Instituto Alberto Luiz Coimbra de Pós-Graduação e Pesquisa de Engenharia
COPPE UFRJ

ESTIMATIVA DE ESFORÇO EM PROJETOS DE ESPECIFICAÇÃO DE REQUISITOS DE SOFTWARE

Verônica Taquette Vaz

Dissertação de Mestrado apresentada ao Programa de Pós-graduação em Engenharia de Sistemas e Computação, COPPE, da Universidade Federal do Rio de Janeiro, como parte dos requisitos necessários à obtenção do título de Mestre em Engenharia de Sistemas e Computação.

Orientador: Guilherme Horta Travassos

Rio de Janeiro
Dezembro de 2013

ESTIMATIVA DE ESFORÇO EM PROJETOS DE ESPECIFICAÇÃO DE REQUISITOS DE SOFTWARE

Verônica Taquette Vaz

DISSERTAÇÃO SUBMETIDA AO CORPO DOCENTE DO INSTITUTO ALBERTO LUIZ COIMBRA DE PÓS-GRADUAÇÃO E PESQUISA DE ENGENHARIA (COPPE) DA UNIVERSIDADE FEDERAL DO RIO DE JANEIRO COMO PARTE DOS REQUISITOS NECESSÁRIOS PARA A OBTENÇÃO DO GRAU DE MESTRE EM CIÊNCIAS EM ENGENHARIA DE SISTEMAS E COMPUTAÇÃO.

Examinada por:

____

Prof. Guilherme Horta Travassos, D.Sc.

____

Prof. Cláudia Maria Lima Werner, D.Sc.

____

Prof. Sérgio Castelo Branco Soares, D.Sc.

RIO DE JANEIRO, RJ - BRASIL

DEZEMBRO DE 2013

Vaz, Verônica Taquette

Estimativa de Esforço em Projetos de Especificação de Requisitos de Software/Verônica Taquette Vaz. – Rio de Janeiro: UFRJ/COPPE, 2013.

XII, 104 p.: il.; 29,7 cm.

Orientador: Guilherme Horta Travassos

Dissertação (mestrado) – UFRJ/ COPPE/ Programa de Engenharia de Sistemas e Computação, 2013.

Referencias Bibliográficas: p. 89-92.

1. Engenharia de Requisitos. 2. Estimativa de Esforço. 3. Engenharia de Software Experimental. I Travassos, Guilherme Horta. II. Universidade Federal do Rio de Janeiro, COPPE, Programa de Engenharia de Sistemas e Computação. III. Título.

À minha mãe, por todo o apoio e pela inspiração em seguir seus passos na vida acadêmica.

AGRADECIMENTOS

Agradeço acima de tudo à minha família, por ter me provido toda a estrutura que me permitiu chegar tão longe.

Aos meus amigos e sócios da Inoa Sistemas, em especial ao Leonardo Arnt, João Pedro Francese e Rafael Zuquim, pela inspiração profissional, por me ajudarem na busca pela maestria na aplicação dos meus conhecimentos no trabalho, e por construírem junto comigo a empresa dos meus sonhos.

À professora Tayana Conte pela amizade, por todos os ensinamentos, e por manter sua dedicação e apoio mesmo estando distante.

A todos os meus companheiros do grupo de engenharia de software experimental, especialmente ao Paulo Sérgio Medeiros, Rafael Mello, Jobson Massolar, Breno França e Pedro Corrêa, pelo apoio e contribuições à minha pesquisa.

Ao meu orientador, Guilherme Travassos, pelos ensinamentos, pela paciência, pela confiança depositada em mim e pelas oportunidades que me foram dadas.

A equipe da secretaria do PESC, por toda a atenção dispensada.

A todos os participantes dos estudos experimentais pela sua colaboração.

À Universidade Federal do Rio de Janeiro e ao Instituto Alberto Luiz Coimbra de Pós-graduação e Pesquisa de Engenharia (COPPE).

À CAPES, pelo apoio financeiro ao longo do mestrado; e ao PESC, pelo apoio financeiro para apresentação de artigos em conferências.

Resumo da Dissertação apresentada à COPPE/UFRJ como parte dos requisitos necessários para a obtenção do grau de Mestre em Ciências (M.Sc.)

ESTIMATIVA DE ESFORÇO EM PROJETOS DE ESPECIFICAÇÃO DE REQUISITOS DE SOFTWARE

Verônica Taquette Vaz

Dezembro/2013

Orientador: Guilherme Horta Travassos

Programa: Engenharia de Sistemas e Computação

Esta dissertação apresenta o trabalho de pesquisa realizado com o intuito de construir um modelo para apoiar a estimativa de esforço em requisitos de projetos, ou seja, projetos em que o produto final esperado é a especificação de requisitos de software em si, e o desenvolvimento do sistema está fora do escopo. Este tipo de projeto é particularmente comum em empresas que contratam fábricas de software, que precisam fornecer à fábrica uma especificação de requisitos de software explícita e completa. Neste cenário, técnicas de estimativa de esforço convencionais não são completamente adequadas, uma vez que a especificação de software corresponde ao produto final esperado, em vez de ser a entrada para apoiar o planejamento futuro. Portanto, esse trabalho foi conduzido com o objetivo de identificar informações baseadas em evidências sobre as variáveis de contexto que podem afetar o esforço envolvido em atividades de requisitos de software, como um primeiro passo de nossa pesquisa para uma técnica de apoio à estimativa de esforço de projetos de especificação de requisitos de software. Todos os estudos e os resultados obtidos são detalhadamente descritos nesta dissertação.

Abstract of Dissertation presented to COPPE/UFRJ as a partial fulfilment of the requirements for the degree of Master of Science (M.Sc.)

EFFORT ESTIMATION IN SOFTWARE REQUIREMENTS SPECIFICATION PROJECTS

Verônica Taquette Vaz

December/2013

Advisor: Guilherme Horta Travassos

Department: Computer Science and System Engineering

This dissertation concerns software technologies to support effort estimation in requirements projects, i.e., projects where the final expected product is the software requirements specification itself, the system development is out of scope. This type of project is particularly common in companies hiring software factories, which need to provide the factory with explicit and complete software requirements specification. In this scenario, conventional effort estimation techniques are not completely adequate, as the software specification corresponds to the expected product rather than the input for support future planning. Therefore, we aimed at identifying evidence-based information on context variables that may affect the effort involved in software requirements activities, as a first step of our research towards a technique to support the effort estimation of software requirements specification projects. All studies and results obtained so far are fully described in this dissertation.

INDEX

INDEX OF FIGURES

INDEX OF TABLES

CHAPTER 1 - INTRODUCTION

1.1 Motivation

Precise effort estimation is a factor of great importance for ensuring that correct planning will be accomplished in software projects. According to MENDES (2007), when executing software projects, effort estimation is the basis for project schedule, cost estimation and resources allocation. A software project schedule is prepared based on the estimated effort distribution over time, respecting the practitioners' calendars, resources, working days and daily working hours' restrictions. Project cost, on the other hand, may also include many other factors (such as hardware costs, environment costs, among others), but it is mostly influenced by the cost of human resources (JORGENSEN AND SHEPPERD 2007). Therefore, project cost estimation is directly related to the estimated effort, multiplied by the cost of the resources to be involved in the activities. For this reason the majority of researchers in this field use the terms cost and effort interchangeably (JORGENSEN AND SHEPPERD 2007). Therefore, the term effort will be used for general purpose in this dissertation.

Current effort estimation techniques (e.g., BOEHM et al. 1995; IFPUG; MENDES 2007) consider the complete software development process and use the software requirements specification as a basis for providing an indication of the features and size of the system that will be developed. This information is used to estimate the development effort, schedule and project cost. In this sense, higher knowledge and thorough software requirements details can assist in obtaining more precise estimates. Therefore, in order to apply the current techniques with acceptable precision, it is necessary to make most of the effort involved in requirements specification activities, prior to estimating the whole project effort (Figure 1-1). In this case, the effort spent in requirements specification activities is, in general, assigned to the project's risks.

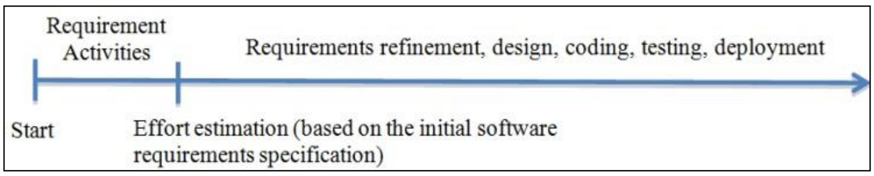


Figure 1-1 – Timeline Representation in Software Development Projects

However, our experience with software projects in the industry has shown that some software companies are working with software-related projects in which the final expected product is the requirements specification. In other words, the project hinges on identifying, analyzing, specifying, verifying, validating, and managing the expected software requirements. In this case, the software development is left out of the project scope. This kind of project is particularly common in companies hiring software factories, which need to provide the factory with an explicit and complete software requirements specification to support software development. Another scenario where requirements projects are also common can be seen in companies working with fixed price and scope projects. In this scenario it is really important to have a deep understanding of requirements prior to project budget and schedule agreement, as the company will be committed to a fixed scope from the beginning. In this case, companies usually run a preliminary project to further analyze and specify the requirements before actually engaging in the development project.

For this type of project covering only requirements activities, the use of current effort estimation techniques is not completely adequate because they take into account different context variables. For instance, requirements specification corresponds to the project deliverable rather than its input (Figure 1-2). Therefore, it is relevant to investigate and define context variables that can affect software requirements specification projects and possible models to support effort estimation in such projects.

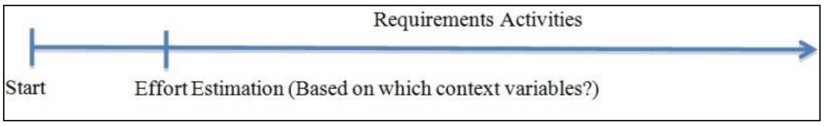


Figure 1-2 - Timeline Representation in Requirements Projects

1.2 Context

ZAVE (1997) defines Requirements Engineering as ‘the branch of Software Engineering concerned with the real-world goals for, functions of, and constraints on, software systems. It is also concerned with the relationship of these factors for the precise specifications of software behaviour, and for their evolution over time and across software families. The subject of Requirements Engineering is inherently broad, interdisciplinary, and open-ended. It concerns translation from informal observations of the real world to mathematical specification languages.’ Requirements Engineering processes usually consist of a set of activities regarding requirements elicitation, analysis, specification, verification, validation, and requirements management.

One of the main topics in the requirements research agenda is estimating software costs, risks, and schedules (ZAVE 1997). JORGENSEN AND SHEPPERD (2007) present a systematic review on current software effort estimation techniques. Although these techniques vary on the method to derive estimations (for instance: regression, analogy, expert judgment, work break-down, function points, classification and regression trees, simulation, neural network, theory, Bayesian, combination of estimates), they all share the perspective of estimating the overall development effort based on the initial knowledge of software requirements. As this initial high-level requirements specification is done prior to effort estimation, the effort involved in this stage is not estimated and is considered as part of the project's risk.

As we could see in our initial searches, few studies have focused so far on supporting the estimation of the requirements stage effort. MALIK and BOEHM (2008) present a study on the prediction of elaboration factors – the ratio of elaboration between each level of abstraction of requirements (e.g., goals → features → use cases → use case steps). In this study, the authors ran a survey on 20 real-client projects to quantify the elaboration factors of each project. They assumed that using historical data on the elaboration factors in previous projects a manager could estimate the requirements specification for a future project from the time the high level requirements have been defined. For example, if a project identifies 20 business goals, and the historical data shows that the elaboration factor between goals and use cases is of 1.5, a software manager can estimate that this project will contain 30 use cases. This notion of size, along with the notion of productivity (number of man-hours needed to specify a use case) could be used as a basis to estimate the effort to be invested in requirement activities.

However, as observed by the same authors, after gathering the elaboration factors of all 20 projects, it was not possible to fit these data points into a single regression line. Otherwise, a stronger relation could be observed when projects were split into three groups: Low Elaboration Factor (LEF), Medium Elaboration Factor (MEF), and High Elaboration Factor (HEF).

This means that if it is possible to know, at the beginning of a project, which regression line the project will fit into (LEF, MEF or HEF), one could use this historical data to estimate the elaboration factor of one's project. This way, the problem then is how to classify *a priori* which EF curve the project will fit into. The authors put forward some suggestions, such as observing the project's complexity and novelty but no conclusive approach has so far been found (MALIK and BOEHM 2008; MALIK et al 2009; MALIK et al 2010; MALIK and BOEHM 2010).

Another approach to estimate requirements effort is presented in MAO et al (2005). In this paper, MAO et al present an artificial neural network model for predicting the effort spent on requirements changes during the later stages of software development (design, coding, testing, and maintenance). This model uses a set of factors (representing some project characteristics) as inputs and the output is the effort in person-days needed to tackle the requirements changes.

This paper was very useful for our work as it suggests a series of factors that can influence the requirements changes effort, and so it has been used as control paper to undertake a systematic literature review, that will be described in the following chapter. However, this work still does not completely answer our research question (which factors may affect the software requirements specification effort?) as it does not deal with the effort needed to build the initial software requirements specification. It only deals with the effort needed to tackle future changes in this specification.

As far as we could look into, no other approach dealing specifically with software requirements effort estimation could be found in the technical literature, which gives us an indication that this research topic is still open to further contribution.

1.3 Problem

The problem tackled by this work is the need to plan and estimate requirements projects in a systematic manner rather than relying just on the tacit knowledge of the experts. Currently available effort estimation techniques are hardly applicable in this kind of project as they require significant knowledge on the software requirements prior to their application, in order to be used with minimally acceptable precision.

To be able to support requirements projects effort estimation we first need to understand which context variables affect requirements activities and how they relate to the project's effort.

1.4 Objective

This work aims at identifying which context variables might be relevant to consider when estimating requirements effort and to understand how these variables relate to requirements effort.

This dissertation discusses the results of this research, presenting the identified context variables and the relations between these variables and the effort involved in requirements activities. It shows how these results were obtained though a secondary study, which was than analysed and synthesized through the application of a qualitative research methodology and evaluated and refined though a survey with researchers and practitioners working with projects that cover software requirements specification.

1.5 Methodology

KITCHENHAM et al. (2004) states that Software Engineering should be based on evidence, just as it is done in other disciplines such as the Medical Sciences, providing means by which the best evidence from research can be integrated with practical experience to support decision-making in software development and maintenance.

MAFRA et. al. (2006) identify the need to undertake secondary studies before the primary studies suggested by (SHULL et. al., 2001), to propose an evidence-based technology. The term secondary study refers to the execution of a study that aims to identify, evaluate and interpret all relevant results in a particular research topic, the phenomenon of interest or research question. A systematic literature review is a type of secondary study (KITCHENHAM 2004; BIOLCHINI et al. 2007).

Therefore, the first step in our research methodology was to conduct a systematic literature review aiming to characterize the current research on requirements effort. This resulted in the extraction of a large amount of qualitative data, which had to be combined and synthesized to make results comprehensible. To support this analysis and synthesis process, we applied the principles of the Grounded Theory methodology, through which we were able to build a set of conceptual maps representing the effort involved in requirements projects and how they relate to the project’s context variables. Such conceptual maps were important to render the results more visible and concise. Finally, we conducted a large-scale survey for the purpose of assessing the adequacy of these results and to supplement it, based on the experience of the practitioners and researchers. Table 1-1 shows the timeline of this work.

Table 1-1 - Research Timeline

Activity	Timeline
Systematic literature review planning	From September/11 to December/11 (4 months)
Systematic literature review execution	From January/12 to June/12 (6 months)
Applying Grounded Theory to analyze and synthesize the results	From July/12 to January/13 (7 months)
Survey planning	From February/13 to May/13 (4 months)
Survey execution	From May/13 (15 days)
Survey data analyses	From June/13 to September/13 (4 months)

1.6 Organization

This dissertation is organized in five chapters, including this first chapter that presented the motivation and context of this research. The remaining of the text is organized as follows:

- **Chapter 2 – Literature Review**: Describes the secondary study that was conducted to identify previous studies that present evidence on context variables that have an influence on the effort involved in requirements projects.
- **Chapter 3 – Literature Review Result - Analysis and Synthesis**: Shows how the results obtained from the literature review were analyzed and synthesized applying the principles of Grounded, a qualitative analysis method.
- **Chapter 4 – Evaluation and Refinement of Results**: Describes the large-scale survey, done to evaluate and refine the results obtained from the literature review.
- **Chapter 5 – Conclusions and Future Work**: Shows the conclusions and contributions of this work and points paths for prospective research.

CHAPTER 2 - SYSTEMATIC LITERATURE REVIEW

This chapter describes the execution of a quasi-systematic literature review with the purpose of identifying the context variables that should be considered when estimating the effort involved in software requirements specification projects and to characterize the influence (increase, decrease, restriction, condition) applied by these variables in the effort. The results of this literature review are presented.

The first step in building a model to estimate project effort is to determine a list of potential / most important effort cost drivers (BOEHM, 1981). Therefore, we decided to investigate the previous studies published in the technical literature that present some evidence on which context variables can influence the effort level needed to carry out Requirements Engineering activities.

With the objective of achieving a degree of scientific rigour, it is recommended to conduct a systematic review of the literature (KITCHENHAM 2004). Unlike informal reviews of the literature, where the researcher does not follow a defined process, a systematic review is performed in a formal way, following a pre-established protocol. Compared with informal reviews of the literature, systematic reviews require greater rigour in their undertaking. In contrast, the outcome tends to be more reliable, as they make use of rigorous methodology and are subject to auditing and repetition.

Thus, we performed a *quasi*-systematic review with the purpose of identifying the context variables that should be considered when estimating the effort involved in software requirements specification projects and to characterize the influence (increase, decrease, restriction, condition) as applied by these variables in the effort. This chapter describes how this review was conducted and the results obtained. By presenting this review, we also mean to provide other researchers with a summary of the current work in the area that serves as a research basis for future work in Requirements Engineering.

2.2 Literature Review on Factors that Influence the Requirements Activities

To achieve an appropriate level of evidence for the characterization of a particular technology, Evidence Based Software Engineering should primarily make use of two types of studies: primary and secondary studies (BIOLCHINI *et al.* 2007).

Primary studies aim at characterizing a technology in use within a specific context. This category includes controlled experiments, case studies, and *surveys*[1] (WÖHLIN *et al.* 2000).

Secondary studies aim at identifying, evaluating and interpreting all relevant results regarding a particular research topic, the phenomenon of interest or research question. A systematic review is a kind of secondary study (KITCHENHAM *et al.* 2004; PAI *et al.* 2004). Results obtained by several related primary studies act as a source of information to be investigated by a secondary study (MAFRA *et al.* 2006).

Systematic reviews are based on a well-defined search strategy, which aims to detect the maximum amount of relevant bibliographic material in a rigorous and auditable manner. Before starting the search for primary studies, one should define a revision protocol that specifies the central research question and the methods that will be used to perform the review. The protocol should indicate explicit inclusion and exclusion criteria and document the search strategy so to allow readers (and other researchers) to assess their degree of accuracy and completeness (BIOLCHINI *et al.* 2007).

The purpose of this literature review was to identify which context variables can influence the effort level needed to carry out the requirements development activities. Such variables can be, for instance: events, configurations, risk factors, characteristics or features related to: the environment, methods, people or project resources, as well as other unknown variables that have to be identified (Figure 2-1).

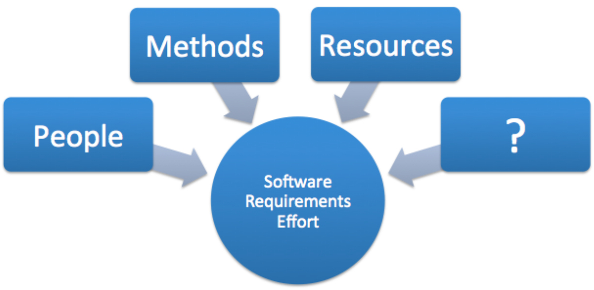


Figure 2-1 - Types of variables that may influence the effort in requirements activities

The goals of this review are summarized in

Table 2-1, according to the GQM paradigm (BASILI and ROMBACH 1988):

Table 2-1 - Goals of the *quasi*-systematic review according to the GQM paradigm

Analyze	Context variables related to the effort, cost, schedule or time, in

[1] Data collection through instruments such as questionnaires

	software requirements specification projects
With the purpose of	Characterizing
Regarding	The influence (increase, decrease, restriction, condition) applied by these variables on the effort, cost, schedule or time
From the point of view of	Software project managers
In the context of	Projects covering software requirements specification as described in the technical literature

This systematic review protocol follows the guidelines described in (BIOLCHINI *et al.* 2007) and uses the PICO approach (PAI *et al.* 2004) for the organization and structure of the search string: **P**opulation of interest, evaluated **I**ntervention, Intervention **C**omparison (if any) and **O**utcome (expected output). This study, in particular, does not have any element or baseline allowing a comparison with the intervention. Therefore, the Comparison dimension is represented as an empty set. Taking this into account and considering the research protocol formalism, it can be classified as *quasi*-systematic literature review (TRAVASSOS *et al.* 2008). Here we present the complete research protocol of this review, to allow other researchers to audit and to reproduce it.

Research question

This study's goal is to identify factors that can influence requirements development activities (identification, analysis, specification, verification, validation, and requirements management). Further, we aim at identifying how these factors can be measured or observed; and which models make use of them.

Main Question: Which factors influence the effort, cost, time or schedule of the requirements development activities?

Secondary Question: How can these factors be measured or observed?

Tertiary Question: Which estimation models use these factors?

- **Population**: Software development projects or processes, or empirical studies.

  Keywords:

  - "Software project", "software development", "software process", "application project", "application development", "application process", "system project", "system development", "system process"

- "Empirical Study", experiment, experimentation, "Empirical Assessment", "Empirical Evaluation", "Experimental Study", "Case Study", survey, "Pilot Study", "field study"

- **Intervention**: Factors influencing the effort, cost, time or schedule of the software requirements development activities.

  Keywords:

  - Factor, variable, “risk factor”, predictor, event, characteristic, attribute, cost driver, parameter
  - Effort, schedule, plan, time, cost, budget, risk
  - “software requirements elicitation”, “requirements identification”, “requirements analysis”, “requirements gathering”, “requirements specification”, “requirements validation”, “requirements management”, requirements verification”, “requirements negotiation”, “Requirements Engineering”, “requirements stage”, “requirements project”, “requirements process”

- **Comparison**: empty
- **Outcome**: influence exerted by these factors.

  Keywords:

  - Increasing, Decreasing, restriction, Constraint, Impact, Influence, Effect, Trend, result

Search Strategy

The search strategy should make explicit the search scope and keywords to be used in it, which are used to compose the search string. These keywords are derived from the population, intervention and outcomes defined in the research question. In this review, apart from the researcher who defined the first version of the protocol, another researcher reviewed it and helped defining the keywords.

To carry out this study, we selected the digital libraries that granted full access to the papers’ complete text through the CAPES portal[2]. In the libraries available in this portal, we selected the EI Compendex[3], IeeeXplore[4] and Scopus[5] databases, as they

[2] www.periodicos.capes.gov.br
[3] www.engineeringvillage2.org
[4] www.ieeexplore.ieee.org
[5] www.scopus.com

provide more coverage for the publications in this area (DIESTE *et al.* 2009). We considered papers that have been published in English, in journals or conference proceedings and that have their complete text available in these electronic databases.

The search string was defined through the combination of the keywords identified in the population, intervention, comparison and outcome of this research protocol, resulting in the following string:

(("software project" OR "software development" OR "software process" OR "application project" OR "application development" OR "application process" OR "system project" OR "system development" OR "system process" OR "Empirical Study" OR experiment OR experimentation OR "Empirical Assessment" OR "Empirical Evaluation" OR "Experimental Study" OR "Case Study" OR survey OR "Pilot Study" OR "field study") AND ((factor OR variable OR predictor OR event OR characteristic OR attribute OR driver OR parameter) AND (effort OR schedule OR cost OR budget OR duration) AND ("requirements elicitation" OR "requirements identification" OR "requirements analysis" OR "requirements gathering" OR "requirements specification" OR "requirements engineering" OR "requirements stage" OR "requirements project" OR "requirements process" OR "requirements activity" OR "requirements development" OR "requirements validation" OR "requirements management" OR "requirements verification" OR "requirements negotiation")) AND (increasing OR increase OR decreasing OR decrease OR affecting OR affect OR restriction OR constraint OR impact OR influence OR effect OR trend OR result))

Inclusion and exclusion criteria

Inclusion Criteria:

- The paper should be available in the Web, and;
- The paper should be written in English, and;
- The paper should present factors that influence (directly or indirectly) the effort, cost, time or schedule of the software requirements activities, and;
- The paper should present which features of the requirements process are affected by these factors.

Exclusion Criteria:

- Not meeting any Inclusion Criterion.

Quality Criteria

The criteria below were used to evaluate the selected papers' quality, regarding this research protocol and our perspective. Therefore, each criterion is marked with a maximum of one or half a point, according to our interpretation of its relevance to the research question. The goal of the evaluation is to highlight those papers that could be more related to the research question and, consequently, provide more reliable final results.

Criteria related to the influence factors:

- Is there a description of how the influence factors were identified? (1.0 pts)
- Is there a description of how the influence factors could be measured or observed? (1.0 pts)
- Is there a description of which features from the requirements activities are influenced by these factors? (1.0 pts)
- Does the paper indicate what the value or direction (e.g. positive or negative) if of such influence? (1.0 pts)
- Is there a description of the organization's or project's features where the factors were identified? (0.5 pts)
- Does the paper present some application or experience report of use of these factors? (0.5 pts)
- Does the paper present any empirical study that evaluates the existence of relationships between the factors identified and any feature from the requirements activities? (1.0 pts)

Criteria related to the estimation model:

- Does the paper present any estimation model that was built based on this set of influence factors? (0.5 pts)
- Is there a description of how this model was built? (0.5 pts)
- Does the paper present an empirical study that evaluates the applicability of such model? (0.5 pts)

Criteria related to the quality of the empirical evidence:

- Does the empirical study (if any) present references to the definitions and guidelines, that are present in the Software Engineering technical literature, in order to justify the empirical design adopted? (0.5 pts)

- Is the empirical study described in an adequate degree of detail, therefore, allowing the comprehension of how the results were obtained, and the identification of its limitations? (0.5 pts)
- Are the study's validity threats identified? (0.5 pts)

Studies' Selection Process

1st Stage:

In the first stage, a researcher performed the searches, organizing the initial list of references.

Table 2-2 shows the number of papers retrieved per search engine, and the total number of unique references.

Table 2-2 - Number of Papers Retrieved per Search Engine

Search Engine	Papers Retrieved
Scopus	145
EI Compendex	162
IEEE Xplore	58
Total	365
Duplicates	134
Total Unique	**231**

We identified a total of 29 duplicates between Scopus and IEEE Xplore result sets, 86 duplicates between Scopus and EI Compendex and 30 duplicates between IEEE Xplore and EI Compendex.

