Adoption of Requirements Engineering Practices in Malaysian Software Development Companies

Badariah Solemon[1], Shamsul Sahibuddin[2], and Abdul Azim Abd Ghani[3]

[1] College of IT, Universiti Tenaga Nasional, Km 7 Jalan Kajang-Puchong, 43009 Kajang, Selangor, Malaysia
badariah@uniten.edu.my
[2] Universiti Teknologi Malaysia, Kuala Lumpur, Malaysia
shamsul@utm.my
[3] Universiti Putra Malaysia, Selangor, Malaysia
azim@fsktm.upm.edu.my

Abstract. This paper presents exploratory survey results on Requirements Engineering (RE) practices of some software development companies in Malaysia. The survey attempted to identify patterns of RE practices the companies are implementing. Information required for the survey was obtained through a survey, mailed self-administered questionnaires distributed to project managers and software developers who are working at software development companies operated across the country. The results showed that the overall adoption of the RE practices in these companies is strong. However, the results also indicated that fewer companies in the survey have use appropriate CASE tools or software to support their RE process and practices, define traceability policies and maintain traceability manual in their projects.

Keywords: Requirements Engineering (RE), RE practices, adoption.

1 Introduction

Good RE practices, according to Davis and Zowghi [1], can "either reduces the cost of the development project or increases the quality of the resulting project when used in specific situation". There exist in the literature several empirical research that study the RE practices in different companies in different parts of the world. Such empirical research include those by El Emam and Madhavji in 1995 [2], a study of 60 (12 interviews and 48 document inspection) cases in Canada; Nikula and colleagues. in 2000 [2], a survey of 15 respondents in twelve small and medium enterprises (SMEs) in Finland; Neill and Laplante in 2003, as explained in [4], a survey of 194 respondents from diverse mix of industries in US; and Damian and colleagues in 2004 [5], a study within a single Australian company. Findings from these studies not only show the different types of RE practices implemented but also provide empirical evidence that improved RE practices help in improving the software development projects.

However, results from most of these surveys may not appropriate to be generalized from such a relatively small samples used. In addition, the situation in Malaysia was

T.-h. Kim et al. (Eds.): ASEA 2010, CCIS 117, pp. 141–150, 2010.

not quite known as there was no research to study both the state of the RE problems and RE practices in the country. Hence, we designed and conducted a survey to find out the current RE problems and practices amongst these software development companies. The survey is adapted from three surveys as reported in [2], [6], and [7]. One of the objectives of the survey is to investigate the pattern of RE problems and practices amongst the software development companies in Malaysia. In this paper, however, the focus is only at presenting the pattern of RE practices implemented in the software companies. In the next two sections, the materials and data collection methods, and the results on valid responses, and the analyses performed to interpret the results of the study are explained.

2 Materials and Methods

To help investigate the patterns of current RE problems and practices, the survey method was used. Self-administered, mailed questionnaire was chosen as the instrument of the survey. The questionnaire was constructed with four sections to capture the required information: Section A, Section B, Section C and Section D. Section A focuses at the background information on the companies, the respondents' profile and attitude to the importance of the RE process and software process improvement (SPI). Section B was intended to identify RE problems, while Section C was looking for information related to the adoption and adequacy of RE practices implementations in the companies. Finally, Section D allows respondents to add other RE practices not addressed in the questionnaire but are important to ensure software development project success in their organizations, and to comment on any aspect they choose. This paper aims to present only the results of section C. Meanwhile, results of the first two sections have been reported in other articles including in [8], [9], and [10].

The survey questionnaires were mailed to 500 randomly selected samples of software development companies in Malaysia. A software development company in the survey is defined following the definition by Sison and colleagues [11] as "[...] a for-profit organization whose main business is the development of software for customers external to the organization. Included in this definition are developers of generic software products (e.g., Microsoft) as well as providers of software solutions for specific domain (e.g., Infosys). Excluded in this definition are IT departments that cater primarily to the needs of the organization of which they are part". As mentioned in [8], the survey data was collected through February and March 2008. Although a total of 90 responses were received, making up 18% response rate, only 64 responses are complete and considered valid for analysis. Most are excluded mainly due to incomplete answers. Although this response rate is fairly low, we decided to proceed with analyzing the responses because according to [12], a low response rate of about 5% would already be sufficient for an exploratory study of this kind. Moreover, the 13% response rate is virtually the same with that reported in [11].

