Requirements Engineering Practices in Very Small Software Enterprises: A Diagnostic Study

Alcides Quispe, Maira Marques, Luis Silvestre, Sergio F. Ochoa, Romain Robbes
Department of Computer Science
Universidad de Chile
Santiago, Chile
{aquispe, mmarques, lsilvest, sochoa, rrobbes}@dcc.uchile.cl

***Abstract*—Requirements engineering practices have been identified as a key issue that affects the success rate of projects in most software organizations. The software engineering community has studied the requirements engineering practices of medium and large-sized organizations extensively, and has produced interesting and suitable solutions. However, several software engineering researchers have shown that most current requirements engineering practices are unsuitable for small and very small software companies. They have also highlighted that there is a lack of knowledge about the requirements engineering practices in these types of companies. This article presents the results of a diagnostic study the authors are performing in very small software companies in Chile. The study tries to identify the state of the practice in this niche and also the potential limitations to adopt appropriate requirements engineering practices in Chilean very small software enterprises.**

Keywords-Requirements Engineering Practices, Diagnostic Study, Very Small Software Enterprises

I. INTRODUCTION

In Latin America, as elsewhere in the world, very small software enterprises (VSSE)—those having fewer than 10 developers [9, 6]—represent a large part of the software industry. In Chile, around 44% of software companies have less than 10 employees [6]; in Canada (Montreal area) 50% of companies are VSSEs [21]; and in the USA this category includes around 78% of software companies [10].

VSSEs have a number of characteristics that distinguish them from larger ones:

a) Project & team size: VSSEs work on small projects (<6 months); their development teams typically range from 3 to 10 persons [1, 34, 43, 37].

b) Resources: They count on scarce resources (human, technological & economical) [1, 7, 4, 35]. Economical issues may be the most critical as they are driven by cash-flow and depend on projects profit [20].

c) Staff quality: The staff has a low level of expertise and training. VSSEs have limitations to hire highly qualified and experienced professionals [25, 7, 33].

d) Process development: Their processes are typically informal and rather immature [2, 41, 40, 15].

e) Project management: VSSEs exhibit high informality in planning, organizing, directing, monitoring and controlling projects [2, 26, 33]

f) Organizational structure: VSSEs have an informal organizational structure [39] with vaguely defined roles and responsibilities [33, 26, 15]. Team members usually play multiple roles [33, 15].

g) Communication and coordination: In the Chilean scenario, most team members usually work in a distributed, asynchronous setting with little time dedicated to the project [33].

Projects performed by VSSE often last longer than planned. After interviewing 70 very small software companies in Austria, Hofer [13] reported that projects were over schedule in nearly 50% of the companies.

Although there are several factors causing low project success rates, the use of poor requirements engineering (RE) practices has been blamed as one of the major factor jeopardizing the success of software projects [36, 14, 42, 19]. It has also been recognized that appropriate RE practices contribute to the success of software projects. Supporting this idea, Nuseibeh and Easterbrook state that gathering and managing requirements properly are key factors to increase the success rate of software projects [32, 24, 8]. Therefore, there is a consensus around the idea that RE practices play an important role in the success and failure of software projects.

However, improving RE practices requires identifying areas of improvement in the current RE process of an organization. The solution will be a particular recipe for each case, as one-size-fits-all approaches do not work in this scenario.

This paper presents the results of a diagnostic study we are performing in Chilean VSSEs. The study consists of a survey and a focus group that were periodically conducted with experienced project managers of VSSEs. Our study intends to obtain a general diagnosis about the state of the RE practices in VSSE. A further goal is to identify common areas of improvement in order to help focus the research efforts on these issues.