Then the same researcher performed the initial papers' classification by reading their title and abstract and applying the previously defined inclusion and exclusion criteria. The papers were marked as: I – Included, E – Excluded or D – Doubt. Afterwards, a second researcher went through all the papers, analyzed the list and assigned his own classification (I2, E2 or D2) to them.

Finally, the following criterion was applied: (a) papers rated "I" by both researchers are directly included; (b) papers rated "E" by both researchers, or "E" and "D" are directly excluded; (c) papers rated "I" and "E", or "I" and "D" should be discussed; and (d) papers rated "D" by both researchers after the discussion should be included for reading.

After applying the process described above, 18 papers were included.

2nd Stage:

In the second stage, both researchers read the full text of the papers included. Inclusion and exclusion criteria were reapplied. It resulted in only seven papers being selected for inclusion: ZOWGHI and NURMULIANI 2002, MAO et al. 2005, DIBBERN et al. 2007, MCGEE and GREER 2009, AMBRÓSIO et al. 2010, BJARNASON et al. 2011, FERNÁNDEZ et al. 2011.

3rd Stage:

The researchers evaluated the papers that were retrieved in the 2nd stage by using the previously defined quality criteria. The results from this quality appraisal are shown in Figure 2-2.

Supplementing the Dataset

As only seven papers were included in the review after the selection process, we applied backward and forward snowballing, i.e., the process of identifying references and citations related to any of the papers included in the systematic review (JALALI and WOHLIN 2012) to see if we could find more relevant work. We analyzed all the references and citations of each included paper, using the same three databases as before. This analysis covered a single depth level, i.e., we did not analyze the references of references, nor the citations of citations. A total of 300 references and 28 citations were analyzed. After applying the same selection process described above (stages 1 to 3) three new papers were selected for inclusion:

- NURMULIANI *et al.* 2004 was included by the analysis of the references from MAO et al. 2005.
- EBERT and DE MAN 2005 was included by the analysis of the references from AMBRÓSIO et al. 2010.
- MCGEE and GREER 2011 was included by the analysis of the citations of MCGEE and GREER 2009.

Quality Appraisal

Figure 2-2 shows the quality appraisal of each paper selected according to the criteria defined in our *quasi*-systematic literature review protocol. Ratings are presented as a percentage of the maximum score to give an idea on the quality of papers, as regards the perspective defined for this research protocol.

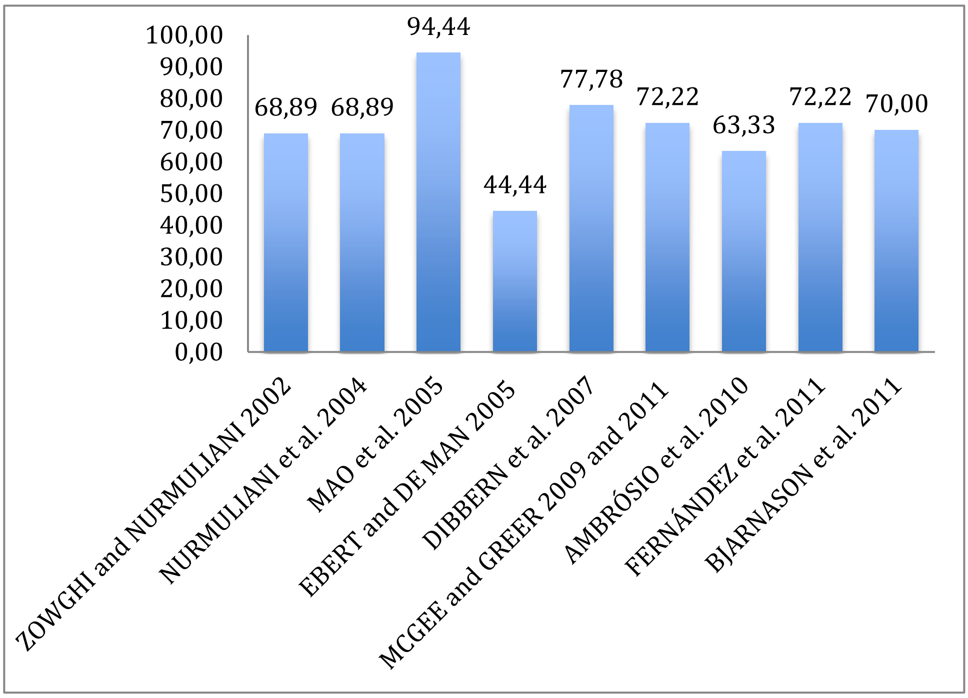


Figure 2-2 - Quality appraisal results of selected papers

The quality appraisal process is important in any systematic review. It is a common part of the research protocol. It gives the researcher the chance to observe the possible ‘strength’ of a paper’s results in the context of the investigation. Please note that Figure 2-2 papers MCGEE and GREER 2009 and MCGEE and GREER 2011 were considered together for the quality appraisal. This is so because these papers covered exactly the same topic (in MCGEE and GREER 2009 the authors propose a taxonomy and in MCGEE and GREER 2011 they present a case study to evaluate it).

None of the papers were removed from the result set after the quality appraisal.

Data Extraction

For each paper approved in the full selection process, we extracted the following information (Table 2-3):

Table 2-3 - Information to be Extracted

Field	Description
Title	Title of the paper
Authors	List of authors, including emails and affiliations.
Year of Publication	Year of Paper Publication
Publication Source	Name of the journal, conference or place where the paper was published.
Abstract	Abstract of the paper.
Influence Factors regarding	List of factors identified.

requirements development activities	
How to measure or observe these factors	Description of how to measure or observe these factors.
Feature(s) influenced by the Factor(s)	Over which feature(s) (e.g. effort, cost, or schedule) of the requirements activities are these factors exerting direct or indirect influence.
Exerted influence	Description of the influence exerted by these factors over the pointed feature.
Models in which these factors were applied	Description of the model in which these influence factors were applied.

The basic information (Title, Authors, Year of Publication, Publication Source, and Abstract) of each paper included paper is provided in Appendix 1.

Table 2-4 shows a short summary of each one of them. The other extracted information is presented below, explaining how it relates to each of our research questions.

Table 2-4 – Summary of Papers in the Final Result Set

Reference	Summary
ZOWGHI and NURMULIANI 2002	This paper shows an investigation on the factors that contribute to the extent of requirement volatility, and of how this volatility impacts software project schedule and costs. The research model was tested using a cross-section survey of 430 software development companies located across Australia. Results show that requirements volatility has a significant impact on project schedule and cost. It also identifies a set of four project features that have a significant impact on requirements volatility.
NURMULIANI *et al.* 2004	This paper aims to identify and characterize requirements volatility problems and its underlying causes during the system development life cycle. An industrial case study was carried out at an ISO 9001 certified software development company that belongs to an international multi-site organization with headquarters and marketing divisions in the US. The analysis considered three main sources of evidence to perform causal analysis of requirements volatility: Change Request (CR) forms, other release documents (i.e., the requirements specification document, configuration management plan, and software product documents), and interview data. Qualitative methods were employed to analyze the data collected and to evaluate the change process. As a result, three root causes for requirements volatility were identified.
MAO et al. 2005	In this paper, MAO et. al. present an artificial neural network model for predicting the effort spent on requirements changes during the

	later stages of software development (design, coding, testing, and maintenance). It uses a set of 23 factors (representing some project characteristics) as inputs, and the output is the effort in person-days needed to tackle the requirements changes. The model was validated though a survey with data from six academic software projects.
EBERT and DE MAN 2005	This paper shows an investigation of influence factors in requirements uncertainties. A field study was conducted inside the Alcatel business group, using data from 246 projects. The projects are representative of Alcatel's overall business in such heterogeneous market segments such as mobile and landline communication networks, mobile handsets, landline and mobile solutions, enterprise communication or transport and satellite infrastructure. The investigation was conducted both through interviews and through an analysis of the reports on the lessons learned on the projects. As a result, four requirements uncertainty root causes were identified.
DIBBERN et al. 2007	This paper shows an investigation on the extra costs involved in outsourcing software projects development to offshore companies. The initial hypothesis about these extra costs is explored using a multiple case study design including six software projects of a large German financial service institution that were offshored to vendors in India. Results highlight the main challenges produced in offshore-outsourced projects, and how these challenges affect the cost of requirements specification, design, knowledge transfer, and team control and coordination.
MCGEE and GREER 2009	This paper shows a software requirements change source taxonomy that highlights the uncertainties or risk factors that may trigger requirements changes. This taxonomy was developed through a collaborative study with a multinational software development company with offices located in the UK and Ireland. This study comprised a set of interviews and workshops used to elicit specialists' knowledge. The resulting taxonomy comprises a set of 32 risk factors that drive the requirements change effort.
AMBRÓSIO et al. 2010	This paper shows a system dynamics model constructed to make it possible for users to better understand the relations among key decision variables in requirements activities. The model was proposed based on a literature review, and it was evaluated using data both from the literature and from a software development company.
FERNÁNDEZ et al. 2011	This paper shows an investigation about the characteristics and strategies used in successful project environments, to propose an approach for tailoring Requirements Engineering processes. As part of this investigation, the authors performed a field study on a set of real software development projects, in which they analyze which RE artefacts were produced in each project and to what extent they were produced. Secondly, they performed a qualitative analysis of

	semi-structured interviews to discover project parameters that relate to the artefacts produced. As a result, they present a set of 31 project parameters and describe how these parameters affect the need for certain specification artefacts and their degree of completeness. As there is an effort involved in producing each artefact, we may consider that these relations also show how project parameters affect the requirements specification effort.
BJARNASON et al. 2011	This paper shows an investigation on the causes and effects of communication gaps in software projects. This investigation was made through a semi-structured interview of nine practitioners at a large market-driven software company. Results include a set of four main root causes of communication gaps, and a discussion on how these gaps affect the software project and, in particular, the requirements activities.
MCGEE and GREER 2011	This paper is a follow-up of the research presented in MCGEE and GREER (2009). It shows how the proposed chance source taxonomy was assessed through a case study where change data was recorded over a 16-month period covering the development lifecycle of a government sector software application. As a result, an extended taxonomy is presented, containing 42 risk factors that drive the requirements change effort.

2.3 Results – Answering the Research Questions

To answer our main research question “Which factors influence the effort, cost, time or schedule of the requirements development activities?” we extracted from each paper: (1) The list of the identified factors (sometimes grouped in categories, sometimes not) and (2) Over which feature(s) (e.g., effort, cost, or schedule) of the requirements activities are these factors exerting direct or indirect influence.

Table 2-5 shows the results of this extraction.

Table 2-5 – Data extracted from each paper to answer our main research question

Reference	Influence Factor Category	Influence Factor	Affected Characteristics
ZOWGHI and NURMULIANI 2002	Project characteristics & practices	Usage of RE methods (1=yes, 0=No).	Requirements Volatility
		Perform Requirements Inspection (1=yes, 0=no).	
		User representatives (Logarithm).	
		Frequency communication (Logarithm).	
	Potential for Change	The changes in business environment for this project were high.	
		The analyst's knowledge of the business environment was excellent.	
	Requirements Instability	Requirements fluctuated in the earlier stages.	
		Requirements fluctuated in the later stages.	
		Difference in requirements identified at the start of the project from the final requirements.	
	Requirements Diversity	It was difficult for stakeholders to reach agreement among themselves on requirements.	
		A lot of effort had to be spent in incorporating the requirements of various users.	

		It was difficult to customize software to one set of users without reducing support for other users.	
NURMULIANI *et al.* 2004	Requirements Volatility Root Causes	Changes in market demands.	Requirements Volatility
		Developers' increased understanding of product domain.	
		(Change in) Organizational considerations.	
MAO et al. 2005	Project Character	The number of sub-systems (or modules).	Requirements Change Effort
		Total number of developers.	
		Project budget.	
		Complexity of techniques and its risk.	
		Reusable resource.	
		Sufficiency of requirements analysis	
	Developers	Developers' experiences.	
		Familiarity with development approaches and tools.	
		Familiarity with the business knowledge.	
		The training for development.	
		Degree of communication.	
		Internal communication and cooperation.	
	Users	Working experience.	

		Definiteness of users' requirements goal.	
		Internal negotiation and consistency.	
		Enthusiasm in the participation.	
		Mastery of computer knowledge.	
		Satisfaction at the initial specification.	
	Project Management	Sufficiency of requirements inspection.	
		Configuration of requirements tools.	
		Formalization of requirements documents.	
		Configuration management for RC	
		Rationality of organization and management.	
EBERT and DE MAN 2005	Requirements Uncertainty Root Causes	Vague product vision and strategy.	Uncertainty (risk)
		Key stakeholders are not involved.	
		Unknown project dependencies.	
		Business case not thoroughly evaluated.	
		Requirements not sufficiently specified and analyzed.	

	Early Project Symptoms	Conflict of interest; commitments not maintained.	
		Unexpected dependencies between requirements.	
		Cost/benefit of individual requirements unclear.	
		Incomplete requirements.	
DIBBERN et al. 2007		Required Client-specific Knowledge	Knowledge transfer & Specification costs
	Vendor-related Characteristics	Lack of Absorptive Capacity.	Knowledge transfer costs
		Personnel turnover.	
	Offshore-Specific Client-Vendor Distance	Geographic distance.	
		Cultural Distance.	Specification effort
AMBRÓSIO et al. 2010	Risks	Percentage of requirements in a software release that are not expected before starting the specification.	Requirements stage cost (man-days spent) and quality (requirements errors %)
		Percentage of requirements change.	
		Percentage of change in requirements delivered in previous releases.	
		Percentage of errors made in requirements activities.	
	Managerial decisions and policies	Attenuation for errors made during errors fixing and requirements change.	
		Turnover.	

		Time for turnover to occur.	
		Initial size of working team responsible for the specification.	
		Percentage of working team’s effort to be allocated to quality assurance.	
		Time used for changing requirements.	
		Time used for increasing schedule.	
		Lower bound of the number professionals in the team.	
		Percentage of extra effort needed, as provided by contracting extra effort from the team.	
		Time used for fixing requirements errors.	
		Time used for decreasing schedule.	
		Number of extra professionals for overstaffing workforce.	
		Maximum increase in effort by team members when there is increase time pressure and a risk of schedule overrun.	
		Time used for contracting extra effort for the team.	
		Time used for increasing team size.	
		Initial schedule.	
		Upper bound for the number of beginners that an experienced professional can train.	
		Time used to have extra effort from the team.	
		Time used to decrease team size	

		Number of Requirements specified and delivered in a software release.	
		Average productivity of professionals in the team.	
FERNÁNDEZ et al. 2011	Customers' Domain	Governmental customer.	Artefacts completeness
		Weak access to business processes.	
		Weak relationship with customers.	
		Good relationship with customers.	
		Weak knowledge of customers' domain.	
		Unavailability of Stakeholders.	
		Weak technical ability of stakeholders.	
		Unreliability of stakeholders.	
	System under construction	High degree of user interaction.	
		High degree of innovation.	
		Emphasis on data flow.	
		Emphasis on control flow.	
		High degree of distribution.	
		Complex dependencies on external systems.	
		Weak knowledge of operative environment.	
		Weak knowledge of operative background.	
		Custom software.	
		Standard software.	
	Cross-cutting Process Aspects	Time-boxing.	
		Existence of external parties.	
		External acceptance tests.	
		High number of requirements.	
		Long project duration.	
		Estimates of functional complexity.	
		Weak documentation provided.	

		Change management established.	
		Standardized design process.	
		Large team size.	
		High team distribution.	
		Weakly experienced team.	
MCGEE and GREER 2009 MCGEE and GREER 2011	Market	Market Stability.	Requirements Volatility
		Differing Customer Needs.	
		Presence of Competitor.	
	Customer Organization	Stability of Customer's Business Environment.	
	Project Vision	All stakeholders involved.	
		Clarity of Shared Product Vision.	
		Unknown customer project dependencies.	
		All stakeholders identified.	
		Semantic Relativism.	
		Degree of change to customers work practice.	
		Involved customer understanding of the problem.	
		Customer Experience in IT.	
		Incorrect stakeholder involved.	
		COTS/framework usage.	
		Logical complexity of problem.	
		Project size.	
		Novelty of application.	
		Synergy of Stakeholder Agenda.	
		Development team knowledge of business area.	
		Analyst skill and experience.	
	Requirements Specification	Communication available with customer/stakeholder.	
		Insufficient sample of user representatives.	

		Quality of communication between Analyst/Customer/Stakeholder.	
		Analysis techniques.	
		Degree of change of customers work practice.	
		Development team knowledge of business area.	
		Analyst skill experience.	
		Development team stability.	
		Quality of development team communication.	
		Low Staff Morale.	
		Incompatible requirements.	
		Logical complexity of problem.	
		Involved customers' knowledge/understanding of the problem.	
		Involved customers' experience in IT.	
		Incorrect user involved.	
		Age of requirements.	
	Solution	Technical uncertainty of solution.	
		Technical complexity of solution.	
		COTS/framework usage.	
BJARNASON et al. 2011		Complex Product and Large Organization.	Communication gaps
		Low understanding of each other's roles.	
		Gaps between roles over time.	
		Weak vision of overall goal.	

This table represents a 'raw' version of the list of influence factors, where factors are presented exactly as they appear in the corresponding paper. As it is possible to see, there are many similarities between factors identified in each paper, for example: MAO et al. 2005 mention the factor "Familiarity with the business knowledge" while FERNÁNDEZ et al. 2011 mention "Weak knowledge of customers' domain". Both factors deal with the aspect of team's knowledge about the problem domain. Therefore, we still needed to interpret and combine these results to get a broader view of the influencing factors and their relations to the requirements effort. This process of analyses and synthesis and the results obtained will be described in the next chapter.

Secondary Question: How can these factors be measured or observed?

Most of the papers from our result set do not give a clear indication of how these factors can be measured.

In ZOWGHI and NURMULIANI 2002, the factors in requirements volatility are described as a statement (for instance: 'It was difficult to customize software to one set of users without reducing support for other users') and a project member should give a score to each statement using a five-point scale: 1=strongly disagree, 3=neutral, and 5=strongly agree.

NURMULIANI et al. 2004 do not mention how the causes of requirements volatility could be measured or observed. In EBERT and DE MAN 2005, there is no indication either.

In MAO et al. 2005, each factor is given a difficulty coefficient ranging from 0 to 1. The valuation of this coefficient can be done by project's developers, administrators or experts in Software Engineering.

In MCGEE and GREER 2009 and MCGEE and GREER 2011, the authors do not give an indication of how the uncertainties that may trigger requirements change could be measured.

In AMBRÓSIO et al. 2010, the user running the simulation model assigns (subjectively) a score to each factor, ranging from 0 to 1.

Finally, FERNÁNDEZ et al. 2011 describe how the factor "artefact completeness" was measured: a project member rates each artefact in the following scale: "(1) Completely specified: The content item can be mapped unambiguously onto a specific element of the analysed documents.; (2) Incompletely specified: The content item can be identified in the analysed documents but displays major deficiencies regarding the defined completeness criteria.; (3) Missing: The content item cannot be identified in the documents analysed”. However, the authors do not mention how to measure the project parameters that influence the artefact completeness.

All in all, we were unable to achieve a satisfactory understanding of how the factors show be measured. Therefore, this research question requires further investigation.

Tertiary Question: Which estimation models use these factors?

The only paper that presented an effort estimation model using the influence factors identified is MAO et al. 2005. This model consists of an artificial neural network that receives the influence factors as inputs, and returns the effort in person-days needed to tackle the requirements changes during the later stages of software development as an output. The model does not deal with the effort needed to build the initial software requirements specification and, to that end, it only partially solves our problem of understanding how to predict the total effort involved in requirements activities.

2.4 Chapter Conclusion and Next Steps

This chapter described the review process used to characterize the current state-of-the-art regarding the research on requirements effort estimation. The extraction of pertinent data from each paper included in this *quasi*-systematic literature review resulted in a large amount of qualitative data that still needed to be analyzed, combined and synthesized. To support this process we applied the principles of Grounded Theory, a qualitative analysis methodology, described in the next chapter.

CHAPTER 3 - SYSTEMATIC LITERATURE REVIEW - RESULT ANALYSIS AND SYNTHESIS

This chapter describes how the results obtained from the quasi-systematic literature review were analyzed and synthesized, applying the principles of Grounded Theory, a qualitative analysis method.

The previous chapter described the execution of a quasi-systematic review that allowed the retrieval of ten papers considered relevant to the research question. After extracting the information from each included paper, a total of 150 requirements effort influence factors were identified. We needed to analyze these factors to understand what they mean, how they can be grouped (when the same factor was identified by more than one paper), and how they relate to the software requirements specification effort. In this analysis we needed to deal with a big amount of information, and use a great deal of interpretation of this qualitative data. To assist in this complex task, we applied the principles of Grounded Theory.

3.1 Grounded Theory

The Grounded Theory methodology (and the analysis techniques proposed by it) is a viable way for conducting qualitative research, whether with the intention of generating a substantive theory or just to make a conceptual assortment (BANDEIRA-DE-MELLO and CUNHA 2003). The essence of the Grounded Theory method is that it emerges through the systematic analysis of the data (BANDEIRA-DE-MELLO and CUNHA 2003). This analysis should be done according to a set of techniques and procedures that aim at mitigating the bias of the researcher in the interpretative process.

Following the principles defined by STRAUSS and CORBIN (1998), this analysis is based on the coding process. This process is divided into three stages: open coding, axial coding, and selective coding. Open coding involves the breakdown, analysis, comparison, conceptualization, and categorization of the data (BANDEIRA-DE-MELLO and CUNHA 2003). The researchers should explore the data through intensive text reading, identifying what they deem relevant to the research question. These incidents are then grouped into codes (representing a concept or theory construct) via the incident–incident comparison. It means that all incidents or events should be classified and grouped according to their meaning. Also in the open coding stage, categories are created to cluster the codes and reduce the number of units the researcher will work with (CONTE et al 2010).

After the identification of the codes and their conceptual categories, the axial coding examines the relations between the categories that form the propositions of the substantive theory (CONTE et al 2010). These propositions should be again tested against data.

Finally, the selective coding refines the entire process of identifying the core theory category to which all the others are related. The core category should be able to integrate all the other coding and express the essence of the social process that takes place between those involved (BANDEIRA-DE-MELLO and CUNHA 2003).

All the coding process was conducted using the Atlas TI tool[6]. This tool supports the analysis by linking every code to the excerpts it is grounded on, allowing an easy retrospect of the open coding stage. It also supports axial coding by allowing the researcher to build networks representing the relation between codes. The sections below describe in detail the application of Grounded Theory principles in the context of our research, and the results are presented in the format and notations provided by the Atlas TI tool.

3.1.1 Open Coding

The first step in the open coding stage was to highlight, in each paper, all the excerpts describing a requirements effort influence factor. This excerpt could be a single phrase with the factor name or a long paragraph detailing the factor, depending on the paper. Next, we proposed a set of codes to represent these factors. Figure 3-1 shows an example of the creation of a new code, based on an excerpt from DIBBERN et al. 2007.

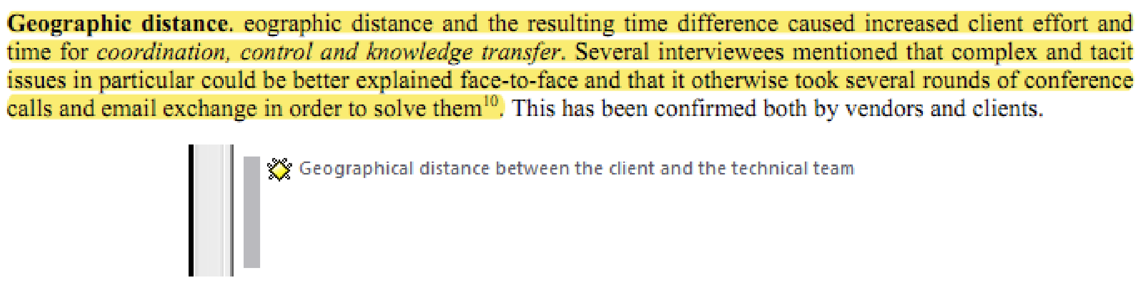

Geographic distance. eographic distance and the resulting time difference caused increased client effort and time for *coordination, control and knowledge transfer*. Several interviewees mentioned that complex and tacit issues in particular could be better explained face-to-face and that it otherwise took several rounds of conference calls and email exchange in order to solve them[10]. This has been confirmed both by vendors and clients.

Figure 3-1 – Open coding example

[6] http://www.atlasti.com/

Whenever two or more papers seemed to talk about the same factor (although using different words and levels of abstraction) we tried to blend them into a single code that could represent both excerpts. This is the process described by the methodology as the incident-incident comparison. An example of such process is shown in Figure 3-2.

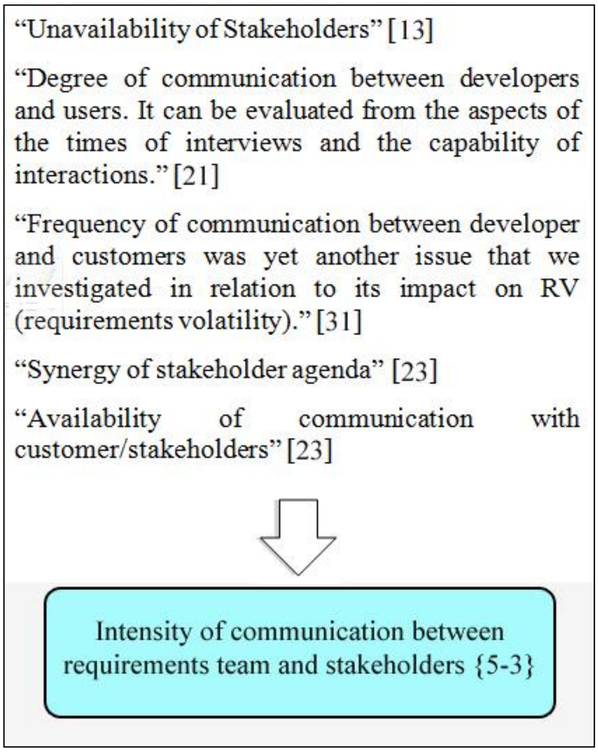


Figure 3-2 – Open coding example

When combining excerpts from various papers we were particularly careful with the terminology. We had to represent all the information contained in each paper about this influence factor in a single code. The use of a dictionary was often necessary to make sure we were using the appropriate terminology. This was a really delicate and time-consuming task accomplished by two researchers working together, to reduce the chance of bias or misinterpretation in the definition of each code.

The final step in the open coding stage is to group the codes into categories. Categories are clusters of concepts joined in a higher degree of abstraction. They are useful to reduce the number of units the researcher will work with and to make the results easier to visualize. As BANKS et al 2000 (*apud* Strauss and Corbin 1990, p. 67) note, categories have to be analytically developed by the researcher, which means they are not extracted explicitly from the papers, but rather proposed by the researcher as a result of one's understanding of the codes. Table 3-1 shows the categories identified in the scope of this work.

