

A Proposed Process Model for Requirements Engineering using Delphi Techniques for Prioritisation

Ishaya P. Gambo

Obafemi Awolowo University, Computer Science & Engineering Department, Ile-Ife, Nigeria
Email: ipgambo@gmail.com

Abimbola H. Soriyan, Rhoda N. Ikono

Obafemi Awolowo University, Computer Science & Engineering Department, Ile-Ife, Nigeria
Email: {hasoriyan, rhoda_u}@yahoo.com

Abstract— Requirements engineering (RE) addresses the first software development step and lays the foundation for a successful system. Consequently, ability to identify problems and suggestions for improvements in the RE process opens up significant potential for increasing the success of software projects. Since RE process is naturally collaborative in nature, the intensiveness from both knowledge and human perspectives opens up the problem of decision making on requirements that can be facilitated by requirements prioritisation. In this regard, the paper opined the need for requirements prioritization techniques that will help the developers to obtain consensus among stakeholders using a suitable technique. In particular, the paper proposed a RE process model using Delphi technique. The Delphi technique was suggested in this paper to facilitate and enhance the process of requirements prioritisation in a multilevel prioritisation dimension. Therefore, the proposed model on implementation will contribute to the formulation of an interactive framework for requirements prioritisation to produce a requirement ordering which complies with the existing priorities.

Index Terms— Requirement engineering, Prioritization, Delphi, Stakeholders, Software engineering, Requirements

I. INTRODUCTION

Every software system needs requirement definition. Requirements drive almost every activity, task, and deliverable in a software development project. Specifically, requirement defines what the software must do to add value for its users; what the software must be to add value for its users; and what limitations there are on the choices that developers have made when implementing the software. The development of this requirement is not just a collection process but a discovery and invention process. In a software development project, these requirements are determined and agreed to by the users (who are legitimate sources of requirements) before the software can be built. These requirements are further engendered through a process. This is with the view to improve system quality and users satisfaction; for easing compliance with standards and systemic regulations; reduction of project cost and delay;

control of complex projects in a way to reduce errors, manage risk and avoid possibly omission and ambiguities; and then to improve team communication. In particular, the success of a software system depends on how well it fits the needs of its users and its environment. Therefore, software requirements comprise these needs, and Requirements Engineering (RE) is the process by which the requirements are determined [1]. If the requirements are not right, the execution of the entire project during development will be faulty. In essence, RE lays the foundation for successful software and system development projects regarding cost and quality [2].

According to the ISO/IEC/IEEE FDIS 29148:2011 [3], requirements are defined as: “statement which translates or expresses a need and its associated constraints and conditions” The word ‘statement’ used in the definition can be captured in tabular form, in diagrammatic form as in notation such as Unified Modeling Language (UML), in formal notations, or in domain-specific notations. In most cases, software developers just talk about requirements without a prior knowledge of the different levels of requirements. Understanding the various levels of requirements is an important step towards system development with effective quality improvement in mind. In [4], the different levels of requirements were presented as shown in figure 1. The user requirements as shown in the figure are utmost in every development process. These requirements are based on the business requirements of the entire system. In particular, the users have vested interest in both the functional and nonfunctional requirements. All of these levels of requirements can be facilitated through the requirements engineering process.

However, in the context of SE, many approaches to the software development process have been formulated, In spite of their differences, virtually all of them includes the Requirements Engineering (RE) phase. This is because, RE addresses the first software development step and lays the foundation for a successful system. According to [5], “one of the main objectives of RE is to enrich systems modeling and analysis potentials so that businesses can better comprehend vital system aspects

before they actually develop the system”. The functional requirements together with quality attributes and other nonfunctional requirements will establish the software requirements specification [6]. RE has been defined as the “the branch of software engineering concerned with the real-world goals for functions of and constraints on software systems” [7]. In [8], RE is defined as “the subset of systems engineering concerned with discovering, developing, tracing, analysing, prioritising, qualifying, communicating and managing requirements that define the system at successive levels of abstraction”. The

authors in [9] defines RE as “a process of identifying the stakeholders and their needs, and documenting these in a form that is amenable to analysis, communication, and subsequent implementation”. In the opinion of [10], “RE is the structured process of eliciting, defining, negotiating, prioritising and validating the requirements of a system”. These definitions lists carefully selected key activities that are considered proper to RE within the scope of this paper, specifically when considering prioritisation issues.