II. CONSEQUENCES OF POOR REQUIREMENTS ENGINEERING PRACTICES

Poor RE practices can cause severe problems to a software project. Some of the common problems are briefly described below.

a) Rework. The major consequence of requirements issues is rework: as changes to the requirements become

apparent, parts of the system have to be updated to reflect the changes in the requirements [5, 42]. These changes can have a direct impact in other parts of the project, causing major delays.

b) Communication and coordination problems. Analysts usually manage requirements using multiple resources; e.g. text documents, spreadsheets, presentation slides, or even e-mails messages [14, 16, 11]. Therefore it becomes difficult to get fast, on-time and accurate information on requirements, resulting in serious communication and coordination issues.

c) Poor visibility of the project status. Many projects lack even the simplest requirements-related metrics to help steer the project towards successful completion, avoid rework, control scope, or manage change during the project [3]. Poor visibility of the current status of a project pushes the project manager to make decisions based on uncertainty.

All of the problems mentioned above have a negative impact in the success rate of a software project performed by VSSE. In order to know which the status of the Chilean VSSE is, the authors have been conducting a diagnostic study with two goals: (1) assessing the state of the practice in RE for local software companies, and (2) identifying the main issues hampering the adoption of appropriate RE practices in Chilean VSSE.

III. Related Work

The study of software engineering practices—and particularly RE practices—in VSSEs is quite new [17, 1]. In 2007, Aranda and Easterbrook studied the RE practices of seven VSSE in Canada [1]. The exploratory study characterizes each software company and reports the results of the RE practices used by these organizations. The preliminary results of the study were that: (1) RE practices in successful VSSE are diverse and work well for the organizations where they were applied; (2) all the companies studied had a strong cultural cohesion; (3) experienced persons were always in charge of the RE process; and (4) requirements errors for these companies were rarely catastrophic. Although these preliminary results are very interesting, the study involved only seven VSSE and all of them were stable (or consolidated) software companies.

After an exhaustive review we have not found similar studies related to RE practices in VSSE. However, several researchers have studied the RE practices used by VSSE in particular work scenarios.

In 2000, Nikula et al [30] presented the results of a study of the current RE practices, development needs and preferred ways of technology transfer of twelve small to medium sized companies in Finland. The study reports the level of adoption (standard, normal, discrete, never) for several RE practices: documentation style, RM methods, and general guidelines for RE practices. Although the study involves small and medium sized companies, it doesn't focus on VSSEs, as only four companies (25%) were VSSEs. Surprisingly, the smaller company (with 5 people) got a score of 0 out of a maximum of 30 points. Referring to this result the author state that: "the survey questions were clearly inappropriate" for this small company. Therefore, the results of this work do not constitute relevant data on the RE practices of VSSEs.

The results of Nikula's research report were used by the authors as a source of RE data for subsequent research [31, 29, 27, 28]. All of this research effort was done toward developing a ready-to-use method for small software companies. In 2004, Nikula presented a basic RE method (BaRE) as part of his Ph.D. thesis [28]. The BaRE method was developed to provide an easy to adopt way to introduce basic systematic RE practices in small and low maturity organizations. The BaRE method includes techniques for requirements development and requirements management whose detailed description can be found in the BaRE Guide. These techniques are based on standard RE techniques explained in detail in the RE literature. Despite the author stating that the BaRE method was developed for small companies, it appears the BaRE method was not developed for the specific needs and characteristics of VSSEs. We arrive to this conclusion as the research work was based on the results of a previous author's work [30]. Therefore, the techniques for requirements development and requirements management presented in BaRE method may not be suitable for VSSEs.

Dorr et al. [7] present a set of 36 RE practices as part of their RE process improvement for small software enterprises called ReqMan. The set of RE practices are classified by requirements phase (elicitation, analysis, specification and verification/validation) and their importance to the RE process (basic practices, advanced practices, optimizing practices and context dependent practices). Despite the positive experience reported by authors on using the set of 36 RE practices as part of the ReqMan model, the research work presents two limitations: (1) only six companies were involved in the study and (2) the size of the companies ranged from 20 to 200 employees. Although the study refers to small companies, VSSEs were not considered.

Recently, Sami Jantunen [15] presented the results of an exploration of software engineering practices in five small and medium-sized organizations. Despite the research work not focusing on RE practices, the study reveals interesting issues about software development practices in small organizations: "the work is done rather informally, heavily relying on collaboration". We believe collaboration issues are important to understand RE practices as RE is a communication-intensive activity. Although the work refers to small and medium-sized organizations, the size of the five companies under study was not mentioned.

