User Requirement Validation: Challenge Exploration in Pre-project Execution

Muhamad Idaham Umar Ong¹*, Mohamed Ariff Ameedeen¹
¹Faculty of Computing, College of Computing and Applied Sciences, University Malaysia Pahang, Gambang, 26300, Malaysia

* Correspondence: idaham@ump.edu.my

Abstract. The need for validation technique for user requirement or in other term Tender Specification at the Request for Tender (RFT) is vital. Without much resource for vendor to dispose of, the outcome of this research will prove useful. The purpose of this article is to determine the gap of initial and finish software product, method of eliminate or mitigate those gaps, list of OO artefacts and their relevancy towards the research. Finding of this article will be the basis of interconnecting user requirement specification and meta-requirement via UML. Based on the result, RE plays an important role in the survivability of a project, there are a few researches being done for solutions in the field of RFT specification validation, UML is the most relevant artefacts and usability of UML diagram ranked based on consistency is relevant toward connecting UML and meta-requirement. It is suggested that further research is to be carried out that will define the connection between user requirement specification in the form of UML to meta-requirement in order to achieve the overall objective of the research.

1. Introduction
There are many software development projects have failed to see the light of day. For project that were delivered to their respective owner, there is usually a problem where the developed product itself doesn’t meet the initial expectation of the owners themselves. Delivered product has gone through more of a devolution rather than evolution from which the first day the initial prototype was presented. This gives software developers a bad reputation in keeping their promises to develop the desired product.

Commonly, software developers on this day continues to produce monstrosity in the form of products that originated from (1) lack of understanding of the nature of the software itself, these include understanding from the aspect of software developer and stakeholders. (2) lack of knowledge management in young and in-experience software developers are struggling to maintain the standards that they predecessors has formally or informally established and (3) Problem in software quality assurance where not many projects willing to invest in latest testing technology and tools but accommodate cost effective testing that uses a lot of regression and unmanaged testing activities. [1]

A static published by [2], software development project failures occurs throughout the industry. It is noted that from the overall percentage of project involve in the study, less than 28% of software development project are being delivered successfully without facing any challenges. Table 1 summarizes the overall statistical result based on the study conduct by [2].
Table 1. Percentage of success/fail projects [2]

| Project Outcome                                      | Size of Stakeholder’s Organization |
|-----------------------------------------------------|-----------------------------------|
|                                                     | Large    | Medium   | Small    |
| Completed on-time and on-budget with all features    | 9%       | 16.2%    | 28%      |
| initially specified                                  |          |          |          |
| Completed with burse time and budget, features       | 61.5%    | 46.7%    | 50.4%    |
| were left-out                                        |          |          |          |
| Project cancelled                                    | 29.5%    | 37.1%    | 21.6%    |

It is also mentioned that some of the main reasons why project budget and timeline are over run its related to the features that are initially specified during the earlier stages of the software development project. Developer having problem to ensure that the features is comply with the need of the system users. Things like; (1) Lack of user input, (2) incomplete requirements and specifications and (3) changing requirements and specifications [2] are the top 3 problem that are being faced by developer. And all those three problems are pointing to one specific area of knowledge in software engineering, and that is requirements engineering.

Software development project failures are not solely tied to the hardworking people of the Information Technology (IT) sector. Failures are wide spread and involves human. When looking into the area of Critical Success Factors (CSF) in software development project, some of the prominent factors are; (1) top management’s commitment and involvement/support, (2) allocation of scarce resources, (3) communications among various stakeholders, (4) team configuration and structure and (5) social cohesion in the team and the complexity of the project and organizational culture. [3]

With earlier intervention by investing more resources and the correct tools and methodology, the problem in software development project could be overcome and projects will be delivered to their respective owner without much of an issue.

The purpose of this publication is to investigate and answer the following research questions present finding on; (1) Why software development fails in the scope of requirement engineering, (2) How developed methods are used to eliminate or mitigate those failures, and (3) Which artefacts are considered as the life-line of a software development project.