Table 3-1 - Categories defined during the Open Coding stage

Category	Description
Solution Domain	characteristics of the solution to be built
Problem Domain	characteristics of the problem domain in which the system is immersed
Environment for Requirements Engineering	characteristics of the work environment used by the Requirements Engineering team
Technical Team	characteristics of the Requirements Engineering team
Stakeholders	characteristics of the project's stakeholders **(not including requirements team)**
Interaction between Stakeholders and Technical Team	characteristics of how the Requirements Engineering team interact with project's stakeholders
Project	the managerial characteristics of the project, such as its budget and schedule restrictions

3.1.2 Axial Coding

The next step was to draw the relations between codes. The papers were re-analyzed by focusing on finding descriptions of how the identified concepts interact. We only represented the relations that were directly extracted from the papers. This means that the arrows (shown in Figure 3-4) are also grounded on data, although the tool used to represent the results does not support explicit linking between an arrow and the excerpt of text where it was grounded on.

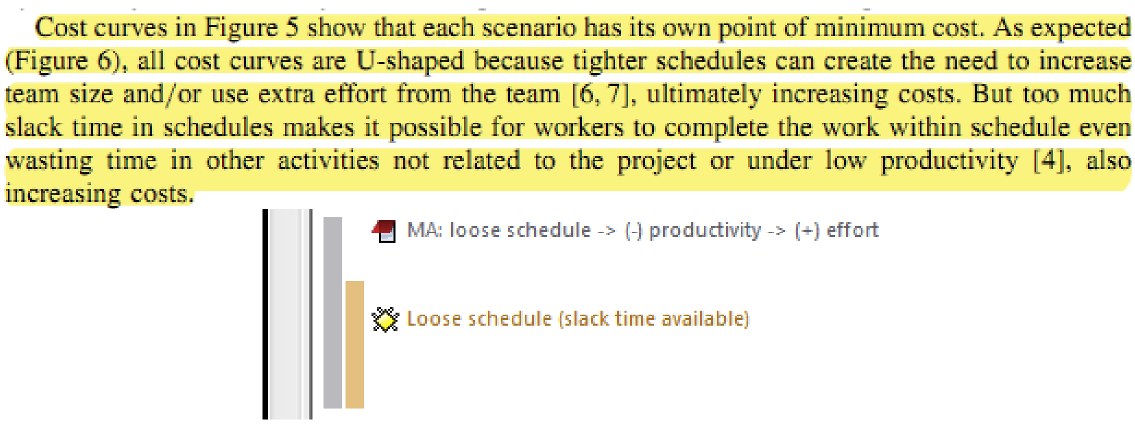


Figure 3-3 - Example of how relations were highlighted

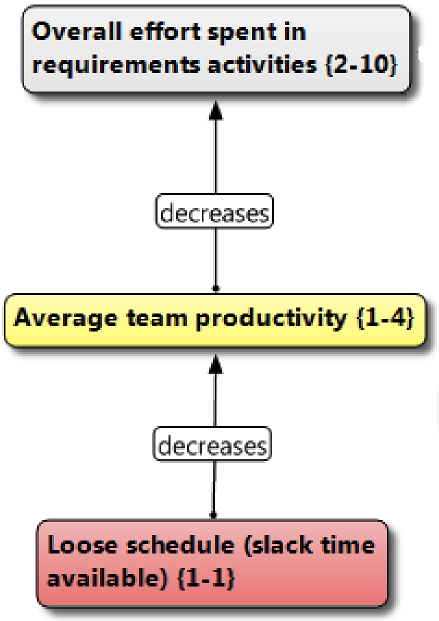


Figure 3-4 - Example of the representation of relations between codes

Note that each code has two associated numbers {*groundedness, density*}. *Groundedness* represents how many citations are related to the code, and *density* represents how many codes it relates to.

The types of relations between codes – represented by the labels over the arrows – can be defined by the very researcher (GLASER 1992). Table 3-2 describes the types of relations used in this work.

Table 3-2 - Types of Relations

Label	Description
Increases	This type of relation was used to represent a direct influence on the other factor, which means that if the factor in the origin of the arrow increases, the other factor will also tend to increase.
Decreases	This type of relation was used to represent an inverse influence on the other factor, which means that if the factor in the origin of the arrow increases, the other factor tends to decrease.
Is part of	This type of relation was used to represent that one factor is a subset of the other factor, e.g., the effort spent in requirements specification *is a part of* the overall effort spent in the requirements activities.

3.1.3 Selective Coding

Finally, in the selective coding stage, the theory is refined by identifying the core category to which all the others are related to. This core category should be able to express the essence of what the research is about. It can be an existing category or a new one (BANDEIRA-DE-MELLO and CUNHA 2003). In this case, we defined a new category representing the effects of all other categories into the requirements activities. Such effects are, for instance, the amount of effort to be spent, the risk of requirements change and the quality of the requirements specification.

3.2 Results

A macro view of the results obtained can be seen in Figure 3-5, the main category representing the effects shown at the center. All other categories relate to it.

Figure 3-5 - Macro view of results

Each of these categories can be seen in finer detail, presenting all the codes contained in the category, how these codes relate to each other and how they relate to the codes in the main category (Figs. 3-4 to 3-10).

The codes from each category are shown in different colours (following the ones used in Figure 3-5), to make results easier to visualize. Furthermore, increasing relationships are shown in red, decreasing relations are shown in blue and "is part of" relations are shown in black, facilitating the distinction of each line.

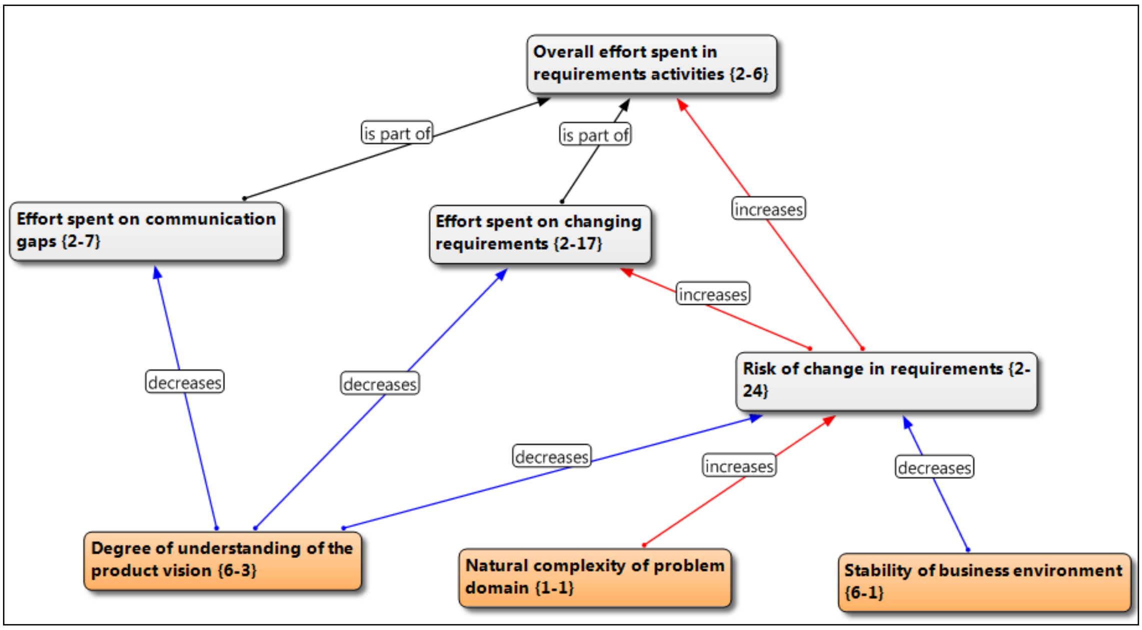


Figure 3-6 - The Problem Domain category

Figure 3-6, shows the Problem Domain category in detail. It has the codes "Stability of business environment", "Natural complexity of problem domain" and "Degree of understanding of the product vision". These codes relate to codes of the main category: "Effort spent on communication gaps", "Effort spent on changing requirements", "Risk of requirements changes" and "Overall effort spent in requirements activities". The arrows represent how each code relates to others. For example: the "Degree of understanding of the product vision" reduces the "Effort spent on communication gaps", which means that a better understanding of the product vision leads to less effort wasted with miscommunication. It also decreases the "Risk of requirements changes" and, therefore, the "Effort spent on changing requirements". The "Stability of business environment" also decreases the "Risk of requirements changes".

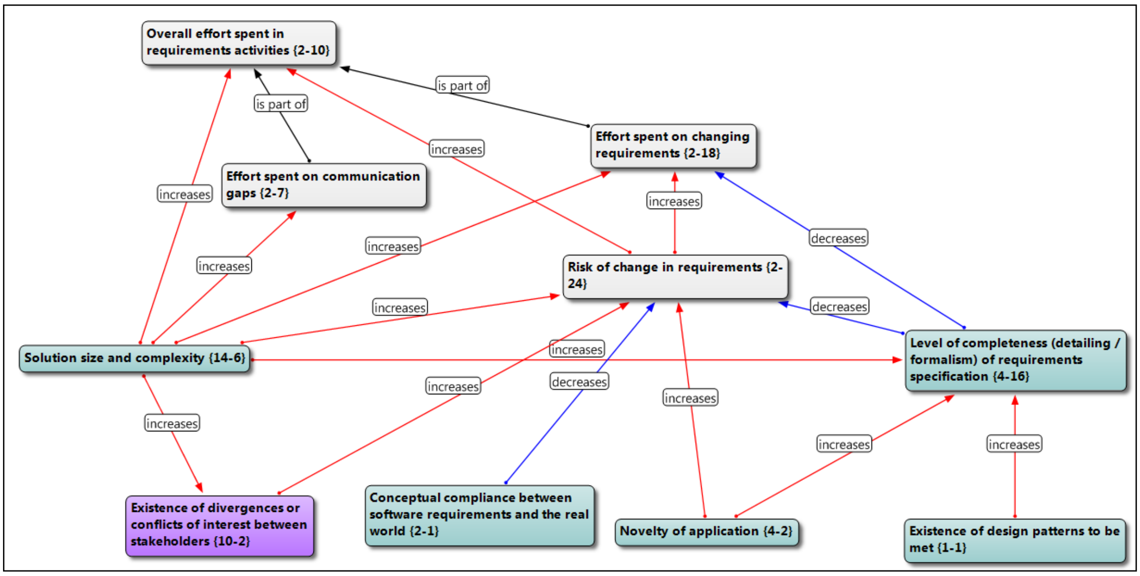


Figure 3-7 - The Solution Domain category

Figure 3-7 shows the Solution Domain category in detail. In this case, we can see that one code of the Solution Domain is related not only with the codes of the main category, but also with a code from the Stakeholders category named "Existence of divergences or conflicts of interest between stakeholders". This shows that the categories are not independent, but relate to each other. For instance, a larger and the more complex solution ("Solution size and complexity") tends to increase the "Existence of divergences or conflicts of interest between stakeholders".

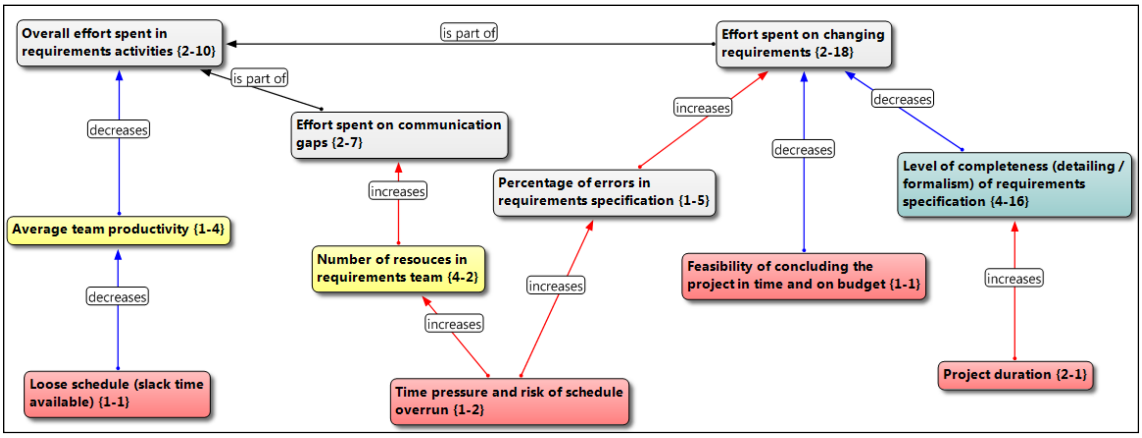


Figure 3-8 - The Project category

The Project category (Figure 3-8) shows the impact of project characteristics such as its time and budget constraints in the project effort. For instance, a too tight schedule (represented by the code "Time pressure and risk of schedule overrun") increases the "Percentage of errors in requirements specifications", and it can create the need to increase team size. However more team resources increase the risk of communication gaps, ultimately increasing the effort spent in requirements activities. A too loose schedule, on the other hand, allows workers to complete the work within schedule, even wasting time in other activities not related to the project or under low productivity, and also increasing costs (AMBRÓSIO et al 2010). Figure 3-8 also shows that longer project duration increases the need for a more detailed requirements specification, as it is important to record things instead of relying on people's memories and communication skills.

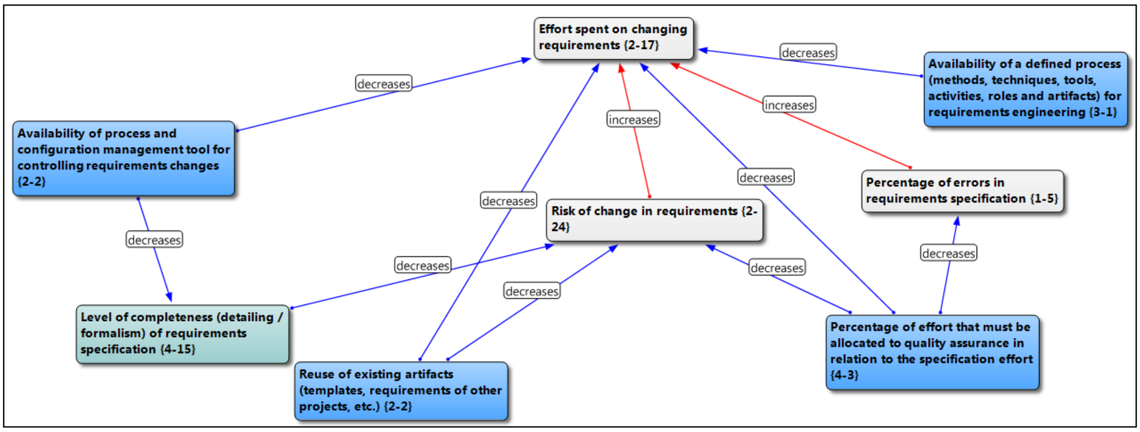


Figure 3-9 - The Environment for the Requirements Engineering category

The Environment for Requirements Engineering Category (Figure 3-9) shows the impact of engineering practices and processes in the project's effort and risks. The "Reuse of existing artefacts" reduces the "Risk of requirements changes" and, therefore, the "Effort spent on changing requirements", based on the assumption that reusable resources are already validated, and so reduce the risk of rework. It also shows that a larger amount of effort dedicated to quality assurance ("Percentage of effort that should be allocated to quality assurance in relation to the specification effort") decreases the "Percentage of errors in requirements specifications" which in its turn reduces the effort spent on rework.

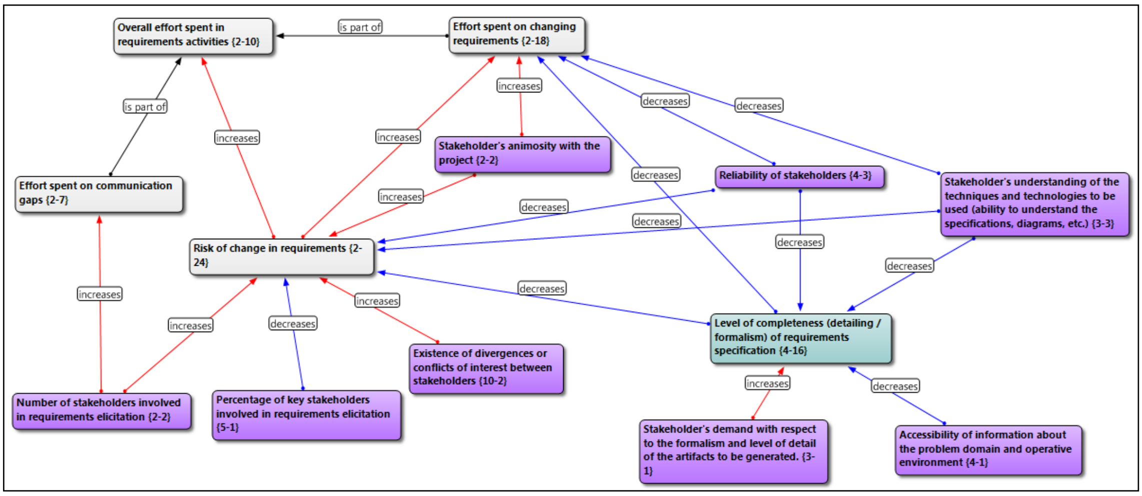


Figure 3-10 - The Stakeholders category

The Stakeholders category (Figure 3-10) shows that the greater the number of stakeholders involved in the elicitation activities, the greater will be the "Effort spent on communication gaps", ultimately increasing the effort. On the other hand, not involving all key stakeholders on the eliciting process increases the risk of having undiscovered requirements, thus increasing the risk of rework. Other factors such as stakeholder's unreliability, animosity with the project and little understanding of requirements specifications, also increases the effort needed to carry out requirements activities. When stakeholders lack the understanding of requirements specification notations and techniques, it might be necessary to produce additional material make the specifications easier to understand (for example, building more detailed and realistic prototypes), increasing the level of completeness of requirements specifications. On the other hand, the existence of accessible information on the problem domain tends to reduce the need for detailed specifications, as some documentation on the business processes and operative environment might already be available.

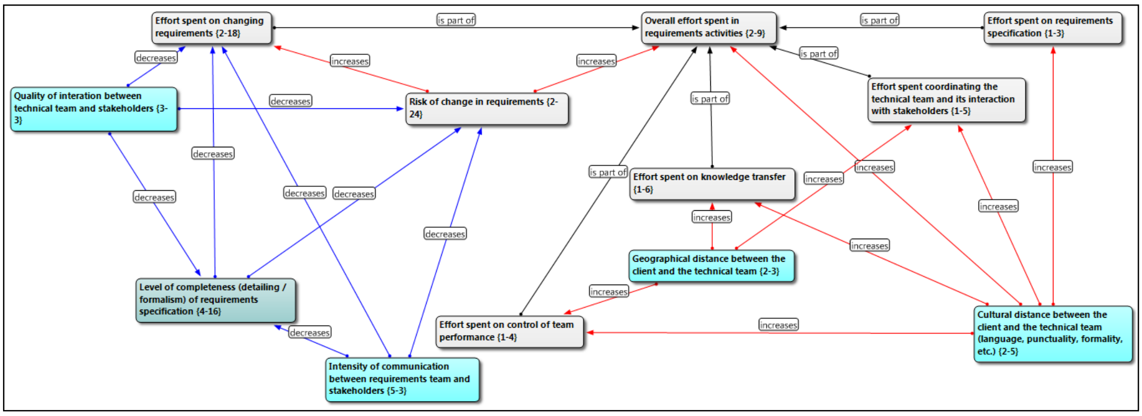


Figure 3-11 - The Interaction between Stakeholders and Technical Team category

Figure 3-11 shows the Interaction between Stakeholders and Technical Team category, with the impact of communication quality, intensity and other communication challenges – such as geographic and cultural distance between stakeholders and the technical team – in the requirements effort.

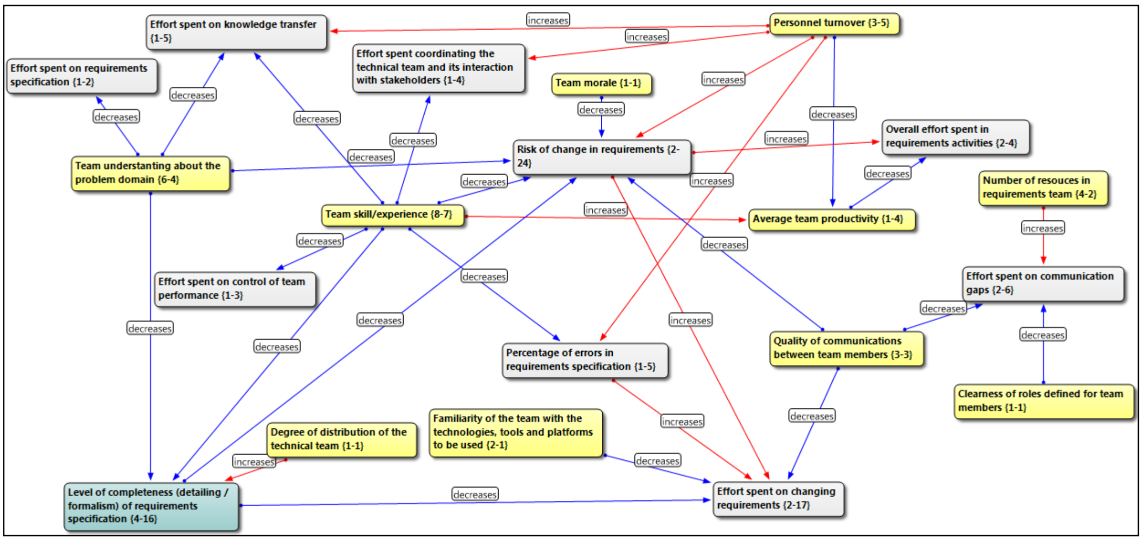


Figure 3-12 - The Technical Team category

Figure 3-12 shows the Technical Team Category, with the impact of team characteristics such as team skills, stability and distribution on requirements error frequency, risk of change and effort. This category also shows an interaction with the Solution Domain category. The geographical distance between team members increases the need for more detailed specifications, to make up for the difficulties in communication in distributed teams. On the other hand, the "Team understanding of the problem domain" may decrease the required level of detail in specifications, as team members are already familiar with the processes and terminologies of the problem domain.

Note that all the effort nodes in this diagram should appear as a “is part of” relation with the “Overall effort spent in requirements activities”, although these relations were omitted from Figure 3-12 to make the diagram easier to visualize.

3.3 Threats to validity

The main threat to the validity of this study is the bias the researcher may introduce when interpreting the content of each paper. To mitigate this threat we invited a second researcher to revisit all the coding process. Also, whenever the meaning of an excerpt from an article was not very clear, we emailed the authors asking for further clarifications, to avoid any misinterpretation. Three out of four emailed authors responded to our questions, and their answers were used to refine or correct our interpretations.

These results represent a complete view of all the influence factors on requirements effort, as found in the technical literature, with no separation of the context in which these factors were observed. This is justified as requirements effort estimation is a relatively unexplored subject. Therefore the goal of this research was not related to mapping the relevant factors in each context, but rather to identify all possible factors about which there is some evidence of their impact on the requirements effort.

Other threats to validity include: (1) Key words used to retrieve the related work might be too restrictive or inadequate. To tackle this threat we selected three control papers to evaluate the search string. The search string applied returned all three of them. (2) The systematic review was limited to three search engines. (3) The inclusion and exclusion criteria may be too restrictive, excluding relevant work from the review; and finally (4) the selected publications might present inaccurate or incorrect results, something that was tackled by applying a quality evaluation to each paper that was included in the review.

3.4 Chapter Conclusion and Next Steps

This chapter describes the analysis of the results of the quasi-systematic review described in the previous chapter. By applying the principles of the Grounded Theory methodology we were able to accomplish a thorough analysis of the current work regarding requirements effort influence factors. The conceptual maps built from this analysis synthesize the results and make the information easier to visualize. We still wanted, however, to assess the appropriateness of these results by allowing practitioners and other researchers to contribute. This was done with the execution of a large-scale survey that will be described in the next chapter.

CHAPTER 4 - EVALUATION AND REFINEMENT OF RESULTS

This chapter describes how the results obtained from the quasi-systematic literature review were assessed and refined through the execution of a large-scale survey.

In order to assess the adequacy of the results obtained from the quasi-systematic literature review and to be able to extend and refine these results, we conducted a survey with practitioners and researchers working in the area of Requirements Engineering. This chapter shows this survey protocol, the description of its execution, how data was analyzed and, finally, the complete results of our research on requirements effort influence factors.

4.1 Survey Plan

In this section we present the survey protocol that describes its goal, hypothesis, analysis guidelines, criteria used for subjects selection, and present the instrument used to collect data.

4.1.1 Global Goal

The global goal of the survey is to assess and refine the results obtained from the quasi-systematic literature review described in the previous chapters.

In order to organize the goals of the survey, we broke these results down into three elements: (1) overall effort involved in requirement activities; (2) effort components that form the overall effort (for instance "effort spent on knowledge transfer" and "effort spent on changing requirements"); and (3) influence factors that affect the requirements effort. Figure 4-1 shows an example of this segmentation based on one of the diagrams presented in the previous chapter (from the Problem Domain category):

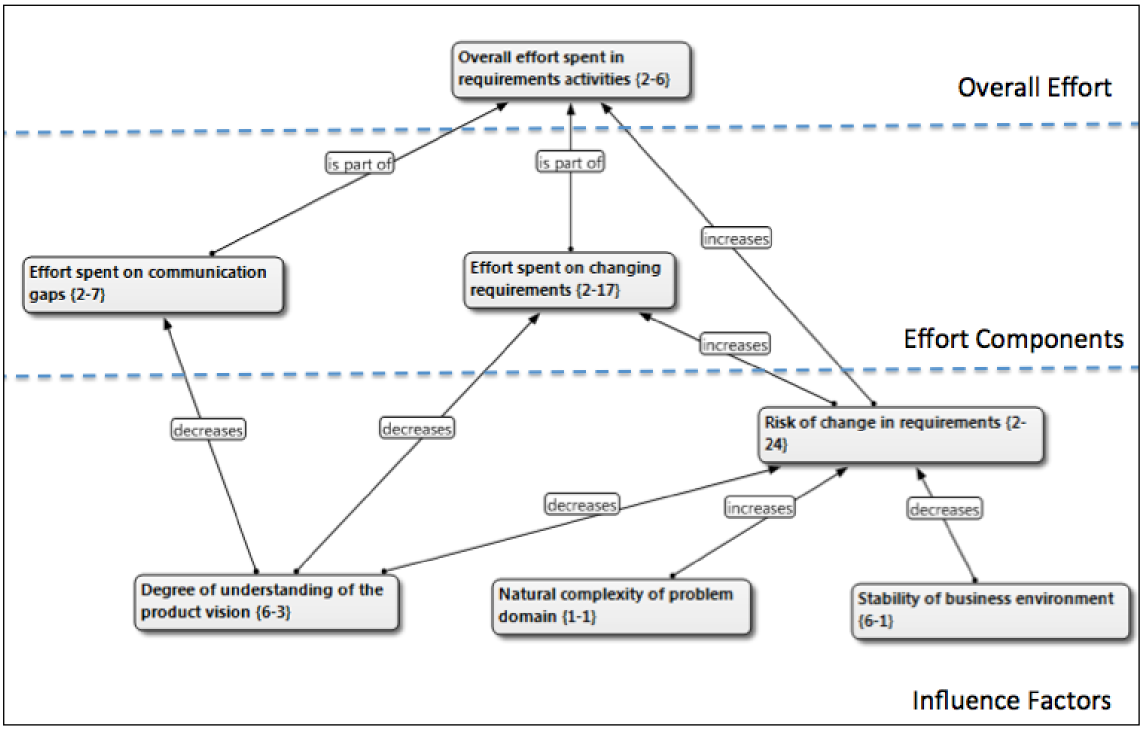


Figure 4-1 - Overall Effort, Effort Components and Influence Factors

Based on this terminology, the goals of this survey can be summarized as:

To Analyze a set of effort components,
With the purpose of characterizing,
Regarding their pertinence to represent the components of the effort involved in requirements activities
From the point of view of practitioners and independent researchers
In the context of their projects covering software requirements specification.