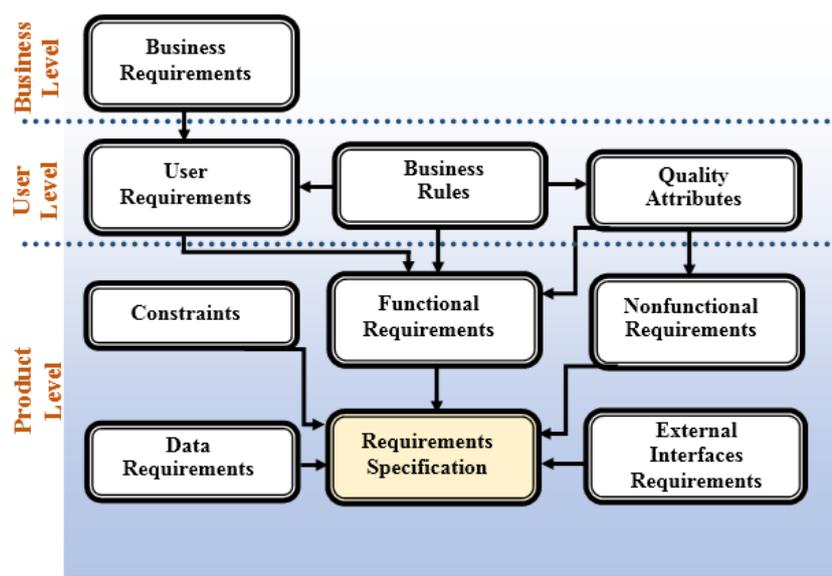


Fig. 1. Levels of Requirements (adapted from: [4])

According to [11], these activities will naturally involve a subset of the stakeholders of the system, such as analysts, developers, system architects, the client, the end users, domain experts, etc. In practice, stakeholders communicate their requirements to the analysts, who may specify them and communicate them back to the origin for validation, and to other stakeholders such as developers, who need the requirements. This initiates a process that needs to be followed in order to ensure quality improvement. The authors in [1] observed that RE is about defining precisely the problem that the software is to solve (i.e., defining what the software is to do). In reality, [12] observed that budget and schedule constraints limit the number of requirements that can be worked on for a software system and is thus necessary to select the most valuable requirements for implementation.

Hence, the justification for ‘prioritisation’. This is true in a situation where there are multiple and conflicting requirements elicited from stakeholders in the process of developing and/or implementing complex and service-oriented systems, for example, the healthcare system. However, selecting from a large number of requirements is a decision problem that requires negotiating with multiple stakeholders and satisficing their value propositions [12]. Requirements will come from many different stakeholders, involve multiple disciplines, and be presented at varying levels of abstraction [13].

Therefore, there is need for an analysis process model that represents requirements in a prioritised multiple dimension.

Finally, this paper is organised into six sections. This section, being the introduction, provides an exhaustive overview of requirements, RE and a brief discussion of requirements prioritisation. Section 2 deals with problem statement and motivation that justifies requirements engineering processes. In section 3, the relationship between requirements engineering process and prioritisation issues were established. The Delphi technique application to requirements prioritisation was discussed in section 4, and the proposed model was presented in section 5. The conclusion and implication of this paper is articulated in the last section.

II. PROBLEM STATEMENT AND MOTIVATION

The ability to identify problems and suggestions for improvements in the RE process opens up significant potential for increasing the success of software projects. In order to improve RE processes, the current practices need to be examined within the scope of a defined software development project.

The authors in [14] opined that “inaccurate, inadequate, or misunderstood requirements are the most common

causes of poor quality, cost overruns and late delivery of software systems". Requirements problems are widely acknowledged to reduce the quality of software and to impact on the effectiveness of the software development process [15]. Despite the importance of requirements engineering, little work has been done on developing ways to improve requirements process. As observed in [16] and [17], Information System (IS) projects deliver 42 to 67% of their original requirements and 48% of Information Technology (IT) development relates to requirements engineering. Consequently, many information system development problems could be traced to RE [18, 19, 20, 21].

Even with the significant varieties of software development processes and project management techniques, software projects may still fail due to vague definitions of project goals, ambiguous requirements, communication and coordination problems (e.g. conflicts that leads to omission in requirements and inconsistencies of requirements) among the stakeholders. From existing literature, prevailing techniques in RE processes do not provide a blended approach of understanding, capturing and representing different stakeholders' expectations in a way to prevent omission, ambiguity and inconsistency of requirements. As a result, requirements are prone to issues of disputes, collision of concerns, disparity and disagreement among the stakeholders [22, 23].

Therefore, there is need for requirements negotiation and prioritization techniques that will help the developers to obtain consensus among stakeholders; and there is need for understanding and modeling current RE processes towards improving RE practice and thereby increasing the success of software development. In fact, when improving the software engineering process, the area, which may have the largest effect on the result, is RE. This is because, requirements are the first things produced, and projects are conducted and finalized in strict concordance with them. So, the ability to improve the software product in order to meet the needs of all stakeholders in software development is a great concern in the software industries. Hence, requirements engineering process is required to provide the appropriate mechanism for quality assurance, resolving ambiguities,

vagueness and fuzziness subject to the conflicting concerns of stakeholders [24, 25].

