Proposed technical activity on requirement engineering process

A P Widodo, R Kusumaningrum, S N Endah, S Adhi
Department of Informatics, Faculty of Science and Mathematics, Diponegoro University, Semarang 50275, Indonesia

corresponding author: arispw@gmail.com

Abstract. Requirement Engineering (RE) aims to provide assurance that a software product can meet user needs. The RE phase is a stage that is carried out early in the software development phase of the System Development Life Cycles (SDLC). The main problem in software development is the accuracy of the requirements specifications generated in the RE phase. Problems due to non-conforming specifications can result in software become of poor quality, development time become longer, and development costs become high. To minimize the obstacles that occur, in this article will be given about the techniques carried out in the RE process including elicitation, analysis, documentation, validation, and management. The techniques are given focus more on the implementation of technical activities and artifacts that are used (if any) in each RE process. Artifacts in the RE process can be used as a tool to simplify the verification process and validation of conformity between the requirements specifications produced with the expectations desired by the user. The resulting artifacts are using UML notation which consists of flow modelling using activity diagrams and prototype modelling using use case diagrams. Based on the proposed technical activities, the RE process can improve the quality of the software, the development time is according to the planned schedule, and the development costs according to the budgeted.

1. Introduction
Requirement Engineering (RE) is the first phase of software development on System Development Life Cycles (SDLC). SDLC has a sequential nature that starts from the first phase, then continues to the next phase after completion [5]. Sequential SDLC has a consequence that the RE phase, which is the first phase must be able to produce software requirements specifications precisely according to user needs. Accuracy in preparing software requirements specifications in the RE phase has the aim of providing assurance that the resulting software product has good quality, the processing time is according to the planned schedule, and development costs do not exceed the planned costs. Therefore RE is the most important stage in determining the success of software development [4]. Failure of software projects can be caused by failure to determine requirements according to user requirements and due to requirement that often changes [8]. Based on data from The Standish Group Report (2014), failure due to inappropriate requirements occurred at 12.3% and due to changing requirements by 11.8%. Failure of software development caused by the requirements is quite high. In order to reduce the failure rate due to the mismatch of requirements, this research discusses the techniques and artifacts produced in the RE process activities. The RE process consists of a process of elicitation, analysis, documentation, validation, and management [1].
This research is more focused on techniques and documentation using an artifact in the documentation process as a result of modelling in the elicitation and analysis process. The proposed artifact documentation uses the Unified Modelling Language notation. The UML diagrams used are activity diagrams and use case diagrams. Activity diagrams are used to capture running business processes to be poured into the software. Use case diagrams are used to provide clarity about functionality based on business processes that have been modeled using activity diagrams. Based on the results of this study, adding an artifact as documentation for the elicitation and analysis process will make it easier to communicate with users to provide reviews and feedback on the resulting raw requirements before becoming final requirements. In addition, adding artifacts as modelling in the RE process can improve the quality of software products in order to better accommodate user needs. It is believed that the model will provide easier communication with users in order to provide feedback on reviews provided by users.

2. Literature review

2.1. Requirement engineering (RE)

RE is a condition that must be met by the system or part of the system to fulfill contracts, standards, specifications, or other formal documents [6]. The technique or stages used to compile requirements is called RE. RE is the most important phase in the software development process because the accuracy of the specifications on the requirements greatly determines the success of software development to be able to accommodate user needs [3]. The RE phase is part of SDLC, whose sequence is in the first part of the SDLC stage so that RE must be able to provide accuracy in defining user needs in order to minimize the risk of software development failure [10]. RE consists of 5 processes, namely: elicitation, analysis, documentation, validation, and management [1], as given in Figure 1. The RE processes in Figure 1 are run sequentially and iteratively. The number of iterations carried out is very dependent on delicacy in capturing user desires. Elicitation is an RE process that focuses on collecting data and information from users based on applicable constraints and standards. Data and information collected can be sourced from formal or non-formal documents in accordance with the objectives defined as raw requirements. The techniques or tools used to conduct elicitation are using observation, interviews, surveys, and Questionnaires [12]. The analysis is a RE process that focuses on analyzing data and information obtained from the elicitation process in both formal and non-formal documents. The analysis process organizes data and information as raw requirements based on the type and process to accommodate user needs appropriately. This analysis process must be carried out carefully so that the results of the analysis can meet the requirements for unambiguous, complete, consistent, verifiable, traceable, relevant, and feasible parameters [13]. Documentation or the specification is a RE process that focuses on preparing a formal document of requirements specifications that have been produced during the elicitation and analysis process stages. The formal specification document must include functional and non-functional requirements obtained from the analysis. Validation is a process that focuses on validating the conformity between the specifications of the requirements generated in the analysis process to be confirmed with the needs desired by the user [2]. The validation process needs to involve users to provide feedback and reviews, which are used to confirm the results of the raw requirement specifications before becoming the final version as outlined in a formal document. The validation process needs to be carried out iteratively until an agreement is obtained between the developer and the user. Process management is a process that focuses on process management and documentation, starting from the elicitation process to the validation process, including any changes that occur. The management process is a process that must be carried out iteratively and continuously, starting from the development process to the maintenance process [7].
2.2. Software development artifact