To Analyze a set of influence factors,
With the purpose of characterizing,
Regarding their pertinence to estimate the effort involved in requirements activities
In the point of view of practitioners and independent researchers
In the context of their projects covering software requirements specification.

Study Object: The initial set of components used to represent the components of the effort involved in requirements projects identified in the literature (Appendix 2).

The initial set of factors identified in the literature to support effort estimation in requirements projects (Appendix 3).

4.1.2 Goal of the Measurement

From an initial set of effort components and influencing factors to analyze:

1. Which components included in the initial set are pertinent to represent the effort involved in requirements projects.
2. Which components included in the initial set are not pertinent to represent the components of the effort involved in requirements projects, and must be excluded from the set of components.
3. Which components not included in the initial set are pertinent to represent the components of the effort involved in requirements projects and must be included in the set of components.
4. The set of components pertinent to represent the components of the effort involved in requirements projects.
5. Which factors included in the initial set are pertinent to estimate the requirements effort.
6. Which factors included in the initial set are not pertinent to estimate the requirements effort, and must be excluded from the set of influencing factors.
7. Which factors not included in the initial set are pertinent to estimate the requirements effort and must be included in the set of influencing factors.
8. The set of factors pertinent for estimating the effort involved in requirements projects.
9. How the influencing factors and effort components relate to each other.

4.1.3 Research Questions

Q1: Are the components presented in Appendix 2 pertinent to represent the effort involved in requirements projects?

Metric 1.1: number of components rated as ‘pertinent’ to represent the effort involved in requirements projects, according to the opinion of the subjects.

Q2: Is there any additional component pertinent to represent the effort involved in requirements projects that is not present in the initial set?

Metric 2.1: number of additional components to be included in the initial set according to the opinion of the subjects.

Q3: Is there any component presented in the initial set that is not pertinent to represent the effort involved in requirements projects?

Metric 3.1: number of components to be removed from the initial set according to the opinion of the subjects.

Q4: Are the factors extracted from the technical literature, and presented in Appendix 3, pertinent to estimate requirements effort?

Metric 4.1: number of factors classified as 'pertinent' to estimate requirements effort, according to the opinion of the subjects.

Q5: Is there any additional factors pertinent to estimate the requirements effort that is not present in the initial set?

Metric 5.1: number of additional factors to be included in the initial set according to the opinion of the subjects.

Q6: Is there any factor presented in the initial set that is not pertinent to estimate the requirements effort?

Metric 6.1: number of factors to be removed from the initial set according to the opinion of the subjects.

Q7: How do the influence factors relate to the effort components?

Metric 7.1: number and direction (positive or negative) of relations between each factor and each effort component.

4.1.4 Definition of Hypotheses

Null Hypothesis 1 (H0 1): The initial set of effort components is complete, that is, every component found in the initial set is relevant or pertinent to represent the components of the effort involved in requirements projects, and no components have been included or removed.

V – Initial set of components

Vr – components rated as ‘not pertinent’ that need to be removed from the initial set in Appendix 2

Vi – components not found in the initial set in Appendix 2 and rated as ‘pertinent’ that need to be included in the initial set.

H0 1:|Vr|=|Vi|=0

Alternative Hypothesis (H1): There are components in the initial set that have been rated as ‘not pertinent’ to represent the components of the effort involved in requirements projects. So, they need to be removed from the set of components.

H1: |Vr| ≠ 0

Alternative Hypothesis (H2): There are components not present in the initial set, classified as ‘pertinent’ to represent the components of the effort involved in requirements projects. Therefore, they need to be included in the set of components.

H2: |Vi| ≠ 0

Null Hypothesis 2 (H0 2): The initial set of influencing factors is complete, that is, every factor present in the initial set is relevant or pertinent to estimate requirements effort, and no factors has been neither included nor removed.

F – Initial set of influencing factors

Fr – factors classified as ‘not pertinent’ that need to be removed from the initial set in Appendix 3

Fi – factors not present in the initial set in Appendix 3 and classified as ‘pertinent’ that need to be included in the initial set.

H0 2:|Fr|=|Fi|=0

Alternative Hypothesis (H3): There are factors in the initial set that have been classified as ‘not pertinent’ to estimate requirements effort. So, they need to be removed from the set of influencing factors.

H3: $|Fr| \neq 0$

Alternative Hypothesis (H4): There are factors not present in the initial set, classified as 'pertinent' to estimate requirements effort. Therefore, they need to be included in the set of influencing factors.

H4: $|Fi| \neq 0$

Null Hypothesis 3 (H0 3): The influence factors identified have no bearing on any of the effort components.

i – Influence factor

v – Effort component

Ii,v – The influence that a factor *i* exerts on component *v* can assume the values:

-1 – inverse influence

0 – no influence

+1 – direct influence

H0 3: $\forall i\ \forall v,\ I_{i,v} = 0$

Alternative Hypothesis (H5): There is at least one factor with influence level different from zero in relation to an effort component.

H5: $\exists i,\ \exists v \mid I_{i,v} \neq 0$

4.1.5 Definition of the Instrumentation

The survey will be filled by researchers and practitioners working on or concerned with the estimating of the effort for requirements specifications.

The researchers will indicate four aspects:

1. If the components presented could be pertinent or not to represent the components of the effort involved in requirements projects, and if it is necessary to include or exclude some components from an initial set (Appendix 2).

2. If the influence factors presented could be pertinent or not to estimate the effort involved in developing software requirements, and if it is necessary to include or exclude some factors from an initial set (Appendix 3).
3. After the definition of the final set of effort components, subjects will be asked to choose which effort component one believes to be the most relevant to build the overall requirements effort.
4. After the definition of the final set of influence factors and of the effort components, it is advisable to define whether these factors are related to any of the effort components. Subjects will be asked to link the factors one believes to exert influence on the effort component one pointed out to be the most relevant component. This influence should be marked as positive (+) or negative (-).

Note that the relations between factors and all other effort components were not covered by this survey. We decided to do this to make the questionnaires shorter, as otherwise the survey would take too long to be completed, reducing the chance of engaging subjects to participate.

The questionnaire was available on the Internet. It was developed using LimeSurvey[7], a free and open source survey tool.

The questionnaire was divided into five parts:

- Characterization of the subject;
- Identification of components pertinent to represent the components of the effort involved in requirements projects;
- Identification of factors pertinent to estimate the effort involved in developing software requirements;
- Identification of the most relevant effort component;
- Definition of how these factors relate to the effort components.

The complete questionnaire is presented in the Section 4.1.10.

4.1.6 Analysis Guidelines

Weighing subjects

To differentiate the answers from the subjects, they are attributed a weight for each subject according to four perspectives: the academic background of the subject, the

[7] http://www.limesurvey.org/

number of papers on requirements effort estimation published by the subject, the experience level regarding the estimating of requirements effort in software projects, the total number of software projects covering software requirements specification one participated in. The formula used to define the subject's weight has been adapted from Farias (2002), as follows:

$S(i)= f(i)+ e(i)+ p(i)+ t(i) + m(i) + k(i) + a(i) + n(i)/ MedianTP$,where:

- *S(i)* is the weight attributed to subject *i*;
- *f* (i) is the academic formation. The options for this field are:
  - *f* (i) = 0, if the subject holds an Undergraduate degree;
  - *f* (i) = 1, if the subject holds a Specialization degree;
  - *f* (i) = 2, if the subject holds a Master degree;
  - *f* (i) = 3, if the subject holds a PhD or DSc degree;
- *e(i) is* the subject's experience level on estimating the requirements effort in software projects. The options for this field are:
  - *e(i)* = 0, if the experience level is low;
  - *e(i)* = 1, if the experience level is medium;
  - *e(i)* = 2, if the experience level is high;
  - *e(i)* = 3, if the experience level is very high;
- *p(i)* is the subject's level of experience on requirements specification in different problem domains.
  - *p(i) = 0,* if the subject only worked with a single problem domain.
  - *p(i) = 1,* if the subject worked with two different problem domains.
  - *p(i) = 2,* if the subject worked with three or more different problem domains.
- *t(i)* is the subject's experience level on various Requirements Engineering techniques and processes.
  - *t(i) = 0,* if the subject only worked with a single Requirements Engineering process.
  - *t(i) = 1,* if the subject worked with two different Requirements Engineering processes.
  - *t(i) = 2,* if the subject worked with three or more different Requirements Engineering processes.

- *m(i)* is the subject's experience level on requirements specification project management.
    - *m(i) = 0,* if the subject only managed a single requirements specification project.
    - *m(i) = 1,* if the subject managed two different requirements specification projects.
    - *m(i) = 2,* if the subject managed three or more different requirements specification projects.
- *k(i)* is the subject's experience level regarding requirements elicitation with various stakeholders.
    - *k(i) = 0,* if the subject worked with one or two different stakeholders.
    - *k(i) = 1,* if the subject worked with three or four different stakeholders.
    - *k(i) = 2,* if the subject worked with five or more different stakeholders.
- *a(i)* is the subject's experience level working with various Requirements Engineering teams.
    - *a(i) = 0,* if the subject only worked with a single team.
    - *a(i) = 1,* if the subject worked with two different teams.
    - *a(i) = 2,* if the subject worked with three or more different team.
- *n(i) is* the estimated number of software projects covering software requirements specification one has taken part in.
- *MedianTP* is the median of the total number of software projects covering software requirements specification considering the answers of all subjects.

After computing the total weight for each subject we normalized the results by dividing it by the maximum possible weight, to fit results in the [0-1] interval, i.e.:

S'(i) = S(i) / max(S)

4.1.7 Context Selection

The study will be performed by the subject, on a website, to fill the questionnaire. Each subject will have unlimited time to fill the questionnaire. Monitoring will not be carried out.

4.1.8 Subjects Selection

The population of this survey consists of authors of scientific papers on Requirements Engineering identified by and referenced in our quasi-systematic literature review, by practitioners with professional experience in Requirements Engineering and also by other researchers in this area.

The subjects were contacted by email, with a link to the website with the questionnaire. We are not able to determine how many people were invited as we used mailing lists whose numbers of members is unknown.

In addition, as part of a MELLO and TRAVASSOS, 2013 work on sample enlargement in empirical Software Engineering studies, we also used LinkedIn[8] to recruit members that participate in groups related to Requirements Engineering. This allowed us not only to increase the number of respondents but also to achieve greater sample heterogeneity (MELLO and TRAVASSOS, 2013) as will be shown in Section 4.3.1.

4.1.9 Variables

Independent Variables:

- The initial set of effort components
- The initial set of influence factors

Dependent Variables:

- The final set of effort components
- The final set of influence factors
- The most relevant effort component
- The relations between factors and components

4.1.10 Instrumentation Details

The instrumentation used in this study was designed to be as simple as possible and to spend a minimal amount of time. The questionnaire is split into nine steps as shown below.

[8] http://www.linkedin.com/

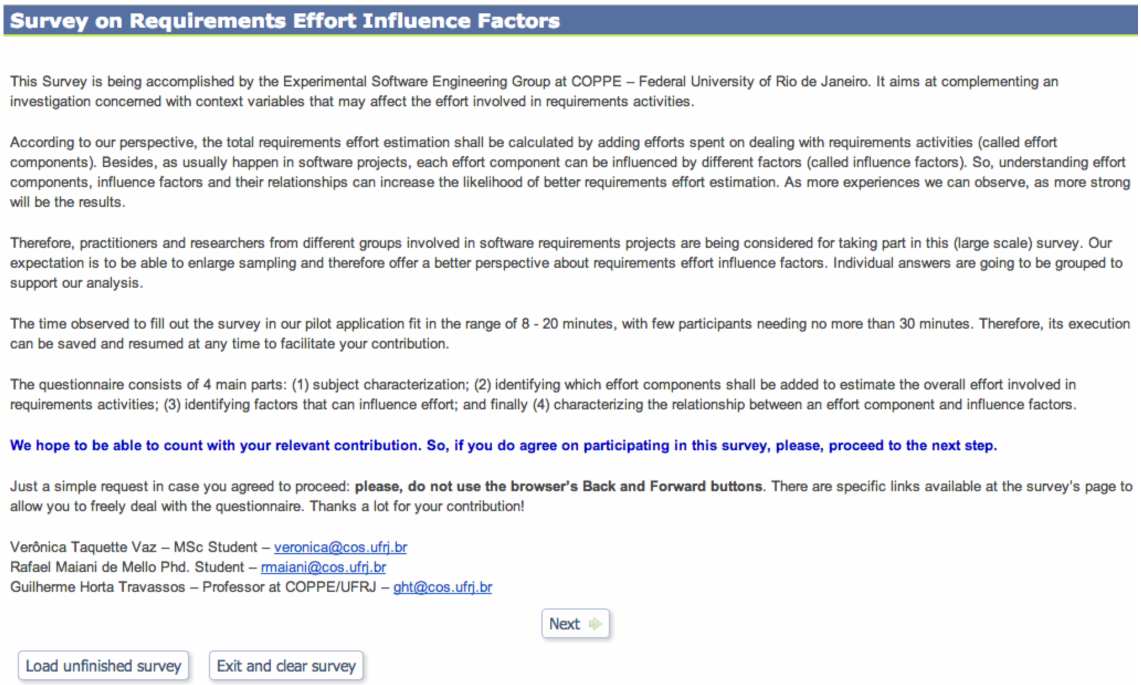


Figure 4-2 - Welcome Screen

The first screen is the welcome screen. It describes the goals of the survey, shows a summary of the steps to be followed in the questionnaire and gives other instructions on using the instrument:

"This Survey is being performed by the Experimental Software Engineering Group at COPPE – Federal University of Rio de Janeiro. It aims at supplementing an investigation concerned with context variables that may affect the effort involved in requirements activities. According to our perspective, the total requirements effort estimation shall be calculated by adding efforts spent on dealing with requirements activities (called effort components). Besides, as it is usually the case in software projects, each effort component can be influenced by different factors (called influence factors). So, understanding effort components, influence factors and their relationships can increase the likelihood of better requirements effort estimation. The more experiences we can observe, the stronger the results will be. Therefore, practitioners and researchers from different groups involved in software requirements projects are being considered to take part in this (large scale) survey. Our expectation is to be able to enlarge sampling and therefore offer a better perspective on the requirements effort influence factors. Individual answers will be grouped to support our analysis. The time observed to fill out the survey in our pilot application fit the range of 8-20 minutes, with a few participants needing no more than 30 minutes. Therefore, its execution can be saved and resumed at any time to facilitate your contribution. The questionnaire consists of 4 main parts: (1) subject characterization; (2) identifying which effort components will be added to estimate the overall effort involved in requirements activities; (3) identifying factors that can influence effort; and finally (4) characterizing the relationship between an effort component and influence factors. We hope to be able to count with your relevant contribution. So, if you agree on participating in this survey, please proceed to the next step. Just a simple request in case you agreed to proceed: please, do not use the browser's Back and Forward buttons. There are specific links available on the survey's page to allow you to freely deal with the questionnaire. We thank you for your contribution!"

The screen below is the first step in the subject's characterization. It aims at finding out where the subject was recruited from (note that the second and third questions of this screen are only shown if the subject chose the "LinkedIn Group invitation" option in the first question).

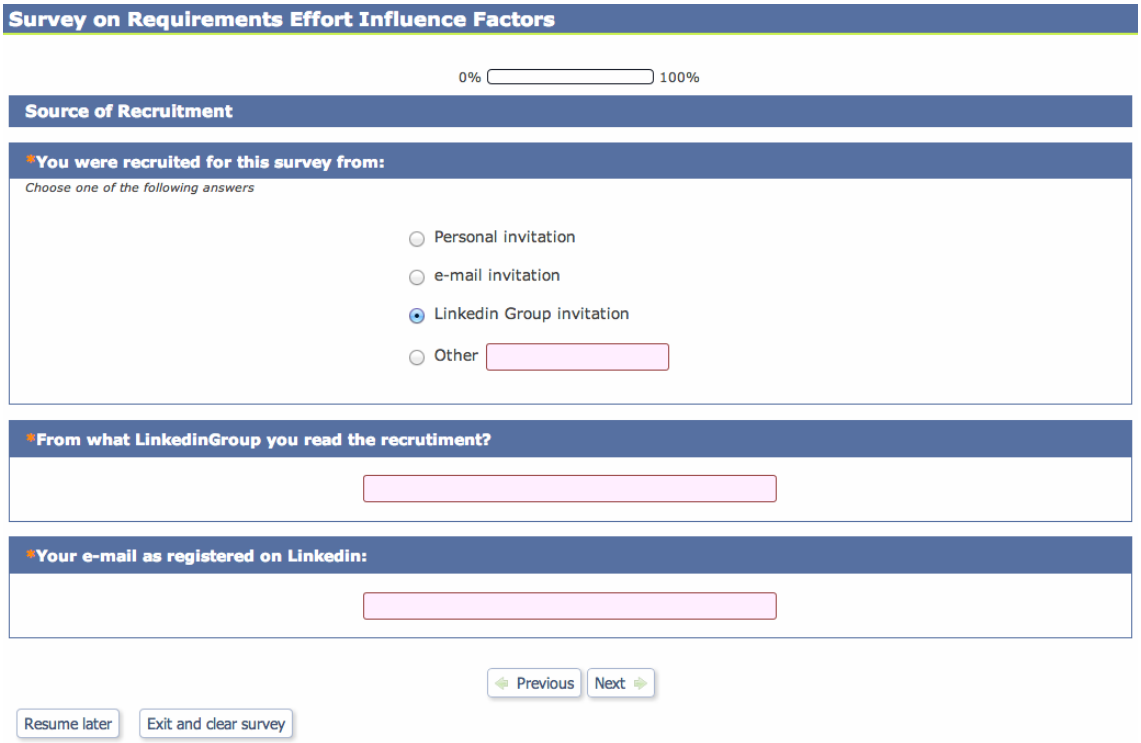


Figure 4-3 - Subject Characterization (step 1)

The characterization is completed through steps two (subjects' country, email, academic degree and parents' names) and three (subjects' experience in Requirements Engineering).

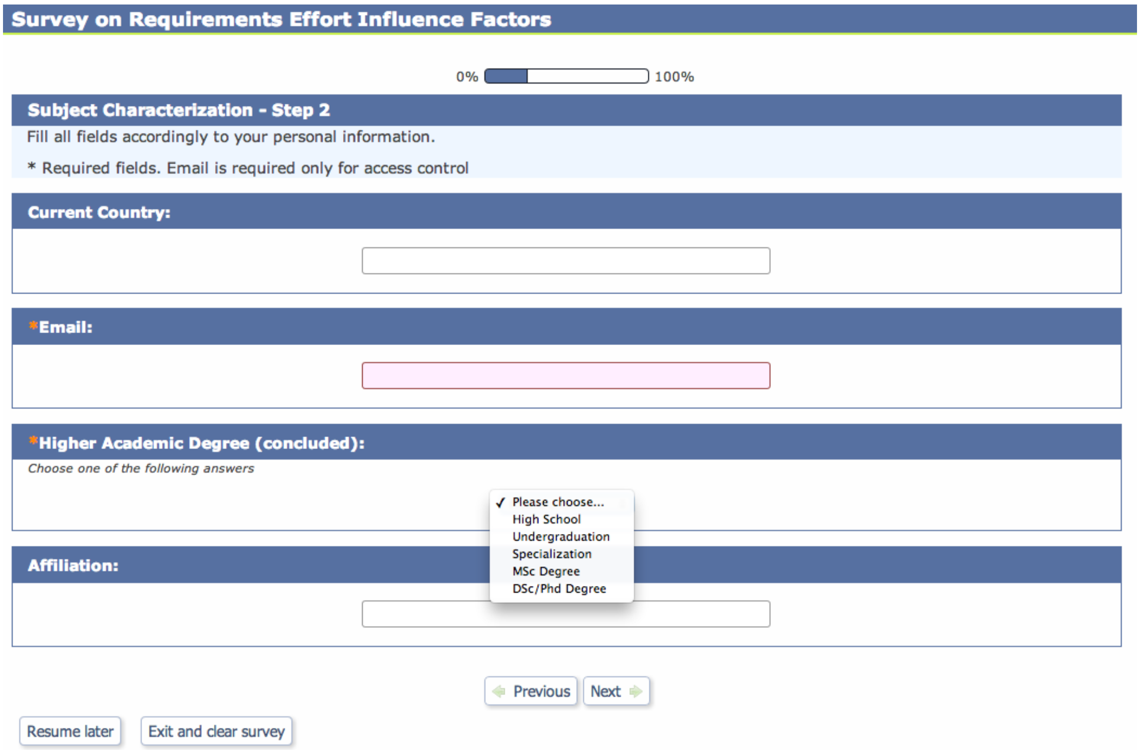


Figure 4-4 - Subject Characterization (step 2)

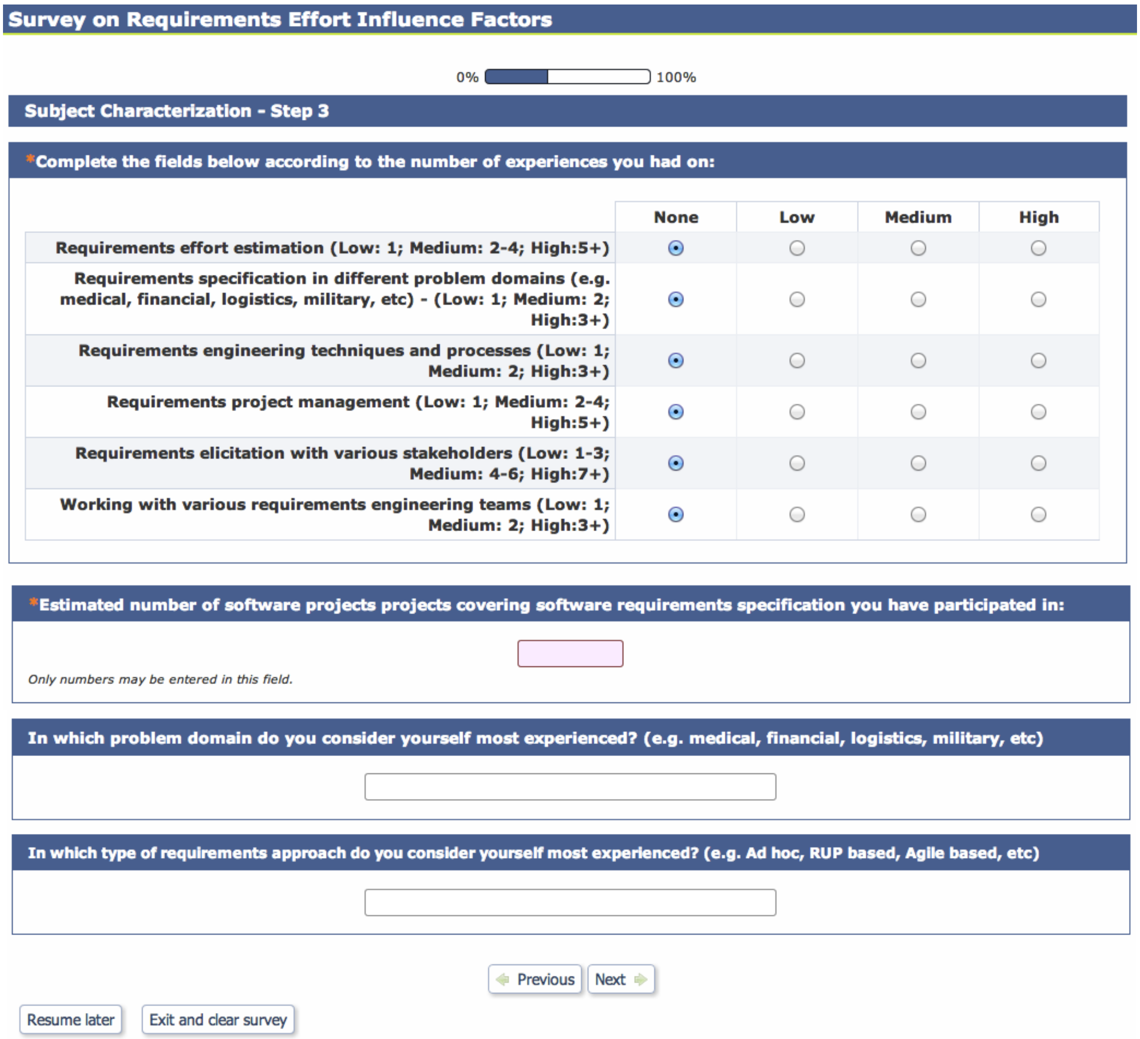


Figure 4-5 - Subject Characterization (step 3)

The next step comprises the identification of pertinent effort components. In this step the subject is asked to check which components from the initial set (Appendix 2) are pertinent to represent the components of the overall effort involved in requirements activities. In the next question he/she can also suggest other components not included in the initial set.

