Exploring software maintenance process characteristics by using inductive thematic analysis

Zeljko Stojanov¹ and Jelena Stojanov²

¹ University of Novi Sad, Technical faculty "Mihajlo Pupin"
Djure djakovica BB, 23000 Zrenjanin, Serbia
zeljko.stojanov@uns.ac.rs

² University of Novi Sad, Technical faculty "Mihajlo Pupin"
Djure djakovica BB, 23000 Zrenjanin, Serbia
jelena.stojanov@uns.ac.rs

Abstract. Software maintenance is the most costly part of software life cycle, deserving more attention of research community. Systematic consideration of software maintenance problems and challenges becomes even more important in small software companies that face several difficulties due to their constraints related to staff, resources and funding. This paper outlines a method for identifying and systematizing knowledge about practice in small organizations. The method is implemented in a micro software company for identifying characteristics of software maintenance processes. Thematic analysis enables identification and systematization of knowledge in a framework that is grounded in the empirical data collected in the company. This knowledge about the maintenance processes is available to the company staff for usage in everyday activities. The method can be easily tailored to other small organizations.

Keywords: process, characteristics, software maintenance, qualitative methods, inductive thematic analysis.

1. Introduction and background

There are many processes involved in software life cycle. Processes in software engineering practice encompass both technical and managerial activities within the software life cycle. Implementation of software life cycle processes requires appropriate infrastructure with available resources such as staff, tools and funding, as well as assigned responsibilities in the processes. Software processes are creative, feedback driven, adaptable and should be observed within the real context [12]. Since software organizations are based on knowledgeable workers, it is common practice to implement process assessment and improvement activities together with organizational learning activities [18], resulting with increased awareness of importance of knowledge assets for overall business performances of organizations [13].

Software maintenance is a set of activities aimed at providing cost-effective support to software [6]. Although it has been recognized as the most costly part of software life
Zeljko Stojanov and Jelena Stojanov

cycle it does not attract enough attention comparing to development [19,26].
Maintenance activities enable: controlling software’s day-to-day functions, controlling software modifications, perfecting existing functions, dealing with security threats, and preventing problems and performance degradation. Since software maintenance is an ongoing process aimed at keeping software useful, the imperative in software maintenance is to reach managed process that will reduce errors through the software life cycle [3]. However, April et al. [2] reported that many software organizations do not have defined processes for their software maintenance activities.

Qualitative research methods have been recently adopted in software engineering for exploring and illuminating everyday practice [11]. Qualitative methods enable deeper understanding of practice and processes from the perspective of involved people, with particular emphasis on social issues in the practice. Qualitative methods have been used in software maintenance for exploring: how developers manage relevant information while solving maintenance tasks [15], documentation usage [17], development agile methodology [20], or effects of inter-smell relations on maintainability [33].

It is well-known that software industry consists of mostly small organizations that rarely adopt best practice proposals, but rather develop their own methods for managing maintenance activities [4,20]. In addition, due to the complexity of software maintenance tasks, many of development techniques, tools, models and processes cannot be directly applied in maintenance [21]. Anquetil et al. [1] stated that the fundamental problem in software maintenance is the lack of various types of knowledge, which is usually only in heads of software engineers. In this course of research, several ontologies, typologies and frameworks were proposed for systematizing knowledge on software maintenance processes [14,10,23,1,32]. All these models and frameworks have been developed in order to direct empirical research in software maintenance, or are suitable for larger organizations that have resources for implementing them in the practice. Therefore, systematization of knowledge that will facilitate knowledge reuse, especially in small software organizations, can help maintainers to more efficiently cope with difficulties in everyday maintenance activities. These observations suggest that further research in the area of software maintenance processes is necessary.

