JOURNAL OF ENGINEERING MANAGEMENT AND COMPETITIVENESS (JEMC)
VOL. 8, NO. 1, 2018, 54-64

SOFTWARE DEVELOPERS' PERCEPTIONS OF SOFT SKILLS IN SOFTWARE REQUIREMENTS ENGINEERING

UDC: 004.41
Original Scientific Paper

Tamara ZORIC[1], **Zeljko STOJANOV**[2]

[1]Republic of Serbia
[2]University of Novi Sad, Technical faculty ”Mihajlo Pupin” Zrenjanin, 23000 Zrenjanin, Đure Đakovića bb, Republic of Serbia
E-mail: zeljko.stojanov@uns.ac.rs

Paper received: 27.02.2018.; Paper accepted: 24.04.2018.

Software requirements express users' needs and constraints for a software system to be constructed in order to solve some real problems. Both researchers and practitioners from industry point out that well performed requirements related activities have very important role for successful realization of software projects. Software developers work on software requirements jointly with software users. Requirements elicitation and specification are especially important since they require software developers with multidisciplinary skills. These activities require not only technical skills but also non-technical or soft skills. This article presents a qualitative study with software developers aimed at inquiring how they perceive soft skills in software requirements activities. The study involved 14 software developers, who were interviewed. Data analysis, based on inductive thematic analysis, resulted in three themes that reflect software developers' perceptions about soft skills. The study findings revealed that software developers are aware of soft skills, recognize the need to improve them and understand their importance in everyday practice. Implications of the research findings for researchers, software developers and managers in software organizations are discussed.

Keywords: Soft skills, Human factor, Software requirements, Software developers, Qualitative study.

INTRODUCTION

Software requirements engineering (SRE) is the field in software engineering concerned with systematic handling of software requirements, which includes (Bourque & Fairley, 2014): (1) elicitation, analysis, specification and validation of software requirements, and (2) management of requirements during the whole life cycle of software products. Software requirements contain descriptions of needs and constraints for software products that solve real world problems. Software development process begins with agreement about what software should do, which is described in software requirements specification document defined through joint work of customers and software organization (supplier). Software requirements activities are of essential importance for success of software projects (Hofmann & Lehner, 2001). The main research challenges in SRE are wider than in other areas of software engineering since requirements engineering is focused on both problem space and solution space (Cheng & Atlee, 2007). Efficient requirements engineering assumes software experts to become familiar with the problem domain, which includes understanding domain terminology, processes, entities and their relationships that should be integrated into software products.

Software requirements engineering, or requirements engineering, relates to process of identifying, analyzing, documenting and checking services (functionality) that software system should implement (Sommerville, 2011). This process is fundamentally interdisciplinary, involving people that have a stake in software. These stakeholders are (Bourque & Fairley, 2014):

- *Users* - heterogeneous group of people who will operate software.

ISSN 2217-8147 (Online)

Available online at http://www.tfzr.uns.ac.rs/jemc

- *Customers* - group of people who commissioned software or who represent target market for software.
- *Market analysts* - group of people working on investigating market for software which does not have commissioned customers.
- *Regulators* - regulatory authorities providing regulations which software should align to in order to be used in specific domains.
- *Software engineers* - group of people having legitimate interest to profit from developing software.

Software requirements are usually elicited from various people in target organization, which ensures that they are grounded in the real needs of the organization requiring a software product for solving existing problems. According to Nuseibeh and Easterbrook (2000), software requirements engineering is multi-disciplinary and human-centered, which means that methods and tools draw upon variety of disciplines, requiring highly skilled engineers. Therefore, software requirements should be written at different levels of description and abstraction, which are suitable for different readers. Software users require high-level descriptions, written in natural language, accompanied with some figures, easy to comprehend. On the other side, software developers need more detailed and formalized descriptions which can be easily transformed into models and code. Sommerville (2011) distinguishes the following two types of requirements depending on the description level:

- *User requirements* are high-level abstract requirements stated in natural language, describing which services software system should provide.
- *System requirements* are more detailed descriptions of software system services and constraints elaborated in a formal document that is a subject of agreement between interested parties (customer and software organization).

People involved in specification of software requirements are presented in Figure 1. People that produce user requirements use domain specific terminology and are concerned with high level descriptions of software products. These people, except system architects who are concerned with architecture of a software product, usually do not possess technical knowledge about software development process. On the other side, system requirements contain implementation details of software design and require knowledge of appropriate software development methods and technologies. System end-users and client engineers are engaged in validating requirements.