3 Results and Discussion

As mentioned earlier, Section C of the questionnaire was looking for information related to the adoption and adequacy of RE practices implementations in the

companies. The information was obtained using a revised list of 72 RE practices items. These items are categorized into 6 key practices: requirements elicitation (Item 3.1-Item 3.13), requirements analysis and negotiation (Item 4.1 – Item 4.9), requirements specifications and documentation (Item 5.1 – Item 5.20), requirements verification and validation (Item 6.1 – Item 6.9), requirements management (Item 7.1 – Item 7.10), and other practices (Item 8.1 – Item 8.11). The RE practices were gathered from the literature such as [7], [13], [14], [15], [16], and [17]. Each item in the list uses a 0 to 3 Likert scale, which will be scored based on the REAIMS assessment (0 = never, 1 = used at discretion of project manager, 2 = normal used, 3 = standardized) as proposed in [14]. A zero scale also mean a non-adoption of the RE practice, whereas the other scale numbers represent different adequacy level of the RE practice implementation in the respondent's organization. However, due to limited space, this paper focuses at presenting only the results related to the adoption level of the RE practices.

3.1 Adoption of RE Practices

First, we queried respondents on their requirements elicitation practices as this key practice is an especially critical part of the RE process [18]. Table A-1 in Appendix A summarizes the results on requirements elicitation practices. The mean overall adoption level of the requirements elicitation practices in the companies in our survey is 96.03%, which is expectably high for such important practices. All 100% of the responding companies define the system's operating environment (Item 3.5), use business concerns to drive requirements elicitation (Item 3.6), and define operational processes (Item 3.12) as part of their requirements elicitation practices. The practices with the lowest three adoption level are prototype poorly understood requirements (Item 3.10), be sensitive to organizational and political considerations (Item 3.2), and look for domain constraints (Item 3.7).

Next, table A-2 summarizes the results of the requirements analysis and negotiation practices. From this table, the mean overall adoption level of the requirements analysis and negotiation practices is relatively high at 89.58%. It can be observed that the most top adopted requirements analysis and negotiation practice is define system boundaries (Item 4.1). The second most adopted practice in this key practice is Item 4.6, prioritize requirements. This is followed by Item 4.7, classify requirements into classes or sections. Meanwhile, the lowest three adopted requirements analysis and negotiation practices are use interaction matrices to find conflicts and overlaps (Item 4.8), provide software or tools to support negotiations (Item 4.4), and assess requirements risks (Item 4.9).

Table A-3 lists the adoption level of the requirements specification and documentation practices. The mean overall adoption level of the requirements specification and documentation practices is high too (92.7%). At least one out of 20 requirements specification and documentation practices (i.e., apply the standard structure to all of your requirements documents) is rated with 100% adoption by the responding companies. The top three most adopted practices are Item 5.3 which is apply the standard structure to all of your requirements, Item 5.8 which is layout the document for readability, and Item 5.12 which is model the system architecture showing the entire system, subsystem and the links between them. Whilst the bottom

three adopted practices are Item 5.2 which is tailor a requirements standard such as IEEE Std 830.1998, Item 5.20 which is specify non-functional requirements quantitatively, and Item 5.15 which is shows traceability between stakeholder requirements and system models.

From table A-4, it can be observed that the mean overall adoption level of the requirements verification and validation practices is 89.06%. All 100% of the responding companies check that the requirements document meets their standard (Item 6.1). The other two top most adopted practices in this RE key practices include perform requirements reviews (Item 6.2), and produce requirements test cases (Item 6.8). Meanwhile, the lowest three adopted requirements verification and validation practices are Item 6.7 which is write a draft user manual at early stage in the process, Item 6.4, which is set-up inspections team and perform formal inspections for requirements and Item 6.9, which is paraphrase system models.