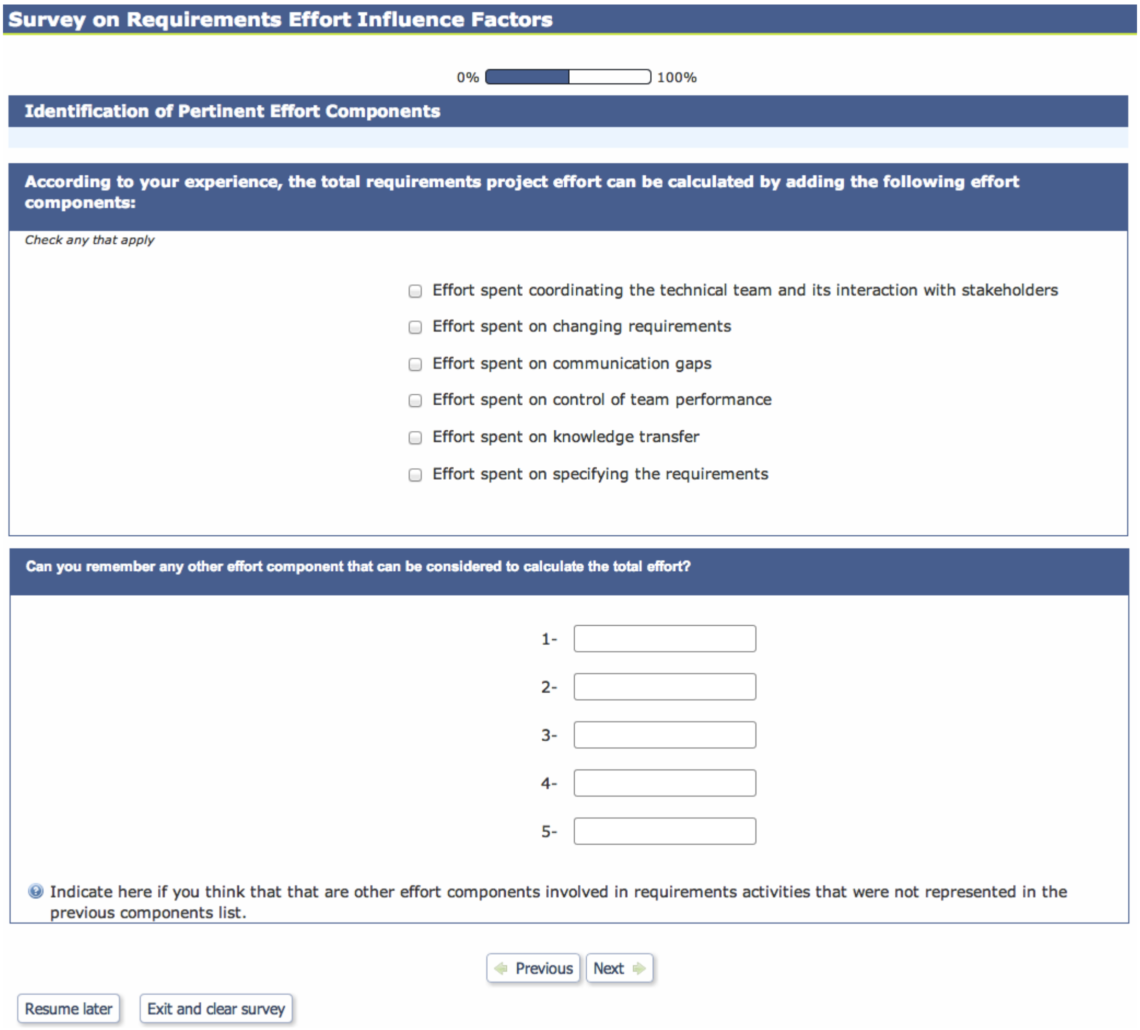
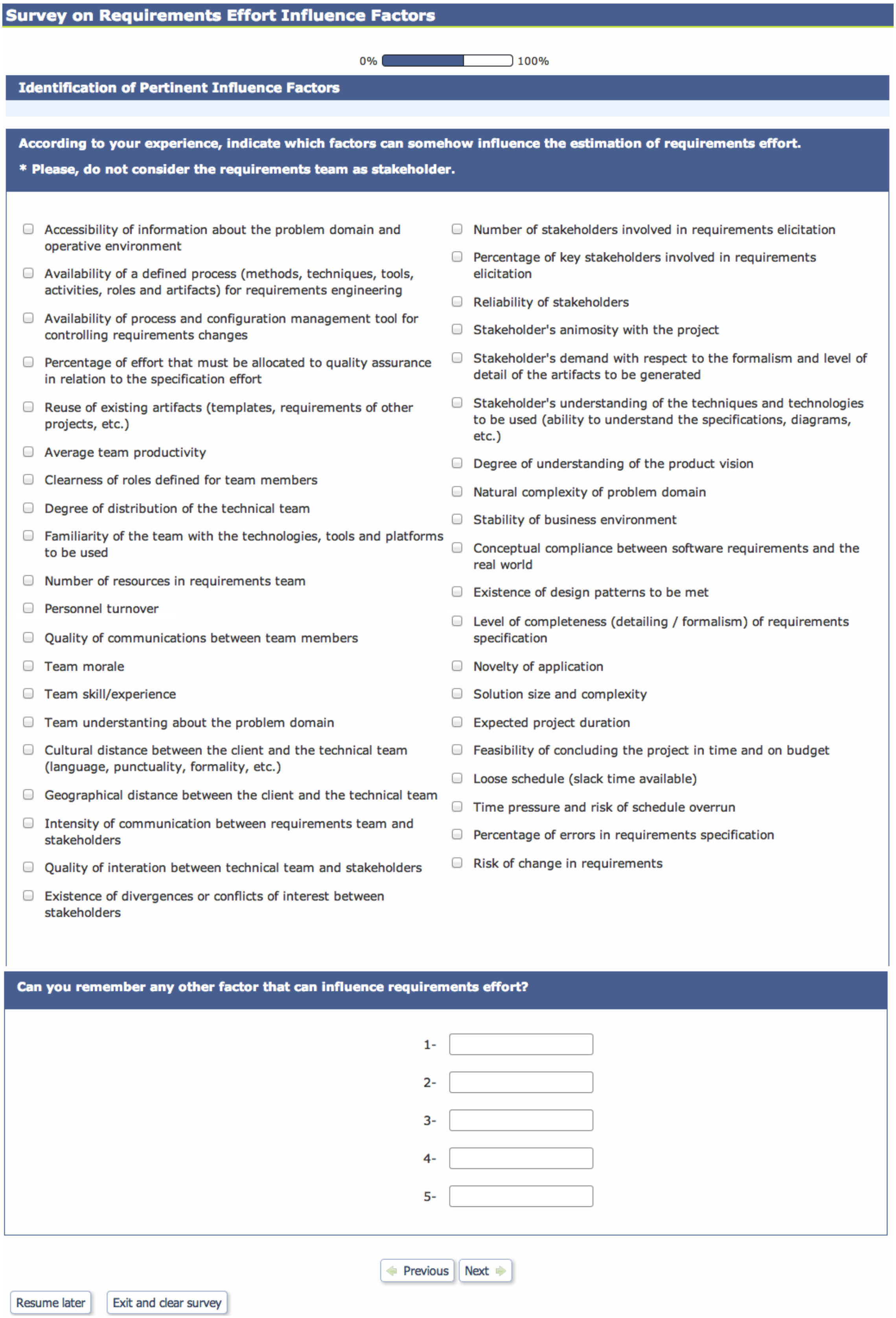


Figure 4-6 - Identification of the Requirements Effort Components

Next is the identification of the pertinent influence factors. Similarly to the previous step, the subject will evaluate the initial set of factors (Appendix 3), also being able to add other factors he/she believes to be missing from the initial set. The factors are presented in alphabetical order and grouped by category. This grouping was done to facilitate the reading and analysis of factors, as the subject can think about all the factors regarding one specific topic at once (for example, all the factors related to Stakeholders).



Survey on Requirements Effort Influence Factors

0% 100%

Identification of Pertinent Influence Factors

According to your experience, indicate which factors can somehow influence the estimation of requirements effort.

*** Please, do not consider the requirements team as stakeholder.**

- [ ] Accessibility of information about the problem domain and operative environment
- [ ] Availability of a defined process (methods, techniques, tools, activities, roles and artifacts) for requirements engineering
- [ ] Availability of process and configuration management tool for controlling requirements changes
- [ ] Percentage of effort that must be allocated to quality assurance in relation to the specification effort
- [ ] Reuse of existing artifacts (templates, requirements of other projects, etc.)
- [ ] Average team productivity
- [ ] Clearness of roles defined for team members
- [ ] Degree of distribution of the technical team
- [ ] Familiarity of the team with the technologies, tools and platforms to be used
- [ ] Number of resources in requirements team
- [ ] Personnel turnover
- [ ] Quality of communications between team members
- [ ] Team morale
- [ ] Team skill/experience
- [ ] Team understanting about the problem domain
- [ ] Cultural distance between the client and the technical team (language, punctuality, formality, etc.)
- [ ] Geographical distance between the client and the technical team
- [ ] Intensity of communication between requirements team and stakeholders
- [ ] Quality of interation between technical team and stakeholders
- [ ] Existence of divergences or conflicts of interest between stakeholders
- [ ] Number of stakeholders involved in requirements elicitation
- [ ] Percentage of key stakeholders involved in requirements elicitation
- [ ] Reliability of stakeholders
- [ ] Stakeholder's animosity with the project
- [ ] Stakeholder's demand with respect to the formalism and level of detail of the artifacts to be generated
- [ ] Stakeholder's understanding of the techniques and technologies to be used (ability to understand the specifications, diagrams, etc.)
- [ ] Degree of understanding of the product vision
- [ ] Natural complexity of problem domain
- [ ] Stability of business environment
- [ ] Conceptual compliance between software requirements and the real world
- [ ] Existence of design patterns to be met
- [ ] Level of completeness (detailing / formalism) of requirements specification
- [ ] Novelty of application
- [ ] Solution size and complexity
- [ ] Expected project duration
- [ ] Feasibility of concluding the project in time and on budget
- [ ] Loose schedule (slack time available)
- [ ] Time pressure and risk of schedule overrun
- [ ] Percentage of errors in requirements specification
- [ ] Risk of change in requirements

Can you remember any other factor that can influence requirements effort?

1-

2-

3-

4-

5-

Previous Next

Resume later Exit and clear survey

Figure 4-7 - Identification of the Requirements Effort Influence Factors

The seventh screen asks the subject to indicate which of the effort components he/she believes to be the most relevant. The list of effort components presented in this section comprises all the components the subject checked in step five (Figure 4-6) as well as the extra components he/she included in the open question.

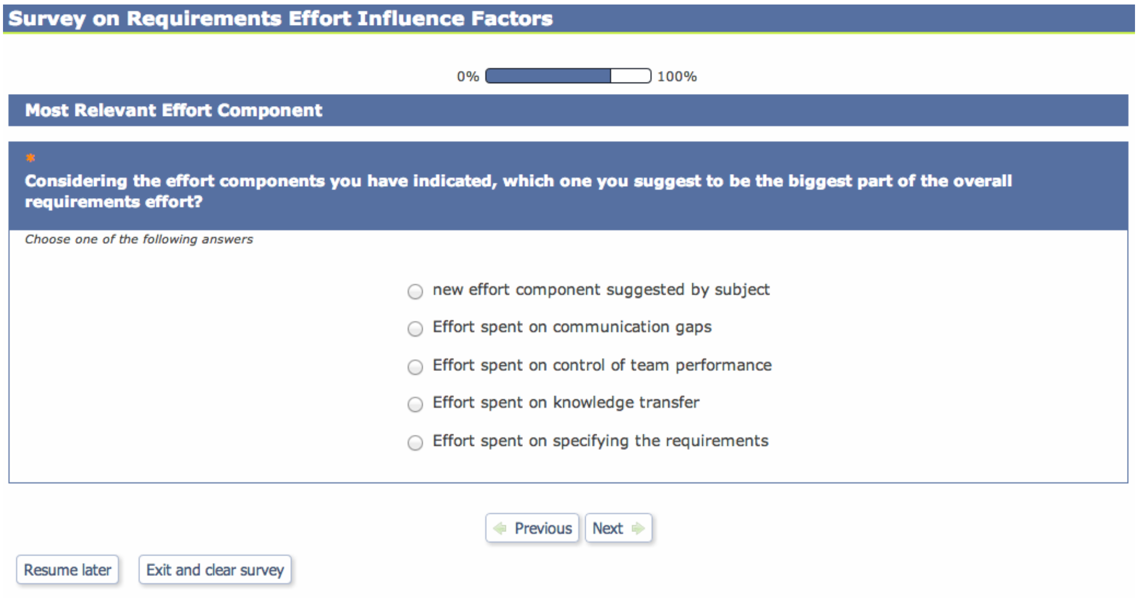


Figure 4-8 - Most Relevant Effort Component

The final step in the questionnaire aims to characterize the relationship between the factors and the effort components. For that, the subject is asked to indicate, for each factor, if the increasing of this factor will increase / decrease / not influence a specific effort component. In order to make the questionnaire shorter, we only ask this question regarding the effort component that was pointed out by the subject as the most relevant.

Similarly to the previous step, the list of influence factors presented in this screen comprises all the factors the subject checked in step six (Figure 4-7) as well as any extra factors one may have included in the open question.

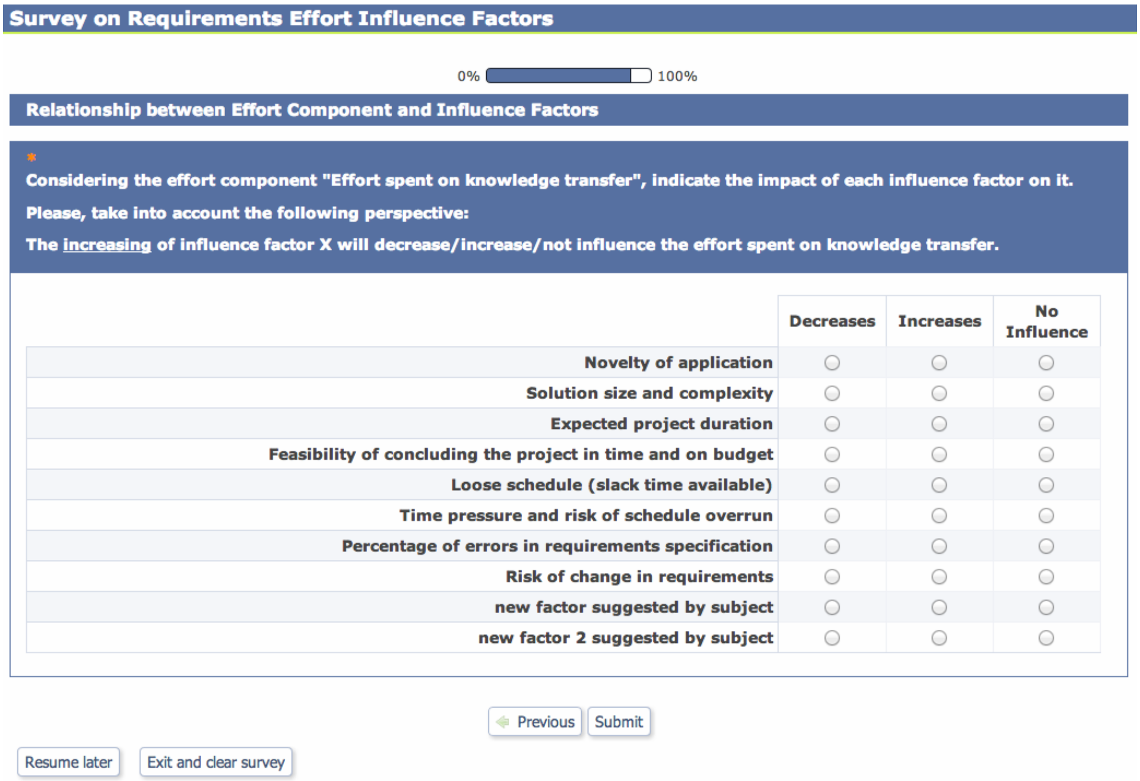


Figure 4-9 - Relationship between Influence Factors and the chosen Effort Component

Finally, the last screen thanks the subject for his/her collaboration and provides our contact information, if the subject has any doubts or suggestions.

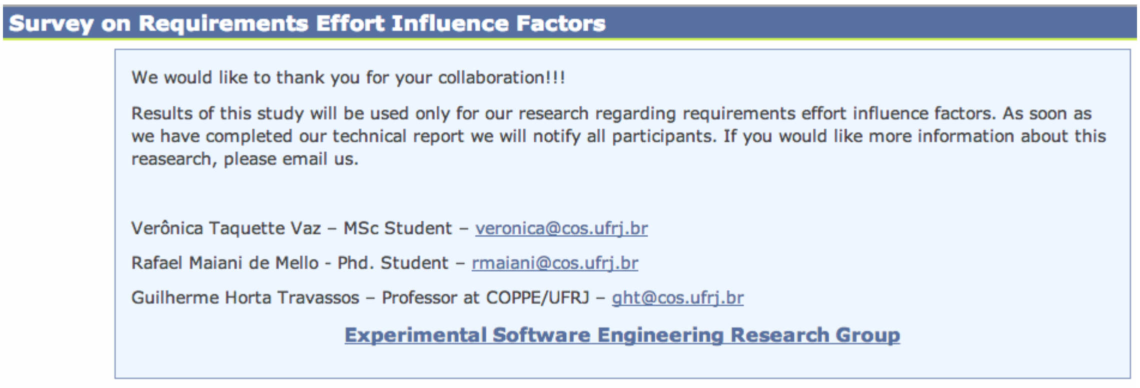


Figure 4-10 - Thank You Screen

4.2 Survey Execution

After the survey plan was complete, we conducted first a preliminary execution using subjects from inside our research group. The answers from this preliminary execution were not considered in the final results. Our goal was to collect the feedback from the participants on the comprehensibility of the questionnaire and to measure the average time to fill it. Based on these answers, we were able to further improve the survey instrument.

Finally, participants were contacted by email and by LinkedIn to fulfil the questionnaire. The survey web site was available for contributions from May 4th to May 19th. We received a total of 84 full responses, being 50 recruited from direct personal invitation and 34 from LinkedIn groups.

4.3 Results

4.3.1 Characterization

To analyze the data from the subjects' characterization we applied the equation described in Section 4.1.6 to obtain the weight associated to each subject. Table 4-1 shows the results obtained, presented in ascending order of normalized final score of each subject. Please note that some subjects' ids are preceded by an 'R' which means the subject came from the LinkedIn recruitment group.

This weight is very important for the analysis of the results, as it allows us to balance the answers for each question based on the respondent's experience.

Table 4-1 – Subjects Weight

Subject id	**Normalized Final Score**
106	0.121621622
22	0.17027027
31	0.172972973
108	0.197297297
137	0.202702703
149	0.221621622
27	0.248648649
69	0.256756757
115	0.27027027
18	0.278378378
52	0.278378378
R57	0.278378378
143	0.281081081
130	0.283783784
R39	0.289189189
32	0.337837838
R37	0.345945946
142	0.351351351
49	0.359459459
101	0.359459459
133	0.367567568
113	0.372972973
112	0.378378378
72	0.391891892
R26	0.405405405
R8	0.416216216
124	0.418918919
42	0.432432432
126	0.440540541
37	0.445945946
66	0.445945946
R34	0.445945946
R24	0.454054054
152	0.459459459
R59	0.462162162
136	0.472972973
R23	0.486486486
R46	0.486486486
R64	0.486486486
50	0.502702703
R68	0.502702703
38	0.510810811
26	0.513513514
146	0.513513514
R7	0.513513514
123	0.521621622
R43	0.521621622
73	0.527027027
102	0.527027027
59	0.532432432
51	0.540540541
74	0.540540541
129	0.540540541
R32	0.540540541
R69	0.551351351
45	0.554054054
78	0.554054054
R10	0.559459459
R52	0.562162162
R21	0.567567568
R54	0.567567568
R71	0.567567568
R28	0.578378378
82	0.581081081
144	0.581081081
R38	0.581081081
35	0.594594595
R16	0.594594595
R74	0.594594595
R75	0.608108108
55	0.621621622
76	0.621621622
R22	0.621621622
R50	0.621621622
R60	0.635135135
R65	0.648648649
R67	0.654054054
15	0.675675676
R4	0.675675676
R62	0.702702703
30	0.72972973
R27	0.72972973
47	0.756756757
88	0.945945946

After weighing all the subjects, we decided to remove the first quartile from the distribution that represented the less experienced individuals, with scores equal to or smaller than 0.36892. We considered that these individuals did not have enough experience to be able to answer the questions with confidence. We also removed one outlier that obtained a score of 0.945945946 (the highest one) to ensure distribution normality. Figure 4-11, Table 4-2, Table 4-3, Table 4-4 and Table 4-5 show the descriptive statistics of the sample after this removal.

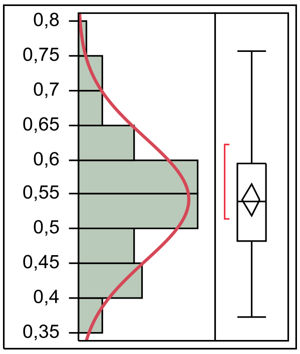


Figure 4-11 - Subjects' Weight Distribution

Normal (0.54263,0.08852)

Table 4-2 - Quantiles

100.0%	maximum	0.757
99.5%		0.757
97.5%		0.741
90.0%		0.669
75.0%	quartile	0.595
50.0%	median	0.541
25.0%	quartile	0.483
10.0%		0.423
2.5%		0.376
0.5%		0.373
0.0%	minimum	0.373

Table 4-3 - Summary Statistics

Mean	0.543
Std Dev	0.0885
Std Err Mean	0.0112
Upper 95% Mean	0.5651
Lower 95% Mean	0.520
N	62

Table 4-4 - Fitted Normal Parameter Estimates

Type	Parameter	Estimate	Lower 95%	Upper 95%
Location	μ	0.542	0.520	0.565
Dispersion	σ	0.0885	0.0752	0.108

Table 4-5 - Goodness-of-Fit (Shapiro-Wilk W Test)

W	Prob<W
0.98527 0	0.6639

Note: Ho = The data is from the Normal distribution. Small p-values reject Ho.

Figure 4-12, Figure 4-13 and Figure 4-14 shows some descriptive statistics from the sample. Regarding the geographic distribution we can observe that there were answers from South, Central and North America, Europe, Asia and Oceania, which show that this survey was able to achieve a high degree of geographic distribution. The sample also shows a high degree of distribution among problem domains.

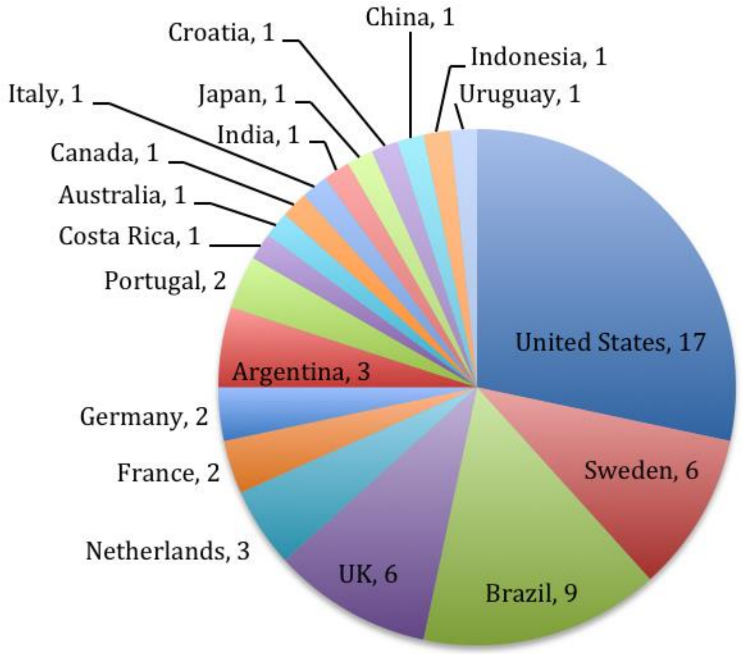


Figure 4-12 - Subject's geographic distribution

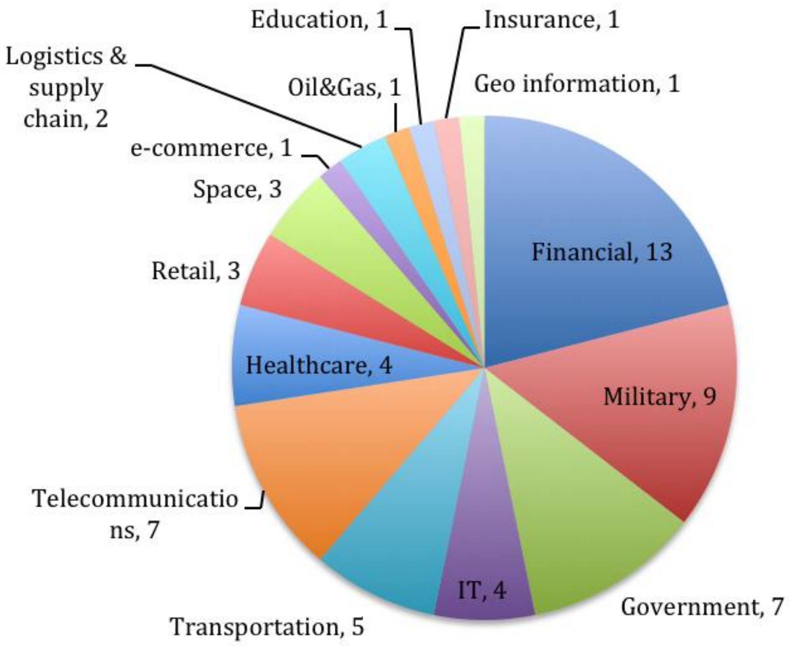


Figure 4-13 - Problem domain distribution

Regarding the requirements engineering approaches most adopted by subjects, RUP and Agile based process are the most frequent ones. The “Other” group show in Figure 4-14 represents approaches that were only mentioned by a single subject. This group includes: “Requirements Lifecycle Management”, “Requirements workshop and scenarios”, “DO-178”, “understanding requirements through simulation”, “ISO29148 based”, “SSADM Based”, “REVEAL (UK Requirements Elicitation Methodology)”, “Performance analysis and modelling” and “Volere”. Two subjects stated they could not answer this question because they use various approaches.

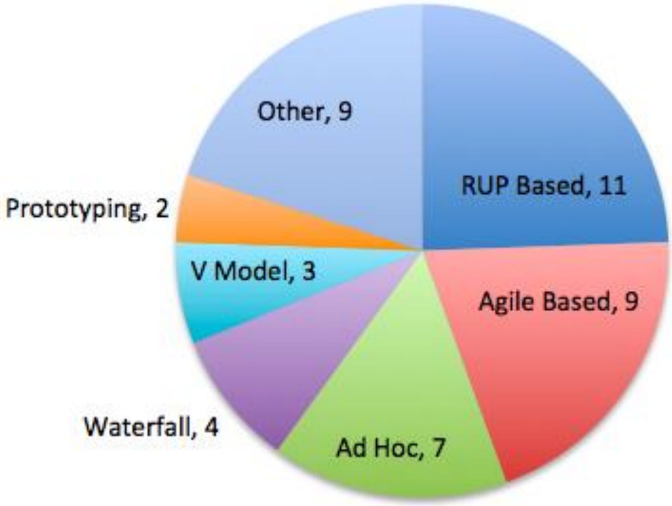


Figure 4-14 - Requirements approach distribution

Regarding the source of recruitment, in the initial set of 84 participants, 50 were recruited from direct personal invitation and 34 from LinkedIn groups. However, after the outlier analysis and the first quartile removal, this distribution changed to: 31 from direct personal invitation and 31 from LinkedIn. This means that invitations from LinkedIn brought more experienced subjects and for so, had less answers removed from the set.

4.3.2 Effort Components

To analyze if an effort component from the initial set should be kept or removed from the set of pertinent components, we computed the sum of the weights of each subject that marked this component as pertinent. Next, this score is divided by the maximum possible score a component could obtain, i.e., the sum of the score of all subjects, obtaining the percentage of the sample that voted for the inclusion of this component. For instance: 42 subjects marked the component C1 (“Effort spent coordinating the technical team and its interaction with stakeholders”) as pertinent. The total score obtained by C1 is equal to 22.35135135 (sum of the 42 subjects’ weight). The maximum score possibly obtained by a component is equal to 33.64324324 (sum of all subjects' weight). So we can say that 66.4% of the sample considered that component C1 should remain in the set of relevant components.

