors that do not pertain to the characteristics of the tasks:

- *Engagement of co-workers* refers to "how engaged the co-workers are, according to one's individual perception". All the four case studies, in isolation, raised the issue that one's state of motivation interacts with (influencing and/or being influenced by) their co-worker's motivation. This phenomenon is explained by the **Inequity Theory**(ADAMS, 1963), which is not explicitly a theory of motivation to work, but states that:

> *"the presence of inequity will motivate Person to achieve equity or reduce inequity (...)Person may increase his inputs if they are low relative to Other's inputs and to his own outcomes (...)* [or] *Person may decrease his inputs if they are high relative to Other's inputs and to his own outcomes." (ADAMS, 1963, p. 427-428)*

In this excerpt, the *other's inputs* can be translated into the *engagement of co-workers*.The inequity theory is generally very well regarded in academy (AMBROSE and KULIK, 1999) but, in contrast to TMS-i, it does not discern work motivation from job satisfaction, so inequity can be manifested in terms of both emotional or behavioural signs. The Job Satisfaction Theory (LOCKE, 1976)posits *equity*as an antecedent of job satisfaction and, consistently, more recent research provided strong evidence to the relationship between general organisational justice and individual health(ELOVAINIO, KIVIMÄKI and VAHTERA, 2002). Our case studies, on the other hand, stand as a complementary evidence of the effect of *inequity* over the desire to work, i.e. work motivation, rather than over job satisfaction.

- *Technical confidence*, is defined as "the individual belief that he/she is technically able to execute a given task". This concept represents a bridge that makes our theoretical framework consistent with the Social Cognitive Theory (SCT) (BANDURA, 1989). That theory defined *Self-efficacy* as"The strength of people's convictions in their own effectiveness" (BANDURA, 1977, p. 193). The JCT and the SCT were built on the precepts of the expectancy theory. However, the SCT was mainly developed around the idea that personal expectations influence one'smotivation, while in the JCT this idea is not explicit. The SCT asserts that the self-efficacy influences choice, as predicted in Vroom's expectancy theory (VROOM, 1964), but also determines "how much effort people will expend and how long they will persist in the face of obstacles and aversive experiences." (BANDURA, 1977, p. 194), which is consistent with our assertion that technical confidence influences engagement. Nevertheless, the TMS-i is underpinned by the JCT, and focuses on workplace factors and task characteristics rather than on individual characteristics. Thus, the idea of self-efficacy in the TMS-SE also appeared task-oriented.
- *Cognitive workload* in our data refers to "how fair and balanced the cognitive workload and the responsibilities are, at work". Our data shows that cognitive *overload* prevents the engineers' concentration. The individual perception of workload unites both the idea of inequity and self-efficacy, because the interpretation of "high" and "low" workload may be guided by how the engineers perceive the workload of their co-workers, or simply by their personal belief of

how much work they are able to effectively carry out at a time. Both arguments are usual in our data.

Finally, our case studies illustrated the pervasive mediation role of individual characteristics, as predicted in the TMS-i. The perception of the antecedents of work motivation and job satisfaction is subjective, and may completely differ from one person to another. Furthermore, individual characteristics mediate the influence of the antecedents on actual work motivation, and the influence of work motivation on performance. However, the present research was not aimed at identifying specifically what individual characteristics mediated these relationships in the studied cases. Thus, our cross-case analysis produced only suggestive evidence about unclear aspects such as*technical orientation* (see *Observation #11*, p. 93) *work experience*(see *Observation #24*, p. 104) and *personal identification* (*Observation #25*, p. 105).

6.1.5 Summary of the theory

All the topics of the present discussion were used to improve our initial theoretical model in three complementary directions:

- First, closing gaps by identifying behavioural traits of motivated software engineers, and by revealing the connection between work motivation and job satisfaction;
- Second, deepening: by showing how, as well as explaining why, task significance, skill variety, autonomy, task identity and feedback are instanced in software engineering contexts.
- Third, expanding: by identifying new workplace factors that influence motivation, and adding elements from other theories to explain why these workplace factors should not be ignored in the software engineering context.

Figure 17 presents an enhanced theoretical model that integrates the aspects discussed in this Section,and the propositions stated below. Henceforth, this enhanced theory is referred to as the Theory of Work Motivation and Job Satisfaction of Software Engineers (TMS-SE).We can interpret the relationships in this model by stating that, moderated by the one's individual characteristics:

(a) *Software engineers value **co-workers' engagement**,so their perception that **co-workers are engaged** positively influences their engagement;*

(b) *Software engineers value **technical confidence**, so their perception that they are **technically confidend** positively influences their motivation to work;*

(c) *Software engineers value **social impact** of the work, so theirperception that the work have **social impact**positively influences their motivation to work;*

(d) *Software engineers value the **acquisition of useful knowledge**, so their perception that they will **acquire useful knowledge**positively influences their own motivation to work;*

(e) *Software engineers value **work variety**, so their perception that a **work is variable**positively influences their motivation to work;*

(f) *Software engineers value **creativity**, so their perception that a work demand**creativity** influences their motivation to work;*

(g) *Software engineers value **well defined work**, so their perception of **how well the work is defined**positively influences their concentration;*

(h) *Software engineers value **cognitiveworkload balance**, so their perception of **cognitive overload** negatively influences their concentration;*

(i) *Motivated engineers are **engaged** while demotivated engineers are **distracted**.*

(j) *An individual's level of work motivation positively influences his/her own **individual performance** at work;*

(k) *An individual's level of work motivation positively influence his/her **collaborative performance** at work, proportionally to the level of **communication and participation**;*

(l) *Software engineers' discrepancy-value perceptions of the **organisational context** account for their happiness;*

(m) *Software engineers' discrepancy-value perceptions of the **agents** account for their happiness*

(n) *Software engineers' discrepancy-value perceptions of their **informed level of performance**, basedon theavailable feedback, influence the software engineers' appraisal of the workplace factors*

(o) *Job satisfaction influences attendance, continuance intention and health*

(p) (q) (r).***Individual characteristics*** *influences the software engineers' appraisal of the workplace factors before and after the actions.*

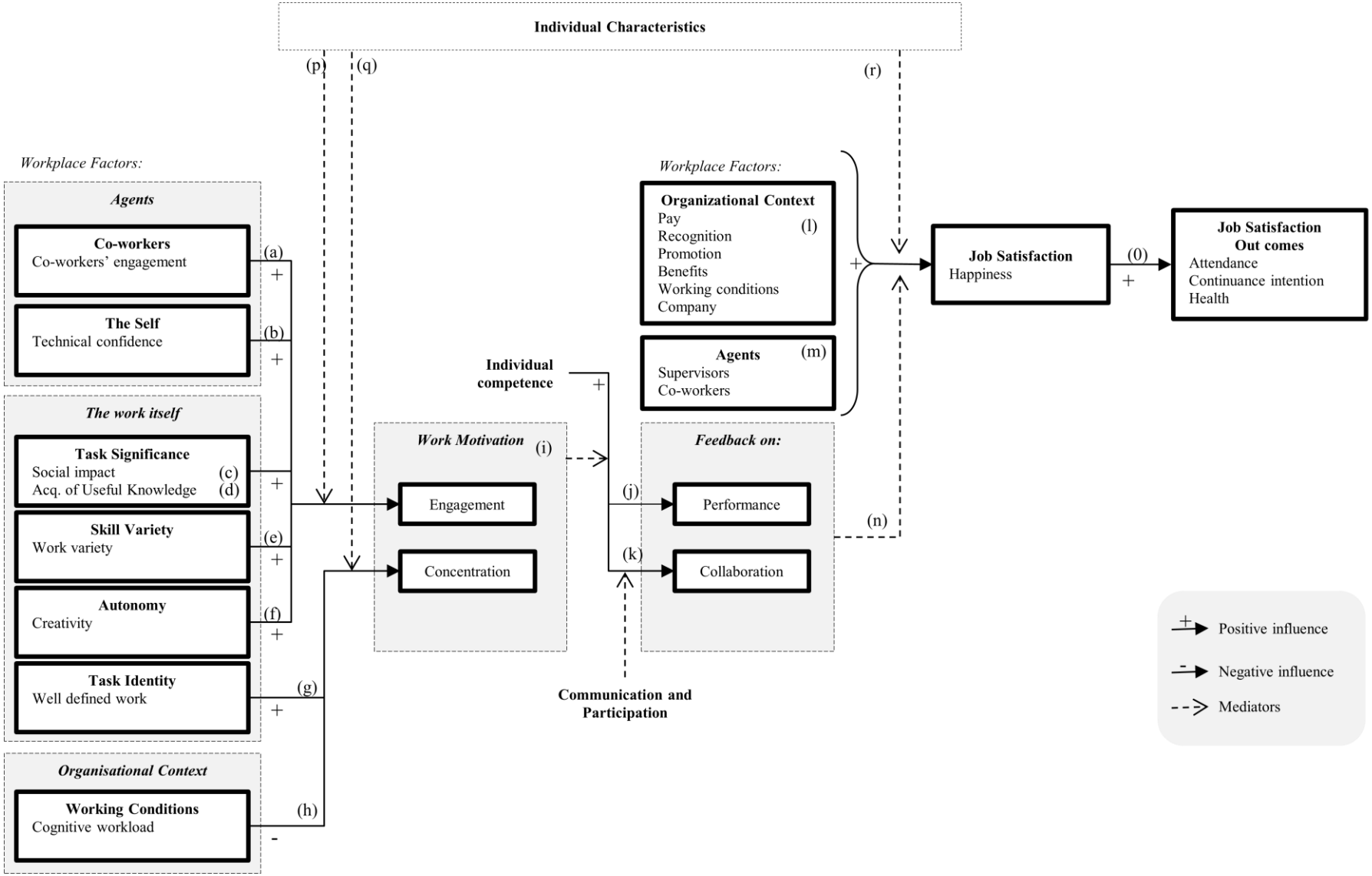


Figure 17–A Model of Work Motivation and Job Satisfaction of software engineers

6.2 Enfolding the Literature

In this subsection, the TMS-SE proposed inthe previous section is compared to the results of other studies that compose the state of art on work motivation in software engineering.

Chapter 3 shows that several studies have been conducted, but they actually presented disconnected knowledge about work motivation of software engineers because: (1) different theories (when some) have been used to underpin these studies; and (2) a wide variety of contexts and cultures have been investigated, without necessarily reporting enough information about the influence of these elements over the investigated aspects. In the following topics, the literature work is thoroughly reviewed and compared to the TMS-SE.Given theoretical and methodological differences, the studies reviewed in this section may not be directly comparable to our cross-case study, so the comparisons drawn in this subsection are limited to only pointing out superficial coincidences and discordances.

6.2.1 Couger and Zawacki's work reviewed

Couger and Zawacki (COUGER and ZAWACKI, 1980) pioneered the research on motivation of software developers. They based their research on the Job Characteristics Theory, and discovered that software engineers from all over the world exhibited similarities regarding their high growth needs (GNS), and hold specific personality traits that limit their social interactions, which causes a low level of feedback among software engineering teams, consequently compromising the motivating potential of software engineering jobs.

Although the JCT represents one of the main pillars of the theory presented in this work, we opted for not adopting the Job Diagnostic Survey questionnaire, for the reasons explained in Chapter 4 . Consequently, the GNS of the participants of our case studies was not measured. Besides, personality profiles of the case studies participants were not assessed too. Therefore, it is not possible directly to affirm that our results are fully consistent with Couger and Zawacki's findings.

Nevertheless, some aspects of the TMS-SE coincide with their thesis. First, the Growth Need Strength (GNS), in the JCT, is interpreted as the degree to which an individual values opportunities for personal growth and development at work (HACKMAN, OLDHAM, *et al.*, 1975). In our case studies, this personal growth and development at work is contained in the factor "acquisition of useful knowledge", which is representative and relevant in all case studies.

Another similarity is that in the four case studies, the engineers accused the organisation of lack of feedback from supervisors and customers. However, the low level of feedback may be either a result of a specific personality trait common in software engineers, as suggested by Couger and Zawacki, or a result of how the software development processes are designed.It is important to point out that we investigated different organisations using distinct development processes, and all four suffered from this problem.

Couger and Zawacki also claim that, given these particular characteristics, specific strategies of job design must be proposed to motivate software engineers, but they do not propose any specific strategy of job redesign. Instead, they recommend that the Goal Setting theory (LOCKE, SHAW, *et al.*, 1980)should be applied in practice, as a means to improve engineers' feedback and, consequently, their motivation. The present work complements these arguments in two aspects: first, we argue that feedback does not impact directly work motivation, but it does impact job satisfaction, which in turn influences the software engineers' perceptions about the workplace factors; second, the case studies illustrated four actual software engineering contexts in which the proposed theory reflects in different organisational practices. We also do not suggest any specific strategy to redesign software engineering jobs, but we provide enough informationto enable future research to do that based on the TMS-SE.

We do not dispute, though, the fact that the Goal Setting approach may be useful to improve work motivation. In fact, the three central elements of the goal setting theory (direction, effort and persistence) converge to our concept of engagement at work. Furthermore, the goal setting attributes act over the technical confidence of software engineers, in accordance to the Social Cognitive Theory that, as already discussed, fits to our expanded theory, the TMS-SE.

6.2.2 The MOCC Model reviewed

As argued in Chapter 3 , the MOCC Model represents the most recent advance about the motivation of software engineers. The general abstract rationale tying the elements of the MOCC are in agreement with the model presented in this work: contextual factors and individual personality influence the characteristics of the software engineers. These characteristics influence the strength of motivators, which in turn influence the outcomes of motivation, mediated by elements of the context again.

However, some aspects of the MOCC are disputed in this research. First, the MOCC model does not distinguish work motivation from job satisfaction, so it suggests that outcomes such as retention, attendance, productivity, budget adherence, project delivery time, and project success are directly influenced by the motivation of software engineers. In contrast, the TMS-SE suggests that work motivation and job satisfaction are distinguishable phenomena, with separable outcomes. The case studies illustrated two situations in which the precepts of the MOCC do not stand: (1) motivated engineers, if not satisfied, may exhibithigh intention to leave; and (2) even motivated engineers, in face of organisational hindrances, may not be productive.

Second, Sharp *et al.*(2008) classified the factors that motivate and demotivate software engineers in *intrinsic*, which come from the pleasure of doing the work itself; and *extrinsic*, refers to workplace factors external to the work. This classificationrationale reminds Herzberg's Motivation-Hygiene Theory, but this theory originally does not use the terms *intrinsic* and *extrinsic* to characterise the workplace factors, it uses *motivational* and *hygienic* in its terminology instead. Psychology research suggests that the use of the dichotomy *intrinsic* and *extrinsic* is naturally ambiguous (GUZZO, 1979), which has been expressed by the several different interpretationsreported in the literature, such as "pertaining/not pertaining to the individual", "pertaining/not pertaining to the task", "directly/indirectly connected to the outcomes of a task" and "directly/indirectly connected to the individuals' expectations".

The MOCC model, however, clearly estates that*intrinsic* and *extrinsic* mean pertaining/not pertaining to the task, but it leads to another problem: *intrinsic* and *extrinsic* are associated, respectively, to Herzberg's idea of motivation and hygiene. Critics tend to affirm that this idea is biased by the fact that people tend to take the credit when things go

well to enhance their self-worth (*intrinsic* elements are *motivational*), while they tend to blame the environment when things go wrong (*extrinsic*factors are *hygienic*) (HOUSE and WIGDOR, 1967).

We believe that this classification scheme is not helpful and, in a different direction, the TMS-SE considers that all workplace factors are *subjective*, i.e. they pertain to a interactive relationship between a person and facts of reality, in which the individuals make use of their functions of *cognition,evaluation* and *regulation* to appraise the work situations (see more details in Chapter 2).

The TMS-SE indeed refers to the Motivation-Hygiene metaphor as used in Herzberg's theory, but notice that (1) the original meaning of the metaphor is maintained and (2) we suggest a clear separation of reinforcing/hygienic factors of work motivation from reinforcing/hygienic factors of job satisfaction.

A limitation of both the TMS-SE and the MOCC model is that they do not clarify how the workplace factors combine to shape the work motivation and the performance outcomes. In the MOCC model, the long list of motivational factors make the investigation of the combination of factors too complex and impracticable. In the present model, it is clearer that each factor is singly necessary to the motivation potential of the work, although it is possible that other elements may have been concealed from our analysis. Our model also providessome support to operationalise the constructs, in order to support future research focused on more specific behavioural traits (engagement, concentration and happiness), which has already been a trend in the organisational behaviour field (AMBROSE and KULIK, 1999).

Finally, the MOCC model integrates research work conducted in many different contexts, cultures, and software development settings. If one agrees that this integration is reasonable, that model would radically describe motivation in a large variety of contexts, or conversely in actually none. In fact, as evidenced in Asghar and Usman (2013) and Melo *et al.*(2012), the representativity of the MOCC model does seem to be very limited.

6.2.3 Previous qualitative exploratory studiesreviewed

It is also possible to point out commonalitiesand differences between the present study and other previous qualitative exploratory studies, found in the systematic reviews detailed in Chapter 3 . These studies are not directly focused on work motivation or job satisfaction, but produce relevant insights into this subject. In order to organise this discussion, the nine studies are classified in three groups, according to their aims: studies that address the characteristics of good and bad software engineers, regarding their level of performance (Table 38); studies that explore HR strategies of successful IT organisations (Table 39); studies interested in occupational aspects of the software engineer professions (Table 40). Other exploratory studies explicitly interested in identifying or evaluating antecedents of work motivation and job satisfaction will be referred on later subsections.

Table 38–Qualitative studies that explore characteristics of high and low performers

Papers* (Year)	Description of the Study	Findings
PS040 (1995)	Aimed to identify behavioural traits of exceptional and non-exceptional software developers. They conducted Critical Incident Interviews with 20 software engineers, from five commercial R&D laoratories and one large company in the US.	Characterization of an exceptional developer: mastery of skills and techniques; maintains the 'big picture' view; proactivity; desire to do; sense of mission; articulate strong convictions; and help others. Non-exceptional developers: desire to contribute; perseverance; seeks help; willingness to confront others; sacrifice design.in face of a schedule pressure.
PS057 (2001)	The authors build a competence model for software developers and project managers, and evaluate the model based on thirty interviews with industry visionaries in Egypt.	The authors deliver a competency list composed of a list of 30 skillsthat make superior performers on the software engineering job, among which appear: motivating skills and self-motivation.
PS101 (2007)	This study aimed at uncovering what are the attributes of a good and a bad developer. The analysis is based on the transcripts of nine semi-structured interviews, conducted in 2004, in a large UK engineering company, that followed the CMMi level 5.	The best developers are technically competent, good communicator, confident, motivated, resourceful, committed to learn, enjoys challenge, resposible and flexible. The worst developers are technically incompetent, insecure, inflexible, over-confident, disorganised, demotivated, unprofessional.

**see the list of SLR primary studies (PS) on page 159*

The three studies in Table 38(PS040, PS057, PS101) generically agree with two aspects ofthe TMS-SE:First, motivation influences performance; and second, there are several factors that lead a software engineer to achieve high performance, and motivation is only one of these factors. High performance software engineers exhibitengagement and concentration, so they appear to be highly motivated.

Notice one difference betweenthe TMS-SE and these three studies: their interpretations of performance rely on a socially-oriented notion of performance, i.e. high

performers are those who are perceived to perform better than others. However, it is not clear whether those high performers would still be perceived as being better than the others even when they are low motivated. The belief that motivated employees perform better than de-motivated workers is a common misunderstanding. The TMS-SE argues that motivation influences the individual performance, i.e. motivated individuals perform as best as they can, which does not assure that they will be perceived as good developers or, at least, better than others. Thus, motivated engineers are not necessarily the best performers, but they will perform better than they would if they were not motivated. "How much better?" is a relevant question for future work to address.

One idea implied in these papers is that the sensitivity to motivational factors is what actually characterizedthe high performers, instead of their level of motivation, but this hypothetical relationship requiresfurther investigation.

Table 39 – Qualitative studies that explore HR strategies of successful IT organisations

Papers* (Year)	Description of the Study	Findings
PS049 (1998)	Describes the characteristics of two US companies with different levels of success seeking for HR practices that help to enable developers' performance and to avoid turnover.	The authors recommends that IS managers do: offer skill-based compensations; constantly monitor salaries against the market; offer trainings according to business needs; understand how to encourage the staff, and what each professional requires to stay interested and happy; build teamwork and sense of belonging.
PS050 (1998) PS062 (2002)	The authors intended to identify effective and innovative HR practices from successful IT organisations. They conducted surveys, and semi-structured phone interviews with professionals from 32 firms from very well regarded companies. Obs.: The two papers refer to the same study.	The study produced a taxonomy of IT retention and development practices, which look at: performance measurement; compensation and benefits systems; work arrangements; employability training and development; longer term career development; opportunities for advancement; opportunities for recognition; quality of leadership; sense of community; life style accommodations; and organisational stability.