Table A-5 summarizes the adoption level of the requirements management practices. The mean overall adoption level of the requirements management practices (87.97%) is the lowest adoption level of all the 6 key practices. Nevertheless, Item 7.1, Uniquely identify each requirement is being implemented by all of the responding companies). Meanwhile, the other two highly adopted requirements management practices are Item 7.10, Define the real (actual) requirements, and Item 7.7, Identify reusable system requirements. The bottom three adopted requirements management practices are Item 7.5, Use a database to manage requirements, Item 7.4, Maintain traceability manual, and Item 7.3, Define traceability policies.

Finally, it can be observed from Table A-6 that the mean overall adoption level of the other RE practices is high too (89.90%). The top three adopted RE practices under this key practice category are define the requirements process using best practices (Item 8.5), assign skilled project managers and team members to RE activities (Item 8.3), and establish and utilize a joint team responsible for the requirements (Item 8.4). The lowest adopted practice under this key practice category is use appropriate CASE tools for RE support and others (Item 8.1), (Item 8.11), and (Item 8.2).

3.2 Top and Lowest Ten RE Practices

The mean overall adoption level of the all the RE practices presented is high (91.21%). Table 1 summarizes the top and lowest 10 RE practices adopted in the responding companies. At least six out of 72 RE practices are rated with 100% adoption by the responding companies. These practices include define the system's operating environment (Item 3.5), use business concerns to drive requirements elicitation (Item 3.6), define operational processes (Item 3.12), apply the standard structure to all of your requirements documents (5.3), check that the requirements document meets your standard (Item 6.1), and uniquely identify each requirements (Item 7.1). Data in the table also suggest that majority of these top 10 RE practices adopted are actually categorized under the requirements elicitation, and requirements specification and documentation key practices. At least two out of the top ten RE practices fall under the requirements verification and validation, and requirements management key practice categories.

According to the survey results, three of the lowest RE practices are related to adoption of software or tool to support RE process and its practices. They are Item 8.1

which is use appropriate CASE tools for RE support and others, Item 4.8 which is use interaction matrices to find conflicts and overlaps requirements and system models, and Item 4.4 which is provide software or tools to support negotiations. It is also interesting to see that at least two out of the ten lowest RE practices are attributed to the traceability related practices i.e., maintain traceability manual (Item 7.4), and define traceability policies (Item 7.3).

Table 1. Top and lowest ten RE practices

Top ten practices	Lowest ten practices
3.5 Define the system's operating environment	5.2 Tailor a requirements standard such as IEEE Std 830.1998
3.6 Use business concerns to drive requirements elicitation	8.1 Use appropriate CASE tools for RE support and others
3.12 Define operational processes	8.2 Use interaction matrices to find conflicts and overlaps
5.3 Apply the standard structure to all of your requirements documents	7.5 Use a database to manage requirements
6.1 Check that the requirements document meets your standard	6.7 Write a draft user manual at early stage in the process
7.1 Uniquely identify each requirements	7.4 Maintain traceability manual
5.8 Layout the document for readability	4.4 Provide software or tools to support negotiations
7.10 Define the real (actual) requirements	5.20 Specify non-functional requirements quantitatively (put a figure on)
3.3 Identify and consult system stakeholders	7.3 Define traceability policies
5.12 Model the system architecture showing the entire system, subsystem and the links between them	6.4 Set-up inspections team and perform formal inspections for requirements

3.3 Findings

The high mean overall adoption level of the all the RE practices, as presented earlier, suggests that the overall adoption of the practices in these companies is actually strong. It also indicates that the companies had implemented almost all of the practices. This is true especially for practices that received 100% rating such as define the system's operating environment, use business concerns to drive requirements elicitation, define operational processes, apply the standard structure to all of your requirements documents, check that the requirements document meets your standard, and uniquely identify each requirements. Data in the table also suggest that majority of these top 10 RE practices are actually categorized under the requirements elicitation, and requirements specification and documentation categories. Two of the top ten activities fall under the requirements verification and validation, and requirements management categories.