The threshold to classify a component as included in the final set, or excluded, was defined through the following procedure: the average score (0.5426) minus one standard deviation (0.08852) of the subjects' weigh distribution is equal to 0.4541. If all subjects from the sample had a weight equal to 0.4541, and all of them marked one component as relevant, then this component would obtain a percentage score of 83.68%.

Percentage Score = total score/maximum score = (0.4541*62/33,643) = 0.8368

We define the threshold for inclusion as half this punctuation: 83.68% / 2 = 41.84%, i.e., if the percentage score of one component is equal to or greater than 41.84% then this component is considered pertinent and will be kept in the set o components. Otherwise, it will be removed.

Applying this threshold to all six components from the initial set, we obtained the following results:

Table 4-6 - Analysis of Pertinence of Components from the Initial Set

Component Code	Component Description	Score	Kept/excluded
C1	Effort spent coordinating the technical team and its interaction with stakeholders	66.4%	Kept
C2	Effort spent on changing requirements	80.6%	Kept
C3	Effort spent on communication gaps	62.4%	Kept
C4	Effort spent on control of team performance	29.4%	Excluded
C5	Effort spent on knowledge transfer	65.9%	Kept
C6	Effort spent on specifying the requirements	92.1%	Kept

So component C4 was removed from the final set of pertinent components.

The next step was the analysis of the extra components suggested by the subjects. Similarly to what was done to the components from the literature, we computed the score of each new component as a percentage of the total subject's weight. In this case, however, we considered that a new component should be included in the set if it obtained a score equal to or greater than 5% of the total sample weight (which corresponds, for instance, to three experienced subjects suggesting the same component for inclusion). After this analysis, four new components obtained sufficient marks inclusion: "Effort spent on requirements elicitation", "Effort spent on managing requirements (tracking current status, traceability and change management)", "Effort spent on requirements analysis and modelling" and "Effort spent on requirements verification and validation". Table 4-2 shows final set of components considered as pertinent to represent the parts of the overall effort involved in requirements activities.

Table 4-7 - Final Set of Requirements Effort Components

Source	Component Description
Coding from technical literature	Effort spent coordinating the technical team and its interaction with stakeholders
Coding from technical literature	Effort spent on changing requirements
Coding from technical literature	Effort spent on communication gaps
Coding from technical literature	Effort spent on knowledge transfer
Coding from technical literature	Effort spent on specifying the requirements
Survey	Effort spent on requirements elicitation
Survey	Effort spent on managing requirements (tracking current status, traceability and change management)
Survey	Effort spent on requirements analysis and modelling
Survey	Effort spent on requirements verification and validation

4.3.3 Influence Factors

To analyze the set of influence factors we applied the same process and criteria described in Section 4.3.2. From the initial set of 40 influence factors obtained from the literature, 28 factors were considered pertinent and 12 factors were removed.

Table 4-8 - Analysis of Pertinence of Factors from the Initial Set

Code	Factor Description	Score	Kept/excluded
F1	Accessibility of information about the problem domain and operative environment	0.722	Kept
F2	Availability of a defined process (methods, techniques, tools, activities, roles and artefacts) for Requirements Engineering	0.684	Kept
F3	Availability of process and configuration management tool for controlling requirements changes	0.501	Kept
23	Percentage of effort that must be allocated to quality assurance in relation to the specification	0.379	Excluded

	effort		
30	Reuse of existing artefacts (templates, requirements of other projects, etc.)	0.707	Kept
F4	Average team productivity	0.339	Excluded
F5	Clarity of roles defined for team members	0.564	Kept
F8	Degree of distribution of the technical team	0.348	Excluded
13	Familiarity of the team with the tools, technologies and platforms to be used	0.499	Kept
21	Number of resources in requirements team	0.506	Kept
26	Personnel turnover	0.470	Kept
27	Quality of communications between team members	0.552	Kept
37	Team morale	0.424	Kept
38	Team skill/experience	0.698	Kept
39	Team understanding of the problem domain	0.594	Kept
F7	Cultural distance between the client and the technical team (language, punctuality, formality, etc.)	0.598	Kept
15	Geographical distance between the client and the technical team	0.480	Kept
16	Intensity of communication between requirements team and stakeholders	0.519	Kept
28	Quality of interaction between technical team and stakeholders	0.477	Kept
11	Existence of divergences or conflicts of interest between stakeholders	0.613	Kept
22	Number of stakeholders involved in requirements elicitation	0.574	Kept
25	Percentage of key stakeholders involved in requirements elicitation	0.502	Kept
29	Reliability of stakeholders	0.576	Kept
34	Stakeholder's animosity with the project	0.443	Kept
35	Stakeholder's demand with respect to the formalism and level of detail of the artefacts to be generated	0.398	Excluded
36	Stakeholders' understanding of the techniques and technologies to be used (ability to understand the specifications, diagrams, etc.)	0.389	Excluded
F9	Degree of understanding of the product vision	0.547	Kept
19	Natural complexity of problem domain	0.532	Kept
33	Stability of business environment	0.476	Kept
F6	Conceptual compliance between software requirements and the real world	0.263	Excluded

10	Existence of design patterns to be met	0.192	Excluded
17	Level of completeness (detailing / formalism) of requirements specification	0.479	Kept
20	Novelty of application	0.335	Excluded
32	Solution size and complexity	0.502	Kept
12	Expected project duration	0.360	Excluded
14	Feasibility of concluding the project in time and on budget	0.262	Excluded
18	Loose schedule (slack time available)	0.213	Excluded
40	Time pressure and risk of schedule overrun	0.368	Excluded
24	Percentage of errors in requirements specification	0.489	Kept
31	Risk of change in requirements	0.603	Kept

The fact that some factors did not obtain the minimum score to remain in the model means that, based on our subjects' experience, in most cases these factors do not have a significant influence on the requirements effort. It is important, however, to notice that these factors might be relevant in some specific contexts, as we have evidence both from the technical literature and from some subjects indicating their relevance. Further research should be carried out to understand which specific context these factors might be relevant in. The contact information of the subjects that marked these factors as relevant provides an opportunity to carry out this investigation.

Relying just on the data available from this first survey, we tried to identify some significant difference among the characterization from the subjects that included each factor versus the ones that excluded it. We analyzed the differences in their experience level, geographic distribution, problem domain and process types they have been working with. We could not identify any significant differences regarding their answers about geographic distribution, problem domain or process types. On the other hand, we could observe a slight difference among their average experience. Table 4-11 shows the results of this analysis. Note that, for all excluded factors, except for F40, the average experience of those who included the factor is greater than the average experience of those who did not. One possible interpretation of this result is that experienced professionals tend to perceive the importance of more factors, as they might have experienced a greater variety of situations and contexts. In fact, if we compare the first quartile of the distribution (the least experienced subjects) they have on average marked 13 factors as pertinent, while the last quartile (the most experienced subjects) presented an average of 24 factors marked that way. This result reinforces our hypothesis that the excluded factors might be relevant in some contexts, although we could not yet see which contexts they are.

Table 4-9 - Excluded Factors Analysis

Factor Code	Average experience among subjects that included it	Average experience among subjects that excluded it
23	0.580	0.522
F4	0.570	0.530
F8	0.585	0.523
35	0.583	0.519
36	0.569	0.527
F6	0.590	0.528
10	0.587	0.533
20	0.594	0.520
12	0.550	0.538
14	0.551	0.540
18	0.597	0.530
40	0.538	0.545

The analysis of extra factors suggested by the respondents did not result in the inclusion of any new factor, as none of the suggested ones obtained the minimum score of 5% of the sample. During this analysis, however, we could notice that some of the factors suggested by participants were already present in the initial set. This might suggest that the subjects did not fully understand the description of the factors. To tackle this problem, we refined the description of these factors.

Table 4-10 shows an analysis of such refinement.

Table 4-10 – Improving some Factors Description

Initial Factor Description	Factor Suggested by Respondents	New Factor Description
Clarity of roles defined for team members	Roles and responsibilities in the organizational set up	Clarity of roles and responsibilities defined for team members
Personnel turnover	Stability of the project team	Personnel (team or stakeholders) turnover
Percentage of effort that must be allocated to quality assurance in	Inspection and review procedures	Percentage of effort that must be allocated to quality assurance (inspection and

relation to the specification effort		review procedures) in relation to the specification effort
Familiarity of the team with the tools, technologies and platforms to be used	Tool training	Team knowledge of the tools, technologies and platforms to be used
Team understanding of the problem domain	Lack of subject matter expertise (business or solution domain)	Team experience with the problem domain
Team skill/experience	Lack of subject matter expertise (business or solution domain)	Team skill/experience with Requirements Engineering

Table 4-11 shows the final set of factors considered as pertinent to represent the context variables that influence the effort involved in requirements activities.

Table 4-11 - Final Set of Requirements Effort Influence Factors

Source	Factor Description
Coding from technical literature	Accessibility of information on the problem domain and operative environment
Coding from technical literature	Availability of a defined process (methods, techniques, tools, activities, roles and artefacts) for Requirements Engineering
Coding from technical literature	Availability of process and configuration management tool for controlling requirements changes
Coding from technical literature	Reuse of existing artefacts (templates, requirements of other projects, etc.)
Coding from technical literature and refined through survey	Clarity of roles and responsibilities defined for team members
Coding from technical literature and refined through survey	Team knowledge of the tools, technologies and platforms to be used
Coding from technical literature	Number of resources in requirements team
Coding from technical literature and refined through survey	Personnel (team or stakeholders) turnover
Coding from technical literature	Quality of communications between team members
Coding from technical literature	Team morale
Coding from technical	Team skill/experience with Requirements Engineering

literature and refined through survey	
Coding from technical literature and refined through survey	Team experience with the problem domain
Coding from technical literature	Cultural distance between the client and the technical team (language, punctuality, formality, etc.)
Coding from technical literature	Geographical distance between the client and the technical team
Coding from technical literature	Intensity of communication between requirements team and stakeholders
Coding from technical literature	Quality of interaction between technical team and stakeholders
Coding from technical literature	Existence of divergences or conflicts of interest between stakeholders
Coding from technical literature	Number of stakeholders involved in requirements elicitation
Coding from technical literature and refined through survey	Percentage of key stakeholders involved in requirements elicitation
Coding from technical literature	Reliability of stakeholders
Coding from technical literature	Stakeholder's animosity with the project
Coding from technical literature	Degree of understanding of the product vision
Coding from technical literature	Natural complexity of problem domain
Coding from technical literature	Stability of business environment
Coding from technical literature	Level of completeness (detailing / formalism) of requirements specification
Coding from technical literature	Size and complexity of solution
Coding from technical literature	Percentage of errors in requirements specification
Coding from technical literature	Risk of change in requirements

4.3.4 Most Relevant Effort Component

Figure 4-15 shows an analysis of the responses on which component corresponds to the largest part of the overall effort involved in requirements activities. The "Effort spent on specifying the requirements" was the most voted, with 41% of the total subjects' weight, followed by: the "Effort spent on knowledge transfer" (12%), the "Effort spent on requirements elicitation" (9%), the "Effort spent on changing requirements" (8%), the "Effort spent coordinating the technical team and its interaction with stakeholders" (7%) and the "Effort spent on communication gaps" (6%). Other components received a single vote each, with percentages varying from 1% to 2% of the total, according to the subjects' weight.

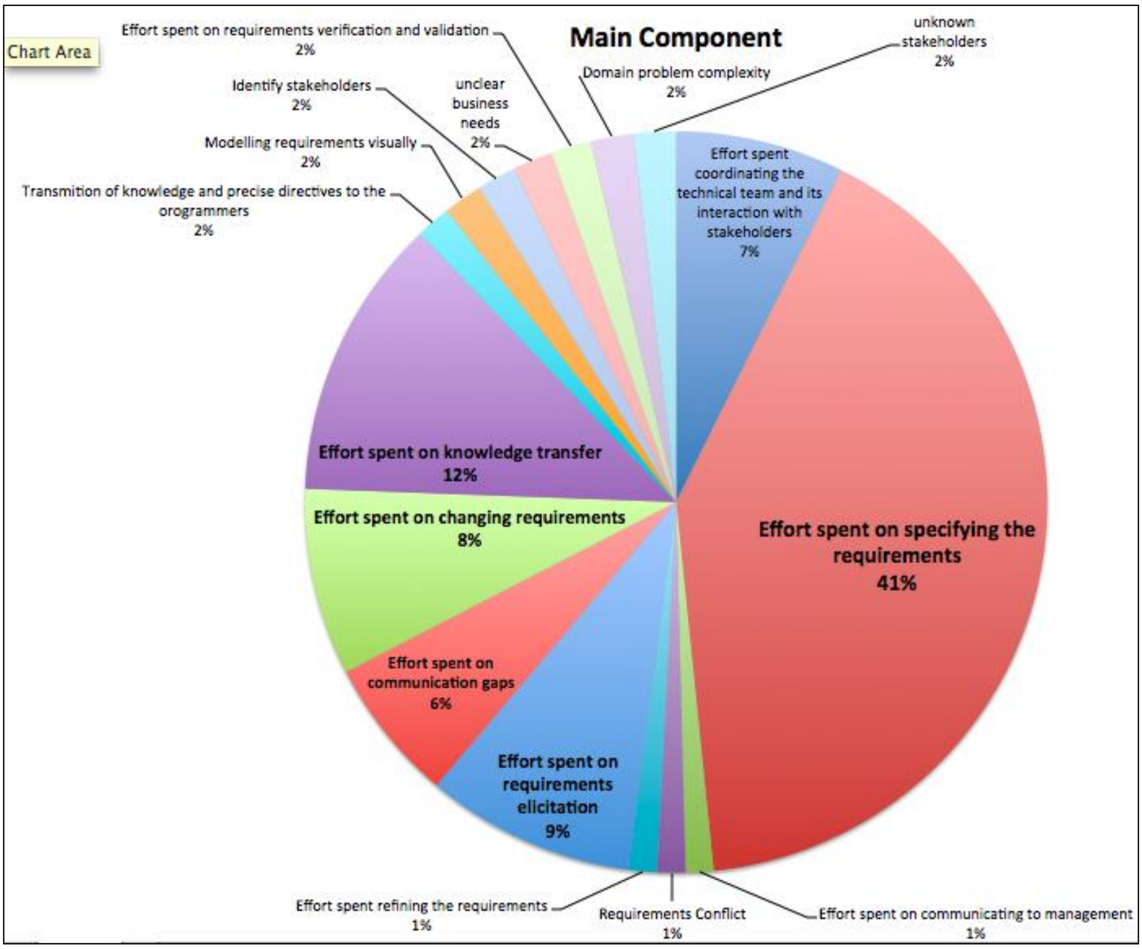


Figure 4-15 - Main Effort Component (weighted analysis)

We also carried the same analysis but now considering just the number of votes for each component, instead of the subjects' weight. Figure 4-16 shows the result of this analysis. Notice that this method of analysis results in subtle difference from the previous one, which shows that there is an even distribution of the subjects' opinion on this matter across experience levels.

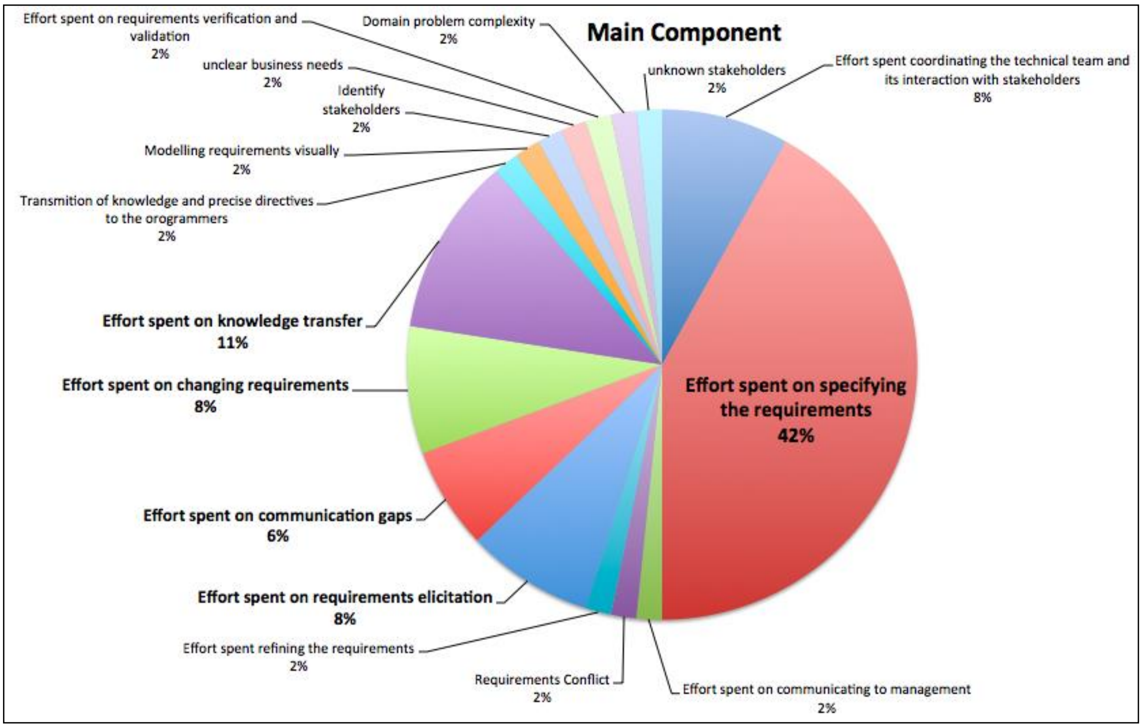


Figure 4-16 - Main Effort Component (vote count analysis)

4.3.5 Relationship between Factors and Effort Components

As mentioned before on the survey protocol, subjects were asked to only relate the influence factors to one single effort component, the component that the subject pointed out as the most relevant one. This means that a component has as many answers about its relations as the number of votes it received in the previous section (main effort component analysis). For instance, component “Effort spent on communication gaps” received three votes, so we only had three subjects answering about these component relations to each influence factor.

With this reduced sample it is harder to obtain a high confidence level on the relations between factors and components. For this reason, we present the results in this section followed by an indication of the confidence level associated to each relationship, in order to clarify the limitations of these results. The confidence level **ε** is computed through the following equation:

$\varepsilon = 100 - 100 * (\sqrt{(N-n)/N*n})$

Where N is the population size (in this case, 62, the number of subjects considered in the survey) and n is the sample size (the number of subjects that answered the question on one specific effort component). Table 4-12 shows the results of this analysis.

Table 4-12 - Confidence level associated with the answers on the relations of each effort component to the influence factors

Component	Confidence Level (%)
Effort spent coordinating the technical team and its interaction with stakeholders	38.86
Effort spent on changing requirements	34.61
Effort spent on communication gaps	24.38
Effort spent on knowledge transfer	**56.90**
Effort spent on specifying the requirements	**78.73**
Effort spent on requirements elicitation	38.13
Effort spent on managing requirements (tracking current status, traceability and change management)	-
Effort spent on requirements analysis and modelling	-
Effort spent on requirements verification and validation	-

We considered a 50% minimum confidence level, which means we only have data on the relations of components "Effort spent on knowledge transfer" and "Effort spent on specifying the requirements".

For each of these effort components, we had answers on the influence of each factor on this component (increases, decreases or does not influence). In order to compute the resulting relation influence of one factor in one effort component, we add the weights of subjects that chose the "increase" option and subtract the sum of weights of subjects that chose the "decrease" option. A positive result indicates an "increase" resultant relation, and a negative result indicates a "decrease" resultant relation. However, if the result falls within the error margin of 5%, then we consider that the direction is undefined, as shown in Figure 4-17.

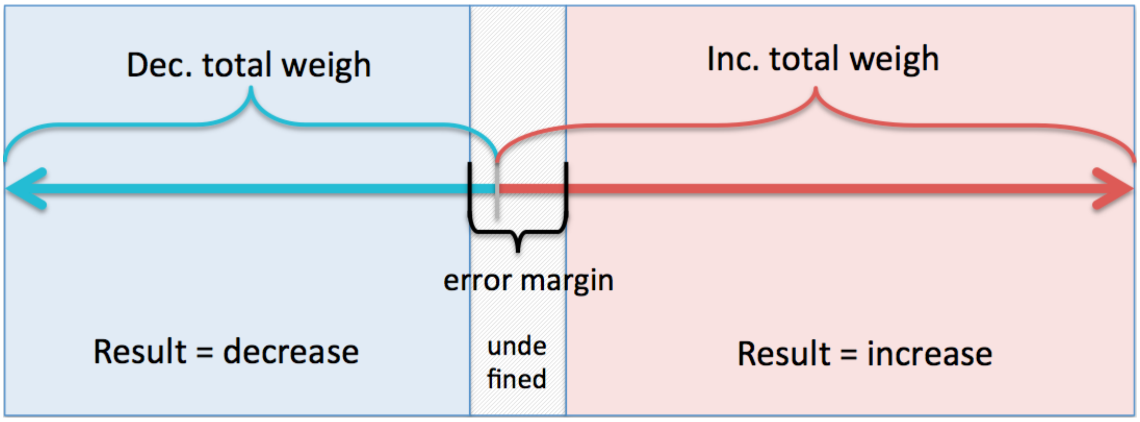


Figure 4-17 - Criteria for computing the resultant relation between an influence factor and an effort component

Table 4-13 shows an example of such analysis. In this example, we compute the following:

Increase total weight = sum of weights of all "increase" votes

Decrease total weight = sum of weights of all "decrease" votes

Result = Increase total weight - Decrease total weight

As this produces a negative value, and this value is less than 5% of the decrease total weight (error margin), than the resulting relation is a "decrease" relation.

Table 4-13 - Analysis of relations between factors and effort components

Subject id	Weight	Relation between F1 and C6
15	0.675675676	Increase
42	0.432432432	Decrease
72	0.391891892	Decrease
73	0.527027027	Decrease
74	0.540540541	Decrease
123	0.521621622	Decrease
126	0.440540541	Increase
129	0.540540541	Decrease
152	0.459459459	Decrease
R10	0.559459459	Decrease
R16	0.594594595	Increase
R21	0.567567568	Increase
R22	0.621621622	Decrease
R24	0.454054054	No Influence
R38	0.581081081	Decrease
R52	0.562162162	Decrease
R62	0.702702703	Decrease
R69	0.551351351	Increase

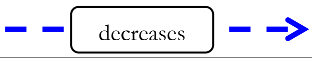
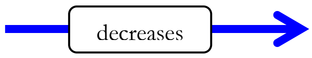
“Increase” total weight		2.82972973
“Decrease” total weight		6.440540541
Result (Inc. – Dec)		**-3.610810811**
5% of Inc. total weight (error margin)		0.141486486
5% of Dec. total weight (error margin)		-0.322027027
Resulting relation		**Decrease**

These relations were added to the diagrams, and results are presented in the next section.

4.3.6 Final Results

Figure 4-18 to Figure 4-31 show the same diagrams introduced in the previous chapter, but now including survey results. The relationships obtained through the survey data are shown in dashed lines, and relationships obtained both from the quasi-systematic literature review and from survey data are shown in thicker lines, as shown in Table 4-14.

Table 4-14 - Relationship Notation

Type of Relation	Source	Notation
Increases	Coding from technical literature	
Decreases	Coding from technical literature	
Increases	Survey	
Decreases	Survey	
Increases	Both	
Decreases	Both	

Note that the Project Category diagram is no longer present, as all codes from this category were removed after the survey analysis.