**see the list of SLR primary studies (PS) on page 159*

In the second group (Table 39) the studies explore HR practices in successful organisations, and draw recommendation towards what other organisations should do to recruit, retain, and develop software engineers. In PS049, in particular, it is not clear whether the recommendations help to enable performance or to avoid turnover, because the author's conclusions do not seem to have been directly raised from the data. PS050 and PS062 are emphatic that the recommendations focus on retention rather than on performance. In light of our theory, the authors generically agree that hygienic factors of satisfaction must be sanitized to avoid turnover, and that the software engineers must be motivated to improve work performance.

Finally, inTable 40, two studies emphasizedthe influence of individual characteristics of the software engineers on their motivation:PS092 illustrated the influence of national cultural and economic variables over individual values;and PS124 illustrated how early life experiences shape personal interests. PS139 focused on testing tasks, and uncovered a subset of elements of our theoretical model: co-workers' engagement, work variety, and sense of responsibility.

Unfortunately, the seven studies from Table 39 and Table 40 are not appropriately underpinned by well established theories of work motivation or job satisfaction, so that the comparison to our data may be limited to sketchy assertions, superficially substantiated by descriptions of factors that are given in the texts. Nevertheless, it shows how the TMS-SE interacts to, and may be useful to guide investigations about, different topics of interest to the management of human resources in software engineering.

Table 40 – Qualitative studies that explore occupation aspects of the Software engineering profession

Papers* (Year)	Description of the Study	Findings
PS087 (2006)	This study explore the occupational culture of IS professionals. Semi-structure interviews were conducted with 10 IS professionals and 11 managers from North American companies.	This study documented some cultural beliefs of IS professionals, and reinforce that IS professionals form a sufficiently coherent community to have cultural characteristics of their own. It also shows how software engineers differ from managers in terms of culture.
PS092 (2006)	This paper describes the findings of a qualitative study which explored the career anchors of IT workers in Nigeria, based on data collected from 30 semi-structure interviews with professionals from the Nigerian Computer Society.	The results of this study point out that Nigerian professionals value: being marketable; being stable; being challenged; being balanced; being free; and being in charge. It also points out that given the economic conditions in Nigeria, IT workers are more likely to opt for jobs because of the monetary rewards rather than the personal satisfaction that it may bring.
PS124 (2009)	This paper reports a series of interviews with 30 IT professionals from Australia, aimed at understanding what drawn those individuals to choose IT as career.	The exposure to technology at an early age was the main aspect that influenced a subsequent interest in technology.
PS139 (2010)	This paper presents a case study that reports the findings of a preliminary ethnographic study (35 days of fieldwork over a period of two months) conducted at a service-based software company in the US. The focus of the study was on understanding the human-dimension and social aspects involved in software testing.	This study shows that: senior's attitudes can significantly influence junior's attitudes towards the testing practice. All participants, but one, found testing (particularly manual testing) to be monotonous after some time. On the other hand, participants felt a high sense of responsibility with the modules that they tested. Thus, this feeling of responsibility contributed to their enthusiasm.

**see the list of SLR primary studies (PS) on page 159*

6.2.4 Previous studies reviewed

According to the review of the technical literature described in Chapter 3 , several surveys aimed at evaluating the relationship between some set of workplace factors and Job Satisfaction (Type A1 papers) support the correlation between task characteristics factor and some measurement of job satisfaction. This thesis provided a theory (TMS-SE)that explains that this relationship is not direct, as it is mediated by the informational feedback provided to engineers about their performance. Type C1 paperspresented empirical studies focusing on antecedents of motivation, and provided further support to the influence of job characteristics on individual work motivation.

The influence of other workplace factors on job satisfaction was also subject of inquiries. As PS016 showed, Interesting work, Challenge, Recognition, Remuneration, and Career prospects influenced enthusiasm in their sample of software engineers. Variations of contexts and subjects may have led different studies to incur in stronger or weaker correlation levels, but this issue may not be even possible to be discussed given the generalized lack of details about the surveyed contexts. Nevertheless, all these elements are encompassed in the TMS-SE, and have different roles that are thoroughly examined and explained in this thesis.

The TMS-SE recalls the importance of treating the two phenomena – Work motivation and Job Satisfaction – separatelynot only because distinct antecedents influence them, but also because their outcomes differ. The case studies presented here evidencedthat motivated engineers may still keep high intention to leave the organisation if they are not satisfied. Conversely, the happiest software engineers may not necessarily be a good performer. Thus, this theory complements the state of art by clearly pointing out the antecedents and outcomes of each phenomenon, so managers can rely on the TMS-SE to design focused strategies to improve individuals' performance or to avoid absence and turnover.

Nevertheless, the biggest challenge for the empirical study of work motivation and job satisfaction has been the operational approach to observe or measure these phenomena. It is not clear in several previous empirical studies whether the investigated phenomena refer to job satisfaction or to work motivation, becausethere are cases in which: (a) studies intended to assess work motivation, using questionnaires that assess job satisfaction (e.g.

PS102); (b) studies intended to assess job satisfaction, using self-designed questionnaires, for which the reliability is unknown (e.g. PS009); (c) studies intended to assess self-defined constructs, which is neither exactly work motivation nor job satisfaction (e.g. PS016). Even studies that consciously focused on one of the two phenomenon, and delivery adequate questionnaires, cannot assure that the individuals were answering it correctly because, as shown in this thesis, software engineering practitioners may not have a clear picture of these phenomena, so they may conflate them when answering questionnaires.It is out of the scope of this research to provide a ready-to-go assessment questionnaire, but it provides a practical framework that can be used in future research to delineate sensible operationalization of the work motivation and job satisfaction constructs.

6.2.5 Recent developments reviewed

The systematic reviews covered a period of thirty years of research about work motivation and job satisfaction in the software engineering field, from 1980 to 2010. However, as Myers (PS024) notice, the software engineering profession is still maturing, and keeps changing over time.

Although the nature of the organisation studied in Šteinberga (2012) and Šteinberga and Šmite (2013)significantly differs from the organisations studied in this thesis, the theoretical bases have common aspects, so do the results. However, as they focused mainly on job satisfaction, their scientific contribution is limited to the instantiation of a working example of aspects covered by the Job Satisfaction theory, in an offshore practice. The TMS-SE would help them to enhance their analysis by discerning what workplace factors effectively contributed to the offshore employees' happiness and retention, and exposing what workplace otherwise influencedoffshore employees' individual performance.

Hernández-Lopez (2012) highlighted the problem of work monotony. He argued that the use of agile methodologies and practices could lead to an increment on job satisfaction because it would avoid monotonous work. However, our case studies disputed this assertion by showing that agile teams can also face the problem of lack of work variety. Jansson (2013) conducted an independent systematic review on motivation theory in research on agile project management since 2001, and found that agile processes:

(a)positively influence work motivation by providing some work variety and clearer goals; but (b) they also negatively influence work motivation by creating a psychological pressure to deliver. Thus, the software engineering process, alone, cannot determine that the engineers will be more or less motivated to work.

Verner *et al.*(2014)suggested two possible interpretations for the relationship between team motivation and overall project success: (a) team motivation contributes to the success of the software projects; or (b) successful projects make software engineers happier. We also hypothesize other two possible interpretations: (c) a third factor is responsible for both team motivation and success of projects; or (d) this relationship is just a coincidence (false-positive). The last option (d) could be ruled out based on the large variety of studies, already discussed in this chapter, that support this relationship following several different approaches. In the present thesis, we showed that the participants of our case studies do not have a clear picture of the phenomena of individual motivation. Assuming that it is also true for their participants regarding the phenomena of team motivation, then it is reasonable to infer that our work indicates that their two interpretations, (a) and (b), are true. However, it is just not clear what the term 'team motivation' refers to, in their participants' perspective.

It is important to remark that the roles of the team and of the teamwork processes received special attention in all these recent studies. In Šteinberga (2012) and Šteinberga and Šmite (2013), the personal relationships appeared as one of the most important predictors of job satisfaction. In Hernández-Lopez (2012), team collaboration was one of the most frequently cited factor of job satisfaction. In Rehman and Mahmoud (2011) and Asghar and Usman (2013), the motivational importance of supportive teams is beyond the cultural specificities of their countries. Verner *et al.*(2014) evidenced the relationship between team motivation and success of software projects. Accordingly, in our work, not only is the relationship with co-workers pointed out as an antecedent of job satisfaction, but also the engagement of co-workers influenced software engineers' work motivation.

In the beginning of the work that resulted in The Job Characteristics Theory, Hackman and Lawler (1971) suggested that the opportunity to interact with others and to make friends at work would also be relevant task characteristics, but their data did not prove so. These elements, therefore, were removed from the final version of the JCT theory. More recently, Oldham and Hackman (2005) reviewed this issue by suggesting that "For research and theory on job design, (...) the very phenomena being studied are changing. (…) *social*

interaction is now much more pervasive and prominent in contemporary work organisations than previously was the case". These two industrial case studies, along with our cross-case analysis of four additional industrial cases, represent a strong evidence for this argument.

6.3 Theoretical considerations

According to Ven (1989), a good theory must be capable of (i) advancing knowledge in a scientific discipline, (ii) guiding research toward crucial questions, and (iii) enlightening the profession of management. There is an increasing interest in the software engineering fields for the development of generalizable theories (MURPHY-HILL and WILLIAMS, 2012) (STOL and FITZGERALD, 2013), and there are known properties that make a theory stronger or weaker. Although there is no generally accepted set of guidelines for the assessment of this type of research (EISENHARDT, 1989, p. 548), in this subsection, the characteristics of the TMS-SE are reviewed according to the criteria suggested by Bacharach (1989) to evaluate organisational theories (Figure 18).

	Falsiability	**Utility**
Constructs	Construct Validity	Construct Scope
Variables	Measurement Issues	Variable Scope
Relationships	Logical Adequacy Empirical Adequacy	Explanatory potential Predictive Adequacy

Figure 18 - A framework for evaluating theories (BACHARACH, 1989, p. 502)

Falsifiability determines whether a theory is constructed such that empirical refutation is possible. According to Bacharach (1989) organisational theories are often stated in a vague way. Theories of motivation such as Maslow's (1954) and Herzberg's (1964) are examples of vagueness and ambiguity, which can be evidenced through the several distinct interpretations for both theories available in the technical literature (HALL and NOUGAIM, 1968)(KING, 1970). The criteria suggested by Bacharach (1989) to evaluate the falsifiability of a theory are:

- *Construct validity*: constructs must be clear and parsimonious. In this research, the central constructs **work motivation** and **job satisfaction** are

explicitly defined in Chapter 2 . Their definitions are constantly referred and repeated along the other chapters. Regarding the antecedents of work motivation identified in Chapter 5 , excerpts of interview and diary data are provided to clarify how the categories were constructed and to avoid misinterpretation of the attributed labels, which, according to Sandelowski and Barroso (2003), is a common source of threat of construct validity in qualitative-interpretive research.

- *Measurement Issues*: As discussed inChapter 2 , previous research on motivation of software engineer usually transferred the resposibility of determining what "motivation" means to their participants, so do questionnaires such as the JDS. As the Job Characteristics Theory treated motivation as internal feelings, it does not provide specific traits for discerning motivation from other internal feelings. In the present thesis, a different approach was adopted: behavioural traits of motivated behaviour were identified in form of syntactical textual element (adjectives) in order to serve as an practical operationalization of the construct; only then were the antecedents of each behavioural trait identified. Nevertheless, we did not set out to purposively investigate issues related to the ten antecedents of job satisfaction, so this theory relies on the Job Satisfaction Theory suggestions regarding other factors beside perfomance. Furthermore, this work does not provide a specific tool for evaluating the behavioural traits, or the antecedents of work motivation and job satsifaction. These measurement issues must be addressed by future research;
- *Logical and empirical adequacy*: In order to attend the logical adequacy criterion, Bacharach (1989) recommends that (a) the theoretical propositions must be nontautological and (b) the relationships between antecedent and consequent must be specified. In a nontautological proposition the existence of an antecedent may not logically imply the existence of the consequent. The concept of GNS in the Job Characteristics theory inserted a problem of tautology in that theory: it states that people with high GNS are motivated by the five characteristics of the work, while defines GNS as the degree to which people value those characteristics of the task. Given the evidence that software engineers generally have high GNS, we have not explictly considered this element of the JCT theory in this research. As a result, we

suggested a new antecedent of work motivation that makes the new theory consistent with the original JCT, and at the same time corrects its tautological issue. We could not identify other specific threats for logical adequacy in the proposed theory. However, in order to achieve a better level of empirical adequacy, the composites that are formed by the antecedents of engagement, concentration or happiness should be clarified by future work, as well as what individual characteristics are truly relevant

Utility refers to the usefulness of a theoretical system (BACHARACH, 1989), i.e. a theory is useful if it can both explain and predict the phenomena. Bacharach (1989) states that theoretical systems in the organisational field are often used to make predictions without providing explanations. This is the same argument with which Locke (1976)criticised the literature on work motivation and job satisfaction,before delivering his Job Satisfaction Theory. According to Bacharach (1989), the criteria used to evaluate the utility of a theory are:

- *Construct and Variable scope*:the constructs must sufficiently, although parsimoniously, tap the domain of the phenomenon in question, and variables included in the theoretical system must sufficiently, although parsimoniously, tap the domain of the constructs in question. In the present work, the phenomena in question are work motivation and job satisfaction of software engineers. Chapter 5 contains parsimonious considerations about the scope of the identified antecedents of work motivation and job satisfaction across the cases, which distinguish representativeness from relevance of the identified factors. The convergence with other theories in this field which are not part of our initial theory (TMS-i), as well as the comparisons with previous research presented in Section6.2 , illustrate that the set of antecedents of work motivation compiled from our analysis may be sufficient to determine work motivation and job satisfaction.
- *Explanatory and predictive power*:The theory presented in this thesis was raised from the cross-case analysis of four practical instances of software engineering organisations, and it provided explanations for the work motivation and job satisfaction of software engineers. Conflicting aspects within the theory, and outside its boundaries have been thoroughly discussed and reviewed over the text, which must have consolidated its explanatory and

predictive power. According to Eisenhardt (EISENHARDT, 1989), the strength of this type of theory relies on the methods and evidence grounding the theory. While there are no concise measures (such as *p* values or F values), nonetheless the reporting of information should provide confidence that the theory is valid. As Merriam (2009) suggests, for this type of research, the "burden of proof" lies with the person who is seeking to make an application elsewhere, and we reported as much as possible information to facilitate the its transference. She argues that "when we give proper weight to local conditions, any generalization is a working hypothesis not a conclusion" (MERRIAM, 2009, p. 225). Murphy-hill and Williams (2012) suggest that the transference of theories in software engineering should take to account similarities of organisations, people, activity and time between the present study and the target context, but for the reasons addressed before, we believe that the predictive power of the TMS-SE can be safely extrapolated to a large variety of software engineering environments.

6.4 Challenges for the Software Engineering practice

Analysing the work motivation and job satisfaction of software engineers from the perspective of the TMS-SE reveals practical challenges for the software engineering practice, which range from the estimation processess to the relationship with users and customers. In this subsection, we look at each proposition of the theory to suggest how software engineers should care about the several characteristics of the workplace.

- *Co-worker's engagement*: our case studies show that a highly motivated engineer may contamine the others, as well as a poorly motivated engineer may negatively influence the others. The first challenge for software engineering team leaders is to identify the poorly motivated engineers before the contamination occurs, to avoid a generalized loss of performance. The second challenge is to identify the higly motivated engineers in order to use this contamination process to leverage the motivation of the other team workers. Both these process must be conducted carefully to avoid the introduction of inequity in the team, because that would set off a risk for their job satisfaction.

- *Technical confidence*: providing software engineers with training and opportunities to experience and learn about new technologies, tools and methods, before actually adopting them,may be the simplest form of enhancing their technical confidence. However, the costs and time consumption associated to training and experimentations may rule out these practices, specially from less mature organisations or/and those that rely on tight budgets. Our case studies show that the estimation process can also be a relevant source of leveraging technical confidence: reliable estimations build up the engineers technical confidence, while unreliable estimations wipe out the engineers' belief that they can deliver timely results.
- *Social impact*: the four organisations studied in this work developedsoftware systems supposed to benefit other people's lives or other organisations' processes efficacy, so it was not difficult to notice in the software engineers a motivated behaviour justified by the sense of responsibility for the social impact caused by their work. This may represent, though, a challenge for software projects that are in their initial stages, with no actual users yet. In one of our case studies, we could evidence that the motivating role of the contact with users was replaced by a strong persuation process from the company directors, because the product had not been sold yet.
- *Acquisition of useful knowledge*: Our case studies showed that, provided that technology as whole is constantly evolving, long term projects tend to lose their novelty appeal over time. Thus, another challenge in the software engineering practice is how to cope with the engineers' need for constant learning in those types of project, whitout introducing technical risks related to the change of technologies. Figuring out what is useful for each engineer may already be a challenge for team leaders. Our case studies show that "useful" may be something that improve the quality of the product, the engineer's future performance, or even enhance their opportunity to find other jobs. Finally, provided that managers find out what "useful" mean to their engineers, they could design specific incentive strategies using the useful knowledge as a reward, alternatively to money or other monetary incentives.
- *Work variety*:There are two challenges concerning work variety. First, how to assure work variety in a single-project setting. The organisation in Case Study 4

has a large product, whose maintenance consumes a significant deal of workforce, and the work variety leans on knowing the different modules of the project. In Case Study 2, in contrast, the work variety leans on finishing short-term projects to start other projects in different domains of knowledge. Both examples represent practical forms of avoiding monotonous work, but the work variety in Case 4 is limited by the project boundaries while in Case 2 there is not alimit in sight. The second challenge is how to assure work variety without stressing the engineers in multi-project settings. Our studies show that changing the allocation of an engineer in a course of a project to which he/she is engaged may actually cause the opposite effect. Accordingly, allocating engineers in simultaneous projects may also hamper their work motivation.

- *Creativity*: An intrinsic characteristic of the productive process in software engineering work is that it is composed by series of activities that continuously alternate from highly creative to non-creative tasks. Although we have not elaborated our analysis upon this issue, it was possible to notice that the participants of our case studies tended to focus on the creative part of a task when referring to their favourite tasks, and tended to focus on the non-creative part when talking about the tasks that they dislike. However, all the tasks seemed to have both creative and non-creative steps. The challenge for software engineers is how to maintain high levels of work motivation during the phases in which they are executing the non-creative parts of their work.
- *Well defined work*: challenges regarding this factor concern several problems common in software engineering projects settings, such as how well the requirements are elicited and documented(WALIA and CARVER, 2009), how well transitory artifacts are understood(CORNELISSEN, ZAIDMAN, *et al.*, 2000), how well the productive process is defined (WANGENHEIM, HAUCK, *et al.*, 2010), etc. It is needless to reinforce how challenging a good definition of the software engineering work is, because these problems *per se* comprise large areas of research in the software engineering field. How to deal with *Workload* is also associated to problems that have been investigated in software engineering since its begginnings, such as in Brook's observations (BROOKS, 1975).
- *Communication, participation and Collaboration*: Our case studies show that software engineers tend to exhibit a more collaborative behaviour when they are

highly motivated to work. However, this influence is mediated by the degree of communication and participation that they have in the team. Previous research has suggested that participation is an antecedent of work motivation (SHARP, BADOO, *et al.*, 2008) but, in fact, the available evidence in software engineering literature only testify weak connections between participation and job satisfaction. The challenge for the practice implied in the TMS-SE is that improving communication channels and participation procedures will not assure higher levels of collaboration, unless the engineers are motivated to work.

- *Feedback*: Recent work on feedback in software engineering (SACH, 2014) uncovered several informational properties that determine the effect of a feedback on an individual job satisfaction, such as the content, the source and the medium of the information. Thus, managers should also figure out how to deal with all these variables in order to administer proper feedbacks for their engineers.

Chapter 7 Conclusions and Future Work

Issues related to work motivation and job satisfaction, since a long time, have attracted the curiosity researchers from all over the world, due not only to the beauty and complexity of the study of human behaviour, but also to the practical business benefits that the enhancement of individuals' performance represent.