III. REQUIREMENTS ENGINEERING PROCESS AND PRIORITISATION ISSUES

Holistically, the RE process is naturally collaborative in nature where validation issues are crucial and mandatory, thereby involving agreements among the stakeholders. According to [26] and [27], "broad mastering is generally infeasible for a single person" in the RE process. As such, an individual cannot claim knowing all that is required for developing any system, since lots of stakeholders with different backgrounds and perspectives are involved. As observed in [28], RE process censoriously has effect on the success of software development process. Figure 2 shows the modified processes from [29] for capturing requirements in a development process. The figure indicates that all other activity from the analysis level to validation can be made to return back to elicitation when need be. In addition, at the analysis level where the desired behaviour need to be understood and modelled, requirements prioritisation can be performed. To check that the specification matches the users' requirements at the validation level, prioritisation can be made useful.

A typical RE process as shown in figure 3, can have a number of inputs and outputs. Inputs to the RE process includes existing system documents, stakeholder's requirements (elicited, gathered or collected through appropriate methods and techniques), business and organisation procedures, domain knowledge. The outputs usually includes the final requirements listing, specification of system and system models. Within the specified phases as shown in figure 3, the requirements analysis phase can be enhanced with the inclusion of prioritisation. This is because, the intensiveness from both knowledge and human perspectives during analysis process opens up the problem of decision making on requirements that can be facilitated by requirements prioritisation. At this level, the requirements are graded or ranked in their order of importance.

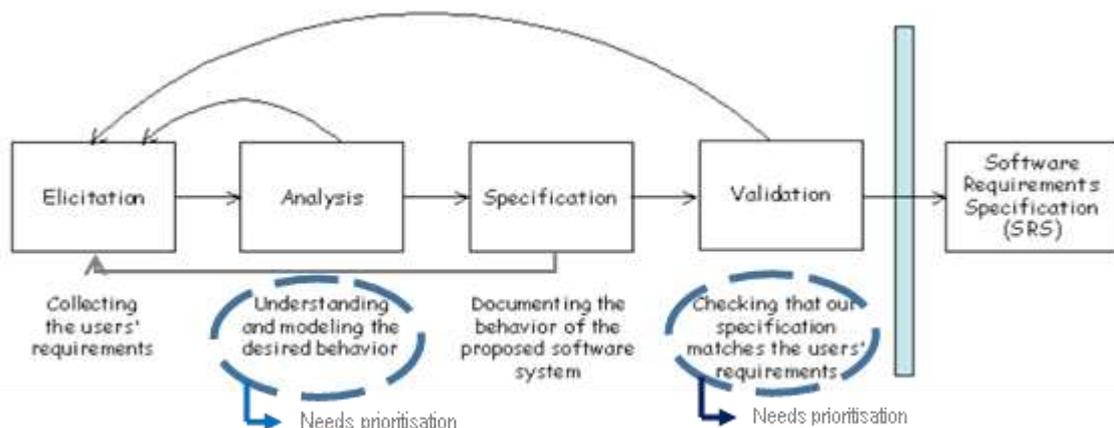


Fig. 2. The Process for capturing requirements (Modified from [29]).

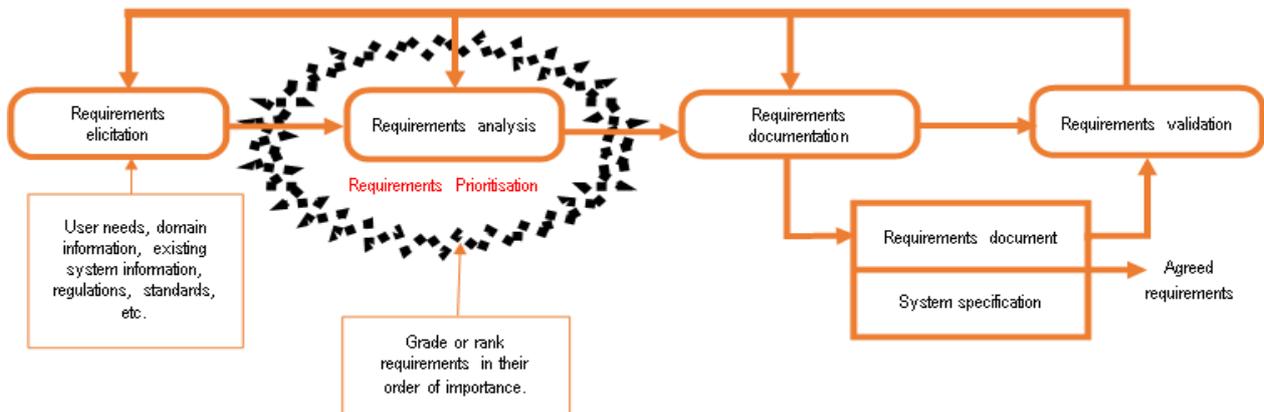


Fig. 3. A typical RE Process showing the need for including prioritisation of requirements at the analysis stage