Software Development Artifacts is one of the products produced in the software development process [11]. Artifact has an important role as part of a formal document in the software development process. Artifact can be in the form of data models, diagrams, scripts, or natural language [9]. Natural language is an artifact in the form of the narrative description using everyday language. The data model is an abstract model for organizing data elements and the relationships between them. The diagram is a representation of symbolic information in two-dimensional form based on visualization techniques. The artifact is a high levels model that is used as a communication tool with users to meet the specifications for defined requirements [12]. Some of the artifacts used in this study consisted of two categories, namely artifacts for the flow section and prototype modelling. Artifact flow consists of natural language, flowchart diagrams, and activity diagrams. Artifact functional prototype consists of natural language, data flow diagrams, and use case diagrams. Artifact, the flow section, is used to provide a description of the business flow (process) of the problem that will be developed into the software. Artifact functional prototype is used to provide a functional description of the software in the black box.

3. Methods

This research was conducted by distributing questionnaires to 30 respondents who acted as users to provide an assessment of the artifact proposed as modelling in the requirements specification preparation process. Respondents based on the description of the questionnaire items to provide a value with a value range of 1 to 4. The assessment items in the questionnaire consisted of the following aspects: ease,
clarity, and simplicity of several alternative artifacts given as a result of the RE analysis process. Artifact presented as modelling is divided into two, namely plot modelling and functional prototype. Artifact modelling of the flow section provided in the form of narration, flowchart, and activity diagram. Artifact modelling of the functional prototype provided in the form of narration and use case diagrams. Respondents, before giving an assessment, are given an explanation of the problem, then the respondent gives an assessment based on the experience obtained by the respondent while receiving explanations and reading documents. The results of the questionnaire were processed by using a weighted average, then given a descriptive analysis to provide an overview of the results obtained. The weighted average for each questionnaire item is calculated using the formula in equation (1).

$$\text{Percentage} = \frac{\text{Amount filled}}{\text{Total number}} \times 100\%$$  \hspace{1cm} (1)

4. Analysis and result

Data collection was carried out by giving questionnaires to respondents to provide a value ranging from 1 to 4 based on the respondent's mastery of knowledge. Before giving a value, the respondent is first given a sample explanation of the problem which is equipped with several alternative solutions in the form of different artifacts for each part of the plot and prototype modelling. Each respondent gives an assessment for all types of artifacts that have been provided. Each value filled in by each respondent is calculated using the weighted average for each artifact both in the flow (natural language, flowchart, and activity diagram) and functional prototypes (natural language, data flow diagrams, and use case diagrams) for each aspect. which includes convenience, clarity, and simplicity. The questionnaire data obtained from 30 respondents were processed using a weighted average value, as given in Table 1. The sum value is obtained by adding up all the values given by each respondent on each aspect and the corresponding part, namely R1 + R2 + R3 +… + R30. Then the Average value is obtained by dividing the sum value with 30 respondents. The weighted average value for flow in the convenience aspect is 2.30, clarity is 3.30, and simplicity is 1.30. Then the weighted average value for the functional prototype for convenience aspect is 2.37, clarity is 3.57, and simplicity is 1.53. Questionnaire data for all artifacts regarding convenience, clarity, and simplicity aspects after processing using a weighted average value are presented in the form of a data summary in Table 2.