Meanwhile, the results in the survey also suggest that there is a relatively low adoption of software or CASE tool to support RE process and practices similar to those findings discovered in other related studies (e.g., [19], and [20]). This low adoption may be due to reasons such as cost, lack of measurable returns and unrealistic expectations as suggested in [21]. According to Kannenberg and Saiedian [22],

requirements traceability has been demonstrated to provide many benefits to software companies. In spite of the benefits that traceability offers to the RE process and software engineering industry, less companies in the survey define traceability policies or maintain traceability manual in their projects. Perhaps this has to do with the low CASE tool adoption amongst these practitioners as poor tool support is the biggest challenge to the implementation of requirements traceability as discussed in [22].

4 Conclusion

As presented earlier, this part of the survey has identified and analyzed the adoption level of RE practices implementation specifically in software development companies operated in Malaysia. The overall adoption of the RE practices, which was described according to the 6 key RE practices, has been presented in this paper. Also, the top ten and the bottom ten RE practices adopted in these companies have been listed. It concludes that the overall adoption of the RE practices in these companies is strong. However, the results also indicated that fewer companies in the survey have use appropriate CASE tools or software to support their RE process and practices, define traceability policies and maintain traceability manual in their projects.

As stated in [23], "surveys are of course based on self-reported data which reflects what people say happened, not what they actually did or experienced". Because we surveyed project managers and software developers, the results are limited to their knowledge, attitudes, and beliefs regarding the RE problems, RE practices and the software development projects with which they have taken part. While every care has been taken to ensure the validity and reliability of the information gathered, its representativeness cannot be 100% guaranteed as the data were obtained from sampled population. To mitigate the possible threats to empirical validity of the survey results, several measures have been taken to evaluate the four criteria (i.e., construct validity, internal validity, external validity, and reliability) for validity suggested in [24]. While we would not assume that the survey results are typical of all software development companies, we believe that they are reasonably typical of software development companies in Malaysia.

Acknowledgement

We would like to thank the respondents for their cooperation and willingness to share their experiences and data. Also, we would like to give special acknowledgment to the Universiti Tenaga Nasional, which provided funding for this work through grant number J5100-14-018.

References

1. Davis, A.M., Zowghi, D.: Good Requirements Practices are neither Necessary nor Sufficient. In: Requirements Eng., vol. 11, pp. 1–3. Springer, London (2006)
2. El Emam, K., Madhavji, N.H.: A Field Study of Requirements Engineering Practices in Information Systems Development. In: 2nd IEEE International Symposium of Requirements Engineering, pp. 68–80. IEEE Press, New York (1995)