With respect to prioritisation issues, the RE process is knowledge-intensive [30] and human-intensive [31], demanding an extensive analysis and tradeoff to be performed in order to prioritise requirements according to certain parameters to overcome the challenge of inconsistency, omission and ambiguity of requirements. This makes the RE process valuable when followed in evolving a system or in a software development project early stage. The activities, which are performed as part of the RE process, aim at the discovery and specification of requirements that unambiguously reflect the purpose of a software system as well as the needs of all relevant stakeholders [9]. It is with this in mind that the paper aimed at proposing a process model for RE using the Delphi technique that allows experts involvement in the prioritisation of requirements in producing the requirements lists and specification. In most cases the problem of unclear descriptions of requirement always occur, which often leads to terminological misalignment between analyst and stakeholders, and then potentially wrong requirements [32] are elicited. As a result, negative consequences of these wrong requirements are felt during all downstream activities such as architecting, design, implementation, and testing. As observed in [33], Poor-quality requirements greatly increase development and sustainment costs and often cause major schedule overruns. More to this problem is the elicitation of partial information due to stakeholder's omissions [34].

However, requirements prioritisation aims at selecting the 'right' requirements from a set of candidate requirements so that all the different key interests, technical constraints and preferences of the stakeholders are fulfilled [10]. Requirements prioritisation involves the allotment of priorities to different stakeholder's requirements. This facilitates requirement engineering process. It also helps requirement engineer's makes crucial decisions about requirements in a software development process. It is used to determine which candidate requirement of a software project should be included in a certain release and different techniques are used to facilitate that. The various techniques used different approaches and consider different factors for prioritisation e.g. cost, value, risk, benefit etc. Specifically, prioritisation is performed to grade or rank requirements in their order of importance and subsequent

implementation releases during RE [35]. As such, it is a major step taken in making crucial decisions so as to increase the economic value of a system. The authors in [36] opined that the priority of a requirement is a characteristic that can be used for different purposes, depending on the program and company needs. According to [10], various approaches are recognised by researchers for requirements prioritisation. In [37], prioritisation using the cost and benefit pertaining to every requirements was established. The authors in [22] opined that a prioritisation method using the relationship matrix utilizes the concept of correlation to compute weighted priorities of the requirements from the multiple perspectives of the stakeholders. In this paper, we are of the opinion that involving experts from the specified domain to prioritise these multiple requirements in a number of rounds will be useful to resolving consensus that will arise from the various stakeholder's expectations and requirements.

IV. THE DELPHI TECHNIQUE AND REQUIREMENTS PRIORITISATION

The underlining philosophy for the choice of using the Delphi technique for prioritisation is based on New Lanchester Theory (NLT) and the fact that the technique involves experts in making consensus based on stakeholders' inputted requirements. NLT links business requirements to market share using quantitative model [38]. The theory postulate that the strength of two or more forces in contention are equal if both suffer the same proportional losses. In specific terms, during requirements prioritisation, the theory can be applied to decide which requirement to specify for the expected software product. We infer from the NLT that getting the necessary requirements for any system requires prioritisation. Secondly, analysing the requirements gathered for a system requires prioritisation in order to avoid conflicts (in terms of inconsistency). In a way, we view requirements gathering and collection as a warfare process where various requirements are to be contended for in terms of making adequate choice on which requirement to use for system development. In such scenario, we are of the opinion that these elicited

requirements need to be prioritised (i.e. graded and ranked) in their order of importance. To achieve this, the Delphi technique becomes suitable for the ranking, where experts are expected to participate in the process. In essence, prioritising the requirements plays a significant role in reducing requirement problems and increasing customer satisfaction.

