

Empirical Study of Analysts' Practices in Packaged Software Implementation at Small Software Enterprises

Issam Jebreen and Ahmad Al-Qerem
Faculty of Information Technology
Zarqa University

ijebreen@zu.edu.jo, ahmad_qerm@zu.edu.jo

Abstract: This study investigates the practices of requirements engineering (RE) for packaged software implementation, as enacted by small packaged software vendors (SPSVs). The research findings lead to introduced new methods of documentation, was not as concerned as general RE practice with looking for domain constraints or with collecting requirements and viewpoints from multiple sources, was more likely to involve live software demonstrations and screenshots to validate user needs, and was more likely to involve the compilation of a user manual. In PSI, prioritising requirements is not a basic practice; instead, analysts collect requirements in a circular process, with managers then directing analysts regarding which requirements to direct most attention toward.

Keywords: Requirement engineering; Packaged software implementation; ERP; Analysts' practices SMEs.

1. Introduction

SMEs generally thrive because they have successfully done something unique within a niche market [1]. For this reason, SMEs may seek to protect their competitive advantage by avoiding any standardization encouraged by packaged software (PS) [1]. Packaged software implementation at SMEs therefore involves challenges related to how best to respond to misalignments between the functionality offered by packaged software and the business needs/environment of SMEs [1,2]. This challenge is one of the key issues in packaged software implementation at SMEs. One question related to this issue is which processes Small packaged software vendors (SPSVs) apply in order to identify misalignments between packaged software functionality and client requirements and how they deal with these misalignments. Implementation consists of the customization, installation, configuration and adaptation of the packaged software acquired according to the needs of the organization, and a better gap/fit analysis between the organization's needs and the functionality of packaged software can be achieved through Requirements Engineering (RE).

However, several software engineering researchers have argued that most current requirements engineering practices are unsuitable for SPSVs [3,4] and that SPSVs are unable to successfully apply RE methods and techniques that have been designed for larger companies [4]. For this reason, research into RE practices should focus more on the investigation of what RE is practiced in smaller companies such as SPSVs and on how the RE methods followed by SPSVs can be improved. This study therefore features a case study of what RE practices of packaged software implementation is enacted in Small to Medium-sized Software Development Companies. In this study,

packaged software is defined as ready-made software products that generally require modification or customization for specific organizations. They are often exemplified by enterprise resource planning (ERP) [5].

2. The Research Method

A case study research method was applied in small packaged software vendors who participated in this study. Organization considered in this study is dominated by the provision of packaged software solutions. Case study is a research strategy which focuses on understanding the research hypothesis and theory [6]. It aims to provide description, test theory generate theory. Case study can involve either single or multiple cases, and numerous levels of analysis [7]. Typically the methods for data collection are archives, interviews, questionnaires, and observations [6]. The interest of this study is to provide description from the analysis of the results. The case study is thought to be "ideally suited to the needs and resources of the small-scale researcher" who might work on one organization, or "one element of such an organization" or "one individual, or a small number of individuals" [7]. Yin [7] defines the scope of a case study as follows: A case study is an empirical enquiry that investigates a contemporary phenomenon within its real-life context, especially when the boundaries between phenomenon and context are not clearly evident. Clearly, the case study research method is particularly well-suited to this research since the object of this discipline is the study of information system in organizations and interest has shifted to organizational rather than technical issues.

2.1. Data Analysis

We assess the enacted RE practices according to a framework adopted from Sommerville & Sawyer [8] and Cox et al. [9]. We use four levels of assessment of RE practices (as theorized by Sommerville & Sawyer [8]). These levels of assessment are the following: standardized use, common use, discretionary use, and never used.

- Standardized use (SU): This practice has a documented standard and is always followed as part of the organization's software development process i.e. it is mandatory.
- Common use (CU): This practice is widely followed in the organization but is not mandatory.
- Discretionary use (DU): This practice is used at the discretion of individual project managers. Some may have introduced the practice for a particular project.
- Never used (NU): The practice is never or rarely applied.