Overall, we can observe a dearth of interest in studying the adoption of RE practices for VSSEs, with only Aranda and Easterbrook reporting findings specific to VSSEs. Given the large proportion of companies that are VSSEs, we decided to perform the preliminary study described below as first step towards better understanding the dynamics of RE for VSSEs.

IV. STUDY DESCRIPTION

Twenty four experimented project managers, from 24 different companies, participated in our study so far; we are continuously adding respondents, subject to their availability. All the companies are located in Santiago. We focus on project managers, as they usually have the most complete high-level view of the projects.

Information gathering was performed through two instruments: a survey and a focus group. The survey form is a Likert scale, including 48 questions that are answered according to the Likert response format [23]. Respondents also participated in a hour-long focus group aimed at identifying through discussions in small groups the common problems affecting the software development process performed by VSSE. In this paper, we discuss seven selected findings extracted from the survey.

V. FINDINGS

Our current findings indicate that: (1) Project specifications are usually met, but the client often finds the solution unsatisfactory; this leads to the conclusion that (2) communication issues with clients cause incomplete specifications; (3) the project's scope expands as clients require additional changes, often with inadequate changes; (4) Requirement specification in VSSEs is mostly an ad-hoc process; (5) this ad-hoc process leads to requirement management issues such as loss of requirements; (6) when uncertainty arise, developers tend to resolve the issue without contacting the clients; and (7) VSSEs are aware of the benefits of RE practices but are not sure they apply in their context. We now detail these findings one by one.

A. Meeting Users Needs

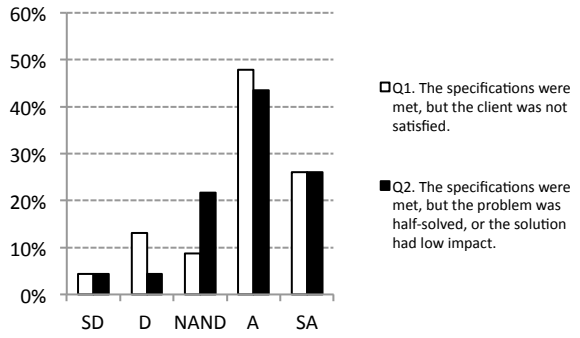

Figure 1. Meeting Users Needs.

Our first finding concerns whether the projects that are delivered actually meet the needs of the customer, even if the specifications, as dictated by the customer and interpreted by the developers, are met.

Item Q1 shows that the majority of the participants (nearly 75%) develop products that end up not satisfying their customers. Further, according to Item Q2, around 70% of the participants report that their products are not resolving the real problem of the customer. Clearly, this situation shows that for our sample, the participants are failing to meet the real needs of the users.

It appears (from Item Q1 and Item Q2) that developers are not having problems in achieving the specification of the product, suggesting that the actual specifications were incorrect. Hence it seems VSSEs do not struggle with implementing a solution, but are rather having difficulties to discover the real problems of the client.

B. Problem identification

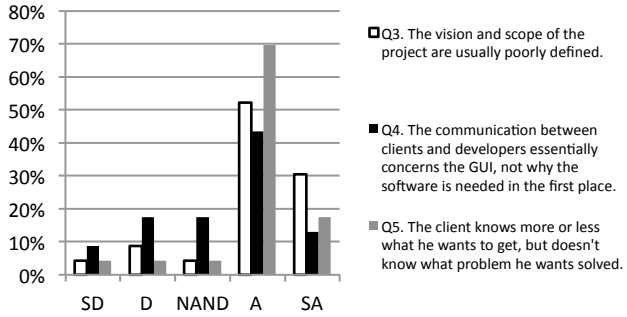

Figure 2. Problem identification.

Given that the recording of the specifications seems to be the issue, we investigated the answers of our respondents with respect to the definition of the problem.