The remaining five sections are as follows. Section 2 presents the background of the study. Section 3 describe the adopted research methodology and threat to validity of study. Section 4 describe briefly the list of reviewed literatures in answering the research questions. Section 5 reports the results and discussion from the study. Finally, Section 6 concludes and outlines some possible future works.

2. Background of Study
This section will describe the list of background of the conducted study.

2.1. The starting point buyer and vendor
This section will describe the process of purchaser from an organization will looking to purchase a software solution from a software vendor. The process starts by a request is being put out internally that a need for a solution to a problem or opportunity is to be cater with the help of a software solution. The request will be advertised through either Express of Interest (EOI), Request for Information (RFI), Request for Proposal RFP), Request for Quotation (RFQ) or Request for Tender (RFT). Software vendors will start engaging with the organization. The RFT advertisement contain requirements, criteria and instructions that are to be abide by the vendors. The vendor will submit a reply either via tender or proposal specifying the service or solution that will be provided, buyers evaluate the vendors and project will be awarded to the winning vendor. Please be informed that the publication will not discuss in depth on the detail knowledge and processes of procurement, but more on how a purchaser’s tender is being initiated and how software vendor approaches the buyers focusing the challenges faced by the software vendors.
2.2. *From Request for Tender to Awarding Project*

Purchaser is in need of a solution for a specific set of problems. Within the scope of solution with the utilization of software, buyers will look into different vendors which could provide the suitable solution to their problem. Some buyers will use different approach in order to gain more insight regarding vendors and suppliers for such request.

Purchaser will always try to balance the price and quality ratio of the solution. It is common that the best solution in the market doesn’t come cheap and plus there is a saying “Nothing is perfect”. In order to overcome the buyer’s problem of finding the suitable solution, tenders are being advertise to encourages business to present their proposal on their take on how to solve the specified problem.

Project proposal is one of the medium widely used by software vendors as a way to communicate their intention and dealing with their potential buyer. The content of a project proposal is straightforward. It may contain information related to the decision maker but not limited to the following information; (1) Problem face by organization, (2) Vision of the project that will ensure that it is in line and closely related with the organizational strategies and vision, (3) Benefit of the project will deliver, (4) Deliverables in the scope of equipment, artefacts and tangible and intangible products, (5) Success criteria that will give confidence to stakeholders when project is executed. Usually referring to success criteria such as Specific, Measurable, Achievable, Realistic and Time bound (SMART) (6) Listing deadlines, overall development plan and approaches used such as the used of external vendors and the used of agile software development, (7) Cost and budget. From all this information, everything will be summarized in an executive summary. Some buyers will sometime be requested separate documentation of proposal are to be prepared, first proposal is related to the technical aspect of the project and second is financial matters.

Decision maker will make the decision to accept or reject the project proposal. First impression is critical at this moment, usually the sales people will try to make a big impact right at the start of this proposal presentation. Problem statement is key, it will explicitly show the current problem faced by the decision maker’s organization which could be overcome had your proposed project has been in production. Also highlighting where opportunity was missed and where risk and cost were incurred that could have been prevented with the use of the proposed project.

2.3. *The Challenge Faced by Software Vendors*

Usually in the event of presenting software development project proposal for the decision makers, a prototype is being used to the express the expected outcome of the project and in a way increasing the decision maker’s level of confidence towards the to-be awarded software vendor. Prototype are usually based on past successful project or it could an out of the wild combination of GUI and functions from a few different software projects. For a software vendor with strong background with similar past successful project, this decision of whether to take on the project or not would be a snap.

In a scenario where a detail list of thousands of specifications of the desired end-product is being listed out in the RFT, it is quite a daunting challenge where producing a highly convincing feedback that will meet and fulfill every single specification of the desired end-product. This is the task that will be led by Sales staff with the help and advice from a team [4] that consist but not limited to; (1) Sales staff, (2) Project Manager, (3) Technical specialists (Requirements Engineer), (4) Implementers, (5) Finance and commercial specialist, and (6) Legal experts. Some of the limitations [5], [6] that are imposed to these team are; (1) Time. This is a case by case scenario whereby the timeframe of between the advertisement of the RFT and project proposal preparation is quite short, (2) Resources. In current situation where the is no guarantee that the software vendor will be awarded with the project, not much resources will be invested in partaking this effort and (3) Competitive. Depending on the RFT itself, the level competitiveness will vary. It is worth mentioning that even large will invest their time in small projects.

