An Empirical Investigation on a Tool-Based Boilerplate Technique to Improve Software Requirement Specification Quality

Umairah Anuar[1], Sabrina Ahmad[2], Nurul Akmar Emran[3]
Centre for Advanced ComputingTechnology (C-ACT)
Faculty of Information and Communication Technology
Universiti Teknikal Malaysia Melaka (UTeM)
Malacca, Malaysia

Abstract—The process of producing software requirements specification (SRS) is known to be challenging due to the amount of effort, skills and experience needed in writing good quality SRS. A tool-based boilerplate technique is introduced to provide assistance in identifying essential requirements for a generic information management system and translating them into standard requirements statements in the SRS. This paper presents an empirical investigation to evaluate the usability of the prototype. Results showed that the tool-based boilerplate technique has high usability, usefulness and ease of use.

Keywords—Empirical investigation; usability; software requirements

I. INTRODUCTION

SRS play a vital role in software development. It is a fundamental document which consists of a set of requirements that forms the foundation of software development process. The quality of the SRS is crucially important because requirement is a basic of a system the molds the shape of the system the need to be developed. Poor quality SRS does not only lead to poor quality software but also increase in development and sustainment costs which cause major schedule overruns [1].

Based on the importance of the SRS, we reviewed literature from year 2000 to 2014 on what efforts that has been done to reduce the problem arise in SRS and we found out boilerplate is one of the method used to improving the SRS quality. According to the research, a boilerplate technique was adopted as one of the semiformal language because boilerplate is proposed as a bridge between formal and informal specification [2].

Boilerplate is a section of text that can be included in many places with little or no alteration. It is a text that can be reused in new contexts or applications without greatly changed from the original. The same understanding is adapted to the software engineering specialization to particularly produce an SRS document to meet some standard. The boilerplate technique ability to produce requirements statements in some sort of controlled environment is seen beneficial to reduce the possibility of defects. The control is made to handle the flexibility of natural language (NL) which usually leads to ambiguity and inconsistency problems.

The purpose of this paper is to present the findings from an empirical investigation on the usability of the tool-based boilerplate technique. Following Introduction, Section II elaborates on the Tool-based Boilerplate Technique. This is followed by Section III which explains the empirical investigation protocol. Next, Section IV presents the results and Section V elaborates on further analysis. Finally, Section VI concludes the paper.

II. TOOL-BASED BOILERPLATE TECHNIQUE

The term boilerplate is used to refer to the sheet steel used to make boilers in the field of printing [3]. Nevertheless, in computer programming, the term boilerplate is used to represent the section of code. Boilerplate is similar to a template that holds the layout and style information. The main advantage of boilerplate is the reusability, where it can be used in several places in a computer program with little (or no) alteration [4]. Boilerplate also used as a preliminary basis for requirement checking. The usefulness of boilerplate is also claimed by Arora et al. [5] who stated that boilerplate provides a simple yet effective way for increasing the quality of requirement by avoiding complex structure, ambiguity and inconsistency in requirements. Our boilerplate technique is an adaptation from Rupp's boilerplate as shown in Fig. 1 [6]. Any requirement may easily be mapped to the Rupp's boilerplate because Rupp's boilerplate structure has a very wide usability and able to fit most of the requirement.
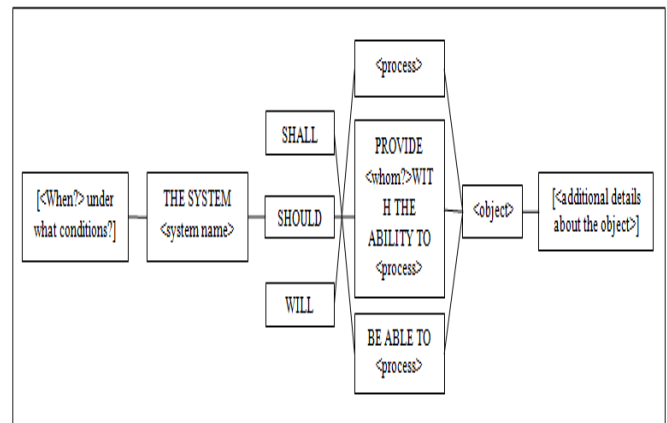

Fig. 1. Rupp's Boilerplate (Taken From [6]).