Table 1. Natural language artifact assessment

| Respondents | Flow  | Functional Prototype |
|-------------|-------|----------------------|
|             | Convenience | Clarity | Simplicity | Convenience | Clarity | Simplicity |
| R1          | 2      | 4        | 2          | 2           | 4       | 2          |
| R2          | 2      | 4        | 1          | 2           | 4       | 1          |
| R3          | 2      | 4        | 2          | 2           | 4       | 2          |
| R3          | 2      | 4        | 1          | 3           | 4       | 1          |
| R5          | 2      | 4        | 1          | 2           | 4       | 1          |
| R6          | 2      | 4        | 1          | 2           | 4       | 1          |
| R7          | 2      | 4        | 1          | 2           | 4       | 2          |
| R8          | 2      | 3        | 1          | 2           | 3       | 2          |
| R9          | 2      | 3        | 1          | 2           | 4       | 1          |
| R10         | 2      | 3        | 1          | 2           | 3       | 1          |
| R11         | 2      | 3        | 1          | 2           | 3       | 2          |
| R12         | 2      | 3        | 1          | 2           | 3       | 1          |
| R13         | 3      | 3        | 1          | 3           | 3       | 2          |
| Respondents | Convenience | Clarity | Simplicity | Convenience | Clarity | Simplicity |
|-------------|-------------|---------|------------|-------------|---------|------------|
| R14         | 3           | 4       | 1          | 3           | 3       | 1          |
| R15         | 3           | 3       | 1          | 3           | 4       | 2          |
| R16         | 3           | 3       | 1          | 3           | 3       | 1          |
| R17         | 2           | 2       | 1          | 2           | 2       | 1          |
| R18         | 2           | 4       | 2          | 2           | 4       | 2          |
| R19         | 2           | 3       | 2          | 4           | 4       | 2          |
| R20         | 2           | 3       | 2          | 2           | 3       | 2          |
| R21         | 2           | 3       | 1          | 2           | 4       | 1          |
| R22         | 3           | 3       | 1          | 3           | 4       | 1          |
| R23         | 3           | 4       | 1          | 3           | 4       | 2          |
| R24         | 3           | 3       | 1          | 1           | 3       | 1          |
| R25         | 3           | 3       | 1          | 3           | 4       | 2          |
| R26         | 2           | 3       | 2          | 2           | 3       | 2          |
| R27         | 2           | 3       | 2          | 2           | 3       | 2          |
| R28         | 2           | 3       | 2          | 2           | 4       | 2          |
| R29         | 2           | 3       | 1          | 3           | 4       | 1          |
| R30         | 3           | 3       | 2          | 3           | 4       | 2          |
| **Sum**     | **69**      | **99**  | **39**     | **71**      | **107** | **46**     |
| **Average** | **2.30**    | **3.30**| **1.30**   | **2.37**    | **3.57**| **1.53**   |

Table 2. Summary questionnaire result

| Aspect       | Natural Language | Flowchart | Activity Diagram | Natural Language | Data Flow Diagram | Use Case Diagram |
|--------------|------------------|-----------|-----------------|------------------|-------------------|------------------|
| Convenience  | 2.30             | 2.57      | 3.70            | 2.37             | 2.67              | 3.63             |
| Clarity      | 3.30             | 3.33      | 3.50            | 3.57             | 3.63              | 3.73             |
| Simplicity   | 1.30             | 1.77      | 3.43            | 1.53             | 2.30              | 3.67             |