Based on the limitation shown on the previous paragraph, the task of Requirements Engineer has somewhat seemed absurd. In the scenario stated, Sales staff and Project Manager will be relying on the input from the requirement engineer to ensure that the proposal will be up-to-par with the expectation
of the decision maker. Based on [7], requirements engineer main target is to achieve the desired output of a software development project by balancing the three main aspect of a software project [7]; (1) Specification, deals with the degree of requirements understanding at a given time (2) Representation, copes with the different representations (informal and formal languages, graphics, sounds etc.) used for expressing knowledge about the system. Within RE there are three categories of representations. and (3) Agreement, deals with the degree of agreement reached on a specification.

With the current attention of a Requirement Engineering is briefly illustrated in Fig. 1, it is acknowledged that with the combination of the situation of proposal presentation of a tender with the standard of work that needed to achieve by Requirement Engineers is quite challenging. Hundreds or even thousands of specifications that are included in the RFT document are also representing the almost finalize user requirement. If the thousands of specifications that are included are being taken lightly, then software development work will projects suffer failure due to user requirements related problem such as changing requirements, unclear, ambiguous and unusable requirements and misunderstood user requirements and the failure to freeze requirements [8]–[14]. Cost and delivery overruns are resource or economic factors. Cost overruns and missed delivery can result in project termination. Publication by [15] includes increase of costs and timeline, actual project expenditures and delivery below the estimates and insufficient budget. These indirect factors may be the reason for the overrun. Failure can also be due to time and delivery below the estimates related to estimation issues in project management. Finally the depletion of funds can result in project termination.

2.4. The Desired Form of User Requirement

During the stage of proposal presentation based on the scenario setup of previous chapter, Requirements Engineer will be facing a conundrum where they will need to analyze the Tender specification that comes in the form of thousands of lines of user requirement to ensure that the content of those specifications are up-to-par with what is needed for development team to work with.

There are three type of user requirement quality that must be evaluated and decided by the project team; (1) correctness of requirements, (2) completeness of requirements, and (3) consistency of requirements. With the high expectation and standards that is defined for Requirements Engineer to work with and with great attention that the result will enable meaningful advice to the sales executive and project manager is seem to bleak. But will the proper mechanism that is developed by future researchers, the hope for a future where a meta-level analysis of tender specification in the form of User Requirement can be realized and will fulfil the needs of software development team.

3. Research Methodology

The research methodology is partially adopting the processes that was define by [16]–[18]. In order to answer the defined research question, the study is based on literature review of publications from publication journals article, conference proceeding and books. It is also taken into account publication that are source from websites and press articles. The information gathered on the stated topic of interest is considered as a mature area of interest and was viewed from a selection of sources that are considered from highly cited and well-established publisher such as SCOPUS indexed and ISI journals.

An initial search using the combination of the following terms: Systematic Literature Review, Systematic Mapping Study, Software Development, Software Engineering, Requirement Engineering, Challenges, Project Management, Tender, Failures, Gaps in Google Scholar. No publication dates were defined in used of restriction which was considered as irrelevant for the task of literature review. Each of the listed publications were reviewed and synthesize to highlight; (1) Failure and success factors of software development projects, (2) proposes method/tool of elimination or mitigating failures and (3) the most critical artefacts of RE.

Following initial analysis of the studies, papers were selected based on their relevance and provided insight specific to the interest. The literature review was structured around the analysis and
synthesis of study findings to identify the relevant information and answering the defined research questions.

3.1. Threat to validity

Even though there is a possible method of producing an absolute literature search and result analysis through a rigorous and highly regressive search in every single source available in the world, there are always limitation that must be abide by your average researcher. In this section, threats to validity are being presented.

Consideration of abstraction of pure RE challenges in pre-project execution is not absolute possible with the close relation of the term computer science majorly in software engineering. Some of the findings presented may or may not be 100% aligned to the research questions but is made available in the study considering the relevancy of the information itself. This will be discussed further in section 5.

