

# Software Requirement Analysis Template with Automation Aided System

Marzanah A. Jabar

Faculty of Computer Science & Information Technology,  
University Putra Malaysia,  
43400 Serdang, Selangor, Malaysia.  
marzanah@fsktm.upm.edu.my

Fatimah Sidi

Faculty of Computer Science & Information Technology,  
University Putra Malaysia,  
43400 Serdang, Selangor, Malaysia.  
fatimah@putra.upm.edu.my

Asyraf Azizan

Faculty of Computer Science & Information Technology,  
University Putra Malaysia,  
43400 Serdang, Selangor, Malaysia.  
asyrafazizan88@yahoo.com

Abdul Azim Abd Ghani

Faculty of Computer Science & Information Technology,  
University Putra Malaysia,  
43400 Serdang, Selangor, Malaysia.  
azim@fsktm.upm.edu.my

**Abstract**— Software engineering practices are becoming increasingly important in order to mitigate any risk of failure that could result in costly downtime, incorrect behavior, or safety failures. Requirement elicitation is the process to systematically extract and identify the requirement of the system from a combination of human stakeholder, the system's environment, feasibility studies, market analyses, business plans, analyses of competing products and domain knowledge. Communication gaps exist between customers, engineers and project managers, and requirements information loss might occur across different software development periods. This paper researches on the variety of approaches used during user requirement analysis in the software development process. A review of user requirement analysis technique from the literature is studied by listing their advantages and limitation. A tabulated summary based on the discussion are produced to form a picture of user requirement technique. Directions for future work are then proposed, based on the review.

**Keywords:** *requirement analysis, requirement elicitation, usability software, requirement determination.*

## I. INTRODUCTION

Since 1960, Requirement Engineering (RE) has become one of the top research topics in software engineering fields [1]. It is very complex and the related technologies are in a great variety [2]. Software engineering typically refers to a regimented and procedural methodology for developing software [3]. The requirement state is the major process in software development life cycle (SDLC) which has a major effect in the quality of the software product. Errors made in this process are extremely expensive to correct when they are detected during implementing or testing period [4]. Affected software quality would give a significant impact on society and economy such as increasing costs and business interruption [5].

This research will explore on the various requirement analysis technique undertaken during the analysis in SDLC. It is widely acknowledge that there is a gap between stakeholders and developer understanding during the requirement gathering process [6]. This problem will cause many effects on the quality of software for the system development. To overcome this problem, a requirement analysis template with an automation aided system is proposed. The main function is to close the understanding gap between stakeholder and the developer in order to construct a better architecture to achieve the usability of software. Usability is the capability of the software to be understood, learned, used and attractive to the user when used under specified conditions [4]. Based on Cusumano survey [7], they suggested that early planning and customer specifications are crucial to productivity, whereas “doing it right the first time” is essential for reducing development time. Therefore it is important to make sure that the requirement process is properly managed.

## II. PROBLEM STATEMENT

Software produced in Malaysia still faced major problems during their development. The problems such as questionable of quality, late delivery, over budget and the need to improve the software or modification is still the main problems for organization in Malaysia [5]. Since 1960, a lot of research in requirement engineering has emerged to overcome these problems [1]. In requirement elicitation and elaboration, using natural language is still the choice for software developer to obtain the software requirement process from the users [8]. But most of the time the users do not know the technicality of computer or what the software developer really need; it is difficult to tell on how the system will function [9]. Failure to gather the

correct requirement, may lead to the lack of usability of the software.

### III. OBJECTIVE

The purpose of the study is to propose a new software requirement analysis template with automation aided system in order to achieve a usability software quality. Quality of software attributes refer to the set of extra-functional quality attributes that are discernable at run time and can be tied to a particular use case [10]. It is believed that the quality of intermediate deliverables like the requirement document is correlated with the quality of the final product [11]. The following are the questions produced to be answered with regard to the problem in eliciting requirement.

The first one is what are the requirement analysis process and its relationship to the usability software product? This question will explain more on the definition of requirement analysis and the process involve. As an early stage in the software development lifecycle, defining the requirement term is important to understand its need and objectives.

Second question is what are the problems with the existing technique? From the literature review, existing technique will be studied, their limitation identified and problems to be understood. This is also an important stage to make sure that there is no others similar research on this topic.

Lastly how to improve the requirement analysis process? This is the main question that will lead to the development of this research. With the new template, gaps from the previous research will be covered.