Item Q5 shows that almost all participants (nearly 90%) have to deal with customers who do not know clearly the problem they want to address with the final software product. Despite this, it is surprising that there is no effort from developers towards the identification of the real problem of the customer. Item Q4 shows that nearly 60% of the participants report that their communications with clients focus mostly on discussion centering on the user interface of the delivered product. As the problem is not well defined by the customers, a strong majority—more than 80%—of the participants work on software projects where the vision and scope are usually poorly defined, as shown by Item Q3. These suggest that managers should steer discussions towards a more accurate description of the problem the client is facing, relegating UI issues to later phases such as prototype demonstrations.

C. Scope Creep

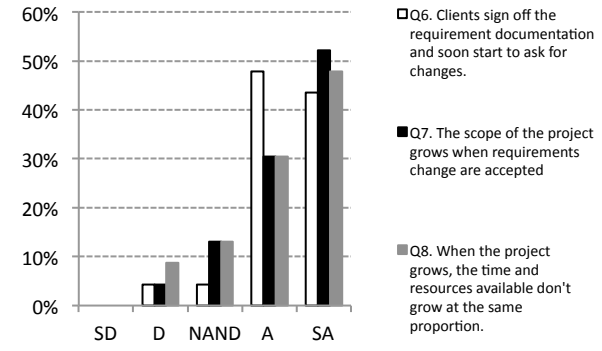

Figure 3. Scope Creep.

When the scope of a project is poorly defined, it has a tendency to expand. Changes to requirements are unfortunately in the nature of software engineering [22]; it has to be controlled and tracked. Change management is necessary to provide facilities for controlling change, so that the consistency of the various components of the application is preserved after each change. Large software companies have dedicated teams whose only concern is software maintenance and evolution. VSSEs do not have that luxury; the development team is also responsible for handling incoming change requests.

Our data shows evidence of scope creep in the experience of our respondents: Item Q6 shows that customers often ask for changes while development has already started. Customers are often eager to have the project delivered and often sign off the requirements document without having done a proper analysis. When early builds of the software are to be delivered or demonstrated, the clients start to realize that the product being built it is not exactly what they want. Another factor is that new ideas emerge—unforeseen requirements—during development. These factors cause scope creep, as demonstrated by Item Q7. The scope of the project grows even more when unforeseen requirements are included. Scope creep is considered one of the major risks in software projects [38], as the clients request changes but do not provide enough additional resources to finish the project on time, despite requesting the changes (Item Q8).

In the next three sections, we detail our findings on the three steps of requirements specification, management, and validation.

D. Requirements Management

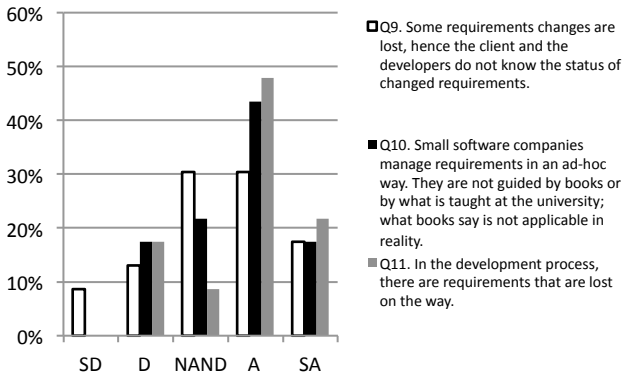

Figure 4. Requirements Management.

Keeping track of all the requirements is necessary to meet specifications, especially in the face of changing requirements as shown above. Our findings show that managing the requirements is not a common strength of VSSEs, especially once development starts.

Item Q10 shows that most respondents think that an ad-hoc management process is sufficient, as we have seen previously. Despite this, Item Q11 shows that a proportion of the requirements end up lost. The situation is not as well marked for changes to actual requirements, but this phenomenon still occurs (Item Q9). As such, they often end up with requirements scattered in different places (Item Q11). This is not without consequences; we discuss this in the next finding.