Another threat to the validity of this research is the humanity aspect of this research. There is no automated tool being used in this research and all information that is compiled for this research are being done through effort of normal human being. With the hope of being able to provide an absolute compilation of all existing meta-requirement creation method in the world, this research is also abiding to the constraints and limitation that is being put on the researchers themselves as normal human being.

4. Reviewed Literature

This section will describe the list of background of the conducted study. The following are the cited literature that are relevant in answering the defined research questions.

| Title                                                                 | Author                                                                 | Year | Publisher                                           | Related RQ |
|----------------------------------------------------------------------|------------------------------------------------------------------------|------|-----------------------------------------------------|------------|
| Perceived Causes of Software Project Failures - An Analysis of Their Relationships [19] | T. O. A. Lehtinen, M. V. Mäntylä, J. Vanhanen, J. Itkonen, and C. Lassenius | 2014 | Information and Software Technology                 | RQ1        |
| Reducing Software Requirement Perception Gaps Through Coordination Mechanisms [20] | H. G. Chen, J. J. Jiang, G. Klein, and J. V. Chen                      | 2009 | Information and Software Technology                 | RQ1        |
| Obsolete Software Requirements [21]                                   | K. Wnuk, T. Gorschek, and S. Zahda                                      | 2013 | Information and Software Technology                 | RQ1        |
| Critical Requirements Engineering Errors Leads to Fails Software Project [22] | M. Talhe                                                                  | 2018 | The Educational Review, USA                          | RQ1        |
| A Study on the Software Requirements Elicitation Issues - Its Causes and Effects [23] | N. Kumari.s and A. S. Pillai                                             | 2013 | 3rd World Congress on Information and Communication Technologies | RQ1        |
| A Quantitative Study to Identify Critical Requirement Engineering Challenges in the Context of Small and Medium Software Enterprise [24] | S. Besrour, L. B. A. Rahim, and P. D. D. Dominic                          | 2016 | 3rd International Conference on Computer and Information Sciences | RQ1        |
5. Result and Discussion
This section will showcase the result of this research. Each finding for respective research question will be presented in each different sub-heading.

5.1. RQ 1: Why software development fails in the scope of Requirement Engineering?
This section will discuss the findings based on the literature listed in section 4. From the view of [19], a software development project consists of people, tasks, methods and project environment where each of these must play its part in making sure everything working together in harmony. Fail to do so will cause the more prone to failure. The study also concluded that weakness identified in teamwork among project team, weak monitoring and management in task backlog, and finally small number of resources were invested in testing were the root causes of the failure.
Also based on the same study, finding on the failures cause and it’s causal are being illustrated. It gives an overview of the different kind of failure that could occur in a project and the inter-relationship between different aspect of a project. Even though the study focuses more on failures that are purely related to the area of software testing, failures that are related to the area of RE are also seems to be playing its major part as few of the reasons impacting the matters related to software project development project failure.

Communication and relationship between stakeholders and developers also gives critical impact in ensuring the project success. A study by [20] confirms that communication among project team members, including stakeholders will surely give major impact in improving the overall performance of project through key area examination of relationships among pre-project association, horizontal coordination, vertical coordination and user-developer perception gaps.

On another study done by [21], establish relationship between the different type of RE problem with obsolete requirements. In other words, the impact of errors occurs in RE process will impact on the system functionality in future where function that are based on problematic user requirement will become irrelevant and most likely will not be used by the users. Figure 1 depict the type of Obsolete Software Requirements (OSR) rating.