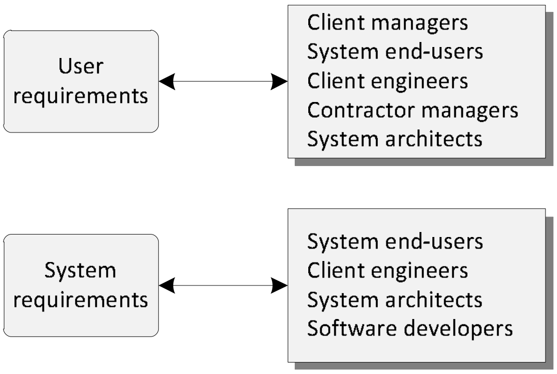

Figure 1: People involved in requirements specification [adapted from Sommerville (2011)]

Wheatcraft et al. (2016) pointed out the importance of clear and unambiguous expression of requirements for communicating the needs into a formal language that can be understood by those engaged in implementing them, by those responsible for proving that system meets the requirements, and by those responsible for proving that system meets the needs of the relevant entity.

Complexity and multidisciplinary nature of SRE require software engineers who posses both technical and non-technical skills. Joseph et al. (2010) suggest that technical skills are insufficient for success in information technology, stressing the importance of acquiring several non-technical or soft skills. It becomes widely accepted that soft skills are of primary importance, while technical skills (e.g. programming, database management, configuring hardware and servers) can be developed through appropriate trainings (Stevens & Norman, 2016). However, due to the complexity of human nature and its impact on software engineering practice, there is a lack of research dedicated to human factor in software engineering, especially in the field of SRE (Ahmed et al., 2013). In addition, the problem with neglecting human factor in software engineering is caused by common, but evidently wrong thinking that there is no place for soft skills in technical disciplines driven by logic and based on mathematics (Capretz, 2014).

In dynamic and evolving software industry, mostly based on agile development approaches, individual competencies and skills become highly important for success (Cockburn & Highsmith, 2001). According to Saiedian and Dale (2000) many software development projects fail even before reaching formal specification stage due to the lack of soft skills of the people involved in requirements elicitation, indicating the importance of human factor in SRE. Human factor has been investigated in software engineering literature (Lieger et al., 2012; Bailey, 2014; Amrit et al., 2014) but mostly focusing on software development in general. Despite these facts, there is a need for more empirical studies on human factor in software engineering (Kosti et al., 2014). Based on the stated observations, there is an evident lack of empirical studies related to human factor in SRE (research on non-technical or soft skills), despite the recognized importance and criticality for success of projects in software industry. This study aims at contributing to the knowledge base of software engineering and management by investigating how software developers from industry perceive soft skills in SRE.

The rest of the article is structured as follows. The next section contains a short review of related work in the field of soft skills in SRE. The third section describes the empirical study, followed with the fourth section presenting the findings of the study. After that, the discussion section outlines some research implications and reflections on validity. The last section presents concluding remarks and further research directions.

RELATED WORK

Soft skills refer to personal attributes and attitudes that drive people’s behavior in a given context. These skills are also named as non-technical skills, people skills, human factor or emotional intelligence in software industry (Ahmed, 2012). It has been recognized in literature that human factor positively contributes to overall business performance (Robles, 2012; Balcar, 2016) and competitiveness of organizations (Óhegyi, 2014). Requirements elicitation in practice involves negotiation and collaboration of all stakeholders, construction of variety of relationships, resulting in highly interactive SRE process (Coughlan et al., 2003).

Growing interest in human issues in software engineering is the consequence of recognizing that everyday workplace activities depend on people characteristics, which resulted in a great interest of research community and a number of publications and special sessions at leading software engineering conferences (Amrit et al., 2014). In software engineering, human factor has been recognized as critical for success, especially in SRE that is based on collaboration of different stakeholders (Pernstål et al., 2015). In this course of thinking, Andreou (2003) stated that rules and focus in collecting software requirements shift from the operations and the data to be processed by the system to human-computer interaction and organizational factors. This trend assumes the use of several social scientific methods, such as ethnography (Macaulay et al., 2000), user stories (Lucassen et al., 2016) or interviews (Hadar et al., 2014; Ferrari et al., 2016) in studies related to software requirements.

According to Gallivan and Keil (2003) majority of prior researches investigating user involvement in software development will ensure clear requirements definitions, resulting in the system which meets user needs. However, Saiedian and Dale (2000) identified the following main difficulties in requirements elicitation: (1) poor communication, (2) resistance as a physical and emotional expression, (3) articulation of problems, and (4) different perspectives on problem to be solved. For coping with the stated difficulties, the authors proposed development of effective communication skills, graphical representation of the user environment, use of customer participatory techniques and prototyping. Davey and Cope (2008) argued that the main problem in requirements elicitation is conversation with clients, and suggest improvements in education and trainings for requirement engineers or business consultants engaged in collecting requirements.

Based on comprehensive survey of SRE job opportunities Ahmed (2012) identified several different roles software experts can play in SRE, such as Software Requirements Engineer, Requirements Writer, Requirements Analysts, Business Analysts etc. Further, detailed statistical analysis of software requirements engineering job opportunities all around the world revealed the following demand for non-technical skills: communication skills (87%), analytical and problem solving (47%), interpersonal skills (48%), organizational skills (36%), ability to work independently (37%), team player (49%), adaptable to changes (30%), innovative (14%) and fast learner (16%). Based on obtained results Ahmed (2012) concluded that communication skills are highly demanded, while some other skills such as analytical and problem solving, interpersonal skills, organizational skills, independent working ability and team player are moderately demanded.