According to Kautz, [18] one of the major problems of VSSE is the lack of metrics measuring the development process; the tracking of change requests is one of the five metrics VSSE should consider adopting. We certainly concur with this opinion, as our data shows that a fair proportion of our respondents occasionally lose track of some of their requirements.

E. Requirements Specification

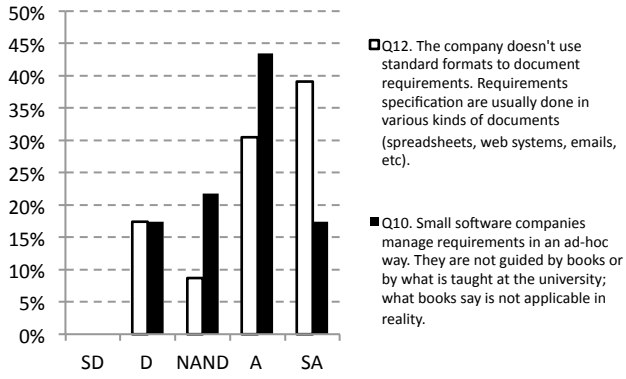

Figure 5. Requirements Specification.

The majority of the companies who participated on this study do not have a structured way of doing requirements specification—adopting an ad-hoc process instead—, despite the literature available on the subject, or the experience of other companies. Item Q10 shows that they feel the courses and the books do not apply to the context of small companies.

According to studies by Graaf et. al. [12] in Europe, VSSE often sees requirements specification, along with design specification, as a way of avoiding additional analyses which would increase the time needed for documentation. This study additionally found that some companies have templates and guidelines for requirements specification, which are unfortunately not used in practice. Our findings concur with the observations of Graaf et. al.

F. Requirements Validation

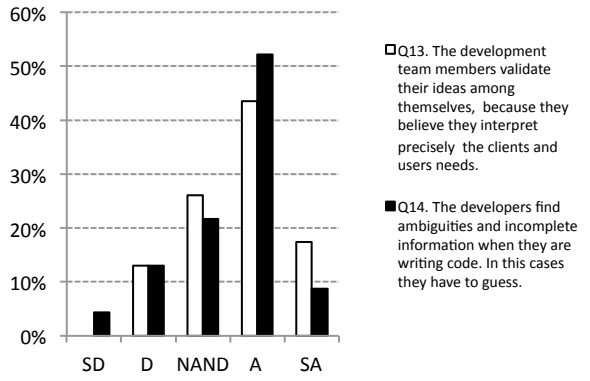

Figure 6. Requirements Validation.

Development starts once the requirements are validated. It is however common those uncertainties appear when developers are building the actual solution, as requirements might in fact not have been specified well enough. Item Q14 shows that these uncertainties happen in practice, making one think that the requirement validation process was hastily performed. This rejoins Item Q6 in finding C; once the customers have been delivered the requirements, they tend to sign off quickly, hoping to hasten product delivery.

If a requirement is unclear, the optimal course of action is to consult the customer and ask for clarifications. However, Item Q14 shows that developers often resort to guesswork instead. This is confirmed by the results of Item Q13, where we see that developers validate their ideas primarily with team-mates, not the customer. These findings reinforce the idea that communication with customers tends to be an issue, as we have seen previously with Items Q4 and Q6 in findings B and C.

Overall, small businesses do not have a validation process compatible with their needs. During the development phase, VSSEs resort to inadequate methods of validating project requirements, due to the short time they devote to validation with the client (Item Q6), and the limited time the client spends answering the developers' request for clarifications—developers often interpret requirements by themselves.

G. About RE practices

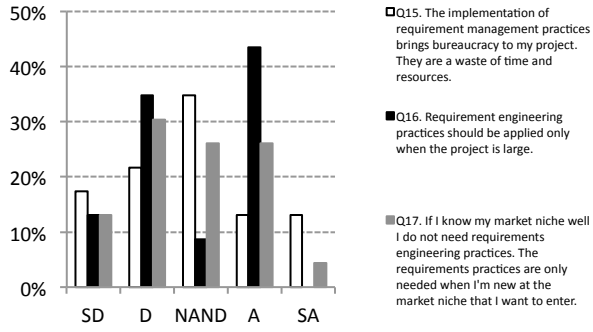

Figure 7. About RE practices

In this finding, we study the general perception of requirement engineering practices in VSSEs. Small businesses are aware of the need and benefit that software engineering practices bring, but do not reach a consensus on whether these practices—known to be beneficial for large companies—are applicable in their context.