Based on a study by [22], the author has listed errors that could occurs during the RE process. The different stages of RE process that was used in the study involves; (1) Elicitation and collection, (2) Analysis and investigation, (3) Specification and description, (4) Verification and (5) Management. Table 3 summarizes the publication findings.

| Type of Error | Description | Common Errors |
|---------------|-------------|---------------|
| Conception    | Arise after the requirement is unwell defined, then the assuming thinking started | Unwell-defined requirement, lack of clarity, unneeded function, unneeded complexity, poor requirement quality, missing requirement, unprepared requirements engineer and inappropriate constraints |
| Specification | Requirement engineer can’t able to describe or specify requirements that has been analysed | Requirements not traced, inadequate verification of requirements quality |
| Implementation| These errors get up when the idea | Inadequate requirements validation, |
was correct, the description was exact, but the requirement was not implemented correctly

**Visualization**

These errors occur when proposed system area was represented by prototypes but not drawn against a result space area

**Prototype Error**

**Requirement Management**

Requirements stored in paper or spreadsheet rather than in requirement repository are difficult if not impossible to create, manipulate, and maintain

**Excessive requirements instability including unmanaged scope creep, inadequate requirements management**

The publication [23] by show the area of requirement engineering where to the concern of answer this publication, it details out the issue reported, the cause and the consequences against the different requirement elicitation issue factors. Table 4 summarizes the overall finding of the [23] findings. Each error that could happen in RE has vast consequences. With the role of RE consider as the main role in providing meaningful and highly sought inputs for a project, depending how good are those inputs will determine the successfulness of the project. As highlighted by the author the consequences of error occur in RE are the same errors occurs over and over, the cost of re-working project output, budget is being overrun, poor quality and progress monitoring systems, feeling of dissatisfaction among stakeholders towards the development progress, project development and process failure and tools that are not alignment with stakeholders environment, uncertainties, poor and ambiguous requirements, high costs of maintenance and operation, high changes frequency, conceptual inconsistency and flaws in resultant system.

Through the literature that was reviewed, some of the more critical challenges of RE are; (1) Vague and ambiguous requirements, (2) Incomplete Requirements, (3) inconsistent requirements, (4) Ownership and communicating requirement to owner and (5) Traceability of requirements [24]. It is also noted that publication in seven root causes related to project failure [25] suggested the following; (1) Incomplete Requirements, (2) User-related factors, (3) Incompetent project planning, (4) Weak support/involvement from management, (5) Lack of resources, (6) Weak business case and (7) Unclear development objectives. The most challenging activity in RE is change management.

**Table 3. Requirements Elicitation Issues and Consequences [23]**

| Elicitation issue factors | Issues | Consequences |
|--------------------------|--------|--------------|
| Changed                  | Management and political rules, acceptance criteria changes, unstable requirements, changes in nature of requirements overtime, user needs and understanding changes | Process overheads, re-work impacted project cost, quality, failure, requirements inconsistency, unusable |
| Communication            | Articulation related, unaware of needs, mis-understanding amongst stakeholders, verbal and presentation skill, requirements-related, culture and perspective related, language barriers, change related | System failure, budget overruns, project failure, coordination issues, misunderstanding, poor definition of needs, broken communication links, imperfect specification, scope creep, abstract communication, low motivation, waste |
| Human                    | Conflicts, ambiguities amongst stakeholders, intra-group conflicts, communication, participation, | Withholding information, recognition failures, sabotage efforts |
## As a conclusion on the finding on RQ 1, it is undeniable that RE is very crucial in ensuring the success of a software development project. Overall finding can be summarize based on Figure 2.
5.2. RQ 2: How are RQ 1 problem are being eliminated or mitigated?
This section will be broken down into primarily two section, managerial solution and technical solution will be further broken down to requirement elicitation phase solution and requirements validation and verification phase solution. Managerial solution will be presenting solution for issues related to RE based on fixes on the project management level and technical will be based on the process and methods of RE.

This section will discuss the findings on the available solution on the managerial-level. Findings is presented based of proposed solution on both the Project Management and Requirements Engineering related activities.