This research is neither the first one to address motivation of software engineers in the theoretical level, nor the first empirical study, nor the first qualitative case study, nor the first to suggest a model for motivation of software engineers. However, to the best of our knowledge, it is the first research work to weave all these elements together. In this thesis, a theory of work motivation and job satisfaction of software engineers (TMS-SE) is proposed, based initially on the Job Satisfaction and Job Characteristics theory, enhanced and adapted for the software development context based on the findings of a multi-case study conducted in Brazil. The present work contributes to the current state of art in several complementary ways:

- By providing a solid theoretical framework, that was initially based on two well-regarded theories – the Job Satisfaction Theory and the Job Characteristics Theory - enhanced to cover the software engineering specificities, borrowing elements from the socio-cognitive theory and from the inequity theory to substantiate the inclusion of additional motivational workplace characteristics: technical confidence and co-workers' engagement respectively.
- By witnessing that software engineering practitioners have no clear picture of the different facets of their organisational behaviour, and are likely to refer to motivation as an umbrella-term that enfolds several different phenomena. In this case, we showed that the participants of our four case studies conflated job satisfaction and work motivation as a single construct, and this may represent a challenge to the interpretation of results from empirical research on this topic, in particular to those studies that rely on poorly designed self-administrable questionnaires. Our work uncovered observable traits of motivated, not motivated, demotivated and satisfied engineers, which have not been addressed by previous research. Although we do not provide questionnaires to assess

work motivation and job satisfaction, we provide as much as possible information to support future research toward the objective operationalization of these constructs.

- By pointing out practical challenges for the software engineering practices, showing how teamwork, training, estimation processes, product content, delivery strategies, technological evolution, well defined processes, quality of tools and artifacts, an other practical issues, may affect the work motivation and job satisfaction of software engineers.

This research effort also contains some elements and attitudes that indirectly contribute to the general research in motivation in software engineering. These elements and attitudes. although are not direct products of this thesis, are worthy mentioning, because:

- it reclaims the importance of treating work motivation and job satisfaction as two distinguishable phenomena, with different antecedents, behavioural signs, and outcomes. This is not a direct result of this thesis, but rather an innovative theoretical approach in the software engineering field. Nevertheless, the cross-case results provided evidence that this theoretical approach helpedus to enhance our discernment about what workplace factors effectively contribute to engineers'happiness and retention as a consequence, as well as what workplace otherwise influenceengineers' individual performance by means of work motivation, which is not clear in previous available models such as in the MOCC model.
- it reinforces that work motivation influences individual performance, including collaboration, and that individual performance contributes to job satisfaction, in agreement to other exploratory studies conducted in this field, but making two contentions to avoid overstatements that are common in this type of research. First,there are several factors that determine higher performance of a software engineer, and work motivation is but one of these factors. Second, work motivation does not assure that software engineers will become the best performers of a group, but will only assure that software engineers will perform as best as they are able to.
- it stresses the pervasive actuation of individual characteristics, which have been pointed out in previous studies as moderators of(1) the influence of workplace factors on work motivation, (2) the influence of work motivation on

performance, and (3) the influence of performance on job satisfaction. Although we have not set out to identify all relevant individual characteristic, we believe that the cultural aspects pointed out in most recent studies can also be included in this category of individual characteristics.

- As discussed all over this thesis, some researchers argue that software engineers hold individual charateristics that distinguishes them from the overall population, while other researchers show conflicting evidence. This work also shows that the nature of software engineering tasks work per secreates specific conditions that alter the motivational structure of these professionals.
- it supports the fact that software engineering projects suffer from generalized a lack of feedback, as suggested by Couger and Zawacki (1980), and by explaining that feedback does impact job satisfaction, which in turn influences the software engineers' perceptions about the workplace factors.
- it serves as a practical and detailed worked example of theory building based on interpretive multi-case studies, for which we believe that there may be not many examples available in the software engineering literature. The software engineering scientific discipline is also starting to care about a more systematical development of theories. Thus, this work also contributes to future research from a methodological perspective, adding to general body of knowledge of Empirical Software Engineering.

Comparing previous theoretical and empirical developments from the last thirty years in the software engineering field, it is possible to assert the generality TMS-SE. However, the theory, as currently stated, still leaves many open questions and opportunities for future research, such as:

- *Towards the development of measurement and/or assessment tools*. Although Locke (1969) warned that "intensity of satisfaction" and "value importance" might not be measured in terms of any known physical or psychological unit, the measurement of subjective phenomena related to the organisational behaviour turned into a large research area, addressing such complex problems (BELFO and SOUSA, 2011). The TMS-SE provideda conceptual approach to work motivation and job satisfaction, that may be useful to support the design of measurement and assessment tools in future research;

- *Towards a combination (and a scale of importance) of the antecedents of work motivation and job satisfaction*: A limitation of the TMS-SE is that it does not explicits how the workplace factors combine to shape the work motivation and the performance outcomes. Some factors may be more or less influent according to extraneous contextual variables, or internal individual characteristics. Further, there may be interactions among the factors. The approach followed in da Silva and França (2012) may be useful to help design research aimed at looking for a combination or a scale of importance among these workplace factors.
- *Towards the design of intervention strategies*: The TMS-SE does not suggest any specific strategy to redesign software engineering jobs.França and da Silva (2010) have made some effort to design managerial strategies to enhance the work motivation of software engineers. Although their methodological approach is valid, their theoretical framework is questionable.Therefore, an adaptation of those motivational schemes to the TMS-SE framework would be sound;
- *Towards quantifying the impact of work motivation on individual performance.* The TMS-SE claims that motivated engineers perform better than they would if not motivated. This claim is supported by enough technical literature studies in, and out of, the software engineering field. However, a question that is still left unsolved is "How much better"? Answering that question would help managers and practitioners in general to quantify their gains or losses in terms of budget, which would attract even more attention to the human aspects in software engineering projects.
- *Towards a deep understanding of the work motivation factors*: Sach (2014) aimed to explore the role of feedback on the motivation of software engineers. He identified several properties of the feedback that mediates its influence on the job satisfaction of software engineers, such as content (positive/negative), source, medium, and others. Similar researches could be conducted to clarify the properties of the other motivators, such as co-workers' engagement, technical confidence, useful knowledge, social impact, work variety, creativity, well defined work and workload.

Finally, considering other themes borderline discussed in this work, other research opportunities emerge, and we recommend that future research should also evolve in the following directions:

- *Towards a better understanding of individual performance in software engineering*. The TMS-SE shows that work motivation is one among the several workplace factors that influence individual performance at work. What are the others? Indeed, there already must be researchers pursuing the answer for this research, andthe TMS-SE may be useful to help research that address other specific elements of performance to isolate human aspects that represent noise, or even baffle their data.
- *Towards a better understanding of the individual characteristics that matters*: Along the cross-case analysis, we have mentioned some individual characteristics that appeared in our data, such as work experience, technical orientation, and national culture, but we opted to put all these aspects together in a single category in the TMS-SE. However, we strongly recommend future research to develop more systematic investigations addressing these characteristics and other, like individual values, personality profile, cognitive styles, career stages, to map what characteristics conditions a better or a worse fit of software engineers to a determined work, and how. As an example, previous research implied that software engineers more sensitive to the workplace motivators tend to be better performers. Would that be true?
- *Towards a better understanding of work motivation in teams*: Teamwork in software engineering is also an area that increasingly attracts academic and industrial attention. Team motivation is "the collective system by which team members coordinate the direction, intensity, and persistence of their efforts." (CHEN and KANFER, 2006, p. 233). According to Chen and Kanfer (2006), the team motivation cannot be understood without having a clear understanding of the work motivation of the members of a team. The TMS-SE can be used to guide investigation in this area. Other researchers may be interested in investigating the effects of team processes (MARKS and MATHIEU, 2001) over interactive aspects of the TMS-SE, such as the perception of co-workers' engagement, or the collaborative performance.

- *Towards the understanding of the motivational role of leadership at work.* Still in this context of teamwork, several studies recall the importance that leaders have on determining the effectiveness of the motivational strategies.The concepts of transformational and transactional leadership of Burns (1978), for example, communicate two types of leaders that adopt different strategies to intervene in the team members' work motivation.The TMS-SE can offer a theoretical framework to interpret and predict how the attitudes and decisions of leaders will influence the work motivation of software engineers.

References

THESIS REFERENCES

ADAMS, J. S. Toward and understanding of inequity. **Journal of abnormal and social psychology**, v. 67, n. 5, p. 422-436, 1963.

ALDERFER, C. P. An empirical test of a new theory of human needs. **Organizational Behavior and Human Performance**, v. 4, n. 2, p. 142-175, 1969.

AMBROSE, M. L.; KULIK, C. T. Old Friends, New Faces: Motivation Research in the 1990s. **Journal of Management**, v. 25, n. 3, p. 231-292, 1999.

ARAÚJO, A. C. M. L. D. **Um estudo de caso sobre motivação em integrantes de equipes de desenvolvimento de software no contexto de uma organização privada**. Universidade Federal de Pernambuco. Recife. 2011.

ARGYRIS, C.; SCHON, D. **Theory in Practice - Increasing Professional Effectiveness**. [S.l.]: Jossey-Bass, 1974.

ASGHAR, I.; USMAN, M. **Motivational and De-motivational Factors for Software Engineers:** an Empirical Investigation. Proceedings of the 2013 11th International Conference on Frontiers of Information Technology. [S.l.]: IEEE Computer Society. 2013. p. 66-71.

BACHARACH, S. B. Organizational Theories : Criteria Some for Evaluation. **Academy of Management Journal**, v. 14, n. 4, p. 469-515, October 1989.

BAKER, T. B.; MORSE, E.; SHERMAN, J. E. The motivation to use drugs: A psychobiological analysis of urges. **Proceedings of the Nebraska Symposium on Motivation**, v. 34, p. 257-323, 1986.

BANDURA, A. Self-efficacy: toward a unifying theory of behavioral change. **Psychological Review**, v. 84, n. 2, p. 191-215, 1977.

BANDURA, A. Human Agency in Social Cognitive Theory. **American Psychologist**, v. 44, n. 9, p. 1175-1184, 1989.

BAYTON, J. Motivation, Cognition, Learning - Basic factors in consumer behavior. **Journal of Marketing**, v. 22, n. 3, 1958.

BECK, K. et al. Twelve Principles of Agile Software. **Agile Manifesto**, 2001. Disponivel em: <http://agilemanifesto.org/principles.html>. Acesso em: 10 fev. 2014.

BEECHAM, S. et al. Motivation in Software Engineering: A systematic literature review. **Information and Software Technology**, v. 50, p. 860-878, 2007. ISSN 10.1016/j.infsof.2007.09.004.

BELFO, F.; SOUSA, R. D. Developing an Instrument to Assess Information. **ENTERPRISE INFORMATION SYSTEMS: Communications in Computer and Information Science**, v. 220, n. 3, p. 230-239, 2011.

BOZIONELOS, N. The big five of personality and work involvement. **Journal of Managerial Psychology**, v. 19, n. 1, p. 69-81, 2004.

BROOKS, F. P. **The Mythical Man-Month:** Essays on Software Engineering. [S.l.]: Addison-Wesley, 1975.

BURN, J. M. et al. **Motivating IT Professionals - The Hong Kong Challenge**. Proceedings of the Twenty-Fourth Annual Hawaii International Conference on System Sciences. Kauai, HI: [s.n.]. 1991. p. 524 - 529.

BURNS, J. M. **Leadership**. New York: Harper & Row, 1978.

CAPRETZ, F. Personality types in software engineering. **International Journal of Human-Computer Studies**, v. 58, n. 2, p. 207-214, 2003.

CARNEIRO, D. E. S. **Motivação em integrantes de equipes de desenvolvimento de software no contexto de uma empresa privada: um estudo de caso**. Universidade Federal de Pernambuco. Recife. 2011.

CARVER, J. C. The Use of Grounded Theory in Empirical Software Engineering. In: BASILI, V.; ROMBACH, D.; SCHNEIDER, K. **Empirical Software Engineering Issues**. [S.l.]: Springer, 2006. p. 42.

CHEN, G.; KANFER, R. Toward a Systems Theory of Motivated Behavior in Work Teams. **Research in Organizational Behavior**, v. 27, n. 6, p. 223-267, 2006.

CORNELISSEN, B. et al. A Systematic Survey of Program Comprehension through Dynamic Analysis. **IEEE Transactions on Software Engineering**, p. 1-20, January 2000.

COUGER, J. D.; ADELSBERGER, H. Comparing Motivation of Programmers and Analysts in Different Socio/Political Environments: Austria Compared to the United States. **ACM Computer Personnel**, v. 11, n. 4, p. 13-17, 1988.

COUGER, J. D.; BOROVITS, I.; & ZVIRAN, M. **Comparison of motivating environments for programmer/analysts and programmers in the US, Israel and Singapore**. Proceedings of the Twenty-Second Annual Hawaii International Conference on System Sciences. Volume IV: Emerging Technologies and Applications Track. Hawaii: IEEE. 1989. p. 316-323.

COUGER, J. D.; CALLAGHAN, R. O. Comparing the motivation of spanish computer personnel with that of computer personnel in Finland and the United States. **European Journal of Information Systems**, v. 3, p. 285-291, 1994. ISSN doi:10.1057/ejis.1994.33.

COUGER, J. D.; HALTTUNEN, V.; LYYTINEN, K. Evaluating the motivating environment in Finland compared to the United States-a survey. **European Journal of Information Systems**, v. 1, n. 2, p. 107-112, 1991. ISSN doi:10.1057/ejis.1991.18.

COUGER, J. D.; ISHIKAWA, A. **Comparing motivation of Japanese computer personnel versus these of the United States**. Proceedings of the Twenty-Eighth Hawaii International Conference on System Sciences. [S.l.]: IEEE Comput. Soc. Press. 1995. p. 1012-1019.

COUGER, J. D.; OPPERMANN, E. B.; AMOROSO, D. L. Changes in Motivation of I.S. Managers: Comparison Over a Decade. **Information Resources Management Journal**, v. 7, n. 2, p. 5-13, 1994.

COUGER, J. D.; ZAWACKI, R. A. **Motivating and Managing Computer Personnel**. New York: Wiley-Interscience, 1980.

CRUZES, D. S. et al. **Case Studies Synthesis:** Brief Experience and Challenges for the Future. Proceeding of the International Symposium on Empirical Software Engineering and Measurement 2011. Banff: [s.n.]. 2011. p. 343-346.

CRUZES, D. S.; DYBÅ, T. **Synthesizing evidence in software engineering research**. Proceedings of the 2010 ACM-IEEE International Symposium on Empirical Software Engineering and Measurement. New York: ACM Press. 2010.

CRUZES, D. S.; DYBÅ, T. **Recommended Steps for Thematic Synthesis in Software Engineering**. International Symposium on Empirical Software Engineering and Measurement (ESEM). Banff, AB: [s.n.]. 2011. p. 275–284.

CRUZES, D. S.; DYBÅ, T. Research synthesis in software engineering: A tertiary study. **Information and Software Technology**, v. 53, n. 5, p. 440-455, 2011. ISSN 10.1016/j.infsof.2011.01.004.

CURTIS, B.; HEFLEY, W. E.; MILLER, S. A. **People Capability Maturity Model (p-CMM)**. Software Engineering Institute, Carnegie Mellon. [S.l.]. 2001. (CMU/SEI-2001-MM-001).

DA SILVA, F. Q. B. et al. Protocolo para Estudos de Caso sobre Motivação de Engenheiros De Software. **HASE - Human Aspects in Software Engineering**, Recife, PE, 2011. Disponivel em: <http://goo.gl/DaLZY>.

DA SILVA, F. Q. B.; FRANÇA, A. C. C. Towards understanding the underlying structure of motivational factors for software engineers to guide the definition of motivational programs. **Journal of Systems and Software (JSS)**, v. 85, n. 2, p. 216-226, 2012.

DE FARIAS JUNIOR, I. H. et al. **Motivational Factors for Distributed Software Development Teams**. Proceedings of the 2012 IEEE Seventh International Conference on Global Software Engineering Workshops (ICGSEW). Porto Alegre, Brazil: [s.n.]. 2012. p. 49-54.

DEARMAN, D.; KELLAR, M.; TRUONG, N. K. **An Examination of Daily Information Needs and Sharing Opportunities**. Conference on Computer supported cooperative work. New York, NY, USA: ACM. 2008.

DEMARCO, T.; LISTER, T. **Peopleware:** Productive Projects and Teams. [S.l.]: Dorset House, 1987.

DENGATE, G.; COUGER, J. D.; WEBER, R. Motivational characteristics of Australian information systems personnel. **Australian Computer Journal**, v. 22, n. 3, p. 77-87, 1990.

DENZIN, N. K.; LINCOLN, Y. S. **The Sage Handbook of Qalitative Research**. Third edition. ed. Thousand Oaks, CA: Sage, 2005.

EASTERBROOK, S. et al. Empirical research methods for software engineering. In: **_____ GUIDE TO ADVANCED EMPIRICAL SOFTWARE ENGINEERING**. [S.l.]: Springer Science + Business Media, 2008. p. 285-311.

EERDE, W. V.; THIERRY, H. Vroom's Expectancy Models and Work-Related Criteria: a meta-analysis. **Journal of Applied Psychology**, v. 81, n. 5, p. 575-586, 1996.

EISENHARDT, K. M. Building Theories From Case Study Research. **The Academy of Management Review**, v. 14, n. 4, p. 532-550, Oct 1989.

EL KHATIB, V. et al. **The Role of Motivational Factors in Distributed Software Development Teams:** An Empirical Investigation. Proceedings of the 2013 21st European Conference on Information Systems (ECIS). [S.l.]: [s.n.]. 2013. p. 12.

ELOVAINIO, M.; KIVIMÄKI, M.; VAHTERA, J. Organizational Justice: Evidence of a New Psychosocial Predictor of Health. **American Journal of Public Health**, v. 92, n. 1, p. 105-108, 2002. ISSN doi: 10.2105/AJPH.92.1.105.

FELIX, A. D. L. C. **Um estudo de caso sobre motivação em integrantes de equipes de desenvolvimento de software no contexto de uma organização pública**. Universidade Federal de Pernambuco. Recife. 2011.

FERRATT, T. W.; SHORT, L. E. Are Information Systems People Different: An Investigation of Motivational Differences. **MIS Quarterly**, v. 10, n. 4, p. 377-387, 1986.

FLYVBJERG, B. Five Misunderstandings About Case-Study Research. **Qualitative Inquiry**, v. 12, n. 2, p. 219-245, 2006.

FRANÇA, A. C. C. et al. **Motivation in software engineering:** a systematic review update. International Conference on Evaluation & Assessment in Software Engineering (EASE). Durham: [s.n.]. 2011. p. 154-163.

FRANÇA, A. C. C. et al. Motivation in Software Engineering Industrial Practice: A Cross-Case Analysis of Two Software Organisations. **Information and Software Technology**, v. Available online, 2013.

FRANÇA, A. C. C.; ARAÚJO, A. C. M. L. D.; DA SILVA, F. Q. B. **Motivation of Software Engineers:** A Qualitative Case Study of a Research and Development Organisation. Proceedings of 6th International Workshop on Cooperative and Human Aspects of Software Engineering (CHASE 2013). San Francisco, CA: [s.n.]. 2013.

FRANÇA, A. C. C.; CARNEIRO, D. E. S.; DA SILVA, F. Q. B. **Towards an Explanatory Theory of Motivation in Software Engineering:** A Qualitative Case Study of a Small Software Company. Proceedings of the Brazilian Symposium of Software Engineering. Natal, Brazil: [s.n.]. 2012.

FRANÇA, A. C. C.; DA SILVA, F. Q. B. **Designing motivation strategies for software engineering teams**. Proceedings of the 2010 International Workshop on Cooperative and Human Aspects of Software Engineering (CHASE). Cape Town, SA: ACM Press. 2010. p. 84-91.

FRANÇA, A. C. C.; DE ARAÚJO, A. C. M. L.; DA SILVA, F. Q. B. **Appendices of : Towards an Explanatory Theory of Motivation in Software Engineering : A Qualitative Case Study of a Research and Development Organization**. HASE - Human Aspects in Software Engineering. Recife. 2012.

FRANÇA, A. C. C.; FELIX, A. D. L. C.; DA SILVA, F. Q. B. **Appendices of : Towards an Explanatory Theory of Motivation in Software Engineering : A Qualitative Case Study of a Government Organization**. HASE - Human Aspects in Software Engineering. Recife. 2012.

FRANÇA, A. C. C.; FELIX, A. D. 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 (EASE). Ciudad Real, Spain: [s.n.]. 2012. p. 72-81.