The tables use guideline classifications relating to 'good requirements practices' that are suggested by Sommerville & Sawyer [8] and Cox et al. [9] to be 'basic', 'intermediate', or 'advanced'. 'Basic' practices can continually be repeated, and it is possible to estimate costs, time, and resources associated with these practices. In our assessment of the RE practices involved in PSI, 'basic' practices match with 'standardized' use. Meanwhile, 'intermediate' practices are more complex and lead to a 'defined' requirements engineering process. They can be considered as aligning with 'common' use. Lastly, 'advanced' practices are designed to help support continuous improvement within any RE process. Some of these practices involve advanced technology and advanced

methods which require specialist knowledge. They may also involve expectations of and guidelines for organizational change. In our assessment of RE practices, 'advanced' practices align with 'discretionary' use.

3. Result and Discussion

The results discussed in the following section were obtained our results relating to specific RE practices by discussing with analysts a list of practices adopted from Sommerville & Sawyer [8] and Cox et al. [9] and asking them to rate their use of each according to the four levels of assessment derived from Sommerville & Sawyer [8]. This helped us to understand the level of use of each named practice. This was understood by the analysts; however, we did run into the issue that the analysts we spoke to did not always refer to the practices they used using the same terminology as ourselves.

3.1. Requirements Documentation Practices

The requirements document itself is a document that effectively communicates requirements to customers, managers and developers. As can be seen from Table 1, the levels of all requirements documentation practices are considered 'basic' by Sommerville & Sawyer [8] and Cox et al. [9]. When the use of these practices in PSI is investigated, it can be found that many of these practices receive Standardized use in PSI. Thus they are practiced at a very similar level to that suggested by Sommerville & Sawyer [8] and Cox et al. [9].

Table 1 Requirements documentation in PSI
Requirements Documentation Practices

No	RE Practices	Type	PSI
RD1	Define a standard document structure	Basic	Standardised use
RD2	Explain how to use the document	Basic	Common use
RD3	Include a summary of the requirements	Basic	Standardised use
RD4	Make a business case for the system	Basic	Standardised use
RD5	Define specialised terms	Basic	Discretionary use
RD6	Make document layout readable	Basic	Common use
RD7	Help readers find information	Basic	Common use
RD8	Make the document easy to change	Basic	Common use

The results show that the most common standardized requirements documentation practices are to define a standard document structure (RD1), to include a summary of the requirements (RD3), and to make a business case for a project (RD4). The practices of explaining how to use the document (RD2), making the document layout readable (RD6), helping readers find information (RD7), and making the document easy to change (RD8), can be considered as 'Common use' practices in PSI. This means that these practices are widely followed in the organizations but are not

mandatory. We also found that PSI documentation practices approached defining specialized terms (RD5) with 'discretionary use'. These findings regarding requirements documentation practices were drawn primarily from the descriptive analysis forms.

We also discovered a range of new practices that were carried out, that are related to requirements documentation. We have listed these 'New Requirements Documentation Practices' in Table 2.

Table 2 New Requirements documentation Practices - PSI

New Requirements Documentation Practices			
No	RE Practices	Type	PSI

RD9	Users' needs/Misalignments specification document	Basic	Standardised use
RD10	Estimating time needed for users' needs document	Basic	Standardised use
RD11	Estimating cost needed for users' needs document	Basic	Standardised use
RD12	Include users' needs validation document	Basic	Standardised use

These new practices involve creating a users' needs/misalignments specification document (referred to as RD9) that analysts sent to developers within their software company. When analysts sent a specification document/request form to developers within their company, Other new practices that the analysts told us of are estimating the time related to creating the users' needs/misalignments document (RD10) and estimating the cost needed for creating a users' needs document (RD11). The scoping process carried out before beginning to develop any software involves software analysis through the discussion of high level modification requirements and new features.