However, [39] opined that the Delphi Technique is based on structural surveys, it makes use of the intuitive available information of the participants, who are mainly experts, and it delivers qualitative as well as quantitative results. The technique is widely recognized as a 'consensus-building tool', which has been applied as a means of cognition and inquiry in a variety of fields, including RE. The Delphi Technique is said to be particularly appropriate in facilitating decision-making by a number of individuals in environments characterized by antagonistic or strongly opposed political or emotional factions, or when personality differences or intellectual style would be distracting in face-to-face settings [40, 41]. The technique consists of a number of rounds of questions to progressively identify, clarify and expand on issues and ideas. In this case, the issue in need of identification, clarification is the requirements. The first round usually consist of general questions to gain a broad understanding of the views, opinions or needs of the range of experts. The responses from this round are collated and summarized and this forms the basis of the next round of questions which will investigate further or clarify the issues raised in round 1. Again the results are collated and used to formulate a third (and usually final) set of questions which will generally consist of obtaining a consensus on the relative importance of the requirements identified in the previous rounds. A key

aspect of Delphi is the anonymized feedback that the experts receive between each round of questioning; general feedback is provided to the experts between rounds and individuals are given the opportunity to revise judgments as a result of the feedback. This actually qualifies the technique to be an iterative method. Whilst in the author in [42] defines the Delphi Technique as 'an interactive and personality free team approach', [43] proclaims its purpose is to 'elicit information and judgments from participants to facilitate problem-solving, planning, and decision making' capitalizing on respondent's creativity, whilst simultaneously maximizing the merits and minimizing the liabilities of group interaction. Since the technique involves non-interactive groups, it is possible to draw upon experts who are based at considerable geographic distances.

V. THE PROPOSED MODEL

Figure 4 shows the conceptual basic flow of the proposed model. The Delphi technique will be used to resolve the inputs arising from the requirements. This makes the approach a multilevel prioritisation technique with the involvement of experts. In contrast to other analysis techniques, Delphi employs multiple iterations designed to develop a consensus of opinion concerning the various requirements elicited. In the proposed model, Delphi involves the use of stakeholder's and experts at various stages or rounds to iteratively prioritise the requirements. The experts' iterative prioritisation is with the view of ensuring the evaluation of requirements in order to produce the specified requirements listing.