GLASER, B.; STRAUSS, A. **The discovery of grounded theory:** Strategies of Qualitative research. London: Wiedenfeld and Nicholson, 1967.

GOLEMBIEWSKI, R. T. **Handbook of Organizational Behavior**. 2. ed. New York: Marcel Dekker Inc, 2000.

GUINAN, P. J.; COOPRIDER, J. G.; FARAJ, S. Enabling Software Development Team Performance Dureing Requirements Definition: A Behavioral Versus Technical Approach. **Information Systems Research**, v. 9, n. 2, p. 101-125, June 1998.

GUZZO, R. A. Types of Rewards, Cognitions, and Work Motivation. **The Academy of Management Review**, v. 4, n. 1, p. 75-86, 1979.

HACKMAN, J. R. Nature of the task as a determiner of job behavior. **Personnel Psychology**, p. 435-444, 1969.

HACKMAN, J. R. Work redesign and motivation. **Professional Psychology**, v. 11, n. 3, p. 445-455, 1980.

HACKMAN, J. R. et al. A New Strategy for Job Enrichment. **California Management Review**, p. 57-71, 1975.

HACKMAN, J. R. J. Richard Hackman. **Scholars at Harvard**. Disponivel em: <http://scholar.harvard.edu/rhackman/>. Acesso em: 06 jan. 2014.

HACKMAN, J. R.; LAWLER III, E. E. Employee reactions to job characteristics. **JOURNAL OF APPLIED PSYCHOLOGY**, v. 55, n. 3, p. 259-286, 1971.

HACKMAN, J. R.; OLDHAM, G. R. **The Job Diagnostic Survey: An instrument for the diagnosis of Jobs and the evaluation of job redesign projects**. Department of Administrative Sciences, Yale University. New Haven, p. 88. 1974.

HACKMAN, J. R.; OLDHAM, G. R. Development of the Job Diagnostic Survey. **Journal of Applied Psychology**, v. 60, n. 2, p. 159-170, 1975.

HACKMAN, J. R.; OLDHAM, G. R. Motivation through the design of work: test of a theory. **Organizational Behavior and Human Performance**, p. 250-279, 1976.

HALL, D. T.; NOUGAIM, K. E. An examination of Maslow's need hierarchy in an organizational setting. **Organizational Behavior and Human Performance**, v. 3, n. 1, p. 12-35, 1968.

HALL, T. et al. A systematic review of Theory use in studies investigating the motivations of software engineers. **ACM Trans. Soft. Eng. Methodol.**, v. 18, n. 3, 2009. ISSN 10.1145/1525880.1525883.

HENNEMAN, E. A.; LEE, J. L.; COHEN, J. I. Collaboration: a concept analysis. **Journal of Advanced Nursing**, v. 21, p. 103-109, 1995.

HERNÁNDEZ-LÓPEZ, A. Satisfaction and Motivation: IT's practitioners' perspective. **International Journal of Human Capital and Information Technology Professionals**, v. 3, n. 4, p. 39-56, October-December 2012.

HERTEL, G. **Management virtueller Teams auf der Basis sozialpsychologischer Theorien:** das VIST-Modell. Sozialpsychologie wirtschaftlicher Prozesse. Lengerich, Germany: Pabst Science Publishers. 2002. p. 174-204.

HERZBERG, F. One More Time: How Do You Motivate Employees? **Harvard Business Review**, 1964.

HERZBERG, F. One More Time: How Do You Motivate Employees? **Harvard Business Review**, p. 5-16, 1987.

HOFSTEDE, G. Motivation, Leadership and Organization: Do American Theories Apply Abroad? **Organizational Dynamics**, v. Summer, p. 42-63, 1980.

HOUSE, R. J.; WIGDOR, L. A. herzberg's dual-factor theory of job satisfaction and motivation: a review of the evidence and a criticism. **Personnel Psychology**, v. 20, n. 4, p. 369-389, 1967.

HUMPHREY, W. S. Why Big Software Project Fail: The 12 Key Questions. In: REIFER, D. J. **Software Management**. Hoboken, NJ: John Wiley & Sons, Inc, 2006. p. 21-26.

IAFFALDANO, M. T.; MUCHINSKY, P. M. Job Satisfaction and Job Performance: a Meta-Analysis. **Psychological Bulltin**, v. 97, n. 2, p. 251-273, 1985.

JANSSON, T. **Motivation theory in research on agile project management (working paper)**. Karlstads Universitet, Faculty of Arts and Social Sciences, Karlstad Business School. Karlstad, Sweeden, p. 17. 2013.

JUDGE, T. A. et al. The job satisfaction-job performance relationship: a qualitative and quantitative review. **Psychological bulletin**, v. 127, n. 3, p. 376-407, 2001.

KATZELL, R. A.; THOMPSON, D. E. Work motivation: Theory and practice. **American Psychologist**, v. 45, n. 2, p. 144-153, 1990.

KHALIL, O. E. M. et al. **What motivates Egyptian IS managers and personnel**. Proceedings of the 1997 ACM SIGCPR conference on Computer personnel research - SIGCPR. New York: ACM Press. 1997. p. 187-196.

KING, N. Clarification and evaluation of the Two-factor Theory of Job Satisfaction. **Psychological Bulletin**, v. 74, n. 1, p. 18-31, 1970.

KITCHENHAM, B. A. et al. **Realising evidence-based software engineering**. Proceedings of the 2005 workshop on Realising evidence-based software engineering - REBSE '05. St. Louis, Missouri, USA: ACM. 2005. p. 1-3.

LATHAM, G. P.; ERNST, C. T. Keys to motivating tomorrow's workforce. **Human Resource Management Review**, 2006, v. 16, n. 2, p. 181-198, 2006.

LOCKE, E. Personnel attitudes and motivation. **Annual review of psychology**, v. 26, p. 457-480, 1975.

LOCKE, E. A. Toward a Theory of Task Motivation and Incentives. **Organizational Behavior and Human Performance**, v. 3, p. 157-189, 1968.

LOCKE, E. A. What is job satisfaction? **Organizational Behavior and Human Performance**, p. 309-336, 1969.

LOCKE, E. A. The nature and causes of job satisfaction. In: DUNNETTE, M. D. **Handbook of Industrial and Organizational Psychology**. Chicago: Rand McNally, 1976. p. 1297-1349.

LOCKE, E. A. Motivation through conscious goal setting. **Applied and Preventive Psychology**, v. 5, n. 2, p. 117-124, 1996.

LOCKE, E. A. Dr. Edwin A. Locke. **Robert H. Smith School of Business**. Disponivel em: <http://www.rhsmith.umd.edu/directory/dr-edwin-locke>. Acesso em: 06 jan. 2014.

LOCKE, E. A. et al. **Goal Setting and Task Performance: 1969-1980**. [S.l.]. 1980.

LOCKE, E. A.; LATHAM, G. P. WORK MOTIVATION AND SATISFACTION: Light at the End of the Tunnel. **Psychological Science**, p. 240-246, 1990.

LOCKE, E. A.; LATHAM, G. P. Building a practically useful theory of goal setting and task motivation: A 35-year odyssey. **American Psychologist**, v. 57, n. 9, p. 705-717, 2002.

LUNENBURG, F. Expectancy Theory of Motivation : Motivating by Altering Expectations. **International Journal of Management, Business and Administration**, v. 15, n. 1, p. 1-6, 2011.

MALIK, M. E.; NAEEM, B. Towards Understanding Controversy on Herzberg Theory of Motivation. **Journal of Applied Science Research**, v. 11, n. 2, p. 11936-11941, 2012.

MARCONI, M.; LAKATOS, E. M. **Metodologia Científica**. 3a. ed. [S.l.]: Atlas, 2004.

MARKS, M. A.; MATHIEU, J. E. A temporally based framework and taxonomy of team processes. **Academy of Management Review**, v. 26, n. 3, p. 356-376, 2001.

MASLOW, A. **Motivation and Personality**. [S.l.]: Harper. 1954.

MASLOW, A. H. Preface to motivation theory. **Psychosomatic Medicine**, p. 85-92, 1943.

MCCLELLAND, D. N Achievement and Enterpreneurship: A longitudinal Study. **Jounal of Personality and Social Psychology**, v. 1, n. 4, p. 389-392, 1965.

MELO, C. D. O.; SANTANA, C.; KON, F. **Developers Motivation in Agile Teams**. Proceedings of the 2012 38th EUROMICRO Conference on Software Engineering and Advanced Applications (SEAA). Cesme, Izmir: IEEE. 2012. p. 376-383.

MERRIAM, S. B. **Qualitative Research:** A Guide to Design and Implementation. 2nd Edition. ed. [S.l.]: Jossey-Bass, 2009. 304 p.

MERRIAM, S. B. **Qualitative Research:** a Guide to Design and Implementation. San Francisco, CA: Jossey-Bass, 2009.

MOCKUS, A.; FIELDING, R. T.; HERBSLEB, J. D. Two case studies of open source software development: Apache and Mozilla. **ACM Transactions on Software Engineering and Methodology**, v. 11, n. 3, p. 309-346, 2002.

MULLER, M. M.; FASANENGARTEN, A. **Case study:** extreme programming in a university environment. Proceedings of the 23rd International Conference on Software Engineering. [S.l.]: [s.n.]. 2001. p. 537 - 544.

MURPHY-HILL, E.; WILLIAMS, L. **How can Research about Software Developers Generalize**. Proceedings of the 5th International Workshop on Cooperative and Human Aspects of Software Engineers (CHASE'2012). Zurich, Switzerland: [s.n.]. 2012. p. 105-109.

NEHER, A. Maslow's Theory of Motivation: A Critique. **Journal of Humanistic Psychology**, v. 31, n. 3, p. 89-112, 1991.

NOBLIT, G. W.; HARE, D. **META-ETHNOGRAPHY:** Synthesizing Qualitative Studies. Newbury Park: SAGE Publications, v. 11, 1988. ISBN 0-8039-3022-4.

OLDHAM, G. R.; HACKMAN, J. R. How Job Characteristics Theory Happened. In: SMITH, K. G.; HITT, M. A. **The Oxford handbook of management theory:** The process of theory development. Oxford: Oxford University Press, 2005. p. 151-170.

OLDHAM, G. R.; HACKMAN, J. R. Not what it was and not what it will be: The future of job design research. **Journal of Organizational Behavior**, v. 31, n. 2-3, p. 463-479, 2010.

PERRY, D. E.; SIM, S. E.; EASTERBROOK, S. M. **Case Studies for Software Engineers**. Proceedings of the 26th International Conference on Software Engineering. [S.l.]: [s.n.]. 2004.

PETERSEN, K.; WOHLIN, C. **Context in industrial software engineering research**. 3rd International Symposium on Empirical Software Engineering and Measurement. Lake Buena Vista, US: [s.n.]. 2009.

PORTO Digital. **Porto Digital**, 2000. Disponivel em: <http://www.portodigital.org>. Acesso em: 10 jan. 2014.

PRITCHARD, R. D.; ASHWOOD, E. **Managing Motivation:** A Manager's Guide to Diagnosing and Improving Motivation. [S.l.]: CRC Press, 2008.

REEL, J. S. Critical Success Factors in Software Projects. **IEEE Software**, v. 16, n. 3, p. 18-23, May/Jun 1999. ISSN 0740-7459.

REHMAN, M. et al. **Motivation in software engineering & personal characteristics of software engineers**. National Postgraduate Conference (NPC). Kuala Lumpur: IEEE. 2011. p. 1-5.

REHMAN, S. et al. Understanding Motivators and De-motivators for Software Engineers – A Case of Malaysian Software Engineering Industry. **Communications in Computer and Information Science**, v. 181, p. 205-214, 2011.

RIEMAN, J. **The Diary Study:** A Workplace-Oriented Research Tool to Guide Laboratory Efforts. Proceedings of the INTERACT '93 and CHI '93 conference on Human factors in computing systems. New York: ACM. 1993.

ROZNOWSKI, M.; HULIN, C. The scientific merit of valid measures of general constructs with special reference to job satisfaction and job withdrawal. In: CRANNY, C. J.; SMITH, P. C.; STONE, E. F. **Job satisfaction**. New York: Lexington Books, 1992. p. 123-163.

RUNESON, P.; HÖST, M. Guidelines for conducting and reporting case study research in software engineering. **Empirical Software Engineering**, 14, n. 2, 2008. 131-164.

SACH, R. **The impact of feedback on the motivation of software engineers (PhD Thesis)**. The Open University. Milton Keynes, UK, p. 385. 2014.

SAEKI, M. **Communication, Collaboration and Cooperation in Software Development - How should we support group work in software development?**Proceedings of the 1995 Asia Pacific Software Engineering Conference (APSEC'95). [S.l.]: IEEE. 1995. p. 12-20.

SALES, E. **Um Estudo de Caso sobre Motivação em Integrantes de Equipes de Desenvolvimento de Software no Contexto de uma Instituição Federal de Ensino Superior**. Universidade Federal de Pernambuco. Recife. 2011.

SANDELOWSKI, M.; BARROSO, J. Classifying the Findings in Qualitative Studies. **Qualitative Health Research**, v. 13, n. 7, p. 905-923, 2003.

SANTANA, A. M. D. et al. **Relationships between communication structure and software architecture:** an empirical investigation of the Conway's law at the Federal

University of Pernambuco. Proceedings of the Third International Workshop on Replication in Empirical Software Engineering Research. [S.l.]: IEEE. 2013. p. 34-42.

SCHUNK, D. H.; MEECE, J. L.; PINTRICH, P. R. **Motivation in Education:** Theory, Research, and Applications. 4th edition. ed. [S.l.]: Pearson Education, 2013.

SCOTT, C.; MANKOFF, J. **When Participants Do the Capturing:** The Role of Media in Diary Studies. Conference on Human factors in computing systems. New York, NY, USA: ACM. 2005.

SHARP, H. et al. Models of motivation in software engineering. **Information and Software Technology**, 2008.

SJOBERG, D. I. K.; DYBA, T.; JORGENSEN, M. **The Future of Empirical Methods in Software Engineering Research**. Proceedings of the Future of Software Engineering (FOSE '07). [S.l.]: IEEE. 2007.

SOFTEX. **MPS.BR - Melhoria de Processo do Software Brasileiro, Guia Geral (Versão 1.2)**. Associação para Promoção da Excelência do Software Brasileiro (Softex). [S.l.]. 2007.

SOHN, T. et al. **A Diary Study of Mobile Information Needs**. Twenty-sixth annual SIGCHI conference on Human factors in computing systems. New York, NY, USA: ACM. 2008. p. 433-442.

STEERS, R. M.; MOWDAY, R. T.; SHAPIRO, D. L. THE FUTURE OF WORK MOTIVATION THEORY. **Academy of Management Review**, p. 379-387, 2004.

ŠTEINBERGA, L. **Measuring Software Engineer Motivation in Globally Distributed Projects**. Proceeding of the 38th Euromicro Conference on Software Engineering and Advanced Applications - Work in Progress / Ph.D. Symposium. Cesme, Izmir, Turkey: [s.n.]. 2012.

ŠTEINBERGA, L.; ŠMITE, D. A Case Study of Job Satisfaction in an Offshore Office: Is software engineers' motivation at risk? **Baltic Journal of Modern Computing**, v. 1, n. 3-4, p. 186-198, 2013.

STOL, K.-J.; FITZGERALD, B. **Uncovering Theories in Software Engineering**. Proceedings of the 2nd SEMAT Workshop on a General Theory of Software Engineering. [S.l.]: 2nd SEMAT Workshop on a General Theory of Software Engineering. 2013.

STRAUSS, A.; CORBIN, J. **Pesquisa Qualitativa:** Técnicas e procedimentos para o desenvolvimento de teoria fundamentada. Tradução de Luciane de Oliveira da Rocha. 2nd. ed. Porto Alegre: Artmed, 2008.

SVENSON, O. Motivation, Decision Theory and Human Decision Making. In: KREITLER, S. **Cognition and Motivation:** forging an interdisciplinary perspective. New York: Cambridge University Press, 2013. p. 307-320.

TOMPKINS, H. K.; COUGER, J. D. **Cross cultural comparison of information system personnel in the Pacific Basin versus the US**. Proceedings of the Twenty-Fourth Annual Hawaii International Conference on System Sciences. [S.l.]: [s.n.]. 1991. p. 530-537.

VEN, A. H. V. D. Nothing is quite so practical as a good theory. **Academy of Management Review**, v. 14, n. 4, p. 486-489, 1989.

VERNER, J. M. et al. Factors that Motivate Software Engineering Teams: A Four Country Empirical Study. **The Journal of Systems and Software**, v. in press, p. online, 2014. ISSN DOI 10.1016/j.jss.2014.01.008.

VROOM, V. H. **Work and Motivation**. New York: Wiley, 1964.

WABBA, M. A. Expectancy Theory in Work and Motivation: Some Logical and Methodological Issues. **Human Relations**, p. 121-147, 1974.

WAHBA, M. A.; BRIDWELL, L. G. Maslow reconsidered: A review of research on the need hierarchy theory. **Organizational Behavior and Human Performance**, v. 15, n. 2, p. 212-240, 1976.

WALIA, G. S.; CARVER, J. A systematic literature review to identify and classify software requirement errors. **Information and Software Technology**, v. 51, p. 1087-1109, 2009.

WALLGREN, L. G.; HANSE, J. J. Job characteristics, motivators and stress among information technology consultants: A structural equation modeling approach. **International Journal of Industrial Ergonomics**, p. 51-59, 2007.

WANGENHEIM, C. G. V. et al. **Systematic Literature Review of Software Process Capability/Maturity Models**. Proceedings of International Conference on Software Process Improvement And Capability dEtermination (SPICE). Pisa, Italy: [s.n.]. 2010.

WEINBERG, G. M. **The Psychology of Computer Programming**. Silver anniversary edition. ed. [S.l.]: Dorset House, 1971. 352 p.

WEINBERG, G. M. **The Psychology of Computer Programming**. Silver anniversary. ed. [S.l.]: Van Nostrand Reinhold, 1971.

WEISS, H. M. Deconstructing job satisfaction: Separating evaluations, beliefs and affective experiences. **Human Resource Management Review**, v. 12, p. 173-194, 2002.

WHITEHEAD, J. **Collaboration in Software Engineering:** a Roadmap. Proceedings of the FOSE '07 2007 Future of Software Engineering. [S.l.]: IEEE Computer Society. 2007. p. 214-225.

WYNEKOOP, J. L.; WALZ, D. B. Revisiting the perennial question: are IS people different? **ACM SIGMIS Database**, v. 29, n. 2, p. 62-72, 1998.

YIN, R. K. The Case Study Crisis: Some Answers. **Administrative Science Quarterly**, v. 26, n. 1, p. 58-65, 1981.

YIN, R. K. **Case Study Research:** Design and Methods. Tradução de Daniel Grassi. 4th. ed. Thousand Oaks, California: SAGE Publications, v. Applied Social Research Methods Series, Volume 5, 2009. ISBN 978-1-4129-6099-1.

ZAWACKI, R. A. Motivating the IS people of the future. **Information Systems Management**, v. 9, n. 2, p. 73-75, 1992. ISSN doi:10.1080/10580539208906871.

SLR: LIST OF PRIMARY STUDIES (PS)

[PS001] LeDuc,A.L.J., "Motivation of programmers", In: the Proceedings of ACM Annual Conference of the Special Interest Group Management Information Systems and Computer Personnel Research (ACM SIGMIS CPR), 1980.

[PS002] Tanniru, M.R.; Taylor, S. M., "Causes of turnover among data processing professionals – some preliminary finding", In: the Proceedings of ACM Annual Conference of the Special Interest Group on Computer Personnel Research (ACM SIGCPR), 1981.

[PS003] Bartol, K.M.; Martin, D. C. , "Managing Information Systems Person- nel: A Review of the Literature and Managerial Implications", Management Information Systems Quarterly1982.

[PS004] Goldstein, D.K.; Rockart, J. F., "An Examination of Work-related Correlates of Job Satisfaction in Programmer/Analysts", Management Information Systems Quarterly1984.

[PS005] Cheney, P.H., "Effects of Individual Characteristics, Organizational Factors and Task Characteristics on Computer Programmer Produc- tivity and Job Satisfaction", Information and Management1984.

[PS006] Toledo, R.A. M.; Unger, E. A., "Another look at motivating data processing professionals,", In: the Proceedings of ACM Annual Conference of the Special Interest Group on Computer Personnel Research (ACM SIGCPR), 1985.

[PS007] Dittrich, J.E.; Couger, J.D.; Zawacki, R.A. , "Perceptions of equity, job satisfaction, and intention to quit among data processing personnel,", Information and Management1985.