The short literature review revealed that SRE is complex and attractive practical field that requires awareness of people characteristics in everyday activities, especially due to the wide range of potential problems in SRE (Zoric & Stojanov, 2017). The presented empirical study, focused on how software developers perceive soft skills in SRE activities, contributed to the knowledge base in SRE and human resource management in software organizations.

CASE STUDY

This study is extracted from a large study aimed at investigating communication and cognitive skills of software developers in software requirements elicitation and specification. The whole study was prepared in November and December 2016. Since the aim was to investigate software developers' opinions about soft skills, the study was designed as a qualitative study (Marshall & Rossman, 2011). The first two months were used for planning activities, which included: (1) the first author of the article investigated SRE literature in order to become acquainted with the terminology, processes and trends, (2) both authors set the study objective and selected the appropriate research methods, and (3) the second author selected the participants based on his contacts with software companies that operate in the region. After that, starting from January 2017, the next three months were used for organizing and conducting interviews with the selected software developers. Data analysis started in April 2017. Until now, data analysis on general attitudes of software engineers towards soft skills, and experiences in problem solving are completed, while the parts related to cognitive and communication skills needs validation through member check interviews (Buchbinder, 2011).

Practically, this study was based on the introductory part of the interviews with the software developers, in which they were asked to talk how they understood and perceived soft skills in general, and importance of soft skills in SRE. Interviewing was based on semi-structured in-depth interviews (Rabionet, 2011), with some general questions prepared in advance for directing conversation during the interviews, while some questions were based on the participants answers. Semi-structured interviews were selected as a method for collecting data because they enable collecting high quality data if they are carefully planned (Hove & Anda, 2005). All interviews were conducted in Serbian. Recorded interviews were transcribed and prepared for qualitative data analysis. During and immediately after each interview, the researchers wrote field notes about the interviewee and the process of interviewing (Wolfinger, 2002). These field notes were used for describing demographic data about the participants and for refining methodological dilemmas during the research process.

Data analysis is based on inductive thematic analysis method proposed by Braun and Clarke (2006) and aided with thematic networks as a tool for presenting themes proposed by Attride-Stirling (2002). The initial phase of data analysis, initial coding, was performed in *QDA Miner Lite* (https://provalisresearch.com), a free and easy-to-use qualitative data analysis software tool for coding unstructured text. The main strengths of *QDA Miner Lite* are: intuitive and easy to use interface, support for organizing and coding multiple documents, and management of memos during the data analysis.

After initial coding, identification and development of themes that reflect the attitudes of software experts were performed by using MS Excel and by drawing thematic network diagrams on the papers. Complete data analysis was performed in Serbian, while selected segments of data, interim results and final findings were translated in English. Writing theoretical memos was of fundamental importance for data analysis and development of the findings (Birks et al., 2008).

Participants

Study participants are software developers working in software companies in the region (Serbian part of Banat region) or working as freelancers. The participants were approached based on their contacts with them or managers of their companies. In the study participated: 5 software developers working in indigenous software companies, 6 software developers working for foreign software companies, and 3 software developers working as freelancers. The average experience in software industry for all participants is 10.29 years, which indicates that the selected participants have enough experience with variety of issues in industrial practice. Detailed demographic data about participants with average experience for participants working in indigenous software companies, foreign software companies and as freelancers are presented in Table 1.

Table 1: Detailed data about the participants' experience

Type of company	**Participant number**	**Years of experience**
Indigenous software companies	Participant 2	25
	Participant 3	12
	Participant 7	8
	Participant 8	17
	Participant 13	30
Average experience for indigenous software companies		**18.40**
Foreign software companies	Participant 5	5
	Participant 6	6
	Participant 9	3
	Participant 10	3
	Participant 11	3
	Participant 12	1
Average experience for foreign software companies		**3.50**
Freelancers	Participant 1	2
	Participant 4	18
	Participant 14	11
Average experience for freelancers		**10.33**
AVERAGE EXPERIENCE FOR ALL PARTICIPANTS		**10.29**

Sampling in the study was based on a suggestion proposed by Braun and Clarke (2006) that data analysis starts after collecting all data (in the case of this study after conducting all interviews), which means that there is no sense to discuss achievement of theoretical saturation in data analysis. Further, Guest et al. (2006) suggested that in most cases twelve interviews will be enough for most of the qualitative thematic studies (the majority of themes or concepts emerge within this number of interviews). Based on a comprehensive literature review of qualitative studies Marshall et al. (2013) proposed a relationship between number of interviews and thematic prevalence in which: (1) 79% of themes are identified if the number of interviews is between 13 and 18, and (2) for the number of interviews larger than 18 the number of newly identified themes is very small. Based on these observations, the authors decided to include 14 interviewees in the study.