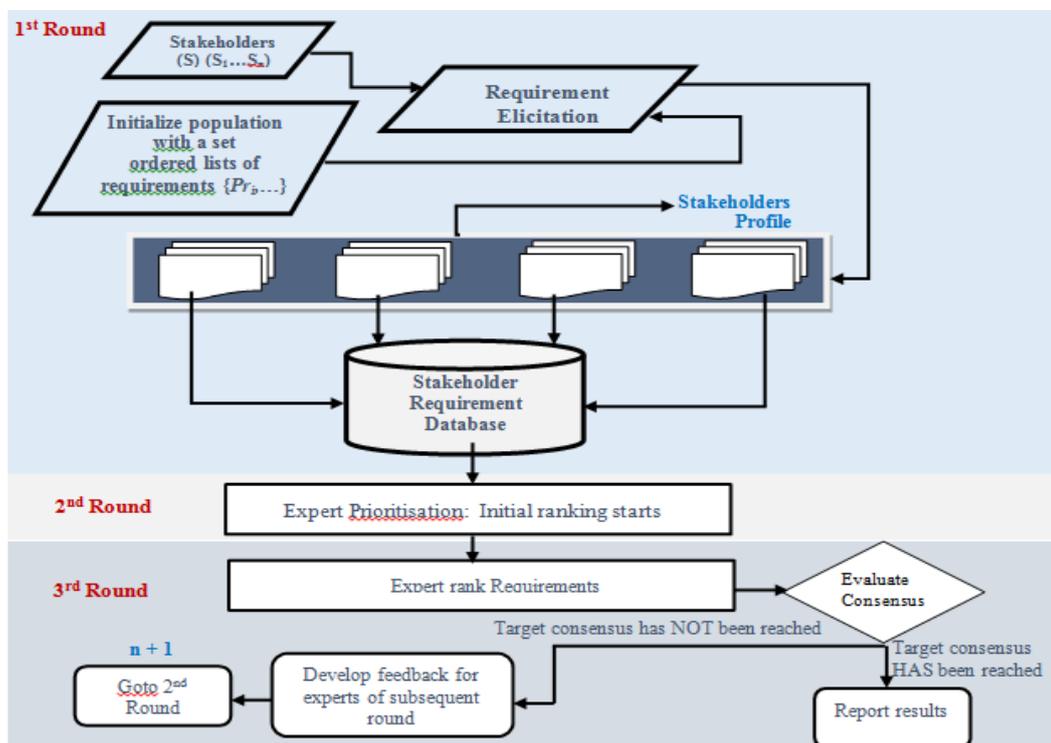


Fig. 4. roposed basic flow of the model

A. Requirements Elicitation

Requirements will be elicited from stakeholders using interview, case study, observation and secondary data. Elicitation process starts at the early stages of RE. In particular, the input to the model comes from the various stakeholder's requirements and/or expectation to be elicited. In addition, iterative and interactive means of eliciting requirements can also be a source of input. At this level, we envisage that appropriate algorithm be defined and used to facilitate requirements elicitation. In particular, we can have other inputs from the algorithm that defines a set of one or more partial orders ($ord_1 \dots ord_n$), derived from the requirement documents. In this case, we specify the priority, value, dependences, etc. from the requirements documents. Another source of input during requirements elicitation in the model can come from the business and organisation procedures. The algorithm can be used to support initialization of the various stakeholders with a set of totally ordered requirements. All of these requirements elicited from the stakeholders and the stakeholders' profiles will be kept in the stakeholders' requirement database. The stakeholders' profile includes a brief description of themselves, their expectations about the system and the system functionalities of their choice. At this stage, the first round of the Delphi technique is initiated. The stakeholders' requirements elicited will be used for the second round of the Delphi technique.

B. Expert Prioritisation

After the first round of requirements elicitation in the Delphi technique, experts are invited in the second round. During the second round, experts reviews the stakeholders' requirements and profiles as elicited. Accordingly, the experts are required to rate or "rank-order the requirements to establish preliminary priorities among the requirements. As a result, areas of disagreement and agreement are identified. It is possible at this level to request that experts justify the rating order of priorities among the various stakeholders' requirements. An interesting aspect of this round is the possibility of establishing consensus.

C. Expert Ranking of Requirements

In this level of prioritisation, the experts are given the opportunity to make further clarifications of both the information and their judgment of the relative importance of the requirements ranked. However, compared to the previous round, only a slight increase in the degree of consensus can be expected [44, 45, 46, 47]. The possibility of providing a final opportunity for the experts to revise their judgments can be establish in the third round of the Delphi technique or allowed to pass on to the fourth round. In a way, Delphi iterations depends largely on the degree of consensus sought by the investigators and can vary from three to five [48, 49]. Specifically, during the experts ranking of requirements, if the target consensus has not been reached a feedback for experts of subsequent round is initiated, otherwise, if target consensus has been reached, the result is reported.

VI. CONCLUSION

Understanding the different kinds of requirements and classifying users input into the appropriate categories during RE process is key to excellent software requirements for a software project. In order to improve the RE processes, the current practices need to be examined within the scope of a defined software development project. The majority projects of any size are created from an initial set of customer needs and then these needs are analyzed into the requirements [50]. As such, it is expedient to seek ways on improving the RE process. In doing this, the paper opined the need to consider prioritizing requirements in the analysis process of requirements engineering. Prioritizing requirements has been significantly discussed in the requirements engineering domain. We observed that the structured model in [51] can be further enhanced through prioritization since it involves decisions making.

Going further, it was generally discovered that the involvement of experts in decision-making on which requirement to pass on for development is crucial. The Delphi technique has been suggested in this paper to facilitate and enhance the process of requirements prioritisation in a multilevel prioritisation dimension, conceptualized in a model. Therefore, the proposed model on implementation will contribute to the formulation of an interactive framework for requirements prioritisation to produce a requirement ordering which complies with the existing priorities.

REFERENCES

- [1] Cheng, B. H., & Atlee, J. M. (2009). Current and future research directions in requirements engineering. In *Design Requirements Engineering: A Ten-Year Perspective* (pp. 11-43). Springer Berlin Heidelberg.
- [2] Broy, M. (2006). Requirements engineering as a key to holistic software quality. In *Computer and Information Sciences—ISCIS 2006* (pp. 24-34). Springer Berlin Heidelberg.
- [3] ISO., (2011). International Organisation for Standardization (ISO): ISO/IEC/IEEE FDIS 29148:2011, Systems and Software engineering – Life cycle processes – Requirements engineering.
- [4] Wiegers, K. E. (2009). *Software requirements*. O'Reilly Media, Inc.
- [5] W. Robinson, S. Pawlowski, and V. Volkov, Requirements Interaction Management, *ACM Computing Surveys*, 35(2):132-190, 2003.
- [6] K. E. Weigers, Karl Wiegers Describes 10 Requirements Traps to Avoid, *Software Testing and Quality engineering*, 2(1):34-40, 2000.
- [7] Zave, P. (1997). Classification of research efforts in requirements engineering. *ACM Computing Surveys (CSUR)*, 29(4), 315-321.
- [8] Hull, E., Jackson, K., and Dick, J. (2011). *Requirements engineering*. Springer.
- [9] Nuseibeh, B., & Easterbrook, S. (2000, May). Requirements engineering: a roadmap. In *Proceedings of the Conference on the Future of Software Engineering* (pp. 35-46). ACM.