[PS008] Honda, K., "Research on work environment for software productivity improvement", In: the Proceedings of Annual IEEE International Computer Software and Applications Conference, 1985.

[PS009] Ferrat, T. W.; Short, L. E., "Are information systems people different: an investigation of motivational difference", Management Information Systems Quarterly1986.

[PS010] Baroudi, J.J.; Ginzberg, M. J., "Impact of the technological environment on programmer/analyst job outcomes", Communication of the ACM1986.

[PS011] Couger, J.D.; McIntyre, S., "Motivating norms for artifical intelligence personnel", In: the Proceedings of Hawaii International Conference on System Sciences, 1987.

[PS012] Couger, J.D.; McIntyre, S., "Motivation Norms of Knowledge Engineers compared to those of Software Engineers", Journal of Management Information Systems1987.

[PS013] Goldstein, D.K., "An updated measure of supervisor-rated job perfor- mance for programmer/analysis", In: the Proceedings of ACM Annual Conference of the Special Interest Group on Computer Personnel Research (ACM SIGCPR), 1988.

[PS014] Garden, A. M., "Behavioural and organisational factors involved in the turnover of high tech professionals", In: the Proceedings of ACM Annual Conference of the Special Interest Group on Computer Personnel Research (ACM SIGCPR), 1988.

[PS015] Couger, J.D.; Adelsberger, H., "Environments: Austria compared to the United States", In: the Proceedings of ACM Annual Conference of the Special Interest Group on Computer Personnel Research (ACM SIGCPR), 1988.

[PS016] Garden, A. M., "Maintaining the spirit of excitement in growing compa- nies", In: the Proceedings of ACM Annual Conference of the Special Interest Group on Computer Personnel Research (ACM SIGCPR), 1988.

[PS017] Rubin, H.I.; Hernandez, E. F., "Motivations and behaviors of software professionals", In: the Proceedings of ACM Annual Conference of the Special Interest Group on Computer Personnel Research (ACM SIGCPR), 1988.

[PS018] Couger, J.D., "Motivators vs. demotivators in the IS environment", Journal of Systems Management1988.

[PS019] Couger, J.D.; Borovits, I.; Zviran, M., "Comparison of motivating environments for program- mer/analysts and programmers in the US, Israel and Singapore", In: the Proceedings of Hawaii International Conference on System Sciences, 1989.

[PS020] Couger, J.D.; Adelsberger, H.; Borovits, I.; Zviran, M.; Motiwalla, J., "Commonalities in motivating environments for programmer/analysts in Austria, Israel, Singapore, and the U.S.A", Information and Management1990.

[PS021] Checchio, R.A., "Creating a motivating environment in software development. Experience with the Management of Software Projects", In: the Proceedings of Proceedings of the Third IFAC/IFIP Workshop. Pergamon. 1990, 1990: p. 81-6., 1990.

[PS022] Im, J.H.; Hartman, S., "Rethinking the issue of whether IS people are different from non-IS people", Management Information Systems Quarterly1990.

[PS023] Fujigaki, Y., "Stress analysis of software engineers for effective management", In: the Proceedings of Proceedings of the Third International Symposium. North-Holland. 1990, 1990: p. 255-8., 1990.

[PS024] Myers, M.E., "Motivation and performance in the information systems field: a survey of related studies", In: the Proceedings of ACM Annual Conference of the Special Interest Group on Computer Personnel Research (ACM SIGCPR), 1991.

[PS025] Moore, J.E., "Personality characteristics of information systems professionals", In: the Proceedings of ACM Annual Conference of the Special Interest Group on Computer Personnel Research (ACM SIGCPR), 1991.

[PS026] Crook, C.W.; Crepeau, R. G.; McMurtrey, M. E., "Utilization of the career anchor/career orientation constructs for management of I/S professionals,", In: the Proceedings of ACM Annual Conference of the Special Interest Group on Computer Personnel Research (ACM SIGCPR), 1991.

[PS027] Crepeau, R.G. ; Crook, C.W.; Goslar, M. D.; McMurtrey, M. E., "Career Anchors of Information Systems Personnel", Journal of Management Information Systems1992.

[PS028] Couger, J.D., "Comparison of motivation norms for programmer/ analysts in the Pacific Rim and the U.S.", International Journal of Information Systems1992.

[PS029] Kym, H.; Park, W. W., "Effect of cultural fit/misfit on the productivity and turnover of is personnel", In: the Proceedings of ACM Annual Conference of the Special Interest Group on Computer Personnel Research (ACM SIGCPR), 1992.

[PS030] Rasch, R.H.; Tosi, H. L., "Factors affecting software developers' performance: an integrated approach", Management Information Systems Quarterly1992.

[PS031] Smits, S.J.; McLean, E.R.; Tanner, J.R., "Managing high achieving information systems professionals", In: the Proceedings of ACM Annual Conference of the Special Interest Group on Computer Personnel Research (ACM SIGCPR), 1992.

[PS032] Burn, J.M.; Couger, J.D.; Ma,L. C. K.; Tompkins, H., "Motivating IT professionals, The Hong Kong challenge", Information and Management1992.

[PS033] Zawacki, R.A. , "Motivating the IS people of the future", Information systems management1992.

[PS034] Klenke, K.; Kievit, K. A., "Predictors of leadership style, organizational commitment and turnover of information systems professionals", In: the Proceedings of ACM Annual Conference of the Special Interest Group on Computer Personnel Research (ACM SIGCPR), 1992.

[PS035] Miller, W.C.; Couger, J.D.; Higgins, L. F., "Comparing innovation styles profile of IS personnel to other occupations", In: the Proceedings of Hawaii International Conference on System Sciences, 1993.

[PS036] Jordan, E.; Whiteley, A. M., "HRM practices in information technology management Proceedings of computer personnel research conference", In: the Proceedings of ACM Annual Conference of the Special Interest Group on Computer Personnel Research (ACM SIGCPR), 1994.

[PS037] Burn, J.M.; Tye,E. M. W.; Ma,L. C. K.; Poon, R. S. K., "Job expectations of IS professionals in Hong Kong", In: the Proceedings of ACM Annual Conference of the Special Interest Group on Computer Personnel Research (ACM SIGCPR), 1994.

[PS038] Igbaria, M.; Meredith, G.; Smith, D. C., "Career orientations of information systems employees in South Africa", The Journal of Strategic Information Systems1995.

[PS039] Couger, J.D.; Ishikawa, A., "Comparing motivation of Japanese computer personnel versus these of the United States. System Sciences", In: the Proceedings of Hawaii International Conference on System Sciences, 1995.

[PS040] Turley, R.T.; Bieman, J. M., "Competencies of exceptional and nonex- ceptional software engineers", Journal of Systems and Software1995.

[PS041] Burn, J.M.; Ma,L. C. K.; Tye,E. M. W., "Managing IT professionals in a global environment,", In: the Proceedings of ACM Annual Conference of the Special Interest Group on Computer Personnel Research (ACM SIGCPR), 1995.

[PS042] Santana, M.; Robey, D., "Perceptions of control during systems development: effects on job satisfaction of systems professionals", In: the Proceedings of ACM Annual Conference of the Special Interest Group on Computer Personnel Research (ACM SIGCPR), 1995.

[PS043] Thomas, S.A.; Hurley, S. F.; Barnes, D. J., "Looking for the human factors in software quality management", In: the Proceedings of SEEP, 1996.

[PS044] McLean, E.R.; Smits, S.J.; Tanner, J.R., "The importance of salary on job and career attitudes of information systems professionals", Information and Management1996.

[PS045] Smits, S.J.; McLean, E.R.; Tanner, J.R., "A longitudinal study of I/S careers: synthesis, conclusion, and recommendations", In: the Proceedings of ACM Annual

Conference of the Special Interest Group on Computer Personnel Research (ACM SIGCPR), 1997.

[PS046] Frangos, S.A., "Motivated humans for reliable software products", Microprocessors and Microsystems1997.

[PS047] Lending, D.; Chervany, N. L., "The changing systems development job: a job characteristics approach", In: the Proceedings of ACM Annual Conference of the Special Interest Group on Computer Personnel Research (ACM SIGCPR), 1997.

[PS048] Khalil, O. E. M.; Zawacki, R.A. ; Zawacki, P.A. ; Selim, A., "What motivates Egyptian IS managers and personnel: Some preliminary results", In: the Proceedings of ACM Annual Conference of the Special Interest Group on Computer Personnel Research (ACM SIGCPR), 1997.

[PS049] Richens, E., "HR strategies for IS professionals in the 21st century", In: the Proceedings of ACM Annual Conference of the Special Interest Group on Computer Personnel Research (ACM SIGCPR), 1998.

[PS050] Agarwal, R.; Ferrat, T. W., "Recruiting, retaining, and developing IT professionals: an empirically derived taxonomy of human resource practices", In: the Proceedings of ACM Annual Conference of the Special Interest Group on Computer Personnel Research (ACM SIGCPR), 1998.

[PS051] Wynekoop, J.L.; Walz, D. B., "Revisiting the perennial Question: Are IS People Different", The Database for Advances in Information Systems1998.

[PS052] Ridings, C.M.; Eder, L. B., "An Analysis of IS technical career paths and job satisfaction", In: the Proceedings of ACM Annual Conference of the Special Interest Group on Computer Personnel Research (ACM SIGCPR), 1999.

[PS053] Linberg, K.R., "Software developer perceptions about software project failure: a case study", Journal of Systems and Software1999.

[PS054] Agarwal, R.; Ferrat, T. W., "Retention and the career motives of IT professionals", In: the Proceedings of ACM Annual Conference of the Special Interest Group on Computer Personnel Research (ACM SIGCPR), 2000.

[PS055] Gambill, S.E.; Clark, W. J.; Wikes, R. B., "Toward a holistic model of task design for IS professionals", Information and Management, Volume 37 Issue 5, Aug.2000.

[PS056] Mak, B.L. ; Sockel, H., "A confirmatory factor analysis of IS employee motivation and retention", Information and Management2001.

[PS057] Kandeel, H.; Wahba, K., "Competency models for human resource development: case of Egyptian software industry", In: the Proceedings of Managing Information Technology in a Global Environment., 2001.

[PS058] Agarwal, R.; Ferrat, T. W., "Crafting an HR strategy to meet the need for IT workers,", Communication of the ACM2001.

[PS059] Niederman, F.; Sumner, M. R., "Job turnover among MIS professionals: an exploratory study of employee turnover", In: the Proceedings of ACM Annual Conference of the Special Interest Group on Computer Personnel Research (ACM SIGCPR), 2001.

[PS060] Nelson, A.C.; LeRouge, C., "Self esteem: moderator between role stress fit and satisfaction and commitment?", In: the Proceedings of ACM Annual Conference of the Special Interest Group on Computer Personnel Research (ACM SIGCPR), 2001.

[PS061] Lee, P.C. B., "Career plateau and professional plateau: impact on work outcomes of information technology professionals.", In: the Proceedings of ACM Annual Conference of the Special Interest Group on Computer Personnel Research (ACM SIGCPR), 2002.

[PS062] Agarwal, R.; Ferrat, T. W., "Enduring practices for managing IT professionals,", Communication of the ACM2002.

[PS063] Agarwal, R.; De, P.; Ferrat, T. W., "Explaining an IT professional's preferred employment duration: Empirical tests of a causal model of antecedents", In: the Proceedings of ACM Annual Conference of the Special Interest Group on Computer Personnel Research (ACM SIGCPR), 2002.

[PS064] Andersen, E.S., "Never the twain shall meet: exploring the differences between Japanese and Norwegian IS professionals", In: the Proceedings of ACM Annual Conference of the Special Interest Group on Computer Personnel Research (ACM SIGCPR), 2002.

[PS065] Thatcher, J. B.; Liu, Y.; Stepina, L. P., "The role of the work itself: An empirical examination of intrinsic motivation's influence on IT workers attitudes and intentions", In: the Proceedings of ACM Annual Conference of the Special Interest Group on Computer Personnel Research (ACM SIGCPR), 2002.

[PS066] Lee, P.C. B., "The social context of turnover among information technology professional", In: the Proceedings of ACM Annual Conference of the Special Interest Group on Computer Personnel Research (ACM SIGCPR), 2002.

[PS067] Agarwal, R.; Ferrat, T. W., "Toward understanding the relationship between IT human resource management systems and retention: An empirical analysis based on multiple theoretical and measurement approaches", In: the Proceedings of ACM Annual Conference of the Special Interest Group on Computer Personnel Research (ACM SIGCPR), 2002.

[PS068] Garza, A.I.; Lunce, S. E.; Maniam, B. , "Career anchors of Hispanic information systems professionals", In: the Proceedings of Annual Meeting of the Decision Sciences Institute, 2003.

[PS069] Hsu, M.K.; Chen, H. G.; Jiang, J. J.; Klein, G., "Career satisfaction for managerial and technical anchored IS personnel in later career stages", In: the Proceedings of ACM Annual Conference of the Special Interest Group Management Information Systems and Computer Personnel Research (ACM SIGMIS CPR), 2003.

[PS070] Niederman, F.; Sumner, M. R., "Decision paths affecting turnover among information technology professionals", In: the Proceedings of ACM Annual Conference of the Special Interest Group Management Information Systems and Computer Personnel Research (ACM SIGMIS CPR), 2003.

[PS071] Ferrat, T. W.; Enns, H. G.; Prasad, J., "Instrument Validation for Investigating a Model of Employment Arrangement Fit for IT Professionals", In: the Proceedings of ACM Annual Conference of the Special Interest Group Management Information Systems and Computer Personnel Research (ACM SIGMIS CPR), 2003.

[PS072] Carayon, P.; Hoonaker, P.; Marchand, S.; Schwarz, J., "Job characteristics and quality of working life in the IT workforce: the role of gender,", In: the Proceedings of ACM Annual Conference of the Special Interest Group Management Information Systems and Computer Personnel Research (ACM SIGMIS CPR), 2003.

[PS073] Peters, L., "Managing the Performance of Software Engineering Professionals", In: the Proceedings of Proceedings of the 16th Conference on Software Engineering Education and Training, 2003.

[PS074] Hertel, G.; Nieder, S.; Herrmann, S., "Motivation of software develop- ers in Open Source projects: an Internet-based survey of contributors to the Linux kernel", Research Policy2003.

[PS075] Tanner, F.R., "On motivating engineers.", In: the Proceedings of International Engineering Management Conference., 2003.

[PS076] Hsu, M.K.; Jiang, J. J.; Klein, G.; Tang, Z., "Perceived career incentives and intent to leave", Information and Management2003.

[PS077] Capretz, L. F., "Personality types in software engineering,", International Journal of Human Computer Studies2003.

[PS078] Almstrum, V.L., "What is the attraction to computing?", Communication of the ACM2003.

[PS079] Mannaro, K.; Melis, M.; Marchesi. M., "Empirical Analysis on the Satisfaction of IT Employees Comparing XP Practices with Other Software Development Methodologies", In: the Proceedings of Proceedings of the Fifth International Conference on Extreme Programming and Agile Processes in Software Engineering (XP 2005), 2004.

[PS080] Ferrat, T. W.; Enns, H. G.; Prasad, J., "Employment arrangement fit for IT professionals: An examination of the importance of fit components,", In: the Proceedings of ACM Annual Conference of the Special Interest Group Management Information Systems and Computer Personnel Research (ACM SIGMIS CPR), 2004.

[PS081] Roberts, J.; Hann, I. H.; Slaughter, S., "Understanding the motivations, participation, and performance of Open Source Software developers: a longitudinal study of the Apache projects.", Carnegie Mellon University Working Paper2004.

[PS082] Sumner, M. R.; Yager, S.; Franke, D., "Career orientation and organiza- tional commitment of IT personnel", In: the Proceedings of ACM Annual Conference of the Special Interest Group Management Information Systems and Computer Personnel Research (ACM SIGMIS CPR), 2005.

[PS083] Darcy, D.P.; Ma, M. J., "Exploring Individual Characteristics and Pro- gramming Performance: Implications for Programmer Selection", In: the Proceedings of Hawaii International Conference on System Sciences, 2005.

[PS084] Morales, A.W., "Salary survey 2005", Software Development2005.

[PS085] Procaccino, J. D.; Verner, J. M.; Shelfer, K. M.; Gefen, D., "What Do Software Practitioners Really Think About Project Success: An Exploratory Study.", Journal of Systems and Software2005.

[PS086] Reid, M. F.; Riemenschneider, C. K.; Allen, M. W.; Armstrong, J. A., "Affective commitment in the public sector: the case of IT employees", In: the Proceedings of ACM Annual Conference of the Special Interest Group Management Information Systems and Computer Personnel Research (ACM SIGMIS CPR), 2006.

[PS087] Ramachandran, S.; Rao, S. V., "An effort towards identifying occupa- tional culture among information systems professionals", In: the Proceedings of ACM Annual Conference of the Special Interest Group Management Information Systems and Computer Personnel Research (ACM SIGMIS CPR), 2006.

[PS088] Smith, D. C.; Speight, H. L., "Antecedents of turnover intention and actual turnover among information systems personnel in South Africa", In: the Proceedings of ACM Annual Conference of the Special Interest Group Management Information Systems and Computer Personnel Research (ACM SIGMIS CPR), 2006.

[PS089] Enns, H. G.; Ferrat, T. W.; Prasad, J., "Beyond Stereotypes of IT Professionals: Implications for IT HR Practices", Communication of the ACM2006.

[PS090] Li, Y.; Tan, C. H.; Teo, H. H.; Mattar, A. T., "Motivating open source software developers: influence of transformational and transactional leaderships", In: the Proceedings of ACM Annual Conference of the Special Interest Group Management Information Systems and Computer Personnel Research (ACM SIGMIS CPR), 2006.

[PS091] Baddoo, N.; Hall, T.; Jagielska, D., "Software developer motivation in a high maturity company: a case study", Software Process: Improvement and Practice 112006.

[PS092] Ituma, A., "The internal career: an explorative study of the career anchors of information technology workers", In: the Proceedings of ACM Annual Conference of the Special Interest Group Management Information Systems and Computer Personnel Research (ACM SIGMIS CPR), 2006.

[PS093] Melnik, G.; Maurer, F., "Comparative Analysis of Job Satisfaction in Agile and Non-agile Software Development Teams", In: the Proceedings of Proceedings of the Sixth International Conference on Extreme Programming and Agile Processes in Software Engineering (XP 2006), 2006.

[PS094] Mahaney, R. C.; Lederer, A. L., "The effect of intrinsic and extrinsic rewards for developers on information systems project success", Project Management Journal2006.

[PS095] Wu, C. G.; Gerlach, J. H.; Young, C. E., "An empirical analysis of open source software developers' motivations and continuance intentions", Information and Management2007.

[PS096] Lounsbury, J. W.; Moffitt, L.; Gibson, L. W.; Drost, A. W.; Stevens, M., "An investigation of personality traits in relation to job and career satisfaction of information technology professionals", Journal of Information Technology2007.

[PS097] Chui, J.; Tong, C.; Mula, M. J., "Assessing the extent to which career development impacts employee commitment: a case study of the ICT industry in Hong Kong", In: the Proceedings of Australasian Conference on Information Systems, 2007.

[PS098] Sumner, M. R.; Franke, D., "Career Orientation and the Global IT Workforce: Research in Progress", In: the Proceedings of ACM Annual Conference of the Special Interest Group Management Information Systems and Computer Personnel Research (ACM SIGMIS CPR), 2007.

[PS099] Ohly, S.; Fritz, C., "Challenging the status quo: What motivates proactive behaviour?", Journal of Occupational and Organizational Psychology2007.

[PS100] Jørgensen, N., "Developer autonomy in the FreeBSD open source project", Journal of Management and Governance2007.

[PS101] Beecham, S.; Sharp, H.; Baddoo, N.; Hall, T.; Robinson, H., "Does the XP environment meet the motivational needs of the software developer? An empirical study", In: the Proceedings of AGILE Conference, 2007.

[PS102] Sharp, H.; Hall, T.; Baddoo, N.; Beecham, S., "Exploring motivational differences between software developers and project managers", In: the Proceedings of ACM Annual Conference of the Special Interest Group on Software Engineering provides, 2007.

[PS103] Mahatanankoon, P., "Exploring the impact of essential IT skills on career satisfaction and organisational commitment of information systems professionals", Int. J. Information Systems and Change Management2007.

[PS104] Bitzer, J.; Schrettl, W.; Schroeder, P. J. H., "Intrinsic motivation in open source software development", Journal of Comparative Economics2007.

[PS105] Kim, S.; Wright, B. E., "IT Employee Work Exhaustion: Toward an Integrated Model of Antecedents and Consequences", Review of Public Personnel Administration2007.