The Rupp's boilerplate starts with optional condition followed by the system name. This is then followed by the three modal verbs which are shall, should or will. Following modal verbs will be the statement of process which able to come directly after the modal verb without further keyword or and information. The final component is an object with optional additional details about the project.

Derived from Rupp's boilerplate idea, we are motivated to develop a tool-based boilerplate that can improve SRS qualities. By using Rupp's guideline, we develop a prototype of tool-based boilerplate and use it to produce an SRS for information management system. This prototype is based on IEEE template [7] and only covers Section 3.2 of SRS which describes functional requirements. We applied the three modal verbs which are SHALL, SHOULD and WILL to indicate the importance of the requirements. Shall is used to indicate a mandatory requirement. Should is used for non-mandatory requirement and Will is used to indicate a statement which is not subjected to verification.

The aim for this tool is to guide the requirements engineer to write the SRS with consistency. This research scopes the tool prototype to cover Section 3.2 of software requirement specification (SRS) only which covers the elaboration of functional requirements. The tool-based boilerplate is designed for a generic information system management which usually has similar essential requirements. The essential requirements or also known as basic functionalities cover registration, login, searching, adding information and maintaining information with edit, delete and update capabilities. These basic functionalities can be utilized by many information system management and further elaborated to make a complete system. Besides, in order to increase the correctness quality of the SRS, a wireframe interface is auto generated to portray the requirements statements stated in the SRS.

III. EMPIRICAL INVESTIGATION PROTOCOL

This section discusses empirical investigation method. An empirical investigation is conducted to evaluate the usability of the tool-based boilerplate from the perspective of novice user.

A. Identifying Participants

The participants for the empirical investigation were identified among senior undergraduate students enrolling Computer Science Degree at Universiti Teknikal Malaysia Melaka. Seventy third year computer science students majoring in system development from Software Engineering Department were carefully selected. However, only sixty-three of them managed to showed up during the evaluation. The participants were purposely selected based on their background knowledge and familiarity with requirements. They took subjects related to software engineering namely Software Engineering (BITP 2213), Software Requirements Engineering (BITP 2233), Software Architecture and Design (BITP 3243) and Software Verification and Validation (BITP 3253). They were also trained to produce software development documents such as Software Development Plan (SDP), Software Requirements Specification (SRS), Software

Design Document (SDD) and Software Test Description (STD).

B. Instruments

In order to deploy the investigation, the participants were provided with a case study, a tool-based boilerplate technique prototype and a questionnaire. The prototype was developed for the usage of producing a Software Requirement Specification (SRS) for a generic Information Management System. Basic functions of an information management system are mainly the same and can simply adapted to another system. In this research, a library system is referred to as a study case. The feedback is obtained by a questionnaire. The questionnaire is designed to measure the participants' opinion in order to know how far the evaluation meets the objectives. The qualities we measured in this investigation are Usability, Usefulness and Ease of Use. These can further be broken into sub-attributes. We select this three quality attributes because these three qualities are the main qualities that lead to a good SRS [8]. Under the usability attributes, the quality we measured is correctness, consistency, learnability, efficiency and simple. We measure these quality based on TAM [9]. Under usefulness, we measure three sub-attributes which is 1) Accomplish task more quickly and correctly, 2) Easier to do job and 3) Enhance the effectiveness of the output. These sub-attributes that we measure, is the quality that contributes to usefulness. Lastly, the ease of use, we measure three sub-attributes which is 1) Find it easy to use the system, 2) Interaction with the system is clear and understandable and 3) I would find the system easy to use. Under ease of use, we measure the experience that participants gain from using this tool.

The definition of the quality and the sub-quality attributes are as listed below:

- Usability test stated by [10] is measured based on user perception of the tool. In order to evaluate the usability, there are five quality attributes which are Correctness, Consistency, Learnability, Efficiency and Simple.

- Correctness [11] is defined when every requirement stated in the SRS is achieved.

- Consistency [12] is defined as a set of requirements contain no internal contradictions.

- Learnability [13] is defined as the ease and speed with which the users get familiar with the use of a new product.

- Efficiency [13] is defined as the ability to do or produce something without wasting materials, time or energy.

- Simple [14] is defined as not hard to understand, not complex or fancy.

- The usefulness is assessed by the tool ability to accomplish the task more quickly and correctly, easier to do the job with and whether the tool can enhance the effectiveness of the output [13].

- The ease of use [14] is assessed by the interaction of participants with the system is clear or not.

C. The Protocol

The empirical investigation started with preparing the environment for the investigation. The investigation was allocated during lab session and the time was limited within 2 hours. A briefing on the research purpose and explanation on the tool-based boilerplate was made to the participants. A demonstration of the tool-based boilerplate was given too. Then, 15 minutes were given to the participants to read the case study. The case study represented an input to the requirements engineering process as if information gathered earlier in order to write a requirement specification. It basically describes stakeholders' needs for the system to be developed. Then, the tool-based boilerplate is given to the participants to write requirements specification based on the case study given. Once done, the participants were required to provide feedback based on the questionnaire. For each question, the participants need to score each attribute based on the 5 points Likert scale whereby 1 is strongly disagree and 5 is strongly agree. The entire process will take no longer than 2 hours as the software project scope is small.

IV. Empirical Investigation Results

All sixty-three feedbacks were gathered and analysed. This empirical investigation was conducted with aims to get a feedback for the usability, usefulness and ease of use of the tool-based boilerplate technique. In this section, we will present the evaluation results in descriptive and inferential analysis.

A. Results

Based on Fig. 2, the results show positive results with respect to the usability of the tool. The result shows that most of the participants agree that the tool is in high usability (49.8%) and only few participants who disagree (5.2%).
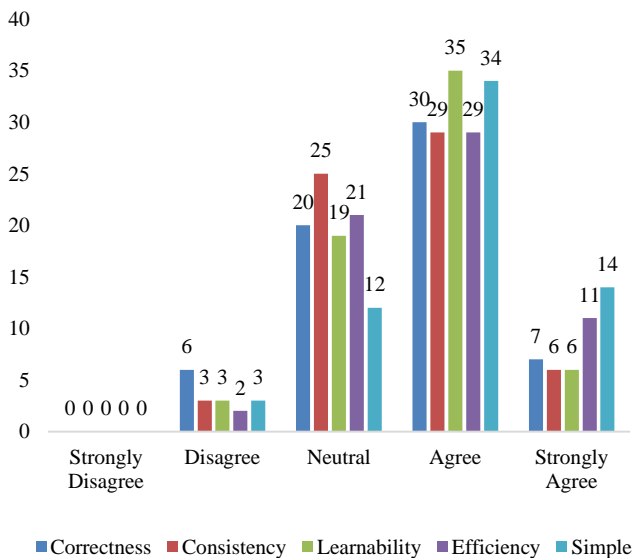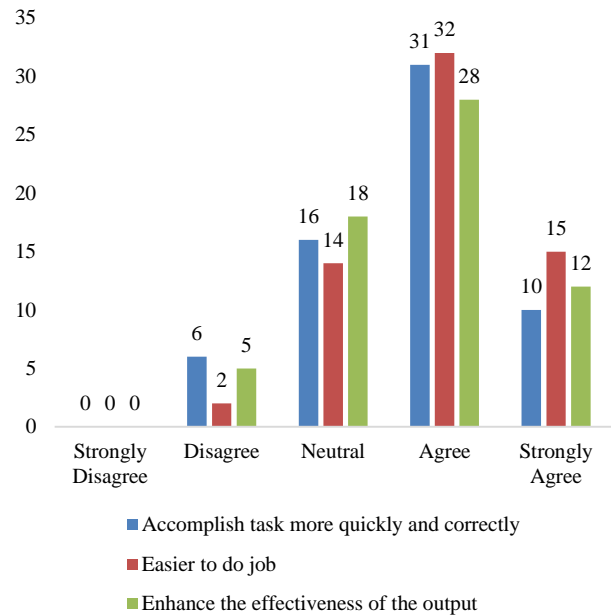

Fig. 3.   Results for Usefulness.

Based on Fig. 3, most of the participants agreed that the tool is useful (48%) based on three attributes which are accomplish task more quickly and correctly, easier to do job and enhance the effectiveness of the output. This result is promising as most of participants gave positive feedbacks.

Fig. 4 shows the evaluation results for Ease of Use. The highest percentage among those five scale is 'Agree' (45.5%). Only one participant strongly disagrees that the interaction with the system is clear and understandable. In summary, the evaluation conducted showed that the tool improves requirement qualities and also can help requirement engineer to capture the requirements with ease.
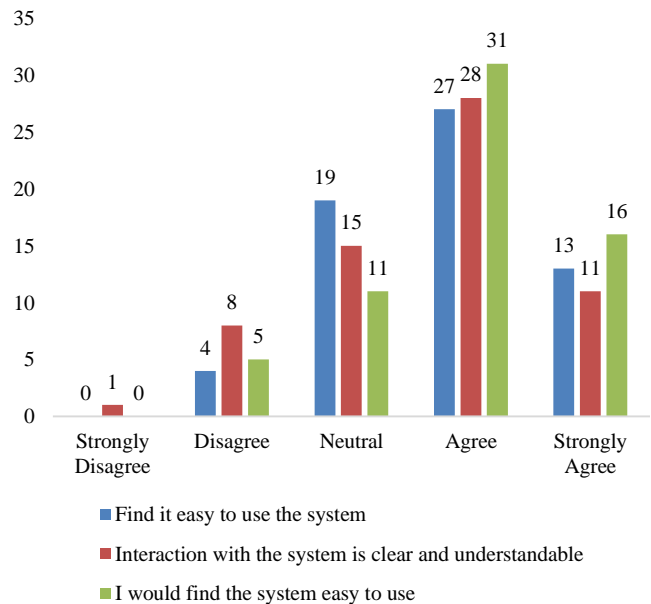

Fig. 2.   Results for Usability.

Fig. 4.   Results for Ease of Use.

V. ANALYSIS

This section presents further analysis on the results derived from the empirical investigation. In addition, results from second evaluation based on expert judgment is included here to show the correlation among the qualities achieved from both evaluation methods. The protocol and details on the expert judgment evaluation can be referred in [15].

A. Descriptive Statistic

We used descriptive statistic to summarize the basic feature data. Measurement includes the mean, median and standard deviation.

Referring to Tables I and II, take a look at mean row. We can see that the mean values for both correctness and consistency from experts' judgements are 4.000 and we can compare that mean values for correctness and consistency from empirical investigation are both 3.603. The mean values show that there is not much difference in correctness and consistency qualities achieved in both evaluations. In empirical investigation, the attributes learnability and simple can be translated into comprehensibility in expert judgment. Table II shows mean values of 3.698 and 3.937 for learnability and simple respectively which has insignificant difference with mean value of comprehensibility attribute (3.900) presented in Table II. The results derived from two evaluations with two different methods were supporting each other and shows that the tool-based boilerplate technique improves SRS quality.

TABLE I. DESCRIPTIVE STATISTIC OF QUALITY ATTRIBUTES DERIVED FROM EMPIRICAL INVESTIGATION

	Empirical Investigation			
	Correct-ness	Consistency	Learn-ability	Simple
N	63	63	63	63
Mean	3.603	3.603	3.698	3.937
Median	4.000	4.000	4.000	4.000
Std. Dev	0.8140	0.8140	0.8140	0.7803

TABLE II. DESCRIPTIVE STATISTIC OF QUALITY ATTRIBUTES DERIVED FROM EXPERT JUDGEMENT

	Expert Judgement		
	Correct-ness	Consistency	Comprehen-sibility
N	10	10	10
Mean	4.000	4.000	3.900
Median	4.000	4.000	4.000
Std. Dev	0.6667	0.4714	0.3162

B. Inferential Statistic

Inferential statistic is prepared to show the relationship between quality attributes derived from both empirical investigation and expert judgement. We run a Pearson's Correlation to measure the relationship of the quality attributes and to indicate if the relationship is either strong or weak.

In this analysis, we were using nominal and ratio for the measurement scale and using bivariate correlation for the scale of measurement. We were using bivariate correlation because it is the suitable statistical test in comparing two sample group and can measure the strength and direction of linear relationship between the two variables which are the quality attributes derived from experts' judgement and empirical investigation.

Table III shows that the Pearson's correlation between Expert Judgement Correctness and Empirical Investigation Correctness is 0.645. This means that there is a strong relationship between results derived from Expert Judgement and Empirical Investigation. Next, we take a look at Sig. (2-Tailed). "Sig." stands for significance level. The value explains if there is a statistically significant correlation between the two variables. In our table, Sig. (2-tailed) value is 0.044. That means, there is a significant difference between results derived from Expert Judgement and Empirical Investigation.

Table IV shows Pearson's Correlation between Expert Judgement Consistency and Empirical Investigation Consistency. We can see that Pearson's r value is 0.488. From the result, we can conclude there is a relationship between Expert Judgement and Empirical Investigation but the relationship is weak. This relationship is not as strong as Correctness. The Sig. (2-tailed) value is 0.153 which is larger than 0.05. That means, there is no significant difference between results derived from Expert Judgement and Empirical Investigation.

TABLE III. PEARSON'S CORRELATION BETWEEN EXPERT JUDGEMENT CORRECTNESS AND EMPIRICAL INVESTIGATION CORRECTNESS

		Correctness EJ	Correctness EI
Correctness EJ	Pearson Correlation Sig.(2-tailed) N	1 10	.645[*] .044 10
Correctness EI	Pearson Correlation Sig.(2-tailed N	.645[*] .044 10	1 63

TABLE IV. PEARSON'S CORRELATION BETWEEN EXPERT JUDGEMENT CONSISTENCY AND EMPIRICAL INVESTIGATION CONSISTENCY

		Consistency EJ	Consistency EI
Consistency EJ	Pearson Correlation Sig.(2-tailed N	1 10	.488[*] .153 10
Consistency EI	Pearson Correlation Sig.(2-tailed) N	.488[*] .153 10	1 63

TABLE V.   PEARSON'S CORRELATION BETWEEN EXPERT JUDGEMENT COMPREHENSIBILITY AND EMPIRICAL INVESTIGATION LEARNABILITY

		Comprehensibility EJ	Learnability EI
Comprehensibility EJ	Pearson Correlation	1	.509
	Sig.(2-tailed)		. 133
	N	10	10
Learnability EI	Pearson Correlation	.509	1
	Sig.(2-tailed)	.133	
	N	10	63

TABLE VI.   PEARSON'S CORRELATION BETWEEN EXPERT JUDGEMENT COMPREHENSIBILITY AND EMPIRICAL INVESTIGATION SIMPLE

		Comprehensibility EJ	Simple EI
Comprehensibility EJ	Pearson Correlation Sig.(2-tailed)	1	.509
	N	10	.133 10
Simple EI	Pearson Correlation	.509	1
	Sig.(2-tailed)	.133	
	N	10	63

Table V shows Pearson's Correlation between Expert Judgement Comprehensibility and Empirical Investigation Learnability. We compare comprehensibility with learnability because they carry the same definition and purpose. The results show the Pearson's r value is 0.506. As the Pearson's r value is close to 1, we can conclude that there is a relationship between comprehensibility and learnability. The Sig. (2 – tailed) value is large than 0.05 which is 0.133 shows that there is no significant difference between results derived from Expert Judgement and Empirical Investigation.

Lastly, Table VI shows Pearson's Correlation between Expert Judgement Comprehensibility and Empirical Investigation Simple. Result shows that learnability (Table V) and simple share the same results. This means that there is a relationship between comprehensibility and simple but there is no statistically significant correlation between results derived from Expert Judgement and Empirical Investigation.

The insignificant difference in the relationship between the quality attributes derived from the two evaluation methods showed that the results are supporting each other to confirm that the tool-based boilerplate technique indeed improve the quality of SRS.

VI. CONCLUSION

This paper presents an empirical investigation to evaluate the usability, usefulness and ease of use of a tool-based boilerplate in order to improve the SRS quality. The art behind the boilerplate technique is explained and the protocol for the empirical investigation is carefully presented. The empirical investigation results showed that the tool-based boilerplate technique does improve the SRS quality in terms of the usability, usefulness and ease of use. In addition, this paper also elaborate on further analysis to compare the results derived from the empirical investigation and another evaluation method which is an expert judgment. The Pearson r value shows that there are relationships between quality attributes achieved from the two evaluation methods. The insignificant correlation between results derived from Expert Judgement and Empirical Investigation show that the results from both methods are confirming each other.

ACKNOWLEDGMENT

REFERENCES

[1] S. Ahmad, I.E.A. Jalil, S.S.S. Ahmad, "An Enhancement of Software Requirements Negotiation with Rule-based Reasoning: A Conceptual Model," Journal of Telecommunication, Electronic and Computer Engineering, vol. 8, no.10, pp. 193-198, 2016.

[2] M. Ortel, M. Malot, A. Baumgart, J.S. Becker, R. Bogusch, S. Farfeleder, N. Gerber, O. Haugen, S. Häusler, B. Josko, J. Mansell, "Requirements Engineering" in CESAR-Cost-efficient Methods and Processes for Safety-relevant Embedded Systems, A. Ajitha, T. Wahl, eds., Springer, pp. 69-143, 2013.

[3] M, Frank Luther, "*A History of American Magazines", v.4*. Cambridge (MA): The Belknap Press of the Harvard University Press. pp. 53–54, 1957.

[4] S. Farfeleder, T. Moser, A. Krall, H. Zojer, and C. Panis, "DODT: Increasing Requirements Formalism using Domain Ontologies for Improved Embedded Systems Development,", 2011.

[5] C. Arora, M. Sabetzadeh, L. C. Briand, F. Zimmer and R. Gnaga, "RUBRIC: A flexible tool for automated checking of conformance to requirement boilerplates." In *Proceedings of the 2013 9th Joint Meeting on Foundations of Software Engineering,* pp. 599-602. ACM, 2013.

[6] K. Pohl and C. Rupp, "Requirements Engineering Fundamentals", 1st ed. Rocky Nook, 2011.

[7] IEEE Recommended Practice for Software Requirements Specifications. IEEE Standard 830-1998. IEEE Computer Society, 1998.

[8] D. Zowghi, & V. Gervasi, "The Three Cs of Requirements: Consistency, Completeness, and Correctness," Proceedings of 8th International Workshop on Requirements Engineering: Foundation for Software Quality, pp. 155–164, 2002.

[9] F. D. Davis, R. P. Bogozzi, & P. R. Warshaw, "User acceptance of computer technology: A comparison of two theoretical models. Management Science", 35, 982-1003, 1989.

[10] N. Bevan, & M. Macleod, "Usability measurement in context. *Behaviour and Information Technology"*, *13*(1–2), 132–145, 1994.

[11] A. Davis, "Software Requirements: Objects", Function and States (Second Edition), Englewood Cliffs, New Jersey: Prentice Hall, 1993.

[12] H. Rombach, "Software Specifications: A Framework", Pennsylvania, Pittsburgh:Software Engineering Instituts, January 1990.

[13] B. Boehm, J. R. Brown, J. R Kaspar, M. Lipow, G.J MacLoed & M. J. Merritt, "Characteristics of Software Quality". TRW Series of Software Technology, Amsterdam, 1978.

[14] D. Leffingwell & D. Widrig, "Managing Software Requirements. A Unified Approach", Addison-Wesley, 2000.

[15] S. Ahmad, U. Anuar, N.A. Emran, "A Tool-based Boilerplate Technique to Improve SRS Quality: An Evaluation". Journal of Telecommunication, Electronic and Computer Engineering, vol. 10, no.2-7, pp. 111-114, 2018.