- [10] Bajaj, P., and Arora, V. (2013). Multi-Person Decision-Making for Requirements Prioritisation using Fuzzy AHP. In ACM SIGSOFT Software Engineering Notes, vol 38, no. 5, pp. 1-6.
- [11] Sommerville I., Sawyer P. and Viller S. (1997) Requirements Process Improvement through the Phased Introduction of Good Practice, *Software Process-Improvement and Practice*, 3, 19-34.
- [12] Kukreja, N. (2013, May). Decision theoretic requirements prioritization: A two-step approach for sliding towards value realization. In *Software Engineering (ICSE), 2013 35th International Conference on* (pp. 1465-1467). IEEE.
- [13] Cheng, B. H., & Atlee, J. M. (2007, May). Research directions in requirements engineering. In *2007 Future of Software Engineering* (pp. 285-303). IEEE Computer Society.
- [14] El Emam, K., & Madhavji, N. H. (1995, March). A field study of requirements engineering practices in information systems development. In *International Symposium on Requirements Engineering, Proceedings of the Second IEEE* (pp. 68-80).
- [15] Sommerville I. (1996) *Software Engineering Fifth Edition*, Addison-Wesley.
- [16] SchrodL, H., and Wind, S. (2011). Requirements Engineering for Cloud Computing, *Journal of Communications and Computers*, Vol. 8, pp. 707-715.
- [17] Standish Group, CHAOS Report, available online <http://www.standishgroup.com>, 2010.
- [18] Rouibah, K., and Al-Rafee, S. (2009). Requirement engineering elicitation methods: A Kuwaiti empirical study about familiarity, usage and perceived value, *Information Management & Computer Security*, Vol. 17, No. 3, pp. 192-217.
- [19] Aurum, A., & Wohlin, C. (Eds.). (2005). *Engineering and managing software requirements*. Springer Verlag, Berlin.
- [20] Beecham, S., Hall, T., & Rainer, A. (2003). Software process improvement problems in twelve software companies: An empirical analysis. *Empirical software engineering*, 8(1), 7-42.
- [21] Leffingwell, D. (1997). Calculating the Return on Investment from More Effective Requirements Management, *American Programmer*, Vol. 10, No. 4, pp. 13-16.
- [22] Ahmad, S. (2008, March). Negotiation in the requirements elicitation and analysis process. In *Software Engineering, 2008. ASWEC 2008. 19th Australian Conference on* (pp. 683-689). IEEE.
- [23] In, H., & Roy, S. (2001). Visualization issues for software requirements negotiation. In *Computer Software and Applications Conference, 2001. COMPSAC 2001. 25th Annual International* (pp. 10-15). IEEE.
- [24] Watson, A. (2008, September). Reflections on Requirements Engineering. In *EDOC*, pp. xxxiii – xxxiii, ISBN: 978-0-7695-3373-5, IEEE.
- [25] Yen, J., & Tiao, W. A. (1997, January). A systematic tradeoff analysis for conflicting imprecise requirements. In *Requirements Engineering, 1997, Proceedings of the Third IEEE International Symposium on* (pp. 87-96). IEEE.
- [26] Damian, D., Izquierdo, L. Singer, J. and Kwan, I. (2007). Awareness in the wild: Why communication breakdowns occur. In *Second IEEE International Conference on Global Software Engineering, 2007. ICGSE 2007*, pages 81-90.
- [27] Kwan, I. and Damian, D. (2011). The hidden experts in software-engineering communication (NIER track). In *Proceedings of the 33rd International Conference on Software Engineering, ICSE '11*, pages 800-803, New York, NY, USA, ACM.
- [28] Dhirendra, P. (2013). Requirement Engineering Research, *International Journal of Computer Science & Engineering Technology (IJCSSET)*, Vol. 4 No. 04, pp. 447-450, ISSN: 2229-3345.
- [29] Sommerville, I., & Kotonya, G. (1998). Requirements engineering: processes and techniques. John Wiley & Sons, Inc.
- [30] Maalej, W., & Thurimella, A. K. (2009, September). Towards a research agenda for recommendation systems in requirements engineering. In *Managing Requirements Knowledge (MARK), 2009 Second International Workshop on* (pp. 32-39). IEEE.
- [31] Castro-Herrera, C., & Cleland-Huang, J. (2010, May). Utilizing recommender systems to support software requirements elicitation. In *Proceedings of the 2nd International Workshop on Recommendation Systems for Software Engineering* (pp. 6-10). ACM.
- [32] McAllister, C. A. (2006). Requirements determination of information systems: User and developer perceptions of factors contributing to misunderstandings. ProQuest.
- [33] Firesmith, D. (2007, May). Engineering Safety and Security Related Requirements for Software Intensive Systems. In *ICSE Companion* (p. 169).
- [34] Sawyer, P., Gervasi, V., & Nuseibeh, B. (2011, August). Unknown known: Tacit knowledge in requirements engineering. In *Requirements Engineering Conference (RE), 2011 19th IEEE International* (pp. 329-329). IEEE.
- [35] Achimugu, P., Selamat, A., Ibrahim, R., & Mahrin, M. N. R. (2014). A systematic literature review of software requirements prioritization research. *Information and Software Technology*.
- [36] Gaur, V., Soni, Anuja., and Bedi, P. (2010). An agent-oriented approach to requirements engineering, in *proceedings 2010 IEEE 2nd International Advance Computing Conference, India*, pp. 449-454. USA: IEEE Press and IEEE Xplore.
- [37] Herrmann, A. (2008). Requirements Prioritisation Based on Benefit and Cost Prediction: An Agenda for Future Research. IEEE, ISBN: 978-0-7695-3309-4, pp. 125-134.
- [38] Fehlmann, T. M. (2008, September). New Lanchester theory for requirements prioritization. In *Software Product Management, 2008. IWSPM'08. Second International Workshop on* (pp. 35-40). IEEE.
- [39] Warth, J., von der Gracht, H. A., & Darkow, I. L. (2013). A dissent-based approach for multi-stakeholder scenario development—the future of electric drive vehicles. *Technological Forecasting and Social Change*, 80(4), 566-583.
- [40] Cline, A. (2000). Prioritization process using Delphi technique. *Carolla Development*. Retrieved March, 8, 2005.
- [41] Rosenthal, E.J. (1976) 'Delphi Technique' in Anderson, S. (ed) *Encyclopedia of Educational Evaluation*, San Francisco: Jossey-Bass: 121-122.
- [42] McNamee, P.B. (1985) '*Tools and Techniques for Strategic Management*', Oxford: Pergamon Press.
- [43] Dunham, Randall, B. (1998) 'Organisational Behaviour: The Delphi Technique', University of Wisconsin School of Business, Retrieved from the World Wide Web, www.instruction.bus.wisc.edu/obdemo/readings/delphi.htm 10th April, 2001.
- [44] Weaver, W. T. (1971). The Delphi forecasting method. *Phi Delta Kappan*, 52 (5), 267-273.
- [45] Dalkey, N. C., & Rourke, D. L. (1972). Experimental assessment of Delphi procedures with group value judgments. In N. C. Dalkey, D. L. Rourke, R. Lewis, & D. Snyder (Eds.). *Studies in the quality of life: Delphi and*