Research process

The research process includes two main phases. The first phase relates to research planning and preparation activities, which includes:

- Investigating relevant literature about human factor in SRE, especially empirical studies that report issues and challenges from industrial practice.
- Setting research objectives.
- Selecting the most suitable methods for implementing all research activities.
- Selecting the study participants that will be able to provide the most valuable insight into practical issues related to stated research objectives.
- Preparing documents that structure the research process and activities: Initial outline of semi-structured interviews, Informed consent, and Initial timeline and schedule of research activities.

The second phase relates to the main research activities, which includes field work with study participants and data analysis activities. Field work assumes arranging and conducting interviews with the participants. The primary data collecting activities are semi-structured interviews with software developers from industry. In addition to interviews, the researchers wrote field notes during the interviews and immediately after them. The interviews were conducted in the places that make participants most comfortable: their companies, the faculty, and cafes. This helped in achieving a pleasant working atmosphere which is necessary for relaxed conversation during the interviews.

Data analysis activities are based on inductive thematic analysis method proposed by Braun and Clarke (2006), aided with the thematic network diagrams proposed by Attride-Stirling (2002).This is a qualitative interpretative methodology that enables data analysis of unstructured text and identification of themes that reflect participants experience about the investigated practice. The steps in data analysis are:

- *Familiarizing with the data*. This step started immediately after conducting each interview by listening to the interview audio record. After the interview transcription, reading and formatting the transcribed text helped in this step.
- *Initial coding*. This step included coding row empirical data imported in software *QDA Miner Lite*, which means assigning meaningful labels

to the segments of the text. For each label, or initial code, a short theoretical memo was written with the researchers remarks about the code and the text segment. Each researcher performed initial coding independently, and after that, the next steps were performed through joint work of both researchers.

– *Searching for themes*. This step assumes collecting relevant text segments for each initial code, and grouping the codes with the similar meaning into themes. This step included printing a list of initial codes and discussing them.
– *Reviewing themes*. This step includes reviewing how collected text segments fit to the identified themes, and development of an initial *thematic network* with all identified themes and their relationships.
– *Defining themes*. This step relates to refining themes, which includes: defining their names and clear descriptions, and associating the text segment from interviews that will be used as verbatim quotations (Corden & Sainsbury, 2006). Verbatim quotations are a common tool for increasing evidential power in qualitative research.
– *Writing the report*. This step relates to writing documents about the research process and findings. It provided an additional opportunity to check the whole research process, and how research findings fit stated research objectives.

Data collecting and analyzing activities are presented in Figure 2, with the focus on the activities presented in the rectangles and products of each activity presented in the rounded rectangles. The final product of data analysis is a thematic framework with identified themes and their relationships.

Memos were written during the whole research process. Methodological memos were used for discussing and clarifying methodological and organizational issues during the research, while theoretical memos were used in data analysis as the main tool for describing initial codes, themes, and the relationships between themes in the developed framework.

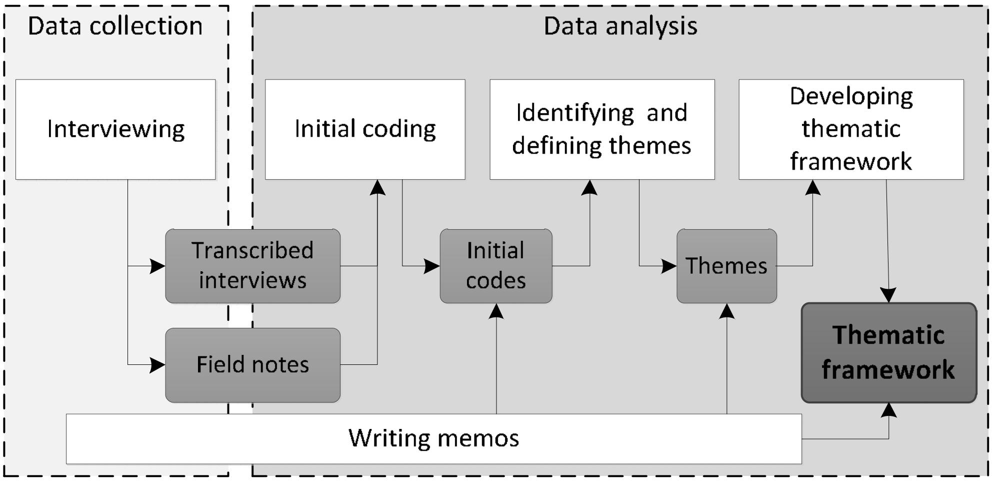

Figure 2: Data collecting and data analyzing activities

The authors selected these research methods because the methods are suitable for the proposed research objectives, and their previous experience in joint work with these and similar methods (Stojanov et al., 2016; Stojanov et al., 2017).

FINDINGS

The research findings of this study were developed as a theoretical framework (network) containing themes explaining how software experts perceive soft skills in SRE. In addition, the framework contains relationship between the identified themes. According to the typology of qualitative findings proposed by Sandelowski and Barroso (2003), these findings can be classified as Conceptual/Thematic description, since the findings are rendered as a set of developed themes and their relationships. Figure 3 presents the developed framework, in which the main themes are presented as eclipses, while the relationships between the themes are presented with arrows.