[PS106] Tessem, B.; Maurer, F., "Job satisfaction and motivation in a large agile team", In: the Proceedings of Proceedings of the Seventh International Conference on Extreme Programming and Agile Processes in Software Engineering (XP 2007), 2007.

[PS107] Hertel, G., "Motivating job design as a factor in open source governance", Journal of Management and Governance2007.

[PS108] Prasad, J.; Enns, H. G.; Ferrat, T. W., "One size does not fit all: Managing IT employees' employment arrangements", Human Resource Management2007.

[PS109] Bahl, K. T.; Gulati, N., "Pay satisfaction of software professionals in India", Vikalpa2007.

[PS110] Tominaga, M.; Asakura, T.; Akiyama, T., "The effect of micro and macro stressors in the work environment on computer professionals' subjective health status and productive behavior in Japan", Industrial Health2007.

[PS111] Pereira, J.; Cerpa, N. ; Verner, J. M.; Rivas, M.; Procaccino, J. D., "What do software practitioners really think about project success: A cross-cultural comparison", Journal of Systems and Software2007.

[PS112] Bhatti, M. W.; Ahsan, A.; Sajid, A., "A framework to identify the'Motivational Factors' of employees: a case study of Pakistan IT industry", Journal WSEAS Transactions on Computers 2008.

[PS113] David, P. A.; Shapiro, J. S., "Community-based production of open-source software: What do we know about the developers who participate?", Information Economics and Policy2008.

[PS114] Oreg, S.; Nov, O., "Exploring motivations for contributing to open source initiatives: The roles of contribution context and personal values", Computers in Human Behavior2008.

[PS115] Subramanyam, R.; Xia, M., "Free/Libre Open Source Software development in developing and developed countries: A conceptual framework with an exploratory study", Decision Support Systems2008.

[PS116] Chen, L. H., "Job satisfaction among information system (IS) personnel", Computers in Human Behavior2008.

[PS117] Ke, W.; Zhang, P., "Motivations for participating in open source software communities: Roles of psychological needs and altruism", In: the Proceedings of PACIS, 2008.

[PS118] Hamel, M. P.; Schweik, C. M., "Open source collaboration: Two cases in the U.S. public sector", First Monday2008.

[PS119] von Krog, G.; Haefliger, S.; Spaeth, S.; Wallin, M., "Open Source Software: What we know (and do not know) about motives to contribute", DIME Working Papers on INTELLECTUAL PROPERTY RIGHTS2008.

[PS120] Korunka, C.; Hoonaker, P.; Carayon, P., "Quality of working life and turnover intention in information technology work", Human Factors and Ergonomics in Manufacturing2008.

[PS121] Lacity, M. C.; Iyer, V. V., "Turnover intentions of Indian IS professionals", Information Systems Frontiers2008.

[PS122] Franca, A.C.C.; da Silva, F.Q.B., "An empirical study on software engineers motivational factors", In: the Proceedings of Empirical Software Engineering and Measurement, 2009.

[PS123] Sharp, H.; Hall, T., "An initial investigation of software practitioners' motivation", In: the Proceedings of International Workshop on Cooperative and Human Aspects of Software Engineering, 2009.

[PS124] Potter, L. E.; von Hellens, L. A.; Nielsen, S. H., "Childhood interest in IT and the choice of IT as a career: the experiences of a group of IT ", In: the Proceedings of ACM Annual Conference of the Special Interest Group Management Information Systems and Computer Personnel Research (ACM SIGMIS CPR), 2009.

[PS125] Mgaya, K. V. ; Uzoka, F. M. E. ; Kitindi, E. G. ; Shemi, A. P., "Examining Career Orientations of Information Systems Personnel in an Emerging Economy Context", In: the Proceedings of ACM Annual Conference of the Special Interest Group Management Information Systems and Computer Personnel Research (ACM SIGMIS CPR), 2009.

[PS126] Calisir, F.; Gumussoy, C. A.; Iskin, I., "Factors affecting intention to quit among IT professionals", In: the Proceedings of The IEEE International Conference on Industrial Engineering and Engineering Management , 2009.

[PS127] Acuña, S. T.; Gómez, M.; Juristo, N., "How do personality, team processes and task characteristics relate to job satisfaction and software quality?", Information and Software Technology2009.

[PS128] Laumer, S., "Non-Monetary Solutions for Retaining the IT Workforce", In: the Proceedings of American Conference on Information Systems, 2009.

[PS129] Song, Y.; Wang, M., "Research on individual motivation model of software engineering", Journal of Communication and Computer2009.

[PS130] Tseng, C. Y.; Wallace, M., "Retention of software employees in the IT industry in Taiwan", In: the Proceedings of Australian and New Zealand Academy of Management, 2009.

[PS131] Wang, Y., "What motivate software engineers to refactor source code? Evidences from professional developers", In: the Proceedings of IEEE International Conference on Software Maintenance, 2009.

[PS132] Lee, K.; Joshi, K.; Bae, M., "An investigation of the influence of the is context on the determinants of turnover intentions in Korea", Journal of Organizational Computing and Electronic Commerce2010.

[PS133] Ghapanchi, A. H.; Aurum, A., "Antecedents to IT personnel's intentions to leave: A systematic literature review", Journal of Systems and Software2010.

[PS134] Sach, R.; Sharp, H.; Petre, M., "Continued involvement in software development: motivational factors", In: the Proceedings of Empirical Software Engineering and Measurement, 2010.

[PS135] Hannay, Jo E.; Arisholm, E.; Engvik, H.; Sjoberg, Dag I.K., "Effects of personality on pair programming", Transactions on Software Engineerinsg2010.

[PS136] Hilkert, D.; Benlian, A.; Hess, T., "Motivational Drivers to Develop Apps for Social Software-Platforms: The Example of Facebook", In: the Proceedings of American Conference on Information Systems, 2010.

[PS137] Wang, Y.; Zhang, M., "Penalty policies in professional software development practice: A multi-method field study", In: the Proceedings of International Conference on Software Engineering, 2010.

[PS138] Kumlander, D., "Semi- and Fully Self-Organised Teams ", Advanced Techniques in Computing Sciences and Software Engineering2010.

[PS139] Shah, H.; Harrold, M. J., "Studying human and social aspects of testing in a service-based software company: Case study", In: the Proceedings of International Workshop on Cooperative and Human Aspects of Software Engineering, 2010.

[PS140] Benbya, H.; Belbaly, N., "Understanding Developers' Motives in Open Source Projects: A Multi-Theoretical Framework", Communications of the Association for Information Systems2010.

ACADEMIC PRODUCTION AND AWARDS

Da Silva, F. Q. B., França, A. C. C., Felix, A. de L. C., de Araújo, A. C. M. L., Carneiro, D. E. S., Salles, E., & Gouveia, T. B. (2011). Protocolo para Estudos de Caso sobre Motivação de Engenheiros De Software. Available at http://goo.gl/yqW9ck.

França, A. C. C., Gouveia, T. B., Santos, P. C. F., Santana, C., & da Silva, F. Q. B. (2011). Motivation in software engineering: a systematic review update. In International Conference on Evaluation & Assessment in Software Engineering (EASE) (pp. 154–163). IET. doi:10.1049/ic.2011.0019

França, a. C. C., & da Silva, F. Q. B. (2012). Towards Understanding Motivation in Software Engineering. In 7th International Doctoral Symposium on Empirical Software Engineering. Lund, Sweedon.

França, A. C. C., Carneiro, D. E. S., & Silva, F. Q. B. (2012). Towards an Explanatory Theory of Motivation in Software Engineering : A Qualitative Case Study of a Small Software Company. In Simpósio Brasileiro de Engenharia de Software.**This work received the honors for the Third Best Paper award.**

França, A. C. C., Felix, A. de L. C., & da Silva, F. Q. B. (2012). Towards an Explanatory Theory of Motivation in Software Engineering : A Qualitative Case Study of a Government Organization. In International Conference on Evaluation & Assessment in Software Engineering (EASE) (pp. 72–81). Ciudad Real, ES: IET.

França, A. C. C., Araújo, A. C. M. L. De, & Silva, F. Q. B. (2013). Motivation of Software Engineers : A Qualitative Case Study of a Research and Development Organisation. In Proceedings of 6th International Workshop on Cooperative and Human Aspects of Software Engineering (CHASE 2013). San Francisco, CA.

França, A. C. C., da Silva, F. Q. B., Felix, A. de L. C., & Carneiro, D. E. S. (2013). Motivation in software engineering industrial practice: A cross-case analysis of two software organisations. Information and Software Technology, online.

França; C., Sharp, H., da Silva, F. Q. B. (2014) Motivated Software Engineers are Engaged, Focused and Collaborative, while Satisfied ones are Happy. *Submitted to IEEE Software on 15/January/2014.*

Appendices

APPENDIX A: SYSTEMATIC LITERATURE REVIEW PROTOCOL

INDEX

A) RESEARCH QUESTIONS

The original study described an extensive systematic literature review of peer reviewed studies focusing on motivation in software engineering. The original study protocol was based on the guidelines presented by Kitchenham (2004). Using the criteria defined by the Centre for Reviews and Dissemination (CDR) Database of Abstracts of Reviews of Effects (DARE), of the York University (Centre for Reviews and Dissemination, 2007, 2010), the Original Study scores 4 points (highest possible score), being considered of excellent quality. The central objective of the original study was defined as "to plot the landscape of current reported knowledge in terms of what motivates developers, what de-motivates them, how existing models address motivation, and whether Software Engineers are indeed a homogeneous group with similar needs" (Beecham, et al., 2008). The original study systematically reviewed published work in software engineering looking for answers to five research questions:

RQ1: What are the characteristics of Software Engineers?

RQ2: What (de)motivates Software Engineers to be more (less) productive?

RQ3: What are the external signs or outcomes of (de)motivated Software Engineers?

RQ4: What aspects of Software Engineering (de)motivate Software Engineers?

RQ5: What models of motivation exist in Software Engineering?

Our objective is to update the Original Study looking for answers to the same five research questions. We use the same review protocol used in the Original Study, with minor changes reported in the sequel.

B) SEARCH STRING AND VALIDATION STRATEGY

Our search string was based on the same keywords of the Original Study, although we did not compose specific strings to match each research question. Instead, we compose only one generic string, detailed in Table 41. We adapted the syntax of the search string according to the rules of each search engine.

Table 41–(Appendix A - SLR Protocol) Search string construction

Keyword	Generic search string
Software	("software" OR "information technology" OR "information system" OR "information systems" OR "computing")
Engineer	AND ("engineer" OR "engineers" OR "developer" OR "developers" OR "programmer" OR "programmers" OR "analyst" OR "analysts" OR "team leader" OR "team leaders" OR "project manager" OR "project managers" OR "practitioner" OR "practitioners" OR "maintainers" OR "maintainers" OR "designer" OR "designers" OR "coder" OR "coders" OR "tester" OR "testers")
Characteristics	AND ("characteristic" OR "characteristics" OR "personality" OR "human factors" OR "psychology" OR "psychological factors" OR "motivator" OR "motivators" OR "behavior" OR "behaviour" OR "behavioral" OR "behavioural")
Motivation	AND ("motive" OR "motivate" OR "motivation" OR "motivational" OR "demotivate" OR "demotivation" OR "demotivational" OR "de-motivate" OR "de-motivation" OR "de-motivational" OR "inspiration" OR "incentive" OR "drive" OR "enthusiasm" OR "stimulus" OR "stimulus" OR "stimuli" OR "impulsion")
Factors	AND ("productivity" OR "factor" OR "factors" OR "output" OR "efficiency" OR "efficient" OR "interact" OR "interaction" OR "yield" OR "production" OR "creativity" OR "prolific" OR "industrious" OR "fruitful" OR "dynamic" OR "hinder" OR "increase" OR "increases" OR "decrease" OR "decreases")

To assure efficacy for the search string, as well as compatibility to the original study search, we adopted three strategies: (1) since our string brought all results from 2006 to 2010, we looked in our results for the seven papers published in 2006 found in the Original Study. All seven papers were retrieved; (2) we previously selected a set of five known papers, published after March 2006, and considered that these papers should be returned from our searches. They were all retrieved; and finally, (3) we run our string again, with no time restrictions, and we looked into our results for a set of 45 randomly selected studies out of the remaining 85 primary studies from the original study, which would assure a confidence level of 95% for finding the others. All 45 papers were retrieved.

C) DATA SOURCES

Regarding the data sources, we used the same digital libraries described in the Original Study (Table 42), and added CiteSeerX, JSTOR, Scopus and SpringerLink. We did not have access to the Inspec Search Engine. Besides the Journals and Conferences proceedings taken from the Original Study, we designed two search strategies to increase coverage: we searched in Google Scholar for (1) all papers in which the Original Study is cited and (2) all the papers in which one or more of the 92 primary studies of the Original Study are cited.

Table 42 - (Appendix A - SLR Protocol) Search sources

Digital Libraries (Automatic search)	Journals and Conference Proceedings (Manual Search)
• ACM Digital Library (ACM) • CiteSeerX (CSX)* • ElCompendex (COMP) • Google Scholar (GS) • IEEE Xplorer (IEEE) • ISI Web of Knowledge (ISI) • UH University's Electronic Library • JSTOR * • Science Direct (SD) • Scopus (SCP) * • SpringerLink (SL) *	• Proceedings for the special interest group for computer personnel research • International Journal of Information Management • International Conference on System Sciences • DIRC Research summary **Reference Search** • Google Scholar

* added sources

D) SELECTION CRITERIA

First, papers retrieved in the automated search were filtered based on Title and Abstract. Then, inclusion and exclusion of material was guided by exactly the same criteria of the Original Study, detailed in Table 43. Each potentially relevant paper was analyzed by two researchers and revised by a third researcher (this thesis' author). Conflicting opinions were resolved in face-to-face meetings. Finally, papers containing results from the same studies, and repeated papers were removed to ensure that there is no duplication. MS Excel® was used to record all steps of the selection process in a set of sheets.

Table 43 – (Appendix A - SLR Protocol) Selection Criteria

Exclusion criteria	Inclusion criteria
Studies were excluded if it: • does not respond to some research questions • is based on cognitive behavior ? is external to software engineering • is personal opinion piece or viewpoint • is in form of books and overhead presentations • is focusing on company structures and hierarchies unless expressly linked to the individual engineer‟s motivation • is focusing on motivating students to learn – even if they are IT students • is focusing on software managers (e.g. Chief Information Officers) not directly producing the software • is focusing on IT group/team dynamics that look at groups rather than individual motivation • is focusing on gender differences (too low level)	Studies were included if it: • answers the research questions • acceptable source • was published between March 2006 and September 2010 • relates to any practitioner directly producing software • focuses on de-motivation as well as motivation • uses students to study motivation to „develop‟ software • focuses on culture in terms of how IT personnel are motivated in different countries or in different software environments (e.g. Open Source Systems, Agile, traditional) • focuses on ‘satisfaction’ in Software Engineering.

E) EXTRACTION AND SYNTHESIS

MS Excel® was used to manage all data extraction, analysis and synthesis procedure. Based on the forms presented in the Original Study protocol, one researcher was allocated to each paper. The following data was extracted: title, authors, year, publisher, source type (conference/journal), study type (empirical/theoretical), keywords, geographical location of subjects, type of subjects (students/practitioners), and text passages whenever the paper provided answers to at least one of the research questions.

In order to analyze data, we transcribed passages answering each research question, from the papers. Then, an open coding procedure was followed on these passages. Then, similar codes were condensed according to themes categories as in an axial coding procedure. We used the same categories of the Original Study whenever made sense. Finally, we accounted the frequencies of citation for each category. As also warned in the Original Study, it is important to notice that these frequencies do not reflect the importance of the category, but only how many papers cite them

APPENDIX B: TERM OF CONSENTMENT (PT_BR)

Termo de Consentimento Livre e Esclarecido (TCLE).

Você está sendo convidado(a) para participar, como voluntário, em uma pesquisa. Após ser esclarecido(a) sobre as informações a seguir, no caso de aceitar fazer parte do estudo, assine ao final deste documento, que está em duas vias. Uma delas é sua e a outra é do pesquisador responsável. Em caso de recusa você não será penalizado(a) de forma alguma. Em caso de dúvida você pode procurar o Comitê de Ética em Pesquisa da Universidade Federal de Pernambuco pelo telefone (81) 2126-8568 ou (81) 2126-8500.

Pesquisadores Responsável:

- Fabio Q. B. da Silva, PhD
  PhD em Ciência da Computação pela University of Endinburgh, Escócia. Professor do Centro de Informática da UFPE (Universidade Federal de Pernambuco) desde 1993. Fundador da Sociedade SOFTEX, do CESAR e do Porto Digital. Ex-presidente do Porto Digital e ex-tesoureiro da Sociedade Brasileira de Computação. Orientou mais de 20 alunos de mestrado e doutorado.
  E-mail: fabio@cin.ufpe.br
- Helen Sharp, PhD
  Professora da Open University, no Reino Unido. Editora da IEEE (Institute of Electrical and Electronics Engineers)
- Alberto César C. França
  Doutorando em Ciência da Computação pela UFPE (Universidade Federal de Pernambuco)
  E-mail: cesarfranca@gmail.com
- <NOME DO ENTREVISTADOR>
  Mestrando em Ciência da Computação pela UFPE (Universidade Federal de Pernambuco)
  E-mail: <E-MAIL DO ENTREVISTADOR>

HASE - Human Aspects in Software Engineering é um Grupo de Pesquisa Empírica que Estuda a Influência de Fatores Humanos na Engenharia de Software. As pesquisas do grupo visam avaliar a influência de fatores humanos como personalidade, comportamento,

motivação, entre outros fatores, em projetos, processos e equipes envolvidas no desenvolvimento de software.

- Esta pesquisa de natureza acadêmica sob o título ***Um Estudo de Caso sobre Motivação de Engenheiros de Software*** tem como objetivo: Analisar o dia-a-dia de engenheiros de software, com o propósito de identificar fatores que influenciem na sua motivação, a partir do ponto de vista dos próprios engenheiros, líderes de equipe e gestores, no contexto de empresas de desenvolvimento de software.
- A pesquisa visa identificar entre os funcionários de organizações de diferentes naturezas, o que motiva ou desmotiva o engenheiro de software, quais sinais e resultados causados por equipes de engenheiros de software motivados ou desmotivados.
- Entre os benefícios esperados da pesquisa espera-se obter o entendimento do fenômeno motivação na organização pesquisada através da: identificação de fatores ou aspectos que motivam ou desmotivam os engenheiros de software; compreensão dos principais resultados atribuídos a pessoas que trabalham motivadas. A partir desse entendimento propor guias e modelos de direcionamentos para os dirigentes ou gestores das organizações, objetivando a identificação do nível de motivação dos seus funcionários e os seus fatores causadores, bem como a realização de ações sistêmicas para aumentar a motivação.
- Os participantes da pesquisa serão submetidos a uma entrevista sobre motivação, teste de personalidade e entrevistas em grupos. O conteúdo das entrevistas e testes só terá valor de avaliação do funcionário para efeito dessa pesquisa, não tendo assim, nenhuma influência na avaliação do funcionário no desempenho das suas atividades na organização. A entrevista em grupo tem como objetivo a troca de opiniões entre os participantes sobre o tema motivação. A entrevista será gravada para posterior documentação. Se o participante sentir-se constrangido durante o andamento da discussão, tem toda a liberdade de sair, sem ser penalizado de nenhuma forma.
- Ao final da pesquisa, os dados serão publicados em eventos de natureza acadêmica, mas os nomes das pessoas envolvidas, bem como seus conteúdos relacionados serão omitidos nas respectivas publicações.

CONSENTIMENTO DA PARTICIPAÇÃO DA PESSOA COMO PARTICIPANTE

Eu, _____, RG/ CPF/ n.º de matrícula do funcionário, abaixo assinado, concordo em participar da Pesquisa Etnográfica sobre Motivação de Engenheiros de Software, como voluntário.

Fui devidamente informado e esclarecido pelo pesquisador _____ sobre a pesquisa, os procedimentos nela envolvidos, assim como os possíveis riscos e benefícios decorrentes de minha participação. Também foi me garantido que posso recusar a participar da pesquisa, ou retirar meu consentimento a qualquer momento, mesmo após o início dos trabalhos, sem precisar justificar, sem que isto leve a qualquer prejuízo em minha relação com a organização.

Estou ciente e fui esclarecido de que minha privacidade será respeitada, ou seja, qualquer informação ou elemento que possa de qualquer forma me identificar será mantido em sigilo.