- decision-making* (pp. 55-83). Lexington, MA: Lexington Books.
- [46] Anglin, G. L. (1991). *Instructional technology past, present and future*. Englewood, CO: Libraries Unlimited Inc.
- [47] Jacobs, J. M. (1996). *Essential assessment criteria for physical education teacher education programs: A Delphi study*. Unpublished doctoral dissertation, West Virginia University, Morgantown.
- [48] Delbecq, A. L., Van de Ven, A. H., & Gustafson, D. H. (1975). *Group techniques for program planning*. Glenview, IL: Scott, Foresman, and Co.
- [49] Ludwig, B. G. (1994). *Internationalizing Extension: An exploration of the characteristics evident in a state university Extension system that achieves internationalization*. Unpublished doctoral dissertation, The Ohio State University, Columbus.
- [50] Alghazzawi, D. M., Siddiqui, S. T., Bokhari, M. U., & Hamatta, H. S. A. (2014). Selecting Appropriate Requirements Management Tool for Developing Secure Enterprises Software. *International Journal of Information Technology and Computer Science (IJITCS)*, 6(4), 49-55.
- [51] Kumar, V., & Thareja, R. (2013). Goal Structured Requirement Engineering and Traceability Model for Data Warehouses. *International Journal of Information Technology and Computer Science (IJITCS)*, 5(12), 78-85.

Authors' Profiles

Ishaya Gambo is pursuing a PhD degree in the area of requirements engineering at the Department of Computer Science and Engineering, Obafemi Awolowo University Ile-Ife, Nigeria.

Abimbola Soriyan is a professor in Computer Science at Obafemi Awolowo University (OAU) Ile-Ife, Nigeria. Her research interest is in Information System, Software engineering, and Health Informatics.

Ikono Rhoda is a lecturer of Computer Science and has a PhD in Computer Science at Obafemi Awolowo University Ile-Ife, Nigeria. Her research interest is in Information System, Software engineering, Health Informatics, and Software Product Usability.

How to cite this paper: Ishaya P. Gambo, Abimbola H. Soriyan, Rhoda N. Ikono, "A Proposed Process Model for Requirements Engineering using Delphi Techniques for Prioritisation", *International Journal of Information Technology and Computer Science (IJITCS)*, vol.7, no.1, pp.73-80, 2015. DOI: 10.5815/ijitcs.2015.01.09