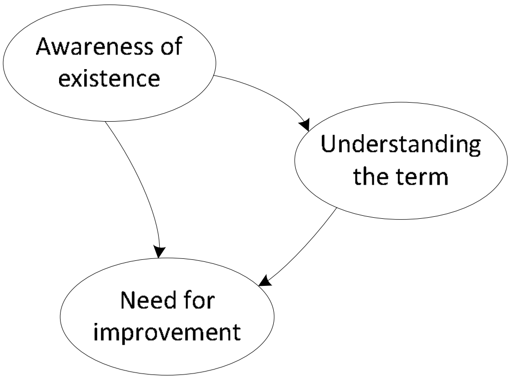

Figure 3: Themes that present how software engineers perceive soft skills in SRE

Figure 3 will be used for organizing the presentation of the findings in the following subsections. Verbatim quotations from the participants answers will be used for illustrating their perceptions (Corden & Sainsbury, 2006), which were used for developing substantial themes during the data analysis process.

Awareness of existence

The theme relates to software developers' awareness about the existence of soft skills. It is very interesting that all participants are aware of soft skills and their importance for everyday practice, but they do not think about them while solving everyday tasks. Some of the participants stated that it is sometimes hard to recognize soft skills because they are mostly focused on technical skills that are usable for solving problems in software development. This is supported with the following quotation:

> *"... sometimes it is hard to recognize soft skills because technical skills are very important for software development, so in some cases soft skills are not considered, which can lead to problems."* [Participant 1]

This quotation revealed that although software developers are not always aware of soft skills, they recognize that neglecting them might have negative consequences on software development practice. Awareness about soft skills is commonly reflected in counting some typical skills related to solving practical problems, which is supported with the following quotation:

> *"... so it is simply a set of some skills, seen from the perspective of our business, how to communicate, how to work, things not related to technique."* [Participant 8]

Understanding the term

The theme relates to how software developers understand the term soft skills. Majority of the participants understand this term. In addition, they are aware of few different terms that are used for soft skills. Two participants have not heard for the term soft skills, but they understood the term "non-technical skills", which they confirmed by counting some of those they used regularly in practice. Majority of the participants do not think about soft skills, but use them in practice, and in most cases they define them as communication skills needed when they work with clients and as the skills needed to fit within the organization. This is illustrated by the following quotation:

> *"Yes, these are skills you need to fit in the company, because each software developer has frequent contacts with managers and clients."* [Participant 10]

Less experienced developers usually understand these skills as those not related to programming, which is illustrated by the quotation:

> *"This is something that does not relate to programming skills."* [Participant 9]

More experienced developers relate these skills as the skills that are not related to techniques and technology, which is illustrated by the following quotations:

> *"This is something necessary to work as software developer. It is important to understand a user, what he wants to say. So, it is not related to technical knowledge."* [Participant 11]

> *"So, these are not technology-related issues, these are not technological skills and knowledge, but it is about how you deal with people."* [Participant 3]

Need for improvement

The theme indicates the importance of soft skills for software developers. This is even more important since software developers pay more attention to techniques and technologies that enable software development, and they mostly improve themselves in these directions. The participants stated that soft skills could be improved at specialized courses or workshops, which are dedicated to specific set of soft skills. This is illustrated by the quotation:

> *"Yes, the courses help. It depends on the course type, starting from verbal communication, non-verbal communication, and workshops dedicated to relationships with clients."* [Participant 8]

In addition, they indicate that it is important to attend these courses several times in the career in order to continuously improve soft skills, as it is the case with technical skills. This is illustrated by the following quotation:

> *"It is normal to attend such courses from time to time, so that you can improve yourself for better quality communication with clients."* [Participant 14]

Relationships between themes

The thematic framework was created after the themes were identified and defined by establishing links between the themes. The themes are interconnected, and the relationships are shown by arrows that point towards the direction of dependence or influence among the themes.

The theme *Awareness of existence* is necessary for defining the other themes in the thematic framework, *Understanding the term* and *Need for improvement*. This theme has a direct impact on the other two themes, as it is presented in Figure 3. The influence of this theme suggests that an individual can understand a concept and identify the need for its improvement, only when he/she becomes aware of the concept. From the aspect of this research, this refers to the fact that software engineers will be able to understand soft skills and identify the need for improving them, only after they become aware of them.

The connection that goes from the theme *Understanding the term* to the theme *Need for improvement* indicates that understanding is necessary in order to identify the need for improvement. It can be concluded that understanding of certain soft skills and awareness of their existence allow software engineers to identify which skills they need, so they could effectively perform their job, and see if it is necessary to improve some of these skills.

DISCUSSIONS

This section presents discussions about issues that are common in qualitative studies, such as research implications, validity and ethical issues. The aim of these discussions is to increase the quality and reliability of the study.