Enfim, tendo sido orientado quanto ao teor de todo o conteúdo aqui mencionado e compreendido a natureza e o objetivo do já referido estudo, manifesto meu livre consentimento em participar, estando totalmente ciente de que não há nenhum valor econômico ou material a receber, ou a pagar, por minha participação.

Local e data: _____

_____

Nome e Assinatura do Participante

_____

Nome e Assinatura do entrevistador

APPENDIX C: INTERVIEW SCRIPT (PT_BR, EN_GB)

GUIA DE ENTREVISTA COM ENGENHEIROS DE SOFTWARE

INTERVIEW GUIDE FOR SOFTWARE ENGINEERS

Apresentação
Presentation

- Auto-apresentação
  Self-presentation
- Agradecimento ao participante
  Thank the participant
- Permissão para gravar o áudio da entrevista
  Ask for permission to record the interview audio
- Estimativa de tempo da entrevista (45 a 60 minutos)
  Time estimation for the interview (45 to 60 minutes)

Sobre o projeto de Pesquisa
Talk about the research project

- Falar sobre o time de pesquisa e citar características das outras organizações participantes

  Talk about the research team and mention the characteristics of the other participant organizations
- Falar sobre a colaboração com o grupo de pesquisas da Helen Sharp, Open University, UK

  Talk about the cooperation with the research group leaded by prof. Helen Sharp, from the Open University, UK

Objetivo da Entrevista
Objective of the interview

Coletar opiniões sobre a rotina dos engenheiros de software a fim de (1) identificar fatores que tornam o seu trabalho prazeroso ou agradável, levando-os a ficar motivado, e (2) identificar os fatores que, por outro lado, pode causar desmotivação.

Our objective is to collect data about your routine as a software engineer, in order to (1) identify factors that make your work work better, influencing your motivation, and (2) identify factors that, conversely, may cause demotivation.

ROTEIRO DA ENTREVISTA (pt_BR)

INTERVIEW SCRIPT (en_GB)

Q1. {**Background**} Fale um pouco de você: sua formação, idade, trajetória profissional

Talk about yourself: your age, graduation, career, etc.

Q2. {**Background**} O que o levou a trabalhar nesta área (Engenharia de Software)?

- *Sondagem*: Teve contato com outras áreas da computação, ou mesmo com outros campos de trabalho antes de trabalhar com Engenharia de Software? Quais?

What led you to work in this area (Software Engineering)?

- *Probe: have you had contact with other areas in computing, or even in other fields before working in Software Engineering? Talk about them…*

Q3. {**Feeling**} Como você se sente atualmente trabalhando como engenheiro de software?

- *Sondagem:* Em comparação com outros campos/áreas, como você avalia o seu trabalho como engenheiro de software? Mais/menos estressante, divertido, puxado, significativo, etc.
- *Sondagem*: Em comparação as outras empresas que você já trabalhou, como se sente nesta?

How do you feel currently working as a software engineer?

- *Probe: Comparing to other fields/areas, how do you evaluate your work as software engineer? More/less stressful, enjoyable, hard, meaningful, etc.*
- *Probe: Comparing to other organizations where you have worked, how do you feel working for this one?*

Q4. {**Background**} Há quanto tempo você trabalha nesta empresa?

How long have you worked for this organization?

Q5. {**Experience**} O que o levou a trabalhar nesta empresa?

What led you to work here?

Q6. {**Opinion**} Dentre as características desta empresa quais te estimulam a trabalhar aqui?

What are the most encouraging characteristics of this organization, for you to work here?

Q7. {**Opinion**} Dentre as características desta empresa, o que te desestimula?

And what are the most discouraging?

Q8. {**Background**} Antes desta função atual, quais outras funções ou atividades você desempenhou nessa empresa?

Before your current role/function, what were the other functions that you have performed in this organization?

Q9. {**Background**} Descreva a sua função atual e responsabilidades.

Please, describe your current function and responsibilities.

Q10. {**Background**} Quais as atividades que você faz no dia-a-dia?

What are your day-to-day activities?

Q11. {**Experience**} Agora imagine um dia extraordinariamente bom. Um dia no qual várias, ou todas, as coisas dão certo. Imagine este dia deste a hora em que você acorda até a hora de dormir. Você pode descrever este dia?

- *Sondagem*: como você se sente ao acordar e sair de casa? Como é sua chegada no trabalho? Pela manhã, quais foram suas principais atividades? Como você se sentiu no final da manhã? Como foi a volta para o trabalho à tarde? Como foi o trabalho na parte da tarde? Como você encerrou o seu dia de trabalho? Como foi a volta para casa? Como você encerrou o seu dia?

Now, imagine an extraordinarily good day. A day in which many, or all things, worked out very well. Can you describe a day like this, from the time you get up, until bedtime?

- *Probe: [after the spontaneous answer] how do you feel when you leave home? How do you feel when you arrive at work? In the morning, what are your main activities? How do you feel in the end of the morning? How is the afternoon? How do you feel in the end of the afternoon? How do you finish your working day? How is it like getting back home? How do you finish your day?*

Q12. {**Opinion**} Dentre as atividades do seu dia a dia, quais são as que você mais gosta?

Among your day-to-day activities, which do you like the most?

Q13. {**Feeling**} Descreva o que estas atividades possuem, que características elas tem, que te deixa estimulado?

What do these activities have that make you like them?

Q14. {**Opinion/Feeling**} Quais atividades você gostaria de fazer e não faz? Como você se sente?

What other activities would you like to do, but currently don't do? how do you feel about it?

Q15. {**Experience**} Agora imagine um dia extraordinariamente ruim. Um dia no qual várias, ou todas, as coisas dão errado. Imagine este dia deste a hora em que você acorda até a hora de dormir. Você pode descrever este dia?

- *Sondagem*: como você se sente ao acordar e sair de casa? Como é sua chegada no trabalho? Pela manhã, quais foram suas principais atividades? Como você se sentiu no final da manhã? Como foi a volta para o trabalho à tarde? Como foi o trabalho na parte da tarde? Como você encerrou o seu dia de trabalho? Como foi a volta para casa? Como você encerrou o seu dia?

Now, imagine an extraordinarily bad day. A day in which many, or all things, do not work at all. Can you describe a day like this, from the time you get up, until bedtime?

- *Probe: [after the spontaneous answer] how do you feel when you leave home? How do you feel when you arrive at work? In the morning, what are your main activities? How do you feel in the end of the morning? How is the afternoon? How do you feel in the end of the afternoon? How do you finish your working day? How is it like getting back home? How do you finish your day?*

Q16. {**Opinion**} Dentre as atividades do seu dia a dia, quais são as que você menos gosta?

Among your day-to-day activities, which do you hate the most?

Q17. {**Opinion**} Considerando outras atividades do projeto que não fazem parte do seu dia a dia, quais NÂO gostaria de fazer de jeito nenhum

What do these activities have that make you hate them?

Q18. {**Feeling**} Descreva o que estas atividades possuem, que características elas tem, que te deixam desestimulado.

- *Sondagem*: Em comparação a funções realizadas anteriormente, como você se sente atuando nesta função?

Besides your day-to-day activities, what other activities would you never want to do?

Q19. {**Background**} Descreva quem são as pessoas da sua equipe com quem você tem relação direta no seu dia-a-dia.

Describe your team people, to whom you relate in your day-to-day activities.

Q20. {**Background**} Como funciona a divisão de trabalho? Como é a dinâmica do trabalho em equipe? Qual o seu papel?

How does the task division work? Describe your teamwork dynamics, and your specific role.

Q21. {**Feeling**} Como você se sente trabalhando nesta equipe?

How do you feel being part of this team?

Q22. {**Opinion**} Na sua opinião, quais são alguns pontos fortes da sua equipe?

In your opinion, what are the strengths of your team?

Q23. {**Feeling**} Como você sente ao porto forte XX, YY, ZZ.

[for each strength] How do you feel about it?

Q24. {**Experience**} Dê-me um exemplo de uma situação que realmente você se sentiu parte desta equipe.

Give me an actual example of a situation when you really felt like belonging to this team.

Q25. {**Experience**} Como você descreveria um colega de trabalho que está claramente motivado com o trabalho?

- *Probe*: to provide stimuli by alerting that these signals may be either behavioural, or related to the individual work outcomes, or even to the team work.

How would you describe a clearly motivated colleague?

- *Probe: how can you make sure that this person is actually motivated?*

Q26. {**Opinion**} De que forma você acha que isso impacta no trabalho da equipe?

In which way do you think that it can affect the teamwork?

Q27. {**Opinion**} De que forma você acha que isso impacta no trabalho dele?

In which way do you think that it can affect the person's individual work?

Q28. {**Opinion**} Na sua opinião, quais são alguns pontos fracos da sua equipe?

In your opinion, what are the weaknesses of your team?

Q29. {**Feeling**} Como você sente ao porto fraco XX, YY, ZZ.

[for each weakness] How do you feel about it?

Q30. {**Experience**} Dê-me um exemplo de uma situação que realmente você não se sentiu parte desta equipe.

Give me an actual example of a situation when you felt like NOT belonging to this team.

Q31. {**Experience**} Como você descreveria um colega de trabalho que está claramente desmotivado com o trabalho?

- *Probe*: to provide stimuli by alerting that these signals may be either behavioral, or related to the individual work outcomes, or even to the team work.

How would you describe a clearly demotivated colleague?

- *Probe: how can you make sure that this person is actually demotivated?*

Q32. {**Opinion**} De que forma você acha que isso impacta no trabalho da equipe?

In which way do you think that it can affect the teamwork?

Q33. {**Opinion**} De que forma você acha que isso impacta no trabalho dele?

In which way do you think that it can affect the person's individual work?

Q34. {**Opinion/Feeling**} Tem alguma outra função ou projeto [dentro da empresa] que você preferiria estar alocado?Como você se sentiria trabalhando lá?

Is there any other function or Project [in the organization] that you would prefer to work? How would you feel working there?

Q35. {**Opinion/Feeling**} E tem alguma outra função ou projeto [dentro da empresa?] que você não gostaria de ser alocado de jeito nenhum? Como você se sentiria trabalhando lá?

Is there any other function or Project [in the organization] where you would never want to be? How would you feel working there?

Q36. {**Opinion**} O que a sua organização oferece ou faz para estimular a motivação dos engenheiros de software?

- *Probe*: What plans. incentives, events, etc. does the organization currently do in order to motivate its software engineers?

What does the organization explicitly offer to encourage the software engineers to work that extra-mile?

- *Probe: What plans. incentives, events, etc. does the organization currently do in order to motivate its software engineers?*

Q37. {**Opinion**} Como essas ações afetam o seu trabalho?

How do these actions affect your work?

Q38. {**Opinion**} O que a sua organização faz (e/ou que não deveria fazer) que mais desmotiva os engenheiros de software?

- *Probe*: What plans. incentives, events, etc. does the organization currently do in order to motivate its software engineers?

What does the organization do (but should not) that actually discourages the software engineers?

Q39. {**Opinion**} Como essas ações afetam o seu trabalho?

- *Sondagem*: Tanto comportamental como profissional.

How do these actions affect your work?

Q40. {**Opinion**} Na sua opinião, o que a empresa deveria/poderia fazer (mas não o faz) para trabalhar melhor a motivação dos engenheiros de software?

- *Probe*: ask for plans, incentives, events, integrations, etc

What else could the organization do (but currently does not) to deal with the motivation of the software engineers?

- *Probe: ask for plans, incentives, events, integrations, etc*

Q41. {**Opinion**} Projetando você daqui a 5 anos, que atividades você gostaria de estar fazendo?No que você gostaria de estar trabalhando ?

Talking about your future career expectations: what activities would you like to be doing in 5 years?

Q42. {**Opinion**} Projetando você daqui a 5 anos, que atividades você não queria fazer de jeito nenhum?E que tipo de projeto não gostaria de trabalhar?

Talking about your future career expectations: what activities would you never want to be doing in 5 years?

Q43. {**Opinion**} Para finalizar, como você definiria o termo "motivação"?

How would you define the term "motivation"?

Q44. Você gostaria de adicionar alguma informação ou observação que não foi perguntada, mas que você considere importante para a motivação de engenheiros de software?

Would you like to add any extra information or observation that we have not asked, but you think can be relevant for us?

Q45. Por favor, faça uma avaliação de dois pontos fortes e dois pontos fracos desta entrevista.

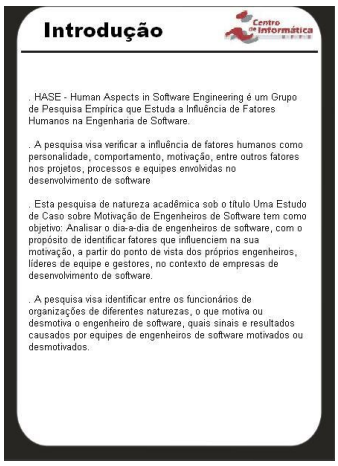
Please, suggest two strengths and two weaknesses of this interview.

APPENDIX D: DIARY NOTEPAD AND ONLINE SYSTEM

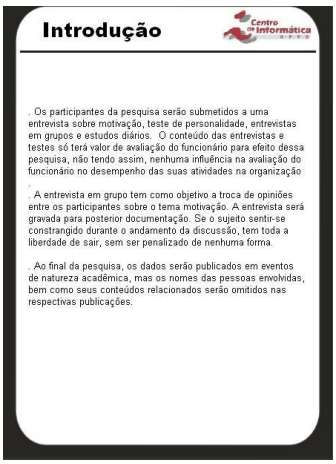
TABLE OF CONTENTS

A) DIARY NOTEPAD

Cover

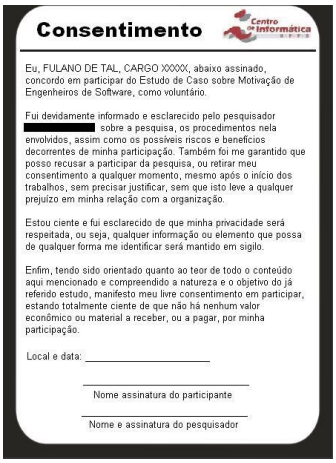


Page 1 – Introduction



Page 2 – Introduction (continued)

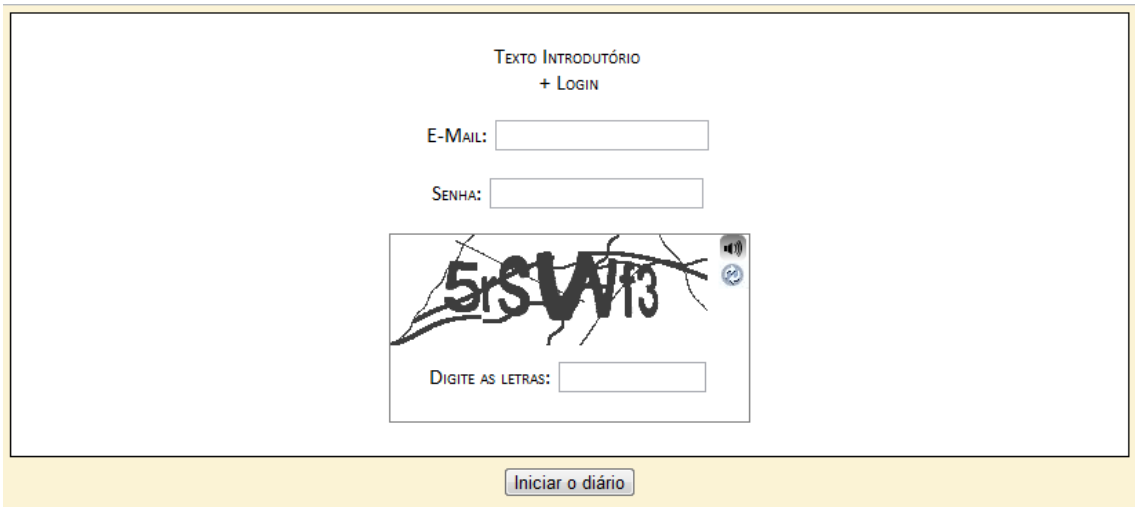
Page 3 – Introduction (continued)



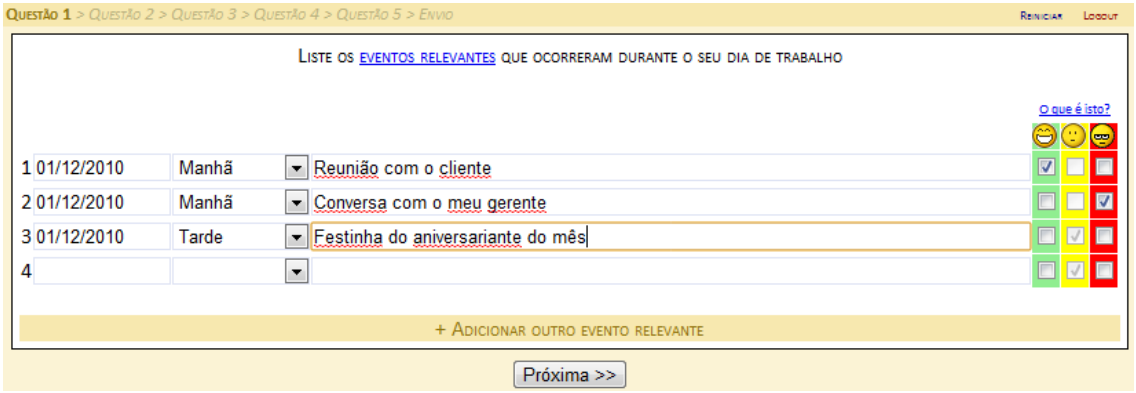
Page 4 – Consentiment Form

Remaining pages– Form

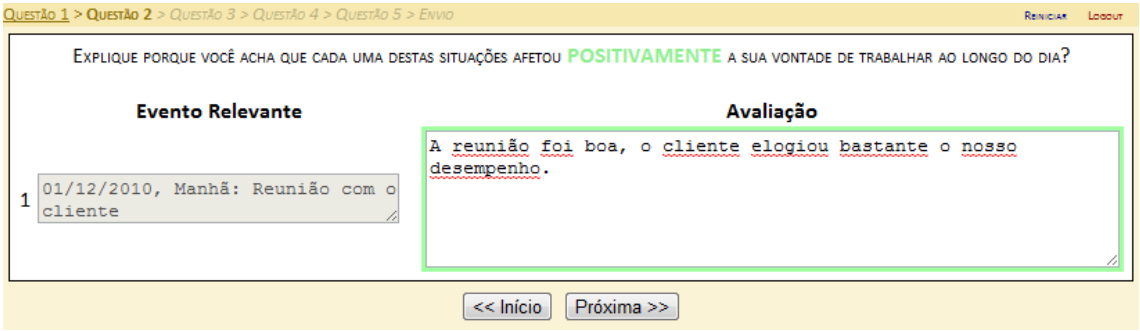
B) ONLINE SYSTEM



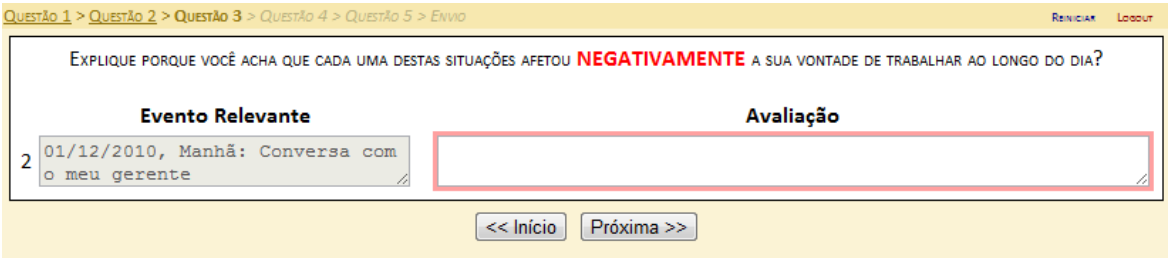
Screen – Login



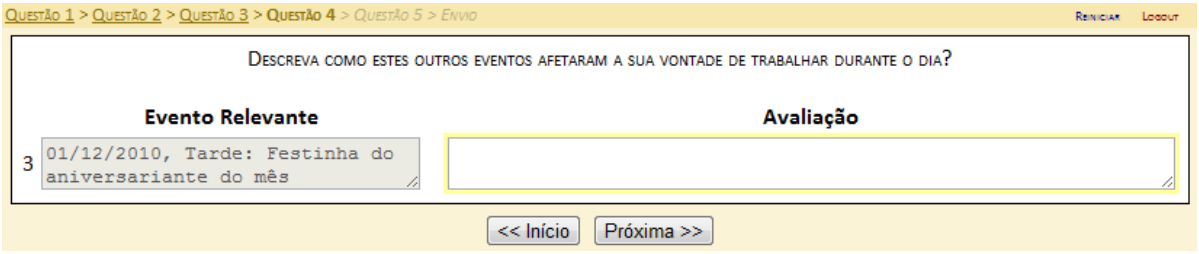
Screen – List of relevant events, date, shift and quick evaluation of the impact