3. Nikula, U., Sajaniemi, J., Kalviainen, H.: A State-of-the-Practice Survey on Requirements Engineering in Small- and Medium-Sized Enterprises. Technical Report. Telecom Business Research Center, Lappeenranta University of Technology, Finland (2000)
4. Laplante, P.A., Neill, C.J., Jacobs, C.: Software Requirements Practices: Some Real Data. In: Proceedings of the 27th Annual NASA Goddard/IEEE Software Engineering Workshop, SEW-27 2002 (2003)
5. Damian, D., Zowghi, D., Vaidyanathasamy, L., Pal, Y.: An industrial case study of immediate benefits of requirements engineering process improvement at the Australian Center for Unisys Software. Empirical Software Engineering Journal 9(1-2), 45–754 (2004)
6. Beecham, S., Hall, T., Rainer, A.: Software Process Improvement Problems in Twelve Software Companies: An Empirical Analysis. Empirical Software Engineering 8(1), 7–42 (2003)
7. Niazi, M., Shastry, S.: Role of Requirements Engineering in Software Development Process: An Empirical Study. In: IEEE INMIC 2003, pp. 402–407 (2003)
8. Solemon, B., Sahibuddin, S., Ghani, A.A.A.: Requirements Engineering Problems and Practices in Software Companies: An Industrial Survey. In: International Conference on Advanced Software Engineering and Its Applications, ASEA 2009 Held as Part of the Future Generation Information Technology Conference, FGIT 2009, Jeju Island, Korea, December 10-12, pp. 70–77 (2009)
9. Solemon, B., Sahibuddin, S., Ghani, A.A.A.: Requirements Engineering Problems in 63 Software Companies in Malaysia. In: International Symposium on Information Technology 2008 (ITSIM 2008), August 26-28 (2008)
10. Solemon, B., Sahibuddin, S., Ghani, A.A.A.: An Exploratory Study of Requirements Engineering Practices in Malaysia. In: 4th Malaysian Software Engineering Conference, Universiti Malaysia Terengganu (UMT), Terengganu (2008)
11. Sison, R., Jarzabek, S., Hock, O.S., Rivepiboon, W., Hai, N.N.: Software Practices in Five ASEAN Countries: An Exploratory Study. In: The 28th International Conference in Software Engineering, ICSE 2006, pp. 628–631. ACM, China (2006)
12. Lethbridge, T.C., Sim, S.E., Singer, J.: Studying Software Engineers: Data Collection Techniques for Software Field Studies. In: Empirical Software Engineering, vol. 10, pp. 311–341. Springer Science + Business Media, Inc., The Netherlands (2005)
13. Kotonya, G., Sommerville, I.: Requirements Engineering. Processes and Techniques. John Wiley & Sons, Chichester (1997)
14. Sommerville, I., Sawyer, P.: Requirements Engineering. A Good Practice Guide. John Wiley & Sons, Chichester (1997)
15. Hofmann, H.F., Lehner, F.: Requirements Engineering as a Success Factor in Software Projects. IEEE Software, 58–66 (2001)
16. Young, R.R.: Effective Requirements Practices. Addison-Wesley, Boston (2001)
17. Beecham, S., Hall, T., Rainer, A.: Defining a Requirements Process Improvement Model. Software Quality Journal 13, 247–279 (2005)
18. Pfleeger, S.L., Atlee, J.M.: Software Engineering. In: Theory and Practice, 3rd edn., Pearson Prentice Hall, New Jersey (2006)
19. Aaen, I., Siltanen, A., Sørensen, C., Tahvanainen, V.-P.: A Tale of Two Countries: CASE Experiences and Expectations. In: Kendall, K.E., Lyytinen, K., DeGross, J. (eds.) The Impact of Computer Supported Technologies on Information Systems Development, IFIP Transactions A-8, pp. 61–91. North-Holland, Amsterdam (1992)
20. Kusters, R.J., ja Wijers, G.M.: On the Practical Use of CASE-tools: Results of a Survey, CASE 1993. In: Proceedings of the 6th International Workshop on CASE, Singapore, pp. 2–10. IEEE Computer Society Press, Los Alamitos (1993)

21. Lending, D., Chervany, N.L.: The Use of CASE Tools. In: Proceedings of the 1998 ACM Special Interest Group on Computer Personnel Research Annual Conference, Boston, Massachessetts, US, pp. 49–58 (1998)
22. Kannenberg, A., Saiedian, H.: Why Software Requirements Traceability Remains a Challenge. The Journal of Defense Software Engineering (July/August 2009), http://www.stsc.hill.af.mil/crosstalk/2009/07/0907KannenbergSaiedian.html (retrieved July 10, 2010)
23. Verner, J., Cox, K., Bleistein, S., Cerpa, N.: Requirements Engineering and Software Project Success: An Industrial Survey in Australia and the U.S. Australasian Journal of Information Systems 13(1), 225–238 (2005)
24. Easterbrook, S., Singer, J., Storey, M.-A., Damian, D.: Selecting Empirical Methods for Software Engineering Research. In: Shull, F., Singer, J. (eds.) Guide to Advanced Empirical Software Engineering, pp. 285–311. Springer, London (2007)

Appendix A

Table A-1. Adoption of requirements elicitation practices

	Item	Valid Responses	Adoption Level
			%
3.1	Assess system feasibility	64	96.88
3.2	Be sensitive to organizational and political considerations	64	92.19
3.3	Identify and consult system stakeholders	64	98.44
3.4	Record requirements sources	64	96.88
3.5	Define the system's operating environment	64	100.00
3.6	Use business concerns to drive requirements elicitation	64	100.00
3.7	Look for domain constraints	64	92.19
3.8	Record requirements rationale	63	93.65
3.9	Collect requirements from multiple viewpoints (sources)	64	96.88
3.10	Prototype poorly understood requirements	64	87.50
3.11	Use scenarios to elicit requirements	64	96.88
3.12	Define operational processes	64	100.00
3.13	Reuse requirements	64	96.88