This paper presents a method suitable for identifying characteristics of software maintenance processes in small software organizations, as well as a case study in a micro software company, resulting with developed thematic framework with systematized characteristics of software maintenance processes. The paper is structured as follows. The next section presents a case study with the description of the research context, methods and findings. The third section contains the discussion of the trustworthiness of the study and the benefits for the company. The last section contains concluding remarks and further research directions.
2. Case study

2.1. Context

A case study was implemented in a local software company focused on local clients in Serbia. According to European commission [8], the company can be classified as a micro enterprise since it has seven employees. The company develops and maintains over 30 business software applications for over 100 clients. Based on the trend analysis of clients’ requests, over 84 percent of the tasks are focused on maintenance activities [28], which strongly emphasizes the importance of software maintenance for the overall business performance of the company. Based on that, the company management recognized the importance of assessing and improving software maintenance processes. In this course of thinking, the company implemented software maintenance process improvement project, as well as identification and systematization of knowledge related to software maintenance activities [31].

By having in mind the well known constraints of small software companies [22,16], the method for identifying and systematizing knowledge was developed as a bottom-up. This means that the identification of knowledge starts with the investigation of the real state of the practice in the company, without attempting to fit the research process and findings to any prescribed strategy, directive, standard or guideline. This method relies on the company staff, who have the best insight into everyday practice. Their knowledge and experience are of the crucial significance for the identification and systematization of the most relevant knowledge. In addition, the method requires full commitment of the company’s management. This approach ensures availability of all necessary resources in the company during the research process.

2.2. Methods

Characteristics of software maintenance processes are identified by using a Lightweight Inductive Method for Knowledge Identification and Systematization (LIM4KIS) [30]. The method enables knowledge identification and systematization without disturbing everyday practice in an organization. Inductive thematic analysis proposed by Braun and Clarke [7] serves as a method for the data analysis and development of the thematic knowledge framework. Fig. 1 presents iterative process of creating knowledge framework about software maintenance practice in the company.

The most comprehensive understanding of the practice assumes combining a variety of different data sources, and using both qualitative and quantitative methods [9]. The main sources of knowledge for identifying characteristics of software maintenance process are interviews with the employees, observations of the everyday practice, the documents available in the company, and data extracted from the internal repository of maintenance requests (MRs). Based on the variety of data sources, different methods for data analysis were used, such as trend analysis [28], regression analysis [29] and fuzzy screening [27] for quantitative data, and thematic analysis for qualitative data. All these
data were prepared for feedback sessions, which were organized as working meetings in the company. The feedback sessions are essential for the successful implementation of the method. The sessions were chaired by the leading researcher, while other participants were invited based on the current state of the research (e.g. an interviewee whose transcribed interview was prepared for the discussion). The transcripts from the feedback sessions were analyzed by using inductive thematic analysis [7]. Methodological memos were used for elaborating all decisions, while theoretical memos were used for developing theoretical constructs (themes, sub-themes and relationships in the framework) during the whole research process [5].

Fig. 1. Iterative process of creating a thematic knowledge framework for software maintenance practice in the company

2.3. Findings

Maintenance activities are organized in order to solve clients’ requests related to sustaining software products usable. These requests are called maintenance requests (MRs). Two themes in this thematic area relate to the process and the features of MRs: Request features and Processing maintenance requests. Fig. 2 shows thematic area with identified themes and sub-themes of MR processing in the company. This thematic area is a segment of thematic knowledge framework describing the maintenance practice in the company [31]. According to the typology of qualitative findings proposed by Sandelowski and Barroso [24], the findings can be classified as conceptual/thematic description, since they were shaped as a set of developed themes and sub-themes integrated into a thematic framework.
Exploring software maintenance process characteristics

Processing maintenance requests. Few types of software maintenance processes can be distinguished in the company. All maintenance processes, regardless of the type, are recorded in the repository with the same attributes. Typical maintenance processes relate to the modification of software products, including enhancements of software with new functionality or solving reported problems, and providing support to the clients through organized training and technical support. The following sub-themes related to processing of MRs were identified:

- **Stages in process.** Each process is adjusted to the current MR and a client that submitted it, starting from receiving a request, analyzing a request, and implementing necessary tasks. The stages in the process are: receiving a request, checking urgency, checking client’s status, assigning a programmer to the request, implementing necessary work, and finishing documentation.