Items Q15, Q16 and Q17 show that overall, practitioners are aware of the need of RE practices, even if a minority remains somewhat doubtful. Only a small minority believe that introducing RE practices incur a large overhead. On the other hand, a strong majority (more than 40%), think that RE practices are only really needed in large projects (Item Q16). Our previous findings show that this is not the case, since a lot of projects—even small projects as done by VSSEs—suffer from scope creep, as well as requirement specification, management, and validation issues. These findings relate also to Item Q10 seen previously, in which we found that a majority of the companies used ad-hoc RE practices in an effort to find a compromise between having full RE practices, or not at all. As we have seen before, these ad-hoc practices may cause issues such as loss of requirements.

Finally—according to Item Q17—the belief that niche products do not need RE practices maintains a somewhat strong (nearly 30%) following. The idea that an extensive domain knowledge might substitute for a part of the initial RE process is attractive, but may provide a sense of false security. It remains to be seen if these companies developed frameworks and/or product lines to exploit their extensive domain knowledge of the area.

VI. Conclusions and Further Work

The state of requirement engineering practices in very small software enterprises (VSSEs) has still been largely unexplored. This paper presented initial results of a study on the RE practices of these companies, based on a Likert scale questionnaire (composed of 48 5-points Likert items) and a series of focus groups. In each case, 24 managers of 24 different VSSEs participated so far; we are continuously adding new respondents. Initial results based on an analysis of some of the Likert items points at two major issues to address: (1) communication between clients and companies is lacking and does not focus on the right issues, yielding imperfect specification, scope creep, and ultimately dissatisfaction with the project; and (2) VSSEs use ad-hoc RE practices—which compounds the earlier issue as requirements are hard to track and may be lost—, out of an impression that RE practices, albeit important, are ill-suited for organizations of their size. All this points to possible solutions that need some investigation: (1) assessing the benefits of practices aimed at improving the communication between the companies and the customers, such as the eXtreme Programming practices of having an onsite customer; and (2) tailoring RE practices for a better acceptance by VSSEs and alleviating their concerns that RE practices are ill-suited for them. Needless to say, additional studies are needed to assess the generality of these findings. We will continue our analysis of RE practices by inspecting the remainders of the items. Likewise, we will inspect the

results of the focus group discussions in order to gather more qualitative data about the experience of VSSEs in with RE practices.

ACKNOWLEDGMENT

The work of Alcides Quispe has been supported by the Computer Science Department, through the NIC Chile scholarship.

REFERENCES

[1] J. Aranda, S. Easterbrook, G. Wilson. “Requirements in the wild: How small companies do it”. Proceedings of the 15th IEEE Requirements Engineering Conference, pp. 39–48, Oct. 15–19, 2007.

[2] J. Batista, A. D. De Figueiredo. “SPI in a Very Small Team: a Case with CMM”, Software Process: Improvement and Practice Journal, Vol. 5, No. 4, pp. 243-250, 2000.

[3] Borland Software Corporation. “Basic Metrics for requirements management”. White Paper, Mar. 2004. URL: http://www.borland.com/resources/en/pdf/white_papers/basic_metrics_for_requirements_management_wp_0125js.pdf

[4] K. Coleman, P. Larsen, M. Shaw, M.V. Zelkowitz. “Software Process Improvement in Small Organizations: A Case Study”. IEEE Software, Vol. 22, No. 6, pp. 68-75, Nov./Dec. 2005.

[5] R.N. Charette. “Why Software Fails”. IEEE Spectrum, Vol. 49, No 9, pp. 42, 2005.