All of these practices are a part of PSI, being practiced with 'standardized use'. Despite their being used so often and the fact that these practices are perceived as having a high value, these practices have not been identified during previous studies of RE and packaged software RE; this study therefore extends the

current framework for requirements practices in term of PSI.

3.2. Requirements Elicitation Practices

Requirements elicitation is defined as a group of practices designed to help discover the requirements for a system. These practices are followed by analysts in order to elicit requirements from the stakeholders related to the system. However, the requirements elicited also depend on the application domain and on the organizational and operational environments of the system.

In Table 3 we see that, several RE elicitation practices are carried out at the 'basic' level; that is, they are almost always practiced. Practices RE1 through RE6 are practiced with 'standardized' use in PSI. However, just over half of the practices operate at the 'intermediate' or 'advanced' levels. Most of those practices that are 'basic' in RE are standard practices in PSI.

Table 3 Requirements elicitation in PSI

Requirements Elicitation Practices			
No	RE Practices	Type	PSI
RE1	Assess system feasibility	Basic	Standardised use
RE2	Be sensitive to organisational and political consideration	Basic	Standardised use
RE3	Identify and consult system stakeholders	Basic	Common use
RE4	Record requirements sources	Basic	Common use
RE5	Define the system's operating environment	Basic	Common use
RE6	Use business concerns to drive requirements elicitation	Basic	Standardised use
RE7	Look for domain constraints	Intermediate	Discretionary use
RE8	Record requirements rationale	Intermediate	Common use
RE9	Collect requirements from multiple viewpoints	Intermediate	Discretionary use
RE10	Prototype poorly understood requirements	Intermediate	Standardised use
RE11	Use scenarios to elicit requirements	Intermediate	Standardised use
RE12	Define operational processes	Intermediate	Discretionary use
RE13	Reuse requirements	Advanced	Standardised use

RE practices RE7 through RE13, shown in Table 3, are practiced rather differently in PSI than how they are presented by Sommerville & Sawyer [8] and Cox et al. [9]. In general RE, a large range of practices could be considered as 'intermediate' practices, that is, they are more complex and not always practiced.

Those practices regarded as 'intermediate' include looking for domain constraints, recording the requirements rationale, collecting requirements from multiple viewpoints, prototyping poorly understood requirements, using scenarios to elicit requirements, and defining operational processes. In PSI, however, these practices are carried out at a range of levels. For example, prototyping poorly understood requirements and using scenarios to elicit requirements are carried out as standardized practices; this difference occurs because software is already developed and might not offer a perfect fit with user requirements.

Looking for domain constraints (RE7), collecting requirements from multiple viewpoints (RE9), and defining operational processes (RE12) are practices that are only carried out with discretionary use in PSI. We can see why looking for domain constraints would only be carried out with discretionary use in PSI when we consider that SPSVs generally work with SME clients. For the same reason, collecting requirements from multiple viewpoints also has discretionary use - if a software development company is working for an SME client, the number of users to be considered is small. One other requirements elicitation practice in this group, recording the requirements rationale (RE8), can be considered as having 'common' use in PSI.

We also identified some new requirements elicitation practices used during PSI. The new practices, listed in Table 4, are using a live software demonstration to elicit the users' needs (RE14), and

using a user manual (RE15). According to Sommerville & Sawyer [8] and Cox et al. [9], these practices are carried out at the 'basic' level in general RE. Our findings showed them to have 'standardized' use in PSI. It is interesting that the practice of holding live software demonstrations (RE14) is repeatedly used in PSI. The reason for frequent use of the live software

demonstration in PSI is that it educates users about the software's functionalities, helps to increase users' participation in discussions, and helps analysts to discover and discuss user needs and any misalignments between these needs and what the software product or system can do [10].