Research implications

Due to its significance for the practice in software industry, the selected research topic and the presented findings have implications for different groups of people. These groups are researchers, software developers and managers in software industry.

Researchers can find guidelines how to organize a qualitative study with experts from industry, how to conduct qualitative data analysis and present the findings of the study. In addition, details about the study design can be used for organizing similar studies for inquiring specific aspects of the practice that is dominantly influenced by the human characteristics and behavior.

Software developers and managers from software industry can find information about the importance of soft skills for everyday practice, especially in the field of SRE. The presented study findings can help them in recognizing soft skills and increasing their awareness of these skills. The managers can get insight into the understanding of soft skills, and organize internal inquiry to find out the state in their organization, which can be used for organizing trainings for their developers. In addition, software managers can find details about using qualitative methods that are commonly used for getting deeper insight into everyday practice, and implement them in their organizations.

Trustworthiness

Trustworthiness relates to confidence toward a study findings, and can be evaluated from the aspect of credibility and transferability (Anney, 2014).

Credibility should ensure reliable information from raw empirical data collected from the participants. In order to ensure reliable development of theoretical constructs from the empirical data, all the steps in inductive thematic analysis were carefully implemented. Besides that, the techniques that have been used are writing reflective memos about all aspects of the research (Watt, 2007), and joint work of the researchers in all stages of data analysis with mutual checks of each other's findings (MacPhail et al., 2016).

On the other hand, transferability is a common limitation of qualitative studies (Anney, 2014), and it was not considered during the study design phase, because the main goal of the study is to investigate the experience in the given context. However, design of the study can be replicated in similar researches involving practitioners from industry.

Ethical issues

This empirical research with the practitioners from the industry should consider several ethical issues, such as informed consent, risks, harms, and benefits from participating in the study (Vinson & Singer, 2008). All ethical issues were considered during the research planning phase, and the participants had been informed about ethical issues before they agreed to participate in the research.

The document called *Informed consent form*, containing elaboration of ethical issues, was presented to each participant before the interview, and signed by both the participant and the researchers. This document addresses confidentiality issues of empirical data and data on participants in this research, as well as issues related to publishing and presenting the research reports. The participation in this research was on a voluntary basis, and the participants had the right to withdraw from the research at any moment without consequences. These issues are also covered by the previously mentioned document.

One of the most important ethical issues was to preserve anonymity of the study participants (Guenther, 2009). This was accomplished by not presenting the names of the participants and their companies. When presenting their words in quotations used for illustrating developed themes, the participants are marked with the associated numbers (e.g. Participant 3).

CONCLUDING REMARKS

Software requirements engineering plays a very important role in the success of software projects, requiring from software engineers to develop and use multidisciplinary skills. This paper presents the findings of the qualitative empirical study focused on awareness of software engineers on the existence and importance of soft skills. The main contribution of this study is to increase awareness of the importance of soft skills among software developers and managers in software organizations, as well as to influence the introduction of teaching contents related to soft skills in software engineering education.

It can be concluded that there is a certain awareness of the existence of soft skills, but also that these skills are often understood as skills that are not related to technical aspects of software engineering. It is clear that majority of software engineers do not think about these skills, but use them unconsciously in practice. The development of skills in software engineering is mainly focused on the development of technical skills, but not so much on the development of soft skills. Considering the fact that these skills are equally important in software requirements engineering activities, in this paper the development and improvement of soft skills are also perceived.

The directions for future research relate to more detailed research into specific soft skills in software industry or specific industry segments (e.g. embedded software systems or Internet-based systems). Future research can also focus on research of soft skills among students of the appropriate study profiles, as well as on the development of software solutions that would allow the systematization of these skills in specific frameworks.

ACKNOWLEDGEMENT

Ministry of Education, Science and Technological Development, the Republic of Serbia, supports this research under the project ”The development of software tools for business process analysis and improvement”, project number TR32044, 2011-2018.

REFERENCES

Ahmed, F. (2012). Software Requirements Engineer An Empirical Study about Non-Technical Skills. *Journal of software*, *7*(2), p. 389-397.

Ahmed, F., Capretz, L. F., Bouktif, S., & Campbell, P. (2013). Soft Skills and Software Development: A

Reflection from Software Industry. *International Journal of Information Processing and Management*, *4*(3), 171-191.

Amrit, C., Daneva, M., & Damian, D. (2014). Human factors in software development: On its underlying theories and the value of learning from related disciplines. A guest editorial introduction to the special issue. *Information and Software Technology, 56*(12), 1537-1542.

Andreou, S. A. (2003). Promoting software quality through a human, social and organisational requirements elicitation process. *Requirements Engineering*, *8*(2), 85-101.

Anney, V. N. (2014) Ensuring the Quality of the Findings of Qualitative Research: Looking at Trustworthiness Criteria. *Journal of Emerging Trends in Educational Research and Policy Studies, 5*(2), 272-281.

Attride-Stirling, J. (2002) Thematic networks: an analytic tool for qualitative research. *Qualitative Research*, *1*(3), 385-405.