Screen – Detailed evaluation of positive events

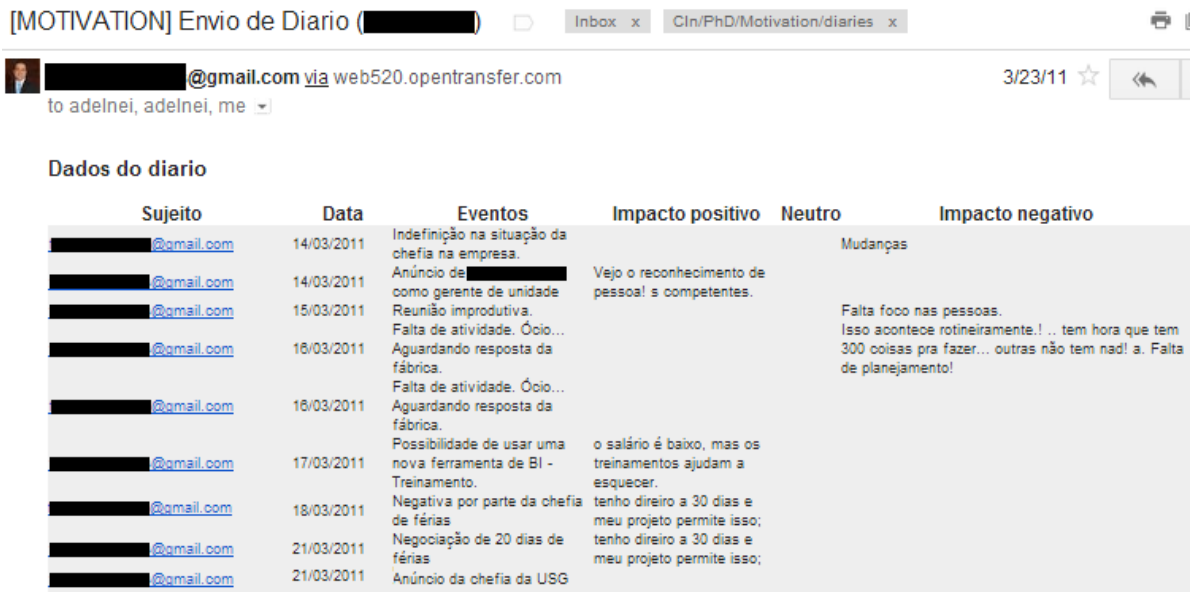


Screen- Detailed evaluation of negative events



Screen – Detailed evaluation of neutral events

C) OUTPUT OF THE ONLINE SYSTEM



Screen – Email sent by the web system (example)

APPENDIX E: CASE I – THE GOVERNMENT ORGANIZATION

TABLE OF CONTENTS

A) CASE DESCRIPTION

This case study was carried out in a government software organization situated in Recife, Brazil, established in 1969 by Government of the State of Pernambuco. Its core mission is to provide Information Technology services to internal customers in several levels of the State Government administration and also to the citizens of the State.

As a government owned organization, it is regulated under the laws and norms of the Brazilian public sector, which have two characteristics that are relevant for this study. First, since the Brazilian Constitution of 1998, public employees must be hired through an open process with universal access, based on objective criteria. This rules out subjective interviews, personality and behavioural assessment, peer indication, and other forms of employee selection found in the private sector. On the other hand, it slows down the process of hiring new employees and, therefore, makes it difficult to produce timely replacement when someone leaves the organization. Second, all public employees have job stability after a probation period of 3 years of work in the public sector (State Law Nº. 6.123/68).

Currently, the organization is structured in 14 main unities distributed in different locations throughout the State. Its employees, including software engineers, are distributed in the main unities and also in over 60 other public administration buildings. By the time this research was performed, the organization held 2,580 employees.

Regarding software development methods and practices, it uses traditional, process-oriented methods, with command and control style of management in most software projects, although some small and isolated agile initiatives could also be found. The organization explicitly stimulates the adoption of open source software in the State administration, and there is one open source project being currently developed.

B) DETAILS ON THE STUDY EXECUTION

Potential participants were initially contacted by e-mail, and invited to participate. The interviews were scheduled and conducted individually, in a meeting room in the organization's premises, between August and December 2010. Fourteen participants were selected: six software engineers, five project managers, and three directors. All interviews were recorded and together added up to 9 hours and 26 minutes of audio time.

Four software engineers were then selected from the set of interviewees to participate in the diary study. We followed the suggestions given by Dearman andTruong.(2008)to avoid low participation. Since the diary study required an additional effort from the participants, they had to be constantly stimulated and receive feedback to continue contributing. Diary data was collected during four weeks, between February and March 2011. Sixty-five events were reported, and retrospective interviews were also recorded, contributing to another 1 hour and 8 minutes of audio time.

Consistently with our sample strategy of maximum variation, we selected participants working in traditional, process-oriented projects, agile projects, and the open source project. Software engineers work as part of teams, assigned to specific development or maintenance project. The type of task (development or maintenance) and the composition of the team were found to be important for motivation, as will be seen below.

This case study has been fully reported in a masters' dissertation (FELIX, 2011) and in a conference paper (FRANÇA, FELIX and DA SILVA, 2012). Its details are available in a technical report (FRANÇA, FELIX and DA SILVA, 2012)

C) PARTICIPANTS' PROFILES

- **P006**: System Analyst, Male, 32 years old, B.Sc. in Computer Science, 10 years of work experience. He has been in different roles in this organization, and he describes himself as introverted and inpatient with people. Participated only in the interview (33' 50").
- **P007**: Developer, Male, 29 years old, B.Sc. in Computer Science, 8 years of work experience. Given his early interest in software programming, his work experience has been primarily as a developer. He states that he likes designing software architecture and he does not like testing. Participated only in the interview (67' 31").
- **P008**: System Analyst, Male, 27 years old, B.Sc. in Computer Science, 5 years of experience primarily in this organization, as he says that he had no previous relevant job experience. His work consisted mainly in generating reports from the database. He wants to have his own business in the future, but he is averse to risk. Participated only in the interview (28' 13").

- **P009**: Developer, Male, 32 years old, B.Sc. in Computer Science, but with previous background in geology and physics, 10 years of overall work experience. He likes technical activities in the software development. Participated only in the interview (63’ 06”).
- **P010**: System Analyst, Female, 29 years old, B.Sc. in Computer Science with specialization in Testing, 7 years of work experience. She works as system analyst simultaneously in three projects and does not like to program. She also does not like to be in charge for others. Participated in the interview (32’ 41”) and in the diary study (17 events).
- **P011**: System Analyst, Male, 39 years old, B.Sc. in Computer Science, 10 years of work experience. He has studied computer science because of the influence of familiars, but he does not like to program. Currently, he is studying business administration and has been recently promoted to a coordination position. Participated only in the interview (72’ 23”).

APPENDIX F: CASE II – THE NOT-FOR-PROFIT R&D ORGANIZATION

TABLE OF CONTENTS

A) CASE DESCRIPTION

This case study was carried out in a private and not-for-profit software development organization, which has unities in three states of Brazil.The organization's headquarters are located in the Porto Digital Science Park (http://www.portodigital.com.br), in Recife, Brazil. This organization was created through the merging of two Foundations, the first one created in 1994.

It operates in many different areas, such as Information Technology, Telecommunications, Industrial Automation, Solutions for the Public Sector, and Energy, by providing support services, workforce supply for third-parties, development of software and hardware products, software factory, product certification tests, and research and development of technological innovative products. The organization had a SW-CMMi level 2 certificate and was targeting the SW-CMMi level 3 at the time of the development of the case study. The management processes broadly followed the PMBOK guide, and managers were certified Project Management Professionals (PMP), but some projects have already been adopting SCRUM agile management practices.

At the time that this research was carried out, the organization had about 300 professionals, 85% part of the technical workforce and 15% allocated in administrative tasks. This case study was limited to the Recife unity, with 40 professionals. This unity had both hardware and software development projects, including web systems, mobile devices and embedded systems, using technologies such as .NET and Java. In this unity, there was no specific human resource management, and project managers performed the activities related to human resources management. This case study has been published as a masters' dissertation (ARAÚJO, 2011), and a conference paper(FRANÇA, ARAÚJO and DA SILVA, 2013).Its details (constructs and data excerpts) are available in a technical report(FRANÇA, DE ARAÚJO and DA SILVA, 2012).

B) DETAILS ON THE STUDY EXECUTION

Interviews were conducted in the organization's premises, between September and November 2010. Fourteen participants were selected: ten software engineers (three of which were team leaders), one project manager, and three directors. All interviews were recorded and together added up to 8 hours and 24 minutes of audio time. Five software engineers were then selected from the set of interviewees to participate in the diary study. Diary data was collected during seventeen business days, between February and March 2011. Eighty-six events were reported.

C) PARTICIPANTS' PROFILES

- **P018**: System Analyst, 36 years old, B.Sc. in Computer Science, 13 years of work experience. He has always liked programming and, even after promoted to System Analyst, he keeps involved in programming tasks at work. Participated in the interview (34' 33") and in the diary study (14 events).
- **P019**: Developer, Male, 2 years of work experience. He holds a technical degree in Information Systems. He is also a part-time firefighter, who started working as a lifeguard but, after getting graduated, migrated to the IT sector in the firefighters department. In the R&D organization, he works in different projects but for the same customer. Participated in the interview (31' 06") and in the diary study (18 events).
- **P020**: System Analyst, Male, in the thirties, 8 years of work experience. He graduated in Electronics, started his career testing telecom systems, and then migrated to software development. In the R&D organization he is currently allocated in different simultaneous projects that are in different stages of development. He likes requirements elicitation, but does not like to deal with contractual negotiations. Participated only in the interview (72' 31").
- **P021**: Tester, 25 years old, B.Sc. in Computer Engineering, 1 year of work experience. He is enrolled in a masters' program in computer science. He is primarily tester, but also helps on development when needed. He likes dealing with people, and describes himself as "restless". Participated only in the interview (39' 27").
- **P022**: Developer, 29 years old, B.Sc. in Computer Engineering, 3 years of work experience. This is his first job contract. He does not like testing and writing documents.Participated only in the interview (36' 12").
- **P023**: Developer, B.Sc. in Computer Science, with specialization in software engineering, 6 years of work experience. He joined the organization as a tester, but given his experience as web developer, migrated to development. He does not like writing documents. Participated only in the interview (32' 00").

APPENDIX G: CASE III – THE SMALL COMPANY

TABLE OF CONTENTS

A) CASE DESCRIPTION

The third case study was carried out in a software company, formally established in 2006 by the initiative of five entrepreneurs from the Information Technology sector, in Recife, Brazil. Its core mission is to support the development of people and organizations with software tools, by means of technical excellence and innovation.This company is specialized in software development for different platforms, with expertise in different programming languages (such as .NET Framework, Java family, LUA, and others). It focuses on the on-demand development of information systems, operating in areas such as management, finance, mining, health, and others. In addition, it also develops its own products. Its flagship product, a corporative social network, stands for intra-organizational innovation management. Currently, it serves national and international customers, usually medium and big companies. Internal products and external projects significantly differ in terms of requirements management process and time pressure. People from both types of projects participated in this research.

The company follows an agile-like software development process, broadly adopting practices such as regular delivery of software, adaptive management style (SCRUM based), small teams, face-to-face meetings, and customer authority. The organizational structure is flat, and the directors eventually act as part of the development teams. The directors themselves, who have software engineering background, instead of management, administer all organizational issues, including the human resource management. At the time that the case study was carried out, the company was composed of 27 people, everyone younger than 30 years (directors included), occupying functions in one of the three types of teams: software development, research and design areas. Some of these people were in the organization for less than six months, while others had more than 3 years along with the team. As an organizational strategy, the company is closely tied to the academy, both physically (its location is near a University) and operationally, since its staff is composed of undergraduate students (trainees) as well as graduated students in software engineering. We sampled participants representing all clusters.

B) DETAILS ON THE STUDY EXECUTION

Interviews were carried out at the company's own facilities, during May 2011. We had access to ten software engineers, plus two project managers and two directors. Each participant was first contacted in advance, and each interview occurred in a private meeting room. All interviews were recorded and together added up to 8 hours and 57 minutes of audio time. According to the original case study protocol, data was supposed to be

complemented by the use of diary studies. After the interviews, six participants were selected to report any relevant event that affected his/her motivation at work, during a period of one month. Even though we have followed suggestions to avoid low participation(DEARMAN, KELLAR and TRUONG, 2008), only 10 relevant events were reported, and retrospective interviews pointed to their high workload as the main cause of low participation.

C) PARTICIPANTS' PROFILES

- **P028**: Developer, Male, 29 years old, B.Sc. in Computer Science, 8 years of work experience. He is interested in computers since his childhood, but he is still in the initial steps on his career. He works for the internal product. Participated only in the interview (38' 51").
- **P029**: Tester, Female, 21 years old, undergraduate, first work experience. She likes to deal with people, and hates programming. She decided to work as tester because of this job opportunity, and she is currently working as part-timer, Participated in the interview (44' 19") and in the diary study (4 events).
- **P030**: Developer, Male, 28 years old, 7 years or work experience, undergraduate. Became software engineer because he believed that could get rich in a short time. Currently work as a front-end engineer. Participated only in the interview (33' 16").
- **P031**: Designer, Male, 22 years old, undergraduate, majoring in Design, 1.5 year of work experience. He works as a part-timer, and this is his first job contract. Participated in the interview (25' 51") and in the diary study (2 events).
- **P032**: Developer, Male, 27 years old, undergraduate, 9 years of work experience, pursuing a degree in Computer Engineering. Formerly, he was interested in Electronics, but after his first contacts with programming, he changed his major. He is interested in embedded systems and audio signal processing. He is quiet and introverted. He does not like testing. He works as part-timer. Participated only in the interview (76' 06").
- **P033**: Part-time Developer, Male, Undergraduate, 23 years old, first job experience. He likes writing code, but feels like he was in the wrong profession because he thinks that programming is stressful. Participated only in the interview (25' 29").
- **P034**: Developer, Male, B.Sc. in Computer Science, 26 years old, 6 years of work experience. He is one of the founders, and shareholders, of the company, but given his technical orientation he prefers to keep working as a developer. He likes programming and does not like *sysadmin* tasks. Participated only in the interview (58' 23").
- **P035**: Part-time developer, Male, Undergraduate, 21 years old, first job experience. He is still learning about software production and, so far, is enjoying the work. Participated only in the interview (26' 40").

- **P036**: Developer, Male, 26 years old, 4 years of work experience. Started a Mechanical Engineering course, but quit. He describes himself as a self-learner, because his technical knowledge has been acquired by studying alone. He likes programming. Participated only in the interview (30’ 16”).
- **P037**: Part-time tester, Male, 23 years old, undergraduate, 1 year of work experience. He is interested in development, and writes code during his free time at home. Participated only in the interview (41’ 25”).

APPENDIX H: CASE IV – THE IT DEPARTMENT OF A UNIVERSITY

TABLE OF CONTENTS

A) CASE DESCRIPTION

This department is accounted for the Information Technology services of a federal university in Recife. It is responsible not only for the maintenance of the software system that holds all the valuable information of that organization (such as academic and patrimonial info), but also for the improvement of this system as well as the development of software to supply all the information needs of the organization. Its core product was released in the early 2000's, and since then has continuously evolved and adapted. It is a web-based system, written in Java, with about 840 functionalities, more than one million Lines of Code, and at the time this case study was carried out, its website received about four thousand hits per month.

The department is mainly organized in three sectors: one responsible for the inception of new projects and products to improve the informational procedures in the university; another is exclusively responsible for the maintenance of the academic module; and a third one is responsible for the elaboration and development of a new module. Regarding the software development process, this department follows an agile SCRUM-based approach. Internal procedures are defined and continuously improved by a study group, which aims to make these internal processes adherent to the MPS.br model[1].

The development process was already stable regarding the configuration management, project management, requirements management, portfolio management and quality assurance. Some initiatives were serving as pilot studies for procedures such as acquisition, measurement, validation and verification. Regarding the human management procedures, this department had 37 professionals, working under three different types of contracts: eighteen public employees, eleven third-parties workforce and eight internships.

[1] MPS.br is a Brazilian model for software development process improvement, compatible with CMMi, developed by SOFTEX (http://www.softex.br).

The former category is composed of government employees, and therefore have the same rights as described in the case I. Third-party workforce are regular employees of another organization that is responsible for supplying workforce for many departments in the university, so they have a regular private employment contract with the third-party organization, but they are 100% allocated in the studied department. Interns are contracted under a standard educational internship contract, with (supposedly) less responsibilities and less work time in the organization.

B) DETAILS ON THE STUDY EXECUTION

In this case, the participant selection was guided by the organisational matrix, which detailed who worked in each sector. Another document provided the entering date of every professional, so we could select professionals with different work experience in the organisation. The interviewer was also informed who were the most and least influent engineers, so we could select a mixed up sample. A total, 10 software engineers, two coordinators (project managers) and one director were interviewed. After the first personal contact with the potential participants, the Interviews were conducted at the company's own facilities, between February and March 2011. Each participant allowed the audio recording in the beginning of the interviews.

Then, the diary participants were selected partly among the participants of the interviews, and partly among the reamining professionals. Six engineers were selected, but onyl four effectively contributed to this activity. The other two withdrawed the activity. During a four-week period between March and April 2011, they submitted 60 diary events. A brief training of 1 hour was provided, and the participants opted for fulfilling directly the online version of the diary, as soon as the events ocurred. They explained that they were on the computer most of the time, so it would not represent a risk for the validity of the collection.

C) PARTICIPANTS' PROFILES

- **P040**: Developer, Male, 26 years old, technical degree in software development and B.Sc. in System Analysis and Development. He decided for this career because his early interests in computers and gaming. He describes himself as having a strong technical orientation, because he likes development. Participated only in the interview (51' 16").
- **P041**: System Analyst, Male, 27 years old, M.Sc. in Computer Science, 6 months of work experience. Given his short professional experience, he does not know exactly what activities he prefers, but he said that he likes to interact

with customers, and does not like writing code. He describes himself as impatient. Participated only in the interview (67’ 43”).

- **P042**: Developer, Male, 25 years old, B.Sc. in Computer Science, 2.5 years of work experience. He likes development. Although he is system analyst, he also has to act as a developer. He describes himself as curious. Participated only in the interview (58’ 28”).
- **P043**: Developer, Male, 23 years old, undergraduate, pursuing the degree of B.Sc. in Computer Science. He has a strong technical orientation. He says that he has always liked programming. Participated only in the interview (39’ 49”).
- **P044**: System Analyst, Male, 29 years old, B.Sc. in Computer Science with specialization in Data Base Administration, 4 years of work experience. He started as developer, and has been promoted to system analyst. However, he is actually interested in data base management. He thinks he does not have leadership profile, and he would not like to be in charge of one of the products in the organization. Participated only in the interview (45’ 56”).
- **P045**: Developer, Male, Technological degree in Systems Analysis and Development, post-grad in Project Management, 7 years of work experience. He started his studies in biology, then migrated to informatics. He likes programming, but he is tired of the repetitive routine of a programmer. Participated only in the interview (73’ 17”).
- **P046**: System Analyst, Male, 27 years old, B.Sc. in Information Systems, 1.5 years of work experience. His early interests were in electronics, in which he has got a technical degree. His work experience began in private companies, and he also has started his own company. He describes himself as an entrepreneur. He likes documenting and does not like programming. Participated in the interview (40‘ 07“) and in the diary study (6 events).
- **P047**: Developer, Female, 40 years old, B.Sc. in Computer Science, 1 year of work experience. This is her first work experience as a developer. Formerly, she used to work as administrative assistant. She likes programming, but also reports interest in requirements. Participated only in the interview (55’ 23”).
- **P048**: Developer, Male, 21 years old, Technical degree in Systems Analysis and Development, 3 years of work experience. Interested in programming. Participated only in the interview (44’ 39”).
- **P049**: Developer, Female, 23 years old, B.Sc. in Computer Science, pursuing a degree of M.Sc. in Computer Science, 2 years of work experience. This is her first job experience. She likes working with computer networks, and she is thinking about leaving this job shortly. She said that she cannot get concentrated easily. Participated in the interview (63’ 45”) and in the diary study (26 events).