Table 2 shows that the natural language artifact from the three aspects has the lowest value compared to other artifacts, both in the flow section or the functional prototype. The difference in value for each artifact on the convenience and simplicity aspects is quite significant, but in terms of clarity, it is not significant. The aspect of clarity is insignificant because for clarity, capture the problem is the main requirement that must be met in order to produce a good specification of requirements. Then in the aspect of convenience and simplicity, the difference in value is very significant because it emphasizes the perception that it is easier to understand a problem if it is given in the form of a notation (diagram), rather than having to be presented in the form of natural language narratives. Diagrams based on non-UML notation (flowcharts and data flow diagrams) compared to diagrams based on UML notation have significant differences in values in the three aspects given above, both in the flow section and the functional prototype section. The difference value in Table 2 shows that the artifact based on UML has a higher value than the artifact based on non-UML notation and natural language. The artifact activity diagram in the flow section has the highest value because it can be presented by dividing the actors involved in the system, which can be presented in the form of a timeline, whereas if the flowchart can only be presented one whole, it cannot be presented based on the actor. The difference in presenting flowcharts and activity diagrams to represent flow has a very big influence on the convenience and
simplicity aspects given by the respondents. This condition also applies to the difference between the data flow diagram and the use case diagram in the functional prototype section. Based on the analysis given, it can be concluded that in order to provide assurance that the requirements specification can accommodate user needs, in order to facilitate communication and feedback, it must be equipped with artifacts based on UML notation. The artifact activity diagram is used to represent the flow of the problem, while the artifact use case diagram is used to represent the functional prototype. Artifacts based on UML notation produced in the RE process carried out in this study can be used to minimize multiple interpretations between users and developers so that the specification requirements produced in the RE process are more in accordance with user needs which can be used to minimize the risk of development time that is not in accordance with a plan, and minimize costs in excess of the planned costs.

5. Conclusions
Artifact, the part of the flow that is most appropriate to use in compiling requirements specifications in the RE process is to use an activity diagram. Activity diagram from convenience aspect has a value of 3.70, clarity aspect has a value of 3.50, and simplicity aspect has a value of 3.43. Based on these three aspects, activity diagrams have the highest value compared to natural language and flowcharts. Artifact, the prototype modelling part that is most appropriate to use in compiling requirements specifications in the RE process is to use a use case diagram. The use case diagram from the convenience aspect has a value of 3.63, the clarity aspect has a value of 3.73, and the simplicity aspect has a value of 3.67. Based on these three aspects, activity diagrams have the highest value compared to natural language and data flow diagrams. Therefore, artifacts that use UML notation are the most appropriate artifacts compared to non-UML artifacts to produce formal documentation of requirements specification in the RE process. By using artifacts in UML notation, it can produce the correct specification of the requirements according to what the user wants so that the risk of failure in the software development process can be minimized. The risk of failure of the software development process can be minimized by using artifacts based on UML notation as an easy and simple communication tool for users to provide feedback on requirements specifications.

Acknowledgments
This work was supported by the Community Service Funding Science and Mathematics Faculty of Diponegoro University 2020.

References
[1] Bernard Y 2012 Journal of Requirements Systems Engineering 15 119
[2] Chemuturi M 2012 Handbook of Requirements Engineering and Management for Software Development Projects (Springer Science & Business Media) p. 135
[3] Lemos R et al 2013 Handbook of Software engineering for self-adaptive systems: A second research roadmap. In Software Engineering for Self-Adaptive Systems II (Springer Berlin Heidelberg) p. 246
[4] Lucia A and Qusef A 2010 Journal of Emerging Technologies in Web Intelligence 2 212
[5] Frank U 2013 Proceeding on 3th International Conference on Information Technology - New Generations (Berlin, Heidelberg) p. 133-157
[6] IEEE Standard 830-1998 1998 Handbook of Recommended Practice for Software Requirements Specifications (New York IEEE Press) p. 372
[7] Geisser M and Hildenbrand T 2006 Proceeding of IFIP World Computer Congress TC (Springer, Boston, MA) p. 108-122
[8] Khan M N A and Khalid M 2013 Journal of Modelling Education Computer Sciences 5 21
[9] Kuhrmann M and Fernandez MD and Groeber M 2013 Proceeding of International Conference on Global Software Engineering (Boston). 108-122
[10] Kumar M Shukla M and Agarwal S 2013 Proceeding of International Conference on Machine Intelligence and Research Advancement (Miami) 515–519
[11] Lieberman H, Nardi B A and Wright D 1998 Proceeding of ACM conference on Human Factors in Computing Systems (Summary Demonstrations) (Los Angeles California) 112-120
[12] Mehmood M and Ijaz B B 2018 Journal of Architecture Engineering Technology 7 45
[13] Widodo A P, Adi K, Nugraheni S A, Indri W 2020 Journal of Physics: Conference Series 1524 012009 (Semarang Indonesia) p. 267-278