Bailey, J. L. (2014). Non-Technical Skills for Success in a Technical World. *International Journal of Business and Social Science*, *5*(4), 1-10.

Balcar, J. (2016). Is it better to invest in hard or soft skills? *The Economic and Labour Relations Review, 27*(4), 453 - 470.

Birks, M., Chapman, Y., & Francis, K. (2008) Memoing in qualitative research: Probing data and processes. *Journal of Research in Nursing*, *13*(1), 68-75.

Bourque, P., & Fairley, R. E. D. (Eds.) (2014). *Guide to the software engineering body of knowledge, version 3.0, SWEBOK*. Piscataway, NJ, USA: IEEE Computer Society.

Braun, V., & Clarke, V. (2006) Using thematic analysis in psychology. *Qualitative Research in Psychology*, 3(2). pp. 77-101.

Buchbinder, E. (2011) Beyond Checking: Experiences of the Validation Interview. *Qualitative Social Work*, *10*(1), 106–122.

Capretz, L. F. (2014) Bringing the Human Factor to Software Engineering. *IEEE Software*, *31*(2), 102-104.

Cheng, B. H. C., & Atlee, J. M. (2007). Research directions in requirements engineering. *In Proceedings of 2007 Future of Software Engineering (FOSE '07)*, pp. 285-303. Minneapolis, MN, USA.

Cockburn, A., & Highsmith, J. (2001). Agile software development, the people factor. *Computer, 34*(11), 131-133.

Corden, A., & Sainsbury, R. (2006) Exploring ’quality’: Research participants’ perspectives on verbatim quotations. *International Journal of Social Research Methodology*, *9*(2), 97–110.

Coughlan, J., Lycett, M., & Macredie, R. D. (2003). Communication issues in requirements elicitation: a content analysis of stakeholder experiences. *Information and Software Technology*, *45*(8), 525-537

Davey, B., & Cope, C. (2008). Requirements Elicitation - What’s Missing? *Issues in Informing Science and Information Technology, 5,* 543-551.

Ferrari, A., Spoletini, P., & Gnesi, S. (2016). Ambiguity and tacit knowledge in requirements elicitation interviews. *Requirements Engineering, 21*(3), 333-355.

Gallivan, M. J., & Keil, M. (2003). The user-developer communication process: a critical case study. *Information Systems Journal*, *13*(1), 37–68.

Guenther, K. M. (2009) The politics of names: rethinking the methodological and ethical significance of naming people, organizations, and places. *Qualitative Research, 9*(4), 411-421.

Guest, G., Bunce, A., & Johnson, L. (2006) How Many Interviews Are Enough? : An Experiment with Data Saturation and Variability. *Field Methods*, *18*(1), 59-82.

Hadar, I., Soffer, P., & Kenzi, K. (2014). The role of domain knowledge in requirements elicitation via interviews: an exploratory study. *Requirements Engineering*, *19*(2), 143-159.

Hofmann, H.F., & Lehner, F. (2001). Requirements engineering as a success factor in software projects. *IEEE Software*, *18*(4), 58-66.

Hove, S. E., & Anda, B. (2005) Experiences from Conducting Semi-structured Interviews in Empirical Software Engineering Research. *In Proceedings of the 11th IEEE International Software Metrics Symposium (METRICS '05)*, pp. 23-32. Como, Italy.

Joseph, D., Ang, S., Chang, R. H. L., & Slaughter, S. A. (2010). Practical intelligence in IT: assessing soft skills of IT professionals. *Communications of the ACM*, 53(2), 149-154.

Kosti, M. V., Feldt, R., & Angelis, L. (2014) Personality, emotional intelligence and work preferences in software engineering: An empirical study. *Information and Software Technology*, 56(8), 973-990.

Lieger, J., Woodward, B., & Martin, N. (2012). Reassessing the Skills Required of Graduates of an Information Systems Program: An Updated Analysis. *In 2012 Proceedings of the Information Systems Educators Conference*, pp. 1-11. New Orleans, LA.

Lucassen, G., Dalpiaz, F., van der Werf, J. M. E. M., & Brinkkemper, S. (2016). Improving agile requirements: the Quality User Story framework and tool. *Requirements Engineering, 21*(3) 383-403.

Macaulay C., Benyon D., & Crerar A. (2000) Ethnography, theory and systems design: from intuition to insight. *International Journal of Human-Computer Studies, 53*(1), 35–60.

MacPhail, C., Khoza, N., Abler, L., & Ranganathan, M. (2016) Process guidelines for establishing Intercoder Reliability in qualitative studies, *Qualitative Research*, *16*(2), 198-212.

Marshall, B., Cardon, P., Poddar, A., & Fontenot, R. (2013) Does Sample Size Matter in Qualitative Research?: A Review of Qualitative Interviews in is

Research. *Journal of Computer Information Systems*, *54*(1), 11-22.