This section will discuss more on the technical-based solution, majoring in RE. Firstly, lets discuss on how the main content of a software development RFT specification (user requirements) are being produce. Based on a study by [26] in the scope of Small and Medium Enterprise (SME), findings were presented on the ranking of most used requirement elicitation techniques based on the 15 different selection that are currently being used widely by the industry. It shows that most requirements are being collected based on Table 5.

| Requirement Technique                  | Level  |
|----------------------------------------|--------|
| Interview                              | 4.39   |
| Joint Application Development (JAD)    | 4.35   |
| ERD-Based Specification                | 4.23   |
| Peer Review                            | 4.17   |
| Use-Case                               | 4.17   |
| Goal-Oriented                          | 4.15   |
| Check-List                             | 4.11   |
| Activity Diagram                       | 4.09   |
| Structured Natural Language (NL)       | 4.08   |
| Misuse-Case                            | 4.06   |
| Software Requirement Specification    | 4.05   |
| Brainstorming                          | 4.02   |
| Ethnography                            | 2.18   |

Figure 2. Summarization of RE Challenges and Impact in Pre-project vs End-product Expectation
| Laddering | 2.04 |
|-----------|------|
| Observation | 1.89 |

With the different method of requirement elicitation is being shown in the table above, Based on a study by [27], they have suggest multiple solution on the different issues that are faced by the four stages of RE; (1) Requirement Elicitation, (2) Requirement Analysis and Negotiation, (3) Requirement Specifications and (4) Requirement Validation.

Focusing on the research question, the next area of interest is the validation of requirements. It is crucial that in the effort of approaching a RFT, vendors must take the necessary action in preparing their winning presentation in order to convince the decision makers to pick them as the awarded vendor. Different method are being introduce by researchers and practitioners to give vendor the extra advantage when presenting their ideas to the decision maker.

For example, a method proposed by [28] suggested that LTesting can be utilize to generate a brief test plan that can be used in the vendor’s proposal presentation. Other than that, it could also support decision-making for vendors to either compete or skip a RFP for providing service of software testing by producing an initial test report based on the specification provided. Based on the Figure 3 highlight the relevant proactive step in the stage of RFP. The detail description of the highlighted steps are; (1) The initiation step if where the RFT is being engage by the vendor, (2) LTesting will verify the test requirements (3) vendor will produce a brief test plan to be presented to the customer, (4) Contract is awarded, (5) A detail test plan is being produced and (6) Resource are being allocated to the newly awarded project.
With the assistance of the correct method or tool, vendor could shine brightly by being different and provide out-of-the-box information that other competing vendors can’t. In the scope of this study, there are multiple techniques that are being proposed by researcher that could perform requirements validation but whether it can be utilize in the situation of RFT is still unknown. A study conduct by [29], their finding highlight which requirements validation technique that can be used in a low resource environment (refer to Table 10). It is suggested that prototyping is one of the most favorable tools that is suitable for the use in a RFT proposal presentation. But other vendors will also use prototyping and depending on the quality differences of those prototype will decide on the winning vendor.

Table 5. Comparison of Requirement Validation Techniques [29]

| Resource Category | Requirements Inspections | Requirements Prototyping | Requirements Testing | Viewpoint Oriented Requirements Validation |
|-------------------|--------------------------|--------------------------|----------------------|--------------------------------------------|
| Team Size         | Large teams              | Small teams              | Large teams          | Small teams                                |

Figure 3. Generic, Dynamic eSourcing Life-cycle Model for the Provisioning of Testing Services [28]
In conclusion, there research and solution being proposed by different researchers, but within the scope of solving this problem in RFT user specification validation is still green. To the authors best knowledge, the is no commercially available solution for the stated scope of problem.

5.3. RQ 3: Which OO artefacts considered as highly critical?
This section will present the finding related to RQ 3, which of the artefacts that are available in OO is considered as the most critical. A few studies has explicit shows the result in answer RQ 3. Based on a finding by [30], the most critical diagram or modelling tool in OO is Use Case and Class diagram in the scope of requirements validation. More detail in the Consistency aspect of requirements. Refer to Figure 4 for more detail on the publication findings.

Another research has also exhibit detail finding on the issue of requirement consistency validation in OO. A study done by [31] summarize the following diagrams (ranked) are being widely use in requirement consistency check; (1) Class Diagram, (2) Interaction Diagram, (3) State Machine Diagram, (4) Use Case Diagram, (5) Activity Diagram, (6) Composite Structure Diagram and (7) Object Diagram as shown in Figure 5.
Figure 5. Summary of rules between UML diagrams presented over the years, coupled with diagrams and Software Engineering activities. [31]

From the findings shown, it is noted that the stated diagrams are being considered as relevant toward the continuation of this research. They will be used to evaluate in term of their relevancy in bridging the gap between user requirements specification and meta-requirements.