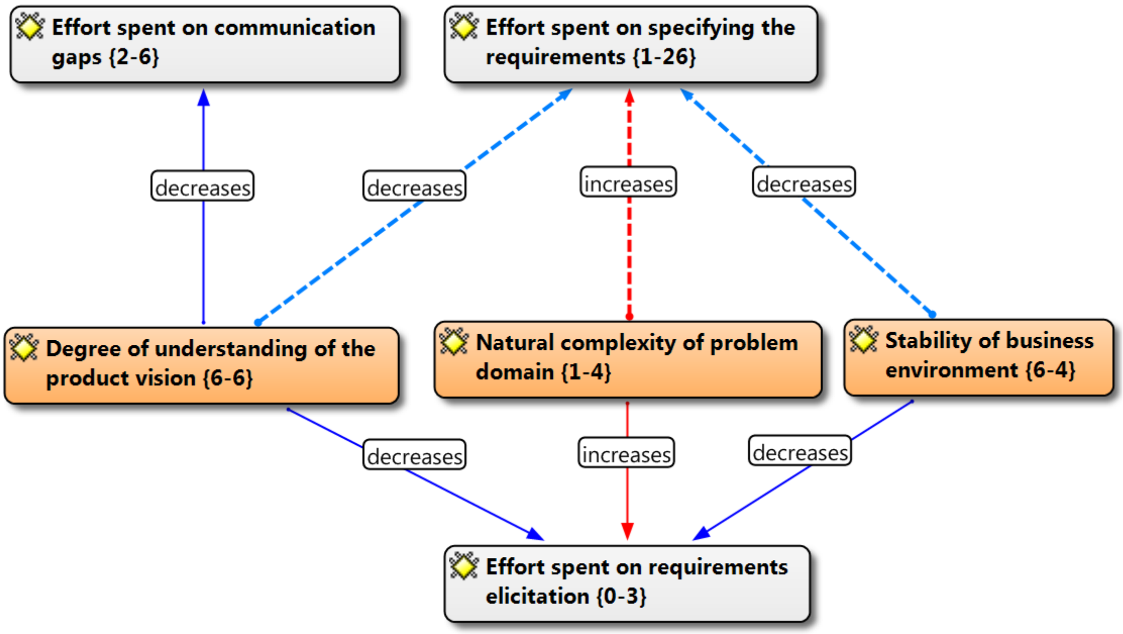


Figure 4-18 - The Problem Domain category (part 1)

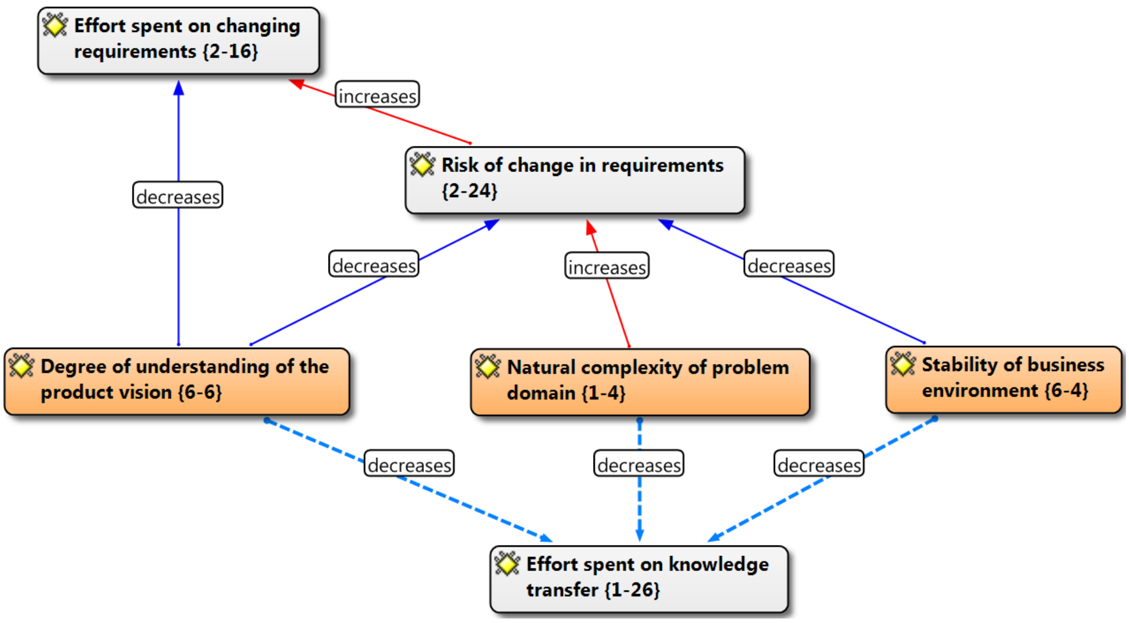


Figure 4-19 - The Problem Domain category (part 2)

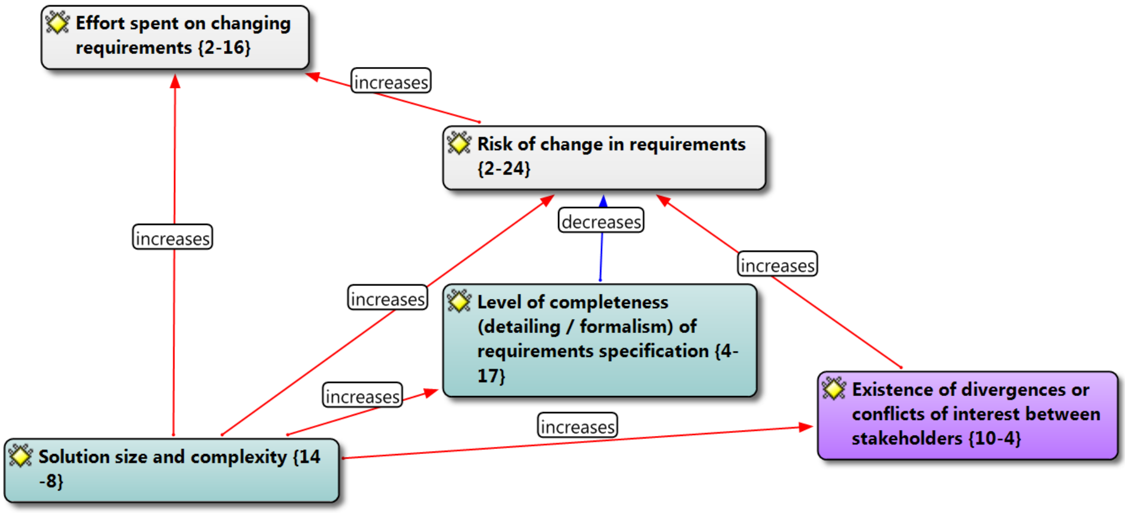


Figure 4-20 - The Solution Domain category (part 1)

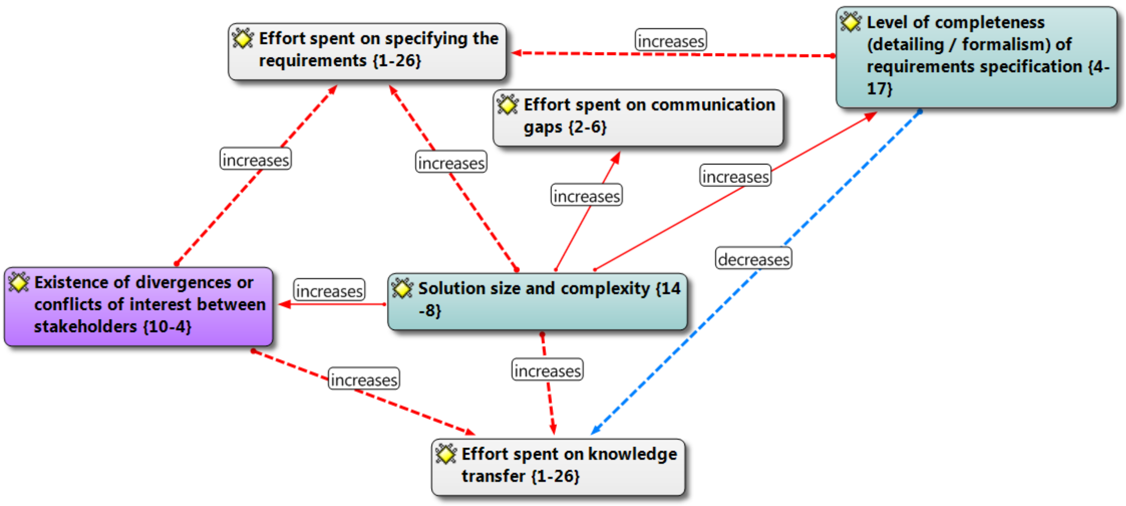


Figure 4-21 - The Solution Domain category (part 2)

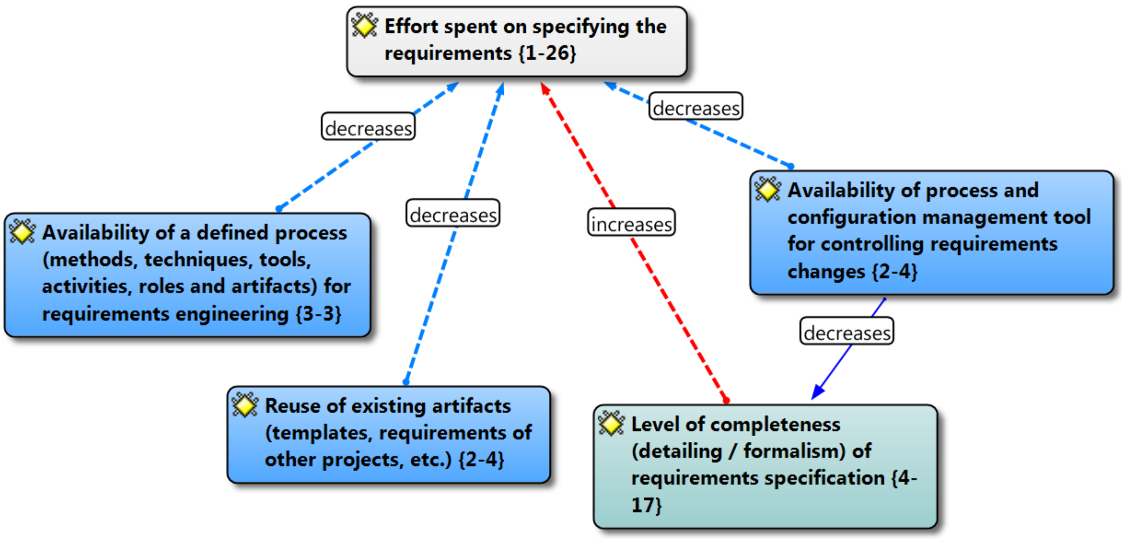


Figure 4-22 - The Environment for Requirements Engineering category (part 1)

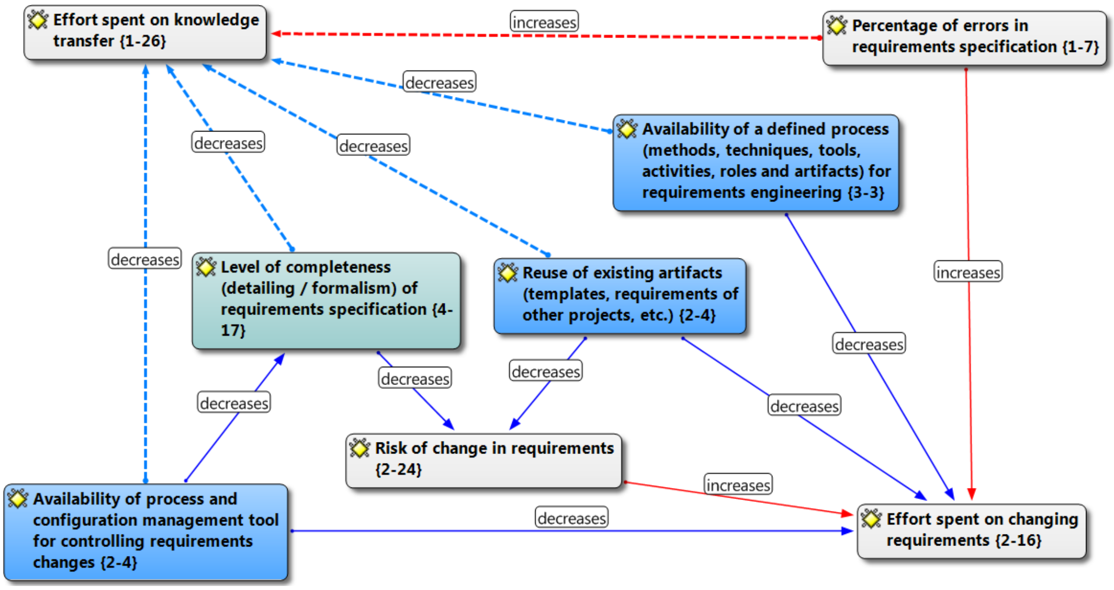


Figure 4-23 - The Environment for Requirements Engineering category (part 2)

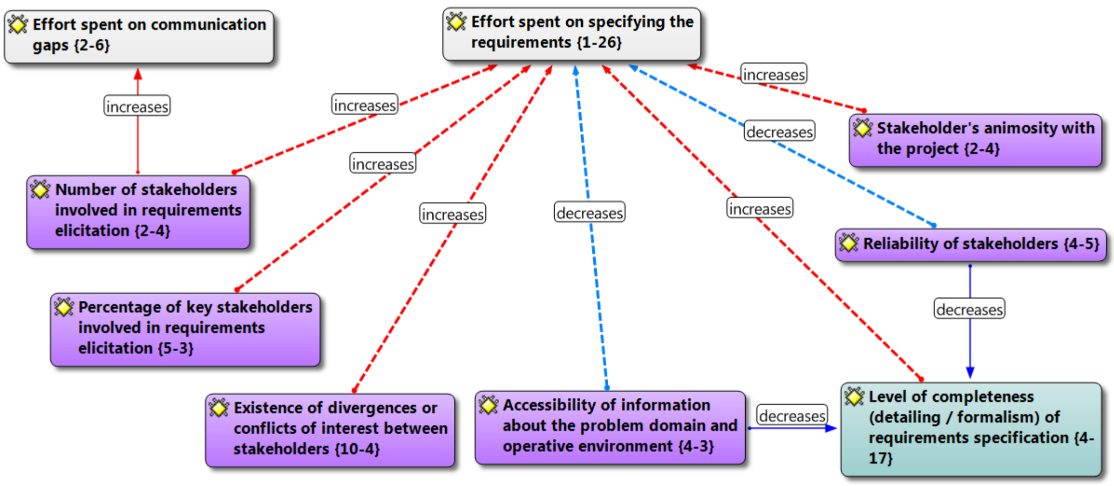


Figure 4-24 - The Stakeholders category (part 1)

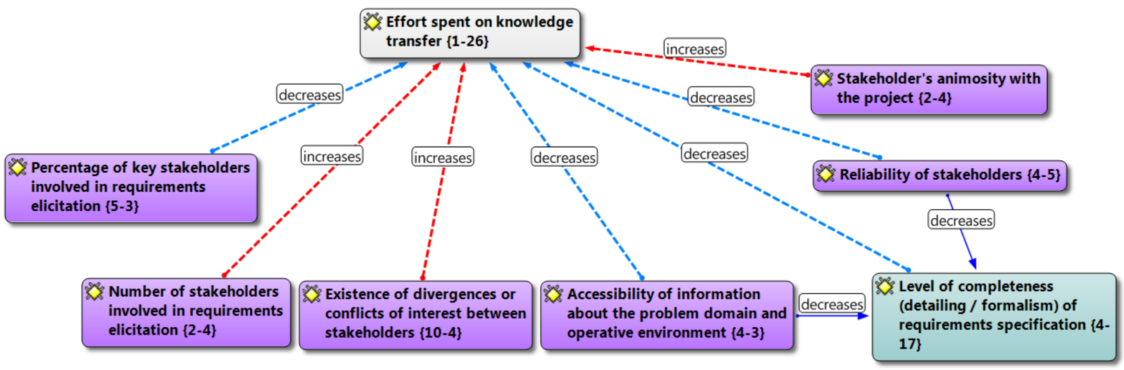


Figure 4-25 - The Stakeholders category (part 2)

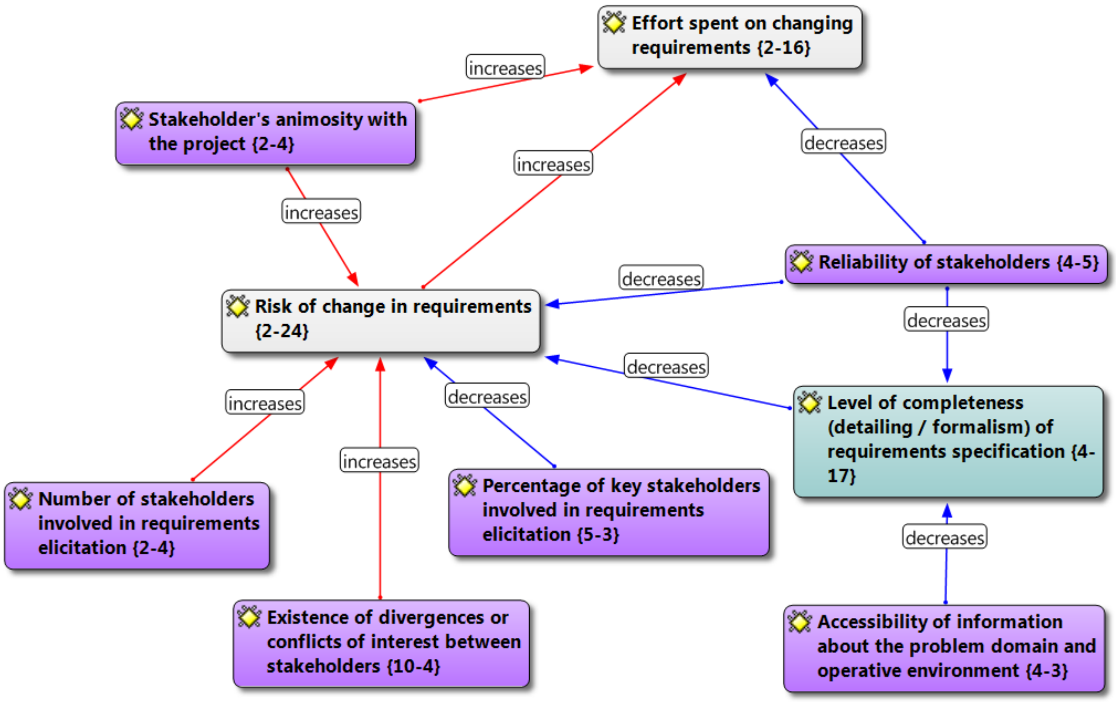


Figure 4-26 - The Stakeholders category (part 3)

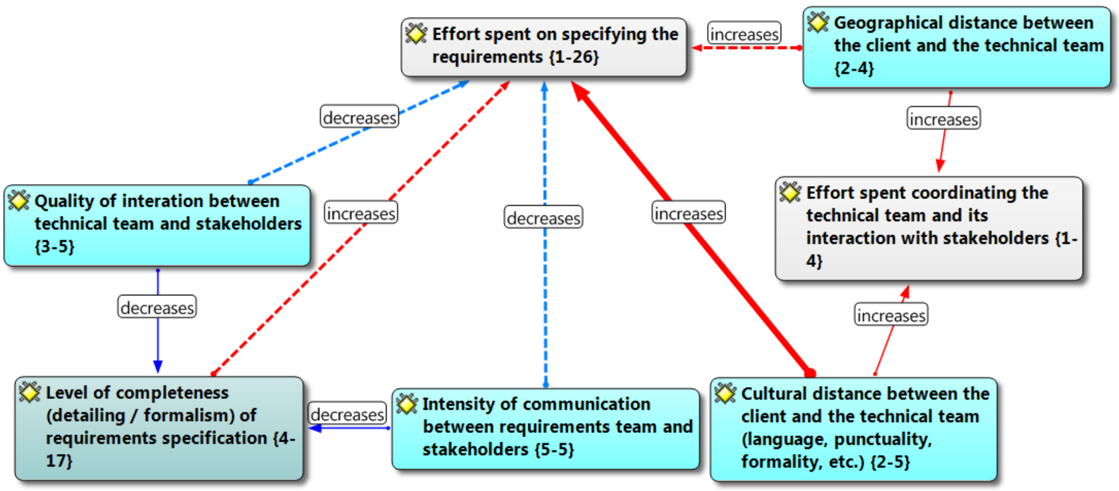


Figure 4-27 - Interaction between Stakeholders and Technical Team category (part 1)

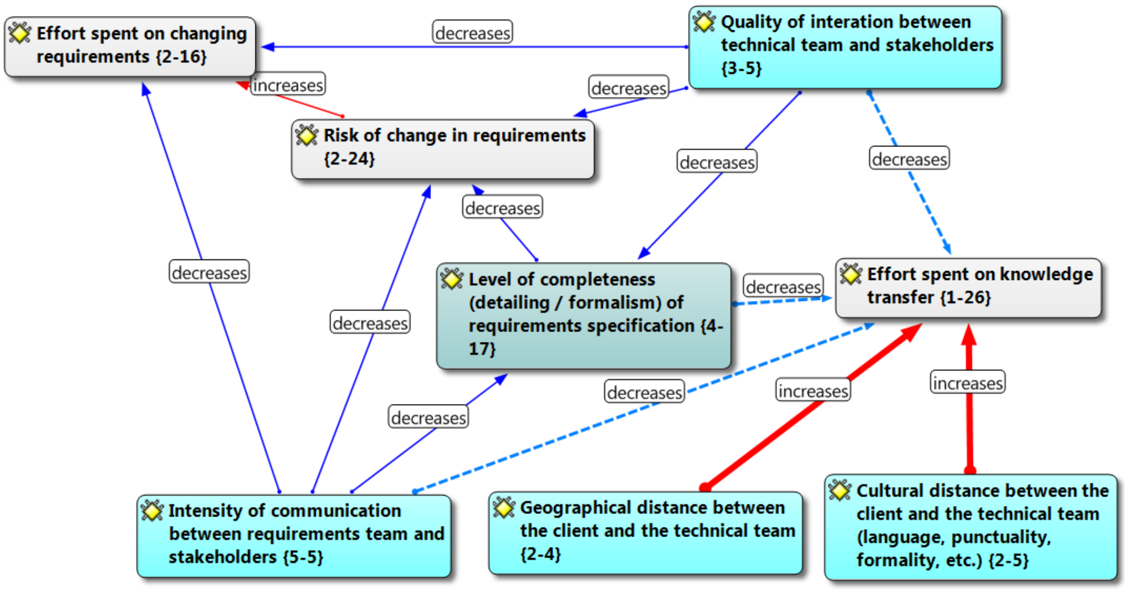


Figure 4-28 - Interaction between Stakeholders and Technical Team category (part 2)

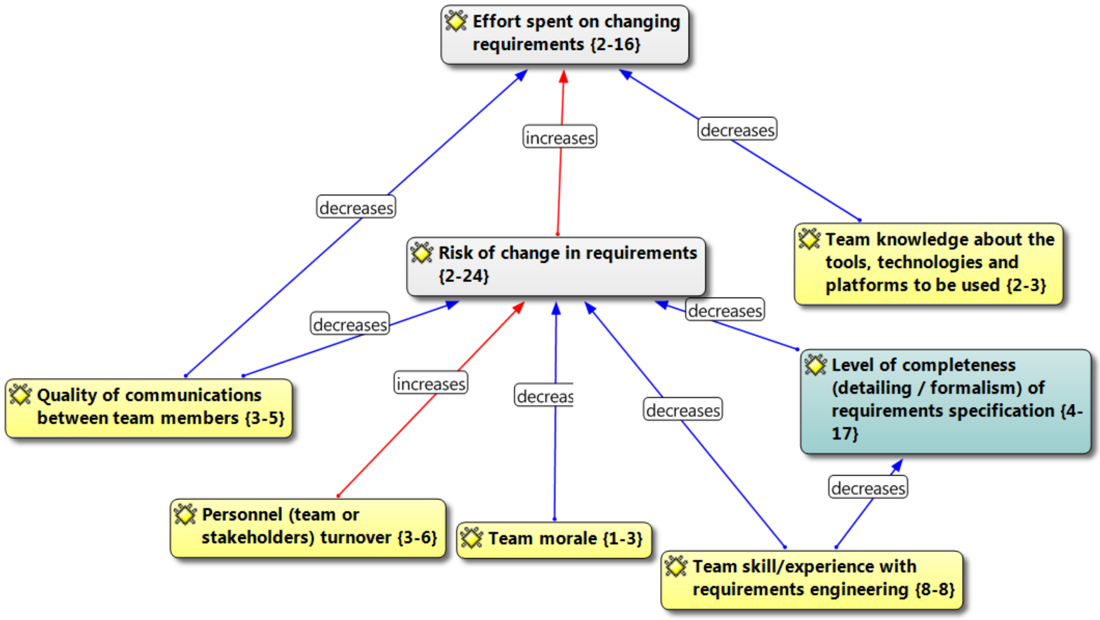


Figure 4-29 - The Technical Team category (part 1)

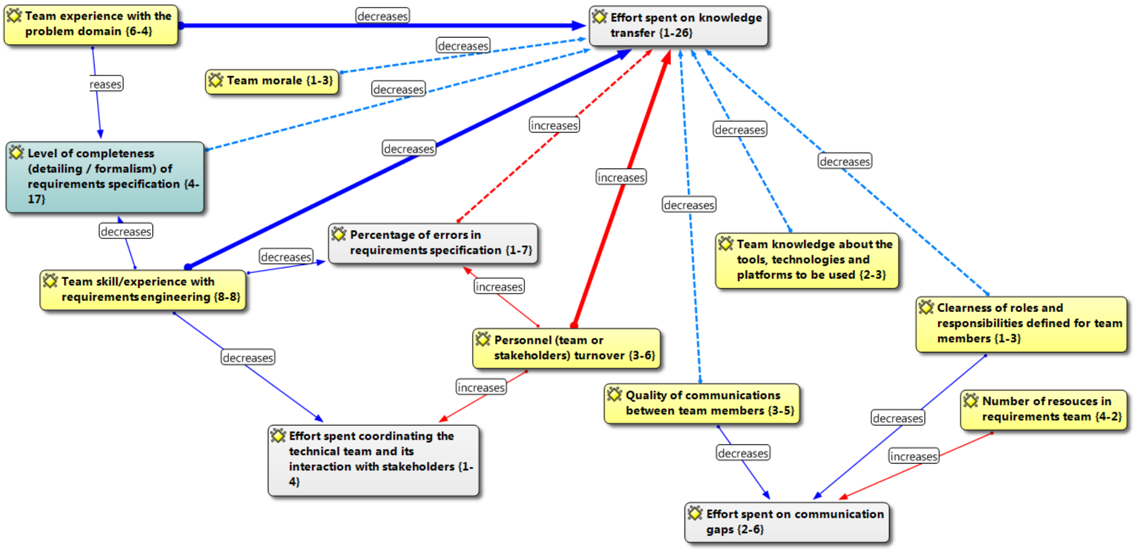


Figure 4-30 - The Technical Team category (part 2)

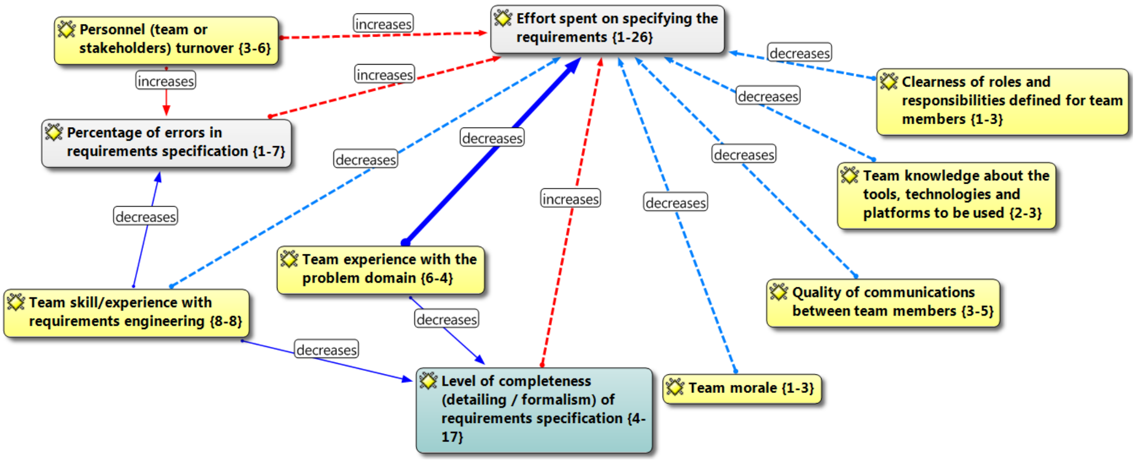


Figure 4-31 - The Technical Team category (part 3)

4.4 Chapter Conclusion

This chapter presents a detailed description of the large-scale survey conducted within the domain of this research. This survey allowed us to refine and supplement the results obtained by the application of Grounded Theory to analyze the results of our quasi-systematic literature review. The resulting conceptual maps of requirements effort influencing factors represent one of the first evidence-based discussions about software requirements projects effort. Although the estimation model is not ready, these results can be already useful to practitioners as it makes explicit which factors might be relevant to analyze, when a project manager needs to estimate the duration of a requirements project. The identified influence factors can be used, for instance, to characterize requirements project, making it possible to drive effort estimations by comparison with previous projects. It can also be used to support risk analysis and to refine the requirements process in such way that these factors can be identified and quantified earlier in the project.

The next chapter shows the conclusions and contributions of this work and points at research prospects.

CHAPTER 5 - CONCLUSIONS AND FUTURE WORK

This chapter provides a conclusion to this work, including some discussion of the results, and highlights its contribution to researchers and practitioners working/concerned with Requirements Engineering. The suggestions of future prospects provide indications of how the work related to the construction of a model to estimate effort for projects requirements can be extended.

5.1 Epilogue

This dissertation presented the results of our research on requirements effort estimation, detailing the steps that led to these results. It allowed us to identify which context variables influence the effort in requirements projects, and to characterize how each of these variables relate to each other, and how they relate to effort. We expect these results to be useful for practitioners and researchers working/concerned with requirements effort estimation.