- **Decisions.** Decisions are regularly made by programmers in all stages and in transitions between stages, while in some cases decisions are expected also by the clients.

- **Classification of requests.** Requests are classified based on the urgency, and after that based on the internal classification of the clients’ organizations.

- **Managing unsolved requests.** Unsolved requests are requests postponed due to the high occupancy of the programmers or inappropriate request specification provided by a client. These requests are either scheduled for solving in the near future or rejected.

- **Communication between stakeholders.** Communication between the programmers and the clients is essential for efficient processing of MRs and occurs during all stages in the process. It includes variety of techniques such as email, phone calls, meetings, etc.

- **Software tools.** Several software tools are used for supporting technical and organizational aspects of MR processing. The most important software tool for processing MRs is internal web based application for managing requests and associated tasks.

Request features. Efficient processing of MRs is based on a conceptual model that includes all relevant attributes related to clients, programmers, tasks and invoices. The
conceptual model is based on several years of experience in solving the clients requests. The following sub-themes related to the features of MRs were identified:

- **Request attributes** are divided in two sets. The first set of attributes defines a MR, which includes its description, the reason for the request, the priority (high, medium and low), and a client that submitted the request. The second set of attributes relates to processing of a MR, with the attributes related to the critical times in processing a MR, the evidence of the performed work, and the information about the assigned programmer.

- **Types of requests** are distinguished for classifying requests. The classification includes modification requests that includes requests for enhancements and problem reports, and requests for technical support including requests for administration of software systems, requests for training and requests for technical assistance to the clients.

3. **Discussions**

3.1. **Benefits for the company**

Active engagement of the company management and the staff in the whole research process resulted with several benefits for them. The first benefit relates to identification and systematization of knowledge on software maintenance processes, which has become available to the staff. The next benefit relates to the increased self importance and motivation of the staff involved in the research. And finally, deep and overall knowledge about processes enables identification of potential improvements.

3.2. **Trustworthiness**

The validity and rigor of qualitative research is based on ensuring that trustworthiness criteria, such as credibility, neutrality and transferability [25]. In this study, the credibility, or internal validity, was increased through careful application of inductive thematic analysis method supported by rich description of the research process and findings, triangulation of data sources, and active participation of the company staff in data analysis. The research findings are grounded in the data collected in the company and validated by the employees, which ensures neutrality of the researchers, whose work served only for creating the most faithful representation of knowledge about the practice.

The main threat to the trustworthiness of this study is transferability of the research findings. However, the aim of this study is not to provide the findings relevant for all similar software organizations, but rather to provide guidelines how to organize a study resulting with the identified characteristics of a selected segment of practice. In
addition, the thick description of used research methods provides guidelines for organizing the similar researches in other small organizations.

4. Conclusions

This paper presents a study aimed at identifying characteristics of software maintenance processes in a micro software company. The study is based on inductive method for identifying and systematizing knowledge about the selected segment of practice, assuming active involvement of the company staff. The findings are presented as a thematic knowledge framework, which is available to all staff in the company. The main contribution of this paper is an inductive lightweight method for knowledge identification and systematization, suitable for small organizations. In addition, the presented segment of the framework can be used as a starting point for deeper investigation of specific aspects of software maintenance.

Several possible directions for further work can be distinguished. The first direction includes implementation of the presented method for identifying characteristics of other segments of the practice, such as requirements engineering or testing, and integration of independently developed frameworks into a general one. Implementation of the method in other small organizations will provide evidence about its usefulness. The most promising direction relates to developing mechanisms that will ensure evolution of developed frameworks aimed at ensuring compliance with the changes in the practice.

Acknowledgement

Ministry of Education, Science and Technological Development, Republic of Serbia, partly supports this research under the project "The development of software tools for business process analysis and improvement", project number TR32044, 2011-2016.

References


Exploring software maintenance process characteristics