6. Conclusion and Future Work
There is still limited work in bridging the gap between user requirement and meta-requirement at present. The process of making sure the relationship of each artifacts that are related to UML and meta-requirement in its core field of software engineering is also another matter that needs to be taken seriously.

Our future work will focus on formulating an algorithm that will function as closing the gap between user requirements specification and meta-requirement. The algorithm itself must be able to work in conjunction to ISDT and OO UML environment taking into account every related artifact in the field. In order to achieve this vision, a systematic formulation and the right approach is needed to produce an algorithm that can work within the stated environment.

References
[1] P. Kapłański, “Ontology-Aided Software Engineering,” pp. 1–161, 2012.
[2] The Standish Group, “Chaos Summary 2014,” 2014.
[3] S. McCarthy, P. O’Raghallagh, C. Fitzgerald, and F. Adam, “Social complexity and team cohesion in multiparty information systems development projects,” J. Decis. Syst., vol. 27, no. sup1, pp. 18–31, May 2018.
requirement engineering challenges in systems development projects. This may be due to several factors including the complexity of the system, the lack of clear requirements, or the involvement of stakeholders from different organizations. Projects that encounter these challenges may face significant difficulties in achieving their goals. One way to address these issues is through effective communication between stakeholders and the use of tools to manage requirements.

In addition to these challenges, there are also problems related to the integration of systems. Integration failures can lead to significant costs and delays, and can be caused by a variety of factors including incompatible systems, inadequate testing, and lack of planning. To prevent these issues, it is important to carefully plan and manage the integration process.

Another important factor is the impact of changes on system development projects. Changes in requirements, design, or technology can have a significant impact on the project, and can lead to delays, cost overruns, and other problems. To address these issues, it is important to have a clear change management process in place.

Finally, there are also issues related to the quality of the software development process. Poor quality can lead to defects and other problems, and can have a significant impact on the success of the project. To address these issues, it is important to have a robust quality management system in place.

Overall, these challenges highlight the importance of effective requirement engineering practices in systems development projects. By addressing these issues, organizations can improve the quality and success of their projects.
2016 3rd International Conference on Computer and Information Sciences, ICCOINS 2016 - Proceedings, 2016, pp. 606–610.

[25] J. Miller, “A total benefits strategy is a valuable approach in HR outsourcing,” Employ. Relat. Today, vol. 34, no. 4, pp. 55–61, 2008.

[26] S. Besrour, L. B. A. Rahim, and P. D. D. Dominic, “Investigating requirement engineering techniques in the context of small and medium software enterprises,” 2016 3rd Int. Conf. Comput. Inf. Sci. ICCOINS 2016 - Proc., pp. 519–523, 2016.

[27] G. Huzooree, “A Systematic Study on Requirement Engineering Processes and Practices in Mauritius,” Int. J. Adv. Res. Comput. Sci. Softw. Eng., vol. 5, no. 2, pp. 40–46, 2015.

[28] Y. Lu and T. Käkölä, “A dynamic life-cycle model for the provisioning of software testing services,” Syst. Sci. Control Eng., vol. 2, no. 1, pp. 549–561, 2014.

[29] U. A. Raja, “Empirical studies of requirements validation techniques,” 2009 2nd Int. Conf. Comput. Control Commun. IC4 2009, 2009.

[30] M. Kamalrudin and S. Sidek, “A review on software requirements validation and consistency management,” Int. J. Softw. Eng. its Appl., vol. 9, no. 10, pp. 39–58, 2015.

[31] D. Torre, Y. Labiche, M. Genero, and M. Elaasar, “A systematic identification of consistency rules for UML diagrams,” J. Syst. Softw., vol. 144, pp. 121–142, Oct. 2018.

Acknowledgments
This work was supported in part by University Malaysia Pahang under Grant RDU RDU191802-2.