5.2 Contributions

The main contributions of this work are:

- It organizes and summarizes the current work in the area, making the information easier to understand and to visualize;
- It defines the context variables that influence the effort involved in requirements projects. These could be useful for:
    - Supporting effort estimation: even if the estimation is done based on the experts' knowledge or by comparison with previous projects, this model can be useful as it helps the expert to identify the pertinent characteristics of a project that should be observed in order to support the estimation;
    - Supporting project risk analysis: the effort influence factors are, in essence, risk factors. For this reason, the results can be used as a checklist of risk factors that should be analysed when evaluating project's risk;

    - Refining requirements processes in such a way that these factors can be quantified earlier in the project.
- It provides an opportunity to treat the requirements stage as a real project, with a planned budget and schedule, instead of treating it as an unpredictable effort invested prior to project estimation. This is important, as one of the main concerns about the techniques to estimate software development effort is that they cannot be applied early in the project with acceptable accuracy. If we start treating the requirements stage as a pre-project, with its own budget and schedule, we create the opportunity to only estimate the development when we have fully detailed requirements specifications, and this might lead to better estimations;
- It makes available its *quasi*-systematic review protocol for other researchers to expand and replicate this research;
- It makes available its survey protocol and instrument for other researchers to expand and replicate this research.

5.3 Discussion of the Results

As a professional working with requirements project effort estimation for the past three years, I would like to expose some impressions on the results. Although the results seem adequate and match my personal experience, there are some problems or limitations I would like to suggest for discussions:

- There might be some overlap between effort components. For instance, components "Effort spent on knowledge transfer" and "Effort spent on communication gaps" may be considered as part of the Effort spent on requirements elicitation. This overlapping might overload the importance of the factors that affect all these components, as their influence on the overall effort will be considered repeatedly. Therefore we should consider revisiting the segmentation of such effort components, to carefully remove (or at least minimize) overlaps.
- There is usually a trade-off between the effort involved in building the initial requirements specification and the maintenance effort represented by the component. The most effort invested in building a more complete specification may lead to lesser effort spent on rework. These trade-offs are represented, for

example, in the relation of influence factor “Percentage of key stakeholders involved in requirements elicitation”: the more key stakeholders involved in requirements elicitation the more effort will be invested in interviews, workshops etc.; on the other hand, it increases the chance of a more complete understanding of the projects requirements, reducing the chance of future rework (Effort spent on changing requirements). This temporal aspect, however, is not very well captured by this model, as requirement changes may occur both during the stage of requirements building and the stage of requirements maintenance.

5.4 Future Work

The execution of this research led to the construction of a model that defines the relations between the project’s context variables and the effort involved in the requirements activities. These results open new perspectives for research that can be explored in future work. Some of these research prospects are detailed in this section.

5.4.1 Calibrating the model depending on context

As mentioned in Section 4.3.3, it is possible that the excluded factors might be relevant in some specific contexts, as we have evidence both from the literature and from some subjects who indicated their relevance. The contact information of the authors and survey participants that indicated the pertinence of these factors provides an opportunity to carry on with this investigation.

5.4.2 Supplementing the relations between influence factors and new effort components

The execution of the survey described in Chapter 4 led to the identification of four new requirements effort components: “Effort spent on requirements elicitation”, “Effort spent on managing requirements (tracking current status, traceability, and change management)”, “Effort spent on requirements analysis and modelling” and “Effort spent on requirements verification and validation”. We were not able, however, to find out how these components are affected by each influence factor. It would therefore be important to go back to the respondents of this survey to capture their understanding on these relations.

5.4.3 Quantify relations

In order to be able to estimate the amount of effort involved in a requirements project (for instance, in person-hours) it is necessary to quantify the influence exerted by each factor in requirements effort. To accomplish this arduous goal we should improve our understanding of how these factors could be measured.

Bibliographical References

AMBRÓSIO, B.G., BRAGA, J.L., RESENDE-FILHO, M.A., 2010. “Modelling and scenario simulation for decision support in management of requirements activities in software projects”. In: Journal of Software Maintenance and Evolution: Research and Practice, Volume 23 Issue 1. Pages 35-50.

BANDEIRA-DE-MELLO, R. and CUNHA, C.J. 2003. “Operacionalizando o Método da Grounded Theory nas Pesquisas em Estratégia: Técnicas e Procedimento de Análise com Apoio do Software Atlas/TI”. In: Encontro de Estudos em Estratégia (3Es), May 2003.

BANKS, S., LOUIE, E., EINERSON, M., 2000. “Constructing personal identities in holiday letters”, Journal of Social and Personal Relationships, 2000, Vol. 17(3): 299–327.

BASILI, V. and ROMBACH, D. 1988. "The tame project: towards improvement-oriented software environments.". In: IEEE Transactions on Software Engineering, 1988, v. 14, n. 6, pp. 758 – 773.

BIOLCHINI, J., MIAN, P., NATALI, A., CONTE, T., ; TRAVASSOS, G. H., 2007. “Scientific research ontology to support systematic review in software engineering”. In: Advanced Engineering Informatics, v. 21, p. 133-151, 2007.

BJARNASON, E., WNUK, K., REGNELL, B., 2011. “Requirements Are Slipping Through the Gaps - A Case Study on Causes & Effects of Communication Gaps in Large-Scale Software Development”. In: Proceedings of the 2011 IEEE 19th International Requirements Engineering Conference, p. 37-46, IEEE Computer Society Washington, DC, USA 2011.

BOEHM, B., 1981. "Software Engineering Economics". Prentice Hall PTR Upper Saddle River, NJ, USA. ISBN: 0138221227.

CONTE, T., CABRAL, R., TRAVASSOS, G.H., 2010. “Applying Grounded Theory in Qualitative Analysis of an Observational Study in Software Engineering – An Experience Report”, INFOCOMP (UFLA. Impresso), v. SE, p. 29-40.

DIBBERN, J., WINKLER, J. HEINZL, A., 2007. “Explaining Variations in Client Extra Costs between Software Projects Offshored to India”. In: MIS Quarterly Special Issue on Information Systems Offshoring, 2007.

DIESTE, O., GRIMÁN, A., JURISTO, N., 2009. “Developing search strategies for detecting relevant experiments”. In: Empirical Software Engineering, October 2009, v. 14, Issue 5, p. 513-539.

EBERT, C. and DE MAN, J., 2005. “Requirements Uncertainty: Influencing Factors and Concrete Improvements”. In: International Conference on Software Engineering, p. 553 – 560, ICSE 2005, Missouri, USA.

FERNÁNDEZ, D., WAGNER, S., LOCHMANN, K., BAUMANN, A., DE CARNE, H., 2011. “Field study on Requirements Engineering: Investigation of artefacts, project parameters, and execution strategies”. In: Information and Software Technology, v. 54 Issue 2, February, 2012, p. 162-178, Butterworth-Heinemann Newton, MA, USA.

GLASER, B., 1992. “Basics of grounded theory analysis”, Mill Valley CA, The Sociology Press.

IFPUG – International Function Point Users Group. “Function Point Counting Practices Manual – Release 4.3.1”. In: http://www.ifpug.org/publications/

JALALI, S. and WOHLIN, C., 2012. "Systematic Literature Studies: Database Searches vs. Backward Snowballing". In: Proceedings 6th International Symposium on Empirical Software Engineering and Measurement (ESEM), pp. 29-38, Lund, Sweden, September 2012.

JORGENSEN, M. AND SHEPPERD, M., 2007. "A Systematic Review of Software Development Cost Estimation Studies". In: Software Engineering, IEEE Transactions, v. 33 , pp. 33 - 53.

KITCHENHAM, B., 2004. Procedures for Performing Systematic Reviews. Joint Technical Report Keele University TR/SE-0401 and NICTA Technical Report 0400011T.1, Keele University and NICTA.

KITCHENHAM, B., DYBA, T., JORGENSEN, M., 2004. "Evidence-Based Software Engineering". In: Proceedings of the 26th International Conference on Software Engineering (ICSE 04), pp. 273-281,

MAFRA, S., BARCELOS, R., TRAVASSOS, G. H., 2006. "Aplicando uma Metodologia Baseada em Evidência na Definição de Novas Tecnologias de Software". In: Proceedings of the 20th Brazilian Symposium on Software Engineering (SBES 2006), v. 1, pp. 239 – 254, Florianopolis. October.

MALIK, A.A. and BOEHM, B.W., “An Empirical Study of Requirements Elaboration”, XXII Simpósio Brasileiro de Engenharia de Software – SBES 2008.

MALIK, A.A., KOOLMANOJWONG, S. BOEHM, B.W., 2009. "An Empirical Study of Requirements-to-Code Elaboration Factors". In: 24th International Forum on COCOMO and Systems/Software Cost Modeling, Cambridge, Massachusetts, November 2009.

MALIK, A.A., BOEHM, B.W., KU, Y. ,YANG, Y., 2010. "An Empirical Study of the Efficacy of COCOMO II Cost Drivers in Predicting a Project's Elaboration Profile", Canadian Society for the Study of Education, CSSE 2010.

MALIK, A.A. and BOEHM, B.W., 2010. "Comparative Analysis of Requirements Elaboration of an Industrial Product". In: 2nd International Conference on Software Technology and Engineering (ICSTE 2010), v1, p. V1-46 - V1-50, San Juan, PR, USA.

MAO, C., LU, Y., XI, W., 2005. "A Study on the Distribution and Cost Prediction of Requirements Changes in the Software Life-Cycle". In: International Software Process Workshop, SPW 2005, p. 136-150, Beijing, China.

MCGEE, S. and GREER, D., 2009. "A Software Requirements Change Source Taxonomy". 4th International Conference in Software Engineering Advances, p. 51 – 58, Porto, Portugal.

MCGEE, S. and GREER, D., 2011. "Software Requirements Change Taxonomy: Evaluation by Case Study". In: IEEE 19th International Requirements Engineering Conference, 2011, p. 25 – 34, Trento, Italia.

MELLO, R., TRAVASSOS, G. H., 2013. "Would Sociable Software Engineers Observe Better?". In: 7th International Symposium on Empirical Software Engineering (ESEM'13), p. 279 – 282, Baltimore, Maryland, USA.

MENDES, E., 2007. "A Comparison of Techniques for Web Effort Estimation". In: First International Symposium on Empirical Software Engineering and Measurement, ESEM 2007, p. 334-343, Madrid, Spain.

NURMULIANI, N., ZOWGHI, D., FOWELL, S., 2004. "Analysis of Requirements Volatility during Software Development Life Cycle". In: Australian Software Engineering Conference (ASWEC'04), p. 28 – 37, Melbourne, Australia.

PAI, M., MCCULLOCH, M., GORMAN, J. D., PAI, N., ENANORIA, W., KENNEDY, G., THARYAN, P., COLFORD JR., J. M., 2004. "Systematic reviews and meta-analyses: An shown, step-by-step guide". In: The National Medical Journal Of India, v. 17, n. 2, pp. 86-95.

TRAVASSOS, G. H., SANTOS, P. S. M. D., MIAN, P. G., DIAS NETO, A. C., BIOLCHINI, J. C. D. A., 2008. "An Environment to Support Large Scale Experimentation in Software Engineering". In: Proceedings of the 13th IEEE International Conference on Engineering of Complex Computer Systems (ICECCS 2008), p. 193-202. March 31 - April 03, 2008, Belfast, Northern Ireland.

STRAUSS, A. and CORBIN, J., 1998. “Basics of Qualitative Research: Techniques and Procedures for Developing Grounded Theory”, 2nd Ed. London, SAGE Publications.

SHULL, F., CARVER, J., TRAVASSOS, G. H., 2001. "An empirical methodology for introducing software processes." ACM SIGSOFT Software Engineering Notes, v. 26, n. 5, p. 288-296.

ZAVE, P., 1997. In: ACM Computing Surveys, v. 29, No. 4.

ZOWGHI, D. and NURMULIANI, N., 2002. “A Study of the Impact of Requirements Volatility on Software Project Performance”. In: Ninth Asia-Pacific Software Engineering Conference, 2002, p. 3 – 11, Gold Coast, Australia.

WÖHLIN, C., RUNESON, P., HÖST, M., OHLSSON, M. C., REGNELL, B., WESSL, A., 2000. Experimentation in Software Engineering: an introduction. Kluwer Academic Publishers.

Appendix 1 – Basic Information Extracted from Included Papers

Field	Description
Title	Requirements Are Slipping Through the Gaps - A Case Study on Causes & Effects of Communication Gaps in Large-Scale Software Development
Authors	Elizabeth Bjarnason, Krzysztof Wnuk and Björn Regnell Department of Computer Science, Lund University, Lund, Sweden E-mail: {Elizabeth.Bjarnason, Krzysztof.Wnuk, Bjorn.Regnell}@cs.lth.se
Year of Publication	2011
Publication Source	IEEE 19th International Requirements Engineering Conference
Abstract	Communication is essential for software development as its efficiency throughout the entire project life-cycle is a key factor in developing and releasing successful software products to the market. This paper reports on findings from an explanatory case study aiming at a deeper understanding of the causes and effects of communication gaps in a large-scale industrial set up. Based on an assumption of what causes gaps in communication of requirements and what effects such gaps have, a semi-structured interview study was performed with nine practitioners at a large market-driven software company. We found four main factors that affect the requirements communication, namely scale, temporal aspects, common views and decision structures. The results also show that communication gaps lead to failure to meet the customers' expectations, quality issues and wasted effort. An increased awareness of these factors is a help in identifying what to address to achieve a more efficient requirements management, and ultimately more efficient and successful software development. By closing the communication gaps the requirements may continue all the way through the project life-cycle and be more likely to result in software that meets the customers' expectations.

Field	Description
Title	Explaining Variations in Client Extra Costs between Software Projects Offshored to India
Authors	Jens Dibbern, Jessica Winkler, Armin Heinzl
Year of Publication	2007

Publication Source	MIS Quarterly Special Issue on Information Systems Offshoring
Abstract	Gaining economic benefits from substantially lower labour costs has been reported as a major reason for offshoring labour-intensive information systems (IS) services to low-wage countries; however, if wage differences are so high, why is there such a high level of variation in the economic success between offshored IS projects? This study argues that offshore outsourcing involves a number of extra costs for the client organization that account for the economic failure of offshore projects. The objective is to disaggregate these extra costs into its constituent parts and to explain why they differ between offshored software projects. The focus is set on software development and maintenance projects that are offshored to Indian vendors. A theoretical framework is developed a priori based on transaction cost economics (TCE) and the knowledge-based view of the firm, complemented by factors that acknowledge the specific offshore context. The framework is empirically explored using a multiple case study design including six offshored software projects in a large German Financial Service institution. The results of our analysis indicate that the client incurs post contractual extra costs for four types of activities: (1) requirements specification and design, (2) knowledge transfer, (3) control, and (4) coordination. In projects that require a high level of client-specific knowledge about idiosyncratic business processes and software systems, these extra costs were found to be substantially higher than in projects were more general knowledge was needed. Notably, these costs most often arose independently from the threat of opportunistic behaviour, challenging the predominant TCE logic of market failure. Rather, the client extra costs were particularly high in client-specific projects because the effort for managing the consequences of the knowledge asymmetries between client and vendor were particularly high in these projects. Prior experiences of the vendor with related client projects were found to reduce the level of extra costs but could not fully offset the increase in extra costs in highly client-specific projects. Moreover, cultural and geographic distance between client and vendor as well as personnel turnover were found to increase client extra costs. Slight evidence was found, however, that the cost increasing impact of these factors was also leveraged in projects with a high level of required client-specific knowledge (moderator effect).

Field	Description
Title	Modelling and scenario simulation for decision support in management of requirements activities in software projects
Authors	Bernardo Giori Ambrósio[1], José Luis Braga[2] and Moisés A. Resende-Filho[3] [1]Universidade Federal de Ouro Preto, João Monlevade, MG, Brazil [2]Universidade Federal de Viçosa, Viçosa, MG, Brazil [3]Universidade Federal de Juiz de Fora, Juiz de Fora, MG, Brazil
Year of Publication	2010
Publication Source	JOURNAL OF SOFTWARE MAINTENANCE AND EVOLUTION: RESEARCH AND PRACTICE
Abstract	There are many tools and techniques readily available to support the work in requirements activities of software development processes. As a consequence, the high frequency of errors still occurring in requirements activities suggests that the misunderstanding of the relationships among key decisions is the probable reason for this. The present work shows a system dynamics model constructed to make it possible for users to better understand the relations among key decision variables in requirements activities. The model was parameterized with data taken from previous studies and from a software development company so as to run two sets of simulations with three scenarios each. Optimistic, baseline and pessimistic scenarios are created on the basis of different assumptions regarding risk factors related to requirements volatility and people turnover. We used our simulation results to foresee the effects of these risk factors on the quality and cost of work in requirements activities. Up-to-date results from the Software Engineering literature strongly support the simulation outcomes obtained in our research. Copyright 2010 John Wiley & Sons, Ltd.

Field	Description
Title	Field study on Requirements Engineering: Investigation of artefacts, project parameters, and execution strategies
Authors	Daniel Mendéz Fernández [a], Stefan Wagner [b], Klaus Lochmann [a], Andrea Baumann [c], Holger de Carne [d] [a] Software & Systems Engineering, Institut für Informatik, Technische Universität München, Boltzmannstr. 3, 85748 Garching, Germany [b] Software Engineering Group, Institute of Software Technology, University of Stuttgart, Universitätsstr. 38, 70569 Stuttgart, Germany [c] Fakultät für Elektrotechnik und Technische Informatik, Universität der Bundeswehr München, Werner-Heisenberg-Weg 39, 85577 Munich, Germany [d] Capgemini Technology Service Deutschland, Carl-Wery-Str. 42, 81739 Munich, Germany
Year of Publication	2012
Publication Source	Information and Software Technology Journal
Abstract	Context: Requirements Engineering (RE) is a critical discipline mostly driven by uncertainty, as it is influenced by the customer domain or by the development process model used. Volatile project environments restrict the choice of methods and the decision about which artefacts to produce in RE. Objective: We aim to investigate RE processes in successful project environments to discover characteristics and strategies that allow us to elaborate RE tailoring approaches in the future. Method: We perform a field study on a set of projects at one company. First, we investigate by content analysis which RE artefacts were produced in each project and to what extent they were produced. Second, we perform qualitative analysis of semi-structured interviews to discover project parameters that relate to the produced artefacts. Third, we use cluster analysis to infer artefact patterns and probable RE execution strategies, which are the responses to specific project parameters. Fourth, we investigate by statistical tests the effort spent in each strategy in relation to the effort spent in change requests to evaluate the efficiency of execution strategies. Results: We identified three artefact patterns and corresponding execution strategies. Each strategy covers different project parameters that impact the creation of certain artefacts. The

	effort analysis shows that the strategies have no significant differences in their effort and efficiency. Conclusions: In contrast to our initial assumption that an increased effort in Requirements Engineering lowers the probability of change requests or project failures in general, our results show no statistically significant difference between the efficiency of the strategies. In addition, it turned out that many parameters considered as the main causes for project failures can be successfully handled. Hence, practitioners can apply the artefact patterns and related project parameters to tailor the RE process according to individual project characteristics.

Field	**Description**
Title	A Study on the Distribution and Cost Prediction of Requirements Changes in the Software Life-Cycle
Authors	Chengying Mao, Yansheng Lu, and Xi Wang College of Computer Science and Technology, Huazhong University of Science and Technology, 430074 Wuhan P.R. China maochy@yeah.net
Year of Publication	2005
Publication Source	International Software Process Workshop
Abstract	Software development is a dynamic process. Requirements change (RC) is inevitable and brings great challenges to the software development. How to precisely predict requirements change is especially important in the field of Requirements Engineering. In this paper, an assessment framework for the factors of RC distribution is constructed firstly. Apart from the rough pre- diction method based on the statistic process control of RCs, an artificial neural network method for predicting RC distribution is presented. In this case, the weight of each factor is calculated by a fuzzy logic method, called experts ranking. Furthermore, we propose a model to pre-evaluate the cost caused by RCs. With some practical projects data, a validation experiment has been drawn, whose result shows that our method and model are practical and efficient to predict the distribution and cost of RCs.

Field	**Description**
Title	A Study of the Impact of Requirements Volatility on Software Project Performance
Authors	Didar Zowghi, N Nurmuliani Faculty of Information Technology University of Technology,

	Sydney P O Box 123 Broadway NSW 2007, Australia {didar,nur}@it.uts.edu.au
Year of Publication	2002
Publication Source	Proceedings of the 9th Asia-Pacific Software Engineering Conference
Abstract	Software development is considered to be a dynamic process where demands for changes seem to be inevitable. Modifications to software are prompted by all kinds of changes including changes to the requirements. This type of changes gives rise to an intrinsic volatility, which has several impacts on the software development lifecycle. This paper describes our findings of an extensive survey based empirical study of requirement volatility (RV) and its impact on software project performance. In particular, findings reveal that requirement volatility has a significant impact on schedule overrun and cost overrun in software projects. Our investigation also examined factors that contribute to the extent of requirement volatility and found that variables such as frequent communications between users and developers and usage of a definable methodology in requirements analysis and modelling have impact on the stability of requirements.

Field	**Description**
Title	A Software Requirements Change Source Taxonomy
Authors	Sharon McGee[1] and Des Greer[2] School of Electronics, Electrical Engineering and Computer Science Queens University Belfast, United Kingdom {[1]smcgee08|[2]des.greer}@qub.ac.uk
Year of Publication	2009
Publication Source	Fourth International Conference on Software Engineering Advances
Abstract	Requirements changes during software development pose a risk to cost, schedule and quality while at the same time providing an opportunity to add value. Provision of a generic change source taxonomy which makes the distinction between factors contributing to requirements uncertainty and events that trigger change will support requirements change risk visibility, and also facilitate richer recording of change data. In this paper we present a collaborative study to investigate and support the management of software requirements volatility within the development lifecycle. Previously published change `causes' are elicited from the literature, consolidated using expert knowledge and classified using card sorting. The resulting change trigger

	taxonomy constructs were initially validated using a small set of requirements change data, and deemed sufficient and practical as a means to collect common requirements change source statistics across multiple projects.

Field	Description
Title	Software Requirements Change Taxonomy: Evaluation by Case Study
Authors	Sharon McGee[1] and Des Greer[2] School of Electronics, Electrical Engineering and Computer Science Queens University Belfast, United Kingdom {[1]smcgee08|[2]des.greer}@qub.ac.uk
Year of Publication	2011
Publication Source	IEEE 19th International Requirements Engineering Conference
Abstract	Although a number of requirements change classifications have been proposed in the literature, there is no empirical assessment of their practical value in terms of their capacity to inform change monitoring and management. This paper describes an investigation of the informative efficacy of a taxonomy of requirements change sources which distinguishes between changes arising from ‘market’, ‘organisation’, ‘project vision’, ‘specification’ and ‘solution’. This investigation was effected through a case study where change data was recorded over a 16 month period covering the development lifecycle of a government sector software application. While an insufficiency of data precluded an investigation of changes produced by the change source of ‘market’, for the remainder of the change sources res ults indicate a significant difference in cost, value to the customer, and management considerations. Findings show that higher cost and value changes arose more often from ‘organ isation’ and ‘vision’ sources; these changes also generally involved the cooperation of more stakeholder groups and were considered to be less controllable than changes arising from the ‘specification’ or ‘solution‘ sources. Overall, the results suggest that monitoring and measuring change using this classification is a practical means to support change management, understanding and risk visibility.

Field	Description
Title	Requirements uncertainty: influencing factors and concrete improvements
Authors	Ebert, C.[1] and De Man, J.[2] [1]Alcatel 54, rue la Alcatel christof.ebert@alcatel.com [2]Francis Wellesplein 1, B-2018 Antwerp Also: Ghent University Boetie F-75008 Paris jozef.de_man@alcatel.be
Year of Publication	2005
Publication Source	27th International Conference on Software Engineering, 2005
Abstract	Practically all industry studies on software project results conclude that good Requirements Engineering plays a pivotal role for successful projects. A key reason for project failures is insufficient management of changing requirements during all stages of the project life cycle. This article investigates one of the root causes for changing requirements, namely requirements uncertainty. In an experimental field study we looked into four underlying drivers for requirements uncertainty. We found several techniques must be used simultaneously to see tangible success. Using only one such technique in isolation doesn't make a difference. The field study is supported by extensive data from well over 200 projects stemming from very different business areas of Alcatel over a period of two years. Results are presented with practical experiences to allow effective transfer.

Field	Description
Title	Analysis of Requirements Volatility during Software Development Life Cycle
Authors	N Nurmuliani, Didar Zowghi, Sue Fowell Faculty of Information Technology University of Technology, Sydney P.O. Box 123 Broadway NSW 2007, Australia {nur, didar, sfowell}@it.uts.edu.au
Year of Publication	2004
Publication Source	Proceedings of the 2004 Australian Software Engineering Conference
Abstract	Investigating the factors that drive requirements change is an important prerequisite for understanding the nature of requirements volatility. This increased understanding will improve the process of requirements change management. This paper mainly focuses on change analysis to identify and characterize the causes of requirements volatility. We apply a causal analysis method on change request data to develop a

	taxonomy of change. This taxonomy allows us to identify and trace the problems, reasons and sources of changes. Adopting an industrial case study approach, our findings reveal that the main causes of requirements volatility were changes in customer needs (or market demands), developers' increased understanding of the products, and changes in the organization policy. During the development process, we also examined the extent of requirements volatility and discovered that the rate of volatility was high at the time of requirements specification completion and while functional specification reviews were conducted.

Appendix 2 – Initial Set of Effort Components

Effort Components
Effort spent coordinating the technical team and its interaction with stakeholders.
Effort spent on changing requirements.
Effort spent on communication gaps.
Effort spent on control of team performance.
Effort spent on knowledge transfer.
Effort spent on requirements specification.

Appendix 3 – Initial Set of Influence Factors

Category	Influence Factor
Environment for Requirements Engineering	Availability of a defined process (methods, techniques, tools, activities, roles and artefacts) for Requirements Engineering
	Availability of process and configuration management tool for controlling requirements changes
	Percentage of effort that must be allocated to quality assurance in relation to the specification effort
	Reuse of existing artefacts (templates, requirements of other projects, etc.)
Interaction between Stakeholders and the Technical Team	Cultural distance between the client and the technical team (language, punctuality, formality, etc.)
	Geographical distance between the client and the technical team
	Intensity of communication between requirements team and stakeholders
	Quality of interaction between technical team and stakeholders
Problem Domain	Degree of understanding of the product vision
	Natural complexity of problem domain
	Stability of business environment
Project	Feasibility of concluding the project in time and on budget
	Loose schedule (slack time available)
	Project duration
	Time pressure and risk of schedule overrun
Solution Domain	Conceptual compliance between software requirements and the real world
	Existence of design patterns to be met
	Level of completeness (detailing / formalism) of requirements specification

	Novelty of application
	Percentage of errors in requirements specification
	Risk of change in requirements
	Solution size and complexity
Stakeholders	Accessibility of information about the problem domain and operative environment
	Existence of divergences or conflicts of interest between stakeholders
	Number of stakeholders involved in requirements elicitation
	Percentage of key stakeholders involved in requirements elicitation
	Reliability of stakeholders
	Stakeholder's animosity with the project
	Stakeholder's demand with respect to the formalism and level of detail of the artefacts to be generated.
	Stakeholder understanding of the techniques and technologies to be used (ability to understand the specifications, diagrams, etc.)
Technical Team	Average team productivity
	Clarity of roles defined for team members
	Degree of distribution of the technical team
	Familiarity of the team with the technologies, tools and platforms to be used
	Number of resources in requirements team
	Personnel turnover
	Quality of communications between team members
	Team morale
	Team skill/experience
	Team understanding about the problem domain