Table 4 New Requirements elicitation Practices - PSI

New Requirements Elicitation Practices			
No	RE Practices	Type	PSI
RE14	Use live software demonstration to elicit users' needs	Basic	Standardised use
RE15	Use a user manual	Basic	Standardised use

In addition, a live software demonstration may help to convince users that there are alternative solutions to any misalignments that are identified [1]: the live software demonstration can be used to show work-around [12]. In packaged software implementation, work-around are used with the intention of minimizing customization, not in order to reduce conflicts between requirements. In the software organization we noticed, 'work-around' were used when the analysts tried to convince clients to use the software as it already was, rather than ask for a full-scale customization. This could be deemed a 'work-around' because the client does end up getting the function or result they want, just not in the particular order they desired, while the software organization avoids having to customize a function. But this kind of work-around is only possible if the client's customization request responds to what is only a 'perceived' misalignment rather than a 'real' one. If a required function is totally missing from the software or can't somehow be supplied by the existing software, then the function has to be built.

4. Conclusion

This study presents an account of packaged software implementation requirements engineering with a focus on analysts' practices within SPSVs. Through this study we offer extensive discussion of how small packaged software vendors approach, and the challenges encountered when conducting the implementation of packaged software at SMEs. We have discussed the different factors involved with finding clients, eliciting requirements and identifying misalignments, creating a packaged software offer, and modifying or customizing existing packaged software. There are three new practices connected to 'Requirements Documents Practices': creating a users' needs/misalignments specification document, estimating the time and cost related to creating the users' needs/misalignments document, and the use of a users' needs validation document. Amongst other new findings, new requirements elicitation practices for PSI were also found, including using a live software demonstration to elicit the users' needs and making use of a user manual. Some of the elicitation practices in PSI that were found included using software

demonstrations, creating live scenarios for testing packages, and prototyping poorly understood requirements. The way in which analysts were expected to priorities requirements in PSI was also considered. It was discovered that analysts working with packaged software collect requirements in a circular process and develop those requirements that are agreed upon at the time (with clients) or that their managers award priority to. In PSI, therefore, prioritizing requirements is a practice used at the discretion of project managers.

References

- [1] O. Zach and B. E. Munkvold, "Identifying reasons for ERP system customization in SMEs: a multiple case study," *Journal of Enterprise Information Management*, vol. 25, pp. 462-478, 2012.
- [2] B. Light, "Going beyond 'misfit' as a reason for ERP package customisation," *Computers in Industry*, vol. 56, 2005.
- [3] I. Jebreen and R. Wellington, "Understanding Requirements Engineering Practices for Packaged Software Implementation," presented at the 4th IEEE International Conference on Software Engineering and Service Sciences, Beijing, China, 2013.
- [4] T. Merten, et al., "Towards a new understanding of small and medium sized enterprises in requirements engineering research," in *Requirements Engineering: Foundation for Software Quality*, ed: Springer, 2011, pp. 60-65.
- [5] L. Xu and S. Brinkkemper, "Concepts of Product Software: Paving the Road for Urgently Needed Research," in *Proceedings of the First International Workshop on Philosophical Foundations of Information Systems Engineering*, 2005.
- [6] K. M. Eisenhardt, "Building theories from case study research," *Academy of management review*, vol. 14, pp. 532-550, 1989.
- [7] R. K. Yin, *Case study research: Design and methods* vol. 5: SAGE Publications, Incorporated, 2008.

- [8] I. Sommerville and P. Sawyer, *Requirements engineering: a good practice guide*: John Wiley & Sons, Inc., 1997.
- [9] K. Cox, *et al.*, "Empirical study of Sommerville and Sawyer's requirements engineering practices," *Software, IET*, vol. 3, pp. 339-355, 2009.
- [10] S. K. Sia and C. Soh, "An assessment of package–organisation misalignment: institutional and ontological structures," *European Journal of Information Systems*, vol. 16, 2007.
- [11] M. Al-Mashari and A. Al-Mudimigh, "ERP implementation: lessons from a case study," *Information Technology & People*, vol. 16, 2003.
- [12] H. M. Khoo, *et al.*, "An exploratory study of the impacts of upgrading packaged software: a stakeholder perspective," *Journal of Information Technology*, vol. 26, 2011.