Marshall, C. & Rossman, G. B. (2011) *Designing Qualitative Research, Fifth Edition*. Thousand Oaks, CA, USA: SAGE Publications.

Nuseibeh, B., & Easterbrook, S. (2000). Requirements engineering: A roadmap. *In Proceedings of the conference on the Future of Software Engineering*, pp. 35-46. Limerick, Ireland.

Óhegyi, K. (2014). Analyzing the impact of human capital factors on competitiveness. *Journal of Engineering Management and Competitiveness*, 4(2), 85-93.

Pernstål, J., Gorschek, T., Feldt, R., & Florén, D. (2015) Requirements communication and balancing in large-scale software-intensive product development. *Information and Software Technology*, 67, November 2015, 44-64.

Rabionet, S. E. (2011) How I Learned to Design and Conduct Semi-structured Interviews: An Ongoing and Continuous Journey. *The Qualitative Report*, *16*(2), 563-566.

Robles, M. M. (2012). Executive Perceptions of the Top 10 Soft Skills Needed in Today's Workplace. *Business Communication Quarterly*, *75*(4), 453-465.

Saiedian, H., & Dale, R. (2000) Requirements engineering: making the connection between the software developer and customer. *Information and Software Technology*, *42*(6), 419-428

Sandelowski, M., & Barroso, J. (2003) Classifying the Findings in Qualitative Studies. *Qualitative Health Research*, *13*(7), 905-923.

Sommerville, I. (2011). Software *engineering, 9th ed*. Boston, MA, USA: Addison Wesley.

Stevens, M., & Norman, R. (2016). Industry expectations of soft skills in IT graduates: a regional survey. *In Proceedings of the Australasian Computer Science Week Multiconference (ACSW '16)*, Article 13 , 9 pages. Canberra, Australia.

Stojanov, Z., Dobrilovic, D., & Zoric, T. (2016) Solving problems in a physical laboratory for computer networks and data security: A conceptual framework with students' experiences. *International Journal of Engineering Education, 32*(6), 2517–2530.

Stojanov, Z., Dobrilovic, D., & Zoric, T. (2017) Exploring students' experiences in using a physical laboratory for computer networks and data security. *Computer Applications in Engineering Education*, *25*(2), 290-303.

Vinson, N. G., & Singer, J. (2008) A Practical Guide to Ethical Research Involving Humans. In: F. Shull, J. Singer, & D. I. K. Sjøberg (Eds.), *Guide to Advanced Empirical Software Engineering* (pp229-256), London : Springer.

Watt, D. (2007) On Becoming a Qualitative Researcher: The Value of Reflexivity. *The Qualitative Report*, *12*(1), 82-101.

Wheatcraft, L. S., Ryan, M. J., & Dick, J. (2016). On the use of attributes to manage requirements. *Systems Engineering*, *19*(5), 448-458.

Wolfinger, N. H. (2002) On writing fieldnotes: collection strategies and background expectancies. *Qualitative Research*, *2*(1), 85-93.

Zoric, T., & Stojanov, Z. (2017) Problem solving in software requirements elicitation and specification: Preliminary results from a qualitative study. *In Proceedings of the 7th International conference on Applied Internet and Information Technologies (AIIT2017)*, pp. 258-266. Zrenjanin, Serbia.

PERCEPCIJE SOFTVERSKIH INŽENJERA O NETEHNIČKIM VEŠTINAMA U INŽENJERINGU SOFTVERSKIH ZAHTEVA

Softverski zahtevi izražavaju korisničke potrebe i ograničenja u izgradnji softverskog sistema namenjenog rešavanju stvarnih problema. Istraživači i stručnjaci iz industrije ističu da korektno obavljene aktivnosti u vezi softverskih zahteva imaju veoma važnu ulogu u uspešnoj realizaciji softverskog projekta. Softverski inženjeri rade na softverskim zahtevima zajedno sa korisnicima softvera. Prikupljanje i specifikacija zahteva su naročito važne aktivnosti jer zahtevaju inženjere koji poseduju multidisciplinarne veštine. Ove aktivnosti zahtevaju ne samo tehničke, već i netehničke veštine. U ovom radu je predstavljena kvalitativna studija sa softverskim inženjerima koja za cilj ima da ispita način na koji oni shvataju netehničke veštine u aktivnostima koje se odnose na softverske zahteve. Studija je bazirana na analizi intervjua sa 14 softverskih inženjera. Rezultat analize podataka, zasnovane na induktivnoj tematskoj analizi, su tri teme koje odražavaju shvatanje netehničkih veština od strane softverskih inženjera. Nalazi istraživanja ukazuju da su softverski inženjeri svesni netehničkih veština, prepoznaju potrebu za njihovim usavršavanjem i razumeju njihov uticaj na svakodnevnu praksu. U radu su diskutovane implikacije nalaza istraživanja za istraživače, softverske inženjere i menadžere u softverskim organizacijama.

Ključne reči: Netehničke veštine, Ljudski faktor, Softverski zahtevi, Softverski inženjeri, Kvalitativna studija.