### IV. RESEARCH METHODOLOGY

For this research, a literature review on several user requirement analysis processes are analyzed based on their requirement model, tools and framework. They will be analyzed and discussed in the viewpoint of their strengths and limitations. The motivation base on the gaps from critical analysis of these finding will direct the future work of the research.

### V. LITERATURE REVIEW

For this research, four papers have been studied and each of it proposed a model for software requirement process.

Based on Chao study [9] used an event-driven in software requirement analysis. Event is the properties on what the user want to do such as send order or make booking. From the event, it will get the detail on what will trigger the event, the source and action taken from the event table. Event table is the process in event-driven technique to get the requirement.

It can be use in structured analysis method and object-oriented method. Structured analysis is a software engineering technique to construct and refine a system

specification from requirements, which has been used for a long time [10]. Object-oriented is a programming paradigm that uses "object" consisting of data fields and method together with their interaction.

For structured analysis method, DFD fragments and 0-level figure are created based on what the event table gets. The DFD diagram then will provide the analytical model on the system. Furthermore the subsystem can be divided through events and external entities interaction, data storage interaction and common point analysis in processing requirement. As for object-oriented method, the event table will develop the Use Case Diagram and Sequence Diagram. Use Case Diagram and Sequence Diagram is the Behavior Diagram in Unified Modeling Language (UML). Behavior diagrams emphasize what must happen in the system being modeled. Since behavior diagrams illustrate the behavior of a system, they are used extensively to describe the functionality of software systems.

This paper uses the case analysis to show how to use the method to carry on structured analysis and complete software requirement modeling. The result shows that the method is easy to use.

Chih-Wei [8] introduced the Model-driven Object-oriented Requirement Editor (MOR Editor) which supports requirement document modeling and model-driven document editing. It support requirement engineering objective requirement artifact, link related requirement artifact, provide reusable requirement template, and create a requirement document model to be integrated with artifacts in analysis, design, and implementation phase.

A case study was done for this model to shows its effectiveness. The model is still in prototype version and it is still in developing process. Part of the objectives of MOR Editor is to enhance the traceability, consistency, completeness, and maintainability of requirement documents.

Researcher showed that [8] an integrated framework for semantic requirement engineering. Semantic is the study of meaning. It typically focuses on the relation between *signifiers*, such as words, phrases, signs and symbols, and what they stand for. It combines domain ontology, enterprise ontology and user ontology to enable semantic representation and reasoning of software requirement. Ontology is a formal representation of the knowledge by a set of concepts within a domain and the relationships between those concepts. It is used to reason about the properties of that domain, and may be used to describe the domain. This is one way to closing the understanding gap between stakeholder and system developer. But the framework never produces any tools yet in order to gather the user requirement.

Another study presents a new approach for linking Requirement Engineering activities into a Process Framework that can be used as a reference for driving concrete Requirement Engineering processes [12]. In order to build a common understanding of the problem context, special emphasis is placed on construction Problem Domain models. It is use to situate user requirement with reference to it, and one of the technique for user requirements

refinement. Process Framework also separates the elicitation, analysis and refinement of user requirement from the construction of a system or software specification.

There has been a tool that uses the Process Framework and it has been tested in several projects from different organization. The tool seems working but still can be enhance it. There are several repeated problem occur based on the paper. The effort required to build the Problem Domain models ant to link the user requirement to them is the problem. Problem Domain is the area of expertise or application that needs to be examined to solve a problem. As from that, many further studies can be made.

## VI. RESULT

Table 1 is the summary of the literature review. It is tabulated into model and for each model the benefits and limitation are highlight.

Model/framework	Benefits	Limitation
Software requirement analysis technology method based on event-driven [9]	<ol style="list-style-type: none"> <li>1. Suitable for structured analysis method and object oriented method.</li> <li>2. Easy and feasible method.</li> </ol>	<ol style="list-style-type: none"> <li>1. Suitable for functional requirement only.</li> <li>2. Users need to have basic skill in software engineering to use this model.</li> </ol>
Model-driven Object-oriented Requirement Editor (MOR Editor) [8]	<ol style="list-style-type: none"> <li>1. Support requirement document modeling.</li> <li>2. Model-driven document editing.</li> <li>3. Support requirement engineering objective requirement artifact.</li> <li>4. Link related requirement artifact.</li> <li>5. Provide reusable requirement template.</li> <li>6. Create a requirement document model which can integrate with artifact in analysis, design and implementation phases</li> </ol>	<ol style="list-style-type: none"> <li>1. The MOR Editor is limit for Model-Driven Object Oriented only.</li> <li>2. Transfers knowledge into object oriented model only.</li> <li>3. No elicitation on the requirement part.</li> </ol>
Integrated framework for semantic requirement engineering [6]	<ol style="list-style-type: none"> <li>1. Combines domain ontology, enterprise ontology and user ontology to enable semantic representation and reasoning of software requirement.</li> </ol>	<ol style="list-style-type: none"> <li>1. No requirement elicitation process involve in the framework.</li> <li>2. It focuses on the integration part between ontology and database.</li> </ol>