[6] CHILE INNOVA Program. “Diagnose of the Information Technologies Industry in Chile” (in Spanish), White Paper, 2003, URL:http://www.estrategiadigital.gob.cl/files/DIAGNOSTICO%20INDUSTRIA%20TIC%202003.pdf

[7] J. Dorr, S. Adam, M. Eisenbarth, M. Ehresmann. “Implementing Requirements Engineering Processes: Using Cooperative Self-Assessment and Improvement”. IEEE Software, Vol. 25, No. 3, pp. 71 – 77, May/Jun 2008.

[8] C. Ebert, J. De Man. “Requirements Uncertainty: Influencing Factors and Concrete Improvements“. Proc. of the 27th Int. Conf. on Software Engineering (ICSE’05), pp. 553 – 560, 15-21 May 2005.

[9] European Commission, “The new SME definition User guide and model declaration”, White Paper, May 2003, URL: http://ec.europa.eu/enterprise/policies/sme/files/sme_definition/sme_user_guide_en.pdf

[10] M.E. Fayad, M. Laitinen, R. P. Ward. “Software Engineering in the Small”. Communications of the ACM, Vol. 43, No 3, pp. 115 – 118, Mar. 2000.

[11] D. Firesmith. “Common Requirements Problems, Their Negative Consequences, and the Industry Best Practices to Help Solve Them”. Journal of Object Technology, Vol. 6, No. 1, pp 17-33, Jan.-Feb. 2007.

[12] B. Graaf, M. Lormans, H. Toetenel. “Embedded Software Engineering The State of the Practice”, IEEE Software, Vol. 20, No. 6, pp. 61-69, Nov./Dec. 2003.

[13] C. Hofer. “Software development in Austria: results of an empirical study among small and very small enterprises”, Proceedings of the 28th Euromicro Conference, pp. 361 – 366, 2002.

[14] H.F. Hofmann, F. Lehner. “Requirements Engineering as a Success Factor in Software Projects”. IEEE Software. Vol. 18, No 4, pp. 58-66, Jul./Aug. 2001.

[15] S. Jantunen. “Exploring software engineering practices in small and medium-sized organizations”, Proceedings of the 2010 ICSE Workshop on Cooperative and Human Aspects of Software Engineering (CHASE '10), pp. 96 – 101, 2010.

[16] N. Juristo, A. M. Moreno, A. Silva. “Is the European Industry Moving toward Solving Requirements Engineering Problems?” IEEE Software, Vol. 19, No 6, pp. 70 – 77, Nov./Dec. 2002.

[17] E. Kamsties, K. Hörmann, M. Schlich. “Requirements engineering in small and medium enterprises”. Journal of Requirements Engineering, Vol. 3, No. 2, pp. 84-90, 1998.

[18] K. Kautz, “Making Sense of Measurement for Small Organizations”, IEEE Software, Vol. 16, No. 2, pp. 14-20, Mar./Apr. 1999.

[19] A. Lamsweerde. “Requirements Engineering: From Craft to Discipline”. Proceedings of the 16th ACM SIGSOFT International Symposium on Foundations of software engineering, pp. 238 – 249, Nov. 9-14, 2008.

[20] C.Y. Laporte, S. Alexandre, R.V. O’Connor. “A software engineering lifecycle standard for very small enterprises”, Proceedings of the 15th European Conference, EuroSPI 2008. Vol. 16, pp. 129 – 141, Sep. 2008.

[21] C.Y. Laporte, S. Alexandre, A. Renault. “Developing International Standards for Very Small Enterprises” IEEE Computer, Vol. 41, No. 3, pp. 98-101, Mar. 2008.

[22] M. M. Lehman, F. N. Parr, “Program Evolution and its Impact on Software Engineering”, Proceedings of the 2nd International Conference on Software Engineering (ICSE’ 76), pp. 350 – 357, 1976.

[23] R. Likert. "A Technique for the Measurement of Attitudes". Archives of Psychology 140, pp. 1–55, 1932.

[24] A. Loconsole. “Empirical Studies on Requirements Management Measures”. Proc. of the 26th Int. Conf. on Software Engineering (ICSE’04), pp. 42-44, May 23-28, 2004.