Table A-2. Adoption of requirements analysis and negotiation practices

	Item	Valid Responses	Adoption Level
			%
4.1	Define system boundaries	64	98.43
4.2	Use checklists for requirements analysis	64	95.31
4.3	Use operational definitions to define requirements	64	93.75
4.4	Provide software or tools to support negotiations	64	78.13
4.5	Plan for conflicts and conflict resolution	64	89.07
4.6	Prioritize requirements	63	96.83
4.7	Classify requirements into classes or sections	64	96.87
4.8	Use interaction matrices to find conflicts and overlaps	64	70.32
4.9	Assess requirements risks	64	89.06

Table A-3. Adoption of requirements specification and documentation practices

	Item	Valid Responses	Adoption Level
			%
5.1	Define a standard requirements document structure	64	96.87
5.2	Tailor a requirements standard such as IEEE Std 830.1998	64	62.50
5.3	Apply the standard structure to all of your requirements documents	63	100.00
5.4	Explain how to use the document	63	85.94
5.5	Include a summary of the requirements (in an overview section)	63	95.32
5.6	Make a business case for the system (showing the systems part in the business)	63	95.31
5.7	Define specialized terms	63	96.88
5.8	Layout the document for readability	63	98.44
5.9	Make the document easy to change	63	95.31
5.10	Develop complementary system models	64	95.31
5.11	Model the system's environment	64	96.88
5.12	Model the system architecture showing the entire system, subsystem and the links between them	64	98.43
5.13	Use systematic approaches for systems modelling	64	93.75
5.14	Use a data dictionary	64	93.76
5.15	Shows traceability between stakeholder requirements and system models	64	82.81
5.16	Define standard templates for describing requirements	64	90.62
5.17	Use simple and concise language	64	96.87
5.18	Use diagrams appropriately	63	95.31
5.19	Supplement natural language with other descriptions of requirements	63	90.62
5.20	Specify non-functional requirements quantitatively (put a figure on)	64	79.69

Table A-4. Adoption of requirements verification and validation practices

	Item	Valid Responses	Adoption Level
			%
6.1	Check that the requirements document meets your standard	64	100.00
6.2	Perform requirements reviews	64	98.43
6.3	Use multi-disciplinary teams to review requirements	64	85.93
6.4	Set-up inspections team and perform formal inspections for requirements	64	81.26
6.5	Define and document validation checklists	64	92.19
6.6	Use prototyping to validate the requirements	64	89.06
6.7	Write a draft user manual at early stage in the process	64	75.01
6.8	Produce requirements test cases	64	96.88
6.9	Paraphrase system models (convert system models into natural language)	64	82.82

Table A-5. Adoption of requirements management practices

	Item	Valid Responses	Adoption Level
			%
7.1	Uniquely identify each requirements	64	100.00
7.2	Define policies for requirements management	64	85.94
7.3	Define traceability policies	64	81.25
7.4	Maintain traceability manual	64	78.12
7.5	Use a database to manage requirements	64	73.44
7.6	Define change management policies	64	89.06
7.7	Identify reusable system requirements	64	96.88
7.8	Identify volatile requirements	64	92.19
7.9	Record rejected requirements	64	84.38
7.10	Define the real (actual) requirements	64	98.44

Table A-6. Adoption of other RE practices

	Item	Valid Responses	Adoption Level
			%
8.1	Use appropriate CASE tools for RE support and others	63	65.62
8.2	Allocate 15% to 30% of total project effort to RE activities	63	85.93
8.3	Assign skilled project managers and team members to RE activities	63	93.75
8.4	Establish and utilize a joint team responsible for the requirements	63	93.75
8.5	Define the requirements process using best practices	63	95.31
8.6	Use and continually improve a requirements process	63	93.74
8.7	Iterate the system requirements and the system architecture repeatedly	63	93.74
8.8	Use a mechanism to maintain project communication	63	89.06
8.9	Select familiar methods and maintain a set of work products	63	90.62
8.10	Conduct careful and objective post-mortem analysis to project	63	89.06
8.11	Conduct Software Quality Assurance (SQA) activities in this RE process	63	82.81