Process framework for requirement analysis and specification [12]	<ol style="list-style-type: none"> <li>1. Can be used as a reference for driving concrete Requirement Engineering processes.</li> <li>2. Separation of the elicitation, analysis and refinement of user requirement from the construction of a system or software specification.</li> <li>3. Supported by a tool and has been tested in several projects from different organization.</li> </ol>	<ol style="list-style-type: none"> <li>1. It has a repeated problem that is the effort required to build the Problem Domain models and to link the user requirements to them.</li> <li>2. Automation system still in research stage.</li> <li>3. Not explicitly differentiate between functional and non-functional requirements.</li> </ol>
---	--	--

## VII. DISCUSSION

Based on what has been discussed above, many attempts have been made by researchers in trying to make the requirement stage easier. But they are still gaps between stakeholder and developer understanding in order to get the actual requirement so that the software that will be developed is acceptable and usability.

We are proposing a tool for software requirement analysis template with automation aided system is the new approach to gather user requirement. Its use the method of "5 W 1 H" that is what, who, when, where, why and how. An interrogation technique used to govern answer extraction range and method, and further affects entire system [13]. Based on the interrogative theory of information and knowledge, it suggests that text is a heterogeneous mixture of data, information, and knowledge, which can be separated and quantified through the interrogatives [14]. Using this technique will create an effective knowledge transfer through visualization [15]. Listing possible answer to these key questions leads to a better understanding of the system.

The reason why we are proposing this tool is because the usage of natural language during the requirement stage is the major part that causes the understanding gaps between stakeholder and developer become apart. We agree that using natural language is still the best way to communicate and gain knowledge, but with this tool, it is aided with the interrogative elements that will make the elicitation process become easier and smooth. The automation process that comes with the tool will facilitate the interrogative process that will give a lot more information during the analysis process. With help from this tool it will assist in processing natural language and enhance the quality of the knowledge while make the gaps become closer.

The interrogative technique is using the method of “5 W 1 H”. The first “W” is what. This is the basic question when getting the user requirement. The question must be directly on what are the user or stakeholder want for their system. As for example “what are the main functions in the Student Registration System?” From the “what” question, the main requirement can be gathered that will involve in the system. It also can detail out information needed. For example if the stakeholder state that “the student should enter their personal information”. Then the “what” question can ask back, “what is the personal information?” From the result, a clear requirement specification can be detail out and reduce the ambiguity.

For the second “W” is who. The source of information must be known. As for example the “enter the personal information”. Then we need to know who will trigger the requirement. The system should response back, “who will enter the personal information?” From that we can know who is the person or actor that involve with requirement.

The third “W” is when. This is the matter of time. When the requirement state that “the student should enter their personal information”, then the system should response back, “when the student can enter their personal information?” or “when the requirement can be trigger out?” With the information collected, the requirement can be more specific and it should help the developer to understand more.

The fourth “W” is where. This will discover the place involve in the requirement. This is important for the developer to know when it involves the data transferring. When the student enters their personal information, the system will ask “where should the information be send?” or “where it should be stored? If the information has been stored, the system can ask “where it should get it?” This is very important when we need to link or pass the data to use in other requirement.

The last “W” is why. This question is to give more understanding for the developer. Using “why” question we can discover out their policies or rules that should be follow. As for example, “why is the personal information is needed?” This will explain more on the flow of the system later. It also will help the stakeholder to elaborate more to make sure that the developer completely understands on what they needs. If the system flow seems to be complicated, as for example, “the student should enter their identity card number and birth date” then the system can ask back “why the student should enter their identity card number and birth date?” as we can identify the birth date from their identity card number. This will help the developer and the stakeholder to discuss more on their system which part can be simplify or enhance. This is where the understanding gap between them is getting closer.

Lastly for the “H” is how. This question is to detail out the requirement. As for example “how long the system should response back?”, “how the information should be stored?” or “how to confirm the information is true?” Answering this question surely will cover more part of the requirement. This will help to understand more.