[25] O. Mondragon, “Addressing infrastructure issues in very small settings”. First International Research Workshop for Process Improvement in Small Settings, Pittsburgh, PA, 2005.

[26] J. Montilva, J. Barrios, M. Rivero. “Requisitos de capacitación y perfiles para ingenieros de software en Micro y Pequeñas Empresas”, XXXV Conferencia Latinoamericana de Informática (CLEI), 2009.

[27] U. Nikula. “Experiences from lightweight RE method evaluations”, IEEE International Workshop of Comparative Evaluation in Requirements Engineering, Monterey Bay, CA, pp. 53 – 60, 2003.

[28] U. Nikula, PhD Thesis: “Introducing basic systematic requirements engineering practices in small organizations with an easy to adopt method”, Lappeenranta University of Technology, 2004.

[29] U. Nikula, J. Sajaniemi. “Basyre: A lightweight combination of proven RE techniques”, Proceedings of International Workshop on Time-Constrained Requirements Engineering, Essen - Germany, 2002.

[30] U. Nikula, J. Sajaniemi, H. Kälviäinen. “A State-of-the-practice Survey on Requirements Engineering in Small-and Medium-sized Enterprises”, TBRC Research Report, Vol. 1, 2000.

[31] U. Nikula, J. SajaniemI, H. Kälviäinen. “Management view on current requirements engineering practices in small and medium enterprises”, Fifth Australian Workshop on Requirements Engineering, Queensland University of Technology, Brisbane – Australia, pp. 81-89, 2000.

[32] B. Nuseibeh, S. Easterbrook. “Requirements Engineering: A Roadmap”. In A. C. W. Finkelstein (ed.) "The Future of Software Engineering". (Companion volume to the proceedings of the 22nd International Conference on Software Engineering, ICSE'00). IEEE Computer Society Press. pp. 35 – 46, Jun. 4-11, 2000.

[33] S.F. Ochoa, J.A. Pino, L.A. Guerrero, C. Collazos. “SSP: A Simple Software Process for Small-Size Software Development Projects”. Proc. of the 1st Conf. on Advanced Software Engineering. Springer Science + Business Media. Vol. 219. Sep. 2006.

[34] M.C. Paulk. “Using the Software CMM in Small Organizations”. Proceedings of the Pacific Northwest Software Quality Conference and the Eighth International Conference on Software Quality, Portland. pp. 350–361, 1998.

[35] I. Richardson, C. G. von Wangenheim. “Why Are Small Software Organizations Different?”. IEEE Software, Vol. 24, No 1, pp. 18 – 22, Jan./Feb. 2007.

[36] J. Ropponen, K. Lyytinen. “Components of software development risk: how to address them? A project manager survey”. IEEE Transactions on Software Engineering, Vol. 26, No 2, pp. 98 – 112, Feb. 2000.

[37] S. Rowe. “Project Management for Small Projects”. Published by Management Concepts, 2006.

[38] L. Wallace, M. Keil. “Software Projects Risks and their Effect on Outcomes”, Communications of the ACM, Vol. 47, No. 4, Apr. 2004.

[39] C.G. von Wangenheim, A. Anacleto, C. F. Salviano. “Helping Small Companies Assess Software Processes”. IEEE Software, Vol. 23, No 1, pp. 91 – 98, Jan./Feb. 2006.

[40] C.G. von Wangenheim, T. Varkoi, C.F. Salviano. “Standard based software process assessments in small companies”, Journal of Software Process: Improvement and Practice, Vol. 11, No. 3, pp. 329-335, 2006.

[41] C.G. von Wangenheim, S. Weber, J.C.R. Hauck, G. Trentin. “Experiences on establishing software processes in small companies”, Information and Software Technology Journal, Vol. 48, No. 9, pp. 890-900, 2006.

[42] K.E. Wiegers. “Software Requirements (Second Edition)”. Microsoft Press, 2003.

[43] E. Yourdon. “Death March”. Prentice Hall; 2nd Edition, 2003.