This is how the new system will close the understanding gap. The requirement gathered then will be stored in repository for further use. With the automation system it will help the user by suggesting the similar requirement that has been used before from the system repository. The system will get the key point to give the suggestion such as “information”, “student”, “enter” and others. The system will automatically suggest some part of the requirement has been used before.

For this research it will only focus on the requirement because it is the most crucial process in requirement engineering since it moulds the shape of the desired end product [11]. The result will satisfy the user requirement and the developer understanding. This research cannot ensure that the stakeholder will not add more requirements when developing process has been started. But it is made to make sure that the requirement is perfectly understand and understandable by both of stakeholder and developer. The end target is the user get what software they want and they use it. For future work, the system can be change the requirement to structured analysis using UML.

## VIII. CONCLUSION

From what has been discussed on the research, a review of each model has been studied and from that a table has been produced to differentiate its advantages and limitations. From the table, a discussion has been made and to overcome the problems, a new tool has been proposed that has an automation function. It is using the inquisitive technique or also known probing technique “5 W 1 H” method that leads to deeply understanding on the requirement. The proposed tool will be followed by the development of the repository to enhance the functionality of the tool.

## REFERENCES

- [1] Badariah Solemon, S. S. a. A. A. G. (2009). "Requirements Engineering Problems and Practices in Software Companies: An Industrial Survey " Advances in Software Engineering, Communications in Computer and Information Science 59: 70-77.
- [2] Jiang, X. (2008). "Modeling and Application of Requirements Engineering Process Metamodel." IEEE International Symposium on Knowledge Acquisition and Modeling Workshop, 2008. KAM Workshop 2008.: 998 - 1001
- [3] Kerry, E. D., S. (2009). "Applying software engineering practices to produce reliable, high-quality and accurate automated test systems." AUTOTESTCON, 2009 IEEE
- [4] Kayed, A. H., N. Samhan, A.A. Alfayoumi, M. (2009). Towards an Ontology for Software Product Quality Attributes. Fourth International Conference on Internet and Web Applications and Services, 2009. ICIW '09.: 200 - 204
- [5] Fauziah Baharom, A. D., Abdul Razak Hamdan (2005). "A Survey on the Current Practices of Software Development Process in Malaysia" Journal of ICT 4: 57-76.

- [6] Li, Y. Y. F. X. W. Z. X. X. Y. L. X. (2008). "Towards Semantic Requirement Engineering." IEEE International Workshop on Semantic Computing and Systems, 2008. WSCS '08.: 67 - 71
- [7] Cusumano, M. M., A.; Kemerer, C.F.; Crandall, B. (2003). "Software development worldwide: the state of the practice." Software, IEEE **20**(6): 28 - 34
- [8] Chih-Wei Lu, C.-H. C., Chu, W.C., Ya-Wen Cheng, Hsin-Chien Chang (2008). "A Requirement Tool to Support Model-based Requirement Engineering." Computer Software and Applications, 2008. COMPSAC '08. 32nd Annual IEEE International 712 - 717
- [9] Chao, Z. Z. L. Y. C. (2009). "Software Requirement Analysis Research Based on Event-Driven." International Forum on Computer Science-Technology and Applications, 2009. IFCSTA '09. **1**: 247 - 250
- [10] O. Preiss, A. Wegmann, and J. Wong, "On quality attribute based software engineering," in *Euromicro Conference, 2001. Proceedings. 27th*, 2001, pp. 114–120.
- [11] Ahmad, S. (2008). "Negotiation in the Requirements Elicitation and Analysis Process." 19th Australian Conference on Software Engineering, 2008. ASWEC 2008.: 683 – 689.
- [12] Garcia Alcazar, E. M., A. (2000). "A process framework for requirements analysis and specification." Proceedings. 4th International Conference on Requirements Engineering, 2000.: 27 - 35
- [13] Liang, Z. L., Zhou; Jia-Jun, Chen; (2007). "Structure Analysis and Computation-Based Chinese Question Classification." Sixth International Conference on Advanced Language Processing and Web Information Technology, 2007. ALPIT 2007.: 39 - 44
- [14] Edward J. Quigley, A. D. (1999). "Interrogative theory of information and knowledge." Proceedings of the 1999 ACM SIGCPR conference on Computer personnel research SIGCPR '99.
- [15] Martin J. Eppler, R. A. B. (2007). "Visual representations in knowledge management: framework and cases." Journal of Knowledge Management